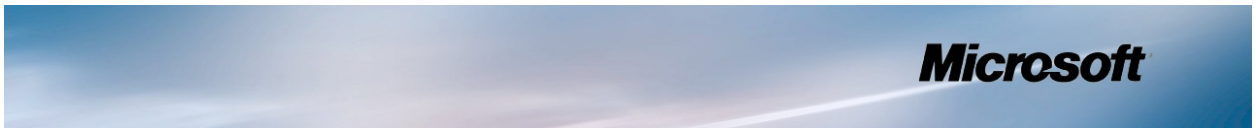


Microsoft Windows

Common Criteria Evaluation

Microsoft Windows 8

Microsoft Windows RT

Security Target

Document Information	
Version Number	1.0
Updated On	December 19, 2014

This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The example companies, organizations, products, people and events depicted herein are fictitious. No association with any real company, organization, product, person or event is intended or should be inferred.

Microsoft, Active Directory, Visual Basic, Visual Studio, Windows, the Windows logo, Windows NT, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

TABLE OF CONTENTS

LIST OF TABLES

1 Security Target Introduction

This section presents the following information required for a Common Criteria (CC) evaluation:

- Identifies the Security Target (ST) and the Target of Evaluation (TOE);
- Specifies the security target conventions and conformance claims; and,
- Describes the organization of the security target.

1.1 Security Target, TOE, and Common Criteria (CC) Identification

ST Title: Microsoft Windows 8 and Windows RT Security Target

ST Version: 1.0; December 19, 2014

TOE Software Identification: The following Windows Operating Systems (OS):

- Microsoft Windows 8 Edition (32-bit and 64-bit versions)
- Microsoft Windows RT

The following security updates and patches must be applied to the above Windows 8 products:

- All critical updates as of October 31, 2013.

The following security updates must be applied to the above Windows RT products:

- All critical updates as of October 31, 2013.

TOE Hardware Identification: The following hardware platforms and components are included in the evaluated configuration:

- Microsoft Surface
- Dell Optiplex GX620
- Dell XPS 8500
- ASUS VivoTab (Windows RT NVidia tablet)
- Dell XPS10 (Windows RT Qualcomm tablet)
- Dell Precision M6300
- Trusted Platform Module

TOE Guidance Identification: The following administrator, user, and configuration guides were evaluated as part of the TOE:

- *Microsoft Windows 8 Microsoft Windows RT Common Criteria Supplemental Admin Guidance* along with all the documents referenced therein.

Evaluation Assurance: As specified in section 5.3, section 5.4, and section 5.5.

CC Identification: CC for Information Technology (IT) Security Evaluation, Version 3.1, Revision 4, September 2012.

1.2 CC Conformance Claims

This TOE and ST are consistent with the following specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, Version 3.1, Revision 4, September 2012, extended (Part 2 extended)
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements Version 3.1, Revision 4September 2012, conformant (Part 3 conformant)
- General Purpose Operating System Protection Profile, Version3.9, December 2012 (OSPP) (draft)
- Evaluation Assurance Activities specified in section 5.3, section 5.4, and CC Part 3 assurance requirements specified in section 5.5

1.3 Conventions, Terminology, Acronyms

This section specifies the formatting information used in the security target.

1.3.1 Conventions

The following conventions have been applied in this document:

- Security Functional Requirements (SFRs): Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations.
    - Assignment: allows the specification of an identified parameter.
    - Selection: allows the specification of one or more elements from a list.
    - Refinement: allows the addition of details.

  The conventions for the assignment, selection, refinement, and iteration operations are described in Section 5.

- Other sections of the security target use a bold font to highlight text of special interest, such as captions.

1.3.2 Terminology

The following terminology is used in the security target:

Term	Definition
Access	Interaction between an entity and an object that results in the flow or modification of data.
Access control	Security service that controls the use of resources[1] and the disclosure and modification of data[2].
Accountability	Tracing each activity in an IT system to the entity responsible for the activity.

[1] Hardware and software
[2] Stored or communicated

Active Directory	Active Directory manages enterprise identities, credentials, information protection, system and application settings through AD Domain Services, Federation Services, Certificate Services and Lightweight Directory Services.
Active Directory Application Mode (ADAM)	Active Directory Application Mode is a LDAP service that runs with user privileges instead of system privileges on a Windows operating system. ADAM is now known as AD Lightweight Directory Services.
Administrator	An authorized user who has been specifically granted the authority to manage some portion or the entire TOE and thus whose actions may affect the TOE Security Policy (TSP). Administrators may possess special privileges that provide capabilities to override portions of the TSP.
Application Filter	An application filter can access the data stream or datagrams associated with a session within the Application Firewall service.
Assurance	A measure of confidence that the security features of an IT system are sufficient to enforce the IT system's security policy.
Attack	An intentional act attempting to violate the security policy of an IT system.
Authentication	A security measure that verifies a claimed identity.
Authentication data	The information used to verify a claimed identity.
Authorization	Permission, granted by an entity authorized to do so, to perform functions and access data.
Authorized user	An authenticated user who may, in accordance with the TOE Security Policy, perform an operation.
Availability	Timely[3], reliable access to IT resources.
Compromise	Violation of a security policy.
Confidentiality	A security policy pertaining to disclosure of data.
Critical cryptographic security parameters	Security-related information appearing in plaintext or otherwise unprotected form and whose disclosure or modification can compromise the security of a cryptographic module or the security of the information protected by the module.
Cryptographic boundary	An explicitly defined contiguous perimeter that establishes the physical bounds (for hardware) or logical bounds (for software) of a cryptographic module.
Cryptographic key (key)	A parameter used in conjunction with a cryptographic algorithm that determines: • the transformation of plaintext data into ciphertext data • the transformation of ciphertext data into plaintext data • a digital signature computed from data • the verification of a digital signature computed from data • a data authentication code computed from data
Cryptographic module	The set of hardware, software, and/or firmware that implements approved security functions, including cryptographic algorithms and key generation, which is contained within the cryptographic boundary.
Cryptographic module	A precise specification of the security rules under which a cryptographic

[3] According to a defined metric

security policy	module must operate.
Custom rule	A claim rule authored using the claim rule language to express a series of complex logic conditions.
Defense-in-depth	A security design strategy whereby layers of protection are utilized to establish an adequate security posture for an IT system.
Discretionary Access Control (DAC)	A means of restricting access to objects based on the identity of subjects and groups to which the objects belong. The controls are discretionary meaning that a subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject.
Edition	A distinct variation of a Windows OS version. Examples of editions are Windows Server 2012 [Standard] and Windows Server 2012 Datacenter.
Emulation	Emulation is the simulation of a processor or device using software facilitated by the Hypervisor.
Enclave	A collection of entities under the control of a single authority and having a homogeneous security policy. They may be logical, or based on physical location and proximity.
Enlightenment	An enlightened guest operating system has knowledge that it is running within a virtualized environment and changes its behavior accordingly.
Entity	A subject, object, user or external IT device.
General-Purpose Operating System	A general-purpose operating system is designed to meet a variety of goals, including protection between users and applications, fast response time for interactive applications, high throughput for server applications, and high overall resource utilization.
Guest Partition	Software running within a non-root partition is referred to as a guest. A guest might consist of a full-featured operating system like Windows Server 2008 R2 or a small, special-purpose kernel. The hypervisor is “guest-agnostic”.
Hypervisor	A hypervisor is a layer of software that sits just above the hardware and beneath one or more operating systems. Its primary job is to provide isolated execution environments called partitions.
Hyper-V Snapshot	A snapshot is a collection of data about a partition and its current state that allows restarting the partition in this state. A Hyper-V snapshot therefore includes all of the information and data that is required to roll back the status of a partition to the state when the snapshot was taken. Information that is collected when taken a snapshot include: • Partition configuration settings (the contents of the .vmc file) • Virtual network settings • The current state of all virtual hard disks (VHDs) that are attached to the partition • State information for the partition
Hyper-V VM Worker Process	The Hyper-V Worker Process provides an emulated system BIOS and a wide variety of emulated devices. VM worker processes are part of the root partition and receive specific notifications that specific events have occurred within a guest partition.
Identity	A means of uniquely identifying an authorized user of the TOE.

Integrated Windows authentication	An authentication protocol formerly known as NTLM or Windows NT Challenge/Response.
Legacy Guest	A legacy guest is an operating system that has no knowledge of the fact that it is running within a virtualized environment. Legacy guests require substantial infrastructure including a system BIOS and a wide variety of emulated devices, which is not provided directly by the Hypervisor.
Microsoft Reputation Service (MRS)	The Microsoft Reputation Service (MRS) is a cloud-based object categorization system designed to provide comprehensive reputation content to enable core trust scenarios across Forefront and Microsoft security and management endpoint solutions.
Named object	An object that exhibits all of the following characteristics: • The object may be used to transfer information between subjects of differing user identities within the TOE Security Function (TSF). • Subjects in the TOE must be able to request a specific instance of the object. • The name used to refer to a specific instance of the object must exist in a context that potentially allows subjects with different user identities to request the same instance of the object.
Object	An entity under the control of the TOE that contains or receives information and upon which subjects perform operations.
Operating environment	The total environment in which a TOE operates. It includes the physical facility and any physical, procedural, administrative and personnel controls.
Partitions	A partition has its own set of physical or virtual hardware resources namely, memory, devices, and processor cycles. The Hypervisor is responsible for controlling and arbitrating access to the underlying hardware where necessary.
Persistent storage	All types of data storage media that maintain data across system boots (e.g., hard disk, removable media).
Public object	An object for which the TSF unconditionally permits all entities “read” access under the Discretionary Access Control SFP. Only the TSF or authorized administrators may create, delete, or modify the public objects.
Resource	A fundamental element in an IT system (e.g., processing time, disk space, and memory) that may be used to create the abstractions of subjects and objects.
SChannel	A security package (SSP) that provides network authentication between clients and servers.
Secure State	Condition in which all TOE security policies are enforced.
Security attributes	TSF data associated with subjects, objects and users that is used for the enforcement of the TSP.
Security-enforcing	A term used to indicate that the entity (e.g., module, interface, subsystem) is related to the enforcement of the TOE security policies.
Security-supporting	A term used to indicate that the entity (e.g., module, interface, subsystem) is not security-enforcing; however, the entity’s implementation must still preserve the security of the TSF.
Security context	The security attributes or rules that are currently in effect. For SSPI, a

	security context is an opaque data structure that contains security data relevant to a connection, such as a session key or an indication of the duration of the session.
Security package	The software implementation of a security protocol. Security packages are contained in security support provider libraries or security support provider/authentication package libraries.
Security principal	An entity recognized by the security system. Principals can include human users as well as autonomous processes.
Security Support Provider (SSP)	A dynamic-link library that implements the SSPI by making one or more security packages available to applications. Each security package provides mappings between an application's SSPI function calls and an actual security model's functions. Security packages support security protocols such as Kerberos authentication and Integrated Windows Authentication.
Security Support Provider Interface (SSPI)	A common interface between transport-level applications. SSPI allows a transport application to call one of several security providers to obtain an authenticated connection. These calls do not require extensive knowledge of the security protocol's details.
Security Target (ST)	A set of security requirements and specifications to be used as the basis for evaluation of an identified TOE.
Subject	An active entity within the TOE Scope of Control (TSC) that causes operations to be performed. Subjects can come in two forms: trusted and untrusted. Trusted subjects are exempt from part or all of the TOE security policies. Untrusted subjects are bound by all TOE security policies.
Target of Evaluation (TOE)	An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.
Threat	Capabilities, intentions and attack methods of adversaries, or any circumstance or event, with the potential to violate the TOE security policy.
Unauthorized individual	A type of threat agent in which individuals who have not been granted access to the TOE attempt to gain access to information or functions provided by the TOE.
Unauthorized user	A type of threat agent in which individuals who are registered and have been explicitly granted access to the TOE may attempt to access information or functions that they are not permitted to access.
Universal Unique Identifier (UUID)	UUID is an identifier that is unique across both space and time, with respect to the space of all UUIDs. A UUID can be used for multiple purposes, from tagging objects with an extremely short lifetime, to reliably identifying very persistent objects across a network.
User	Any person who interacts with the TOE.
User Principal Name (UPN)	An identifier used by Microsoft Active Directory that provides a user name and the Internet domain with which that username is associated in an e-mail address format. The format is [*AD username*]@[associated *domain*]; an example would be *john.smith@microsoft.com.*
Uniform Resource Locator (URL)	The address that is used to locate a Web site. URLs are text strings that must conform to the guidelines in RFC 2396.
Version	A Version refers to a release level of the Windows operating system.

	Windows 7 and Windows 8 are different versions.
Virtualization	Virtualization provides multiple logical instances of processors and other hardware resources by the Hypervisor. These logical instances are then mapped onto physical hardware resources.
Vulnerability	A weakness that can be exploited to violate the TOE security policy.
Windows Communication Foundation (WCF)	The Microsoft unified programming model for building service-oriented applications. Developers can use WCF to build secure, reliable, transacted solutions that integrate across platforms and interoperate with existing programs.

1.3.3 Acronyms

The acronyms used in this security target are specified in **Appendix A: List of Abbreviations**.

1.4 ST Overview and Organization

The Windows 8 and Windows RT, known hereafter as "Windows", is a general-purpose operating system that provides controlled access between subjects and user data objects. The Windows TOE has a broad set of security capabilities including

- Single logon to the network (using a password)
- Access control and data encryption
- FIPS 140-2 validated cryptography
- Extensive security audit collection
- Host-based firewall and IPsec to control information flow
- Built-in standards-based network security protocols such as
  - Kerberos[4]
  - Transport Layer Security (TLS)/Secure Sockets Layer (SSL)[5]
  - Digest[6]
  - Internet Key Exchange (IKE)/IPsec[7]
  - Light-weight Directory Access Protocol (LDAP) Directory-based resource management[8]

The Windows TOE provides the following security services:

[4] See http://msdn.microsoft.com/en-us/library/cc233855(PROT.10).aspx for more information about the Windows implementation of Kerberos.
[5] See http://msdn.microsoft.com/en-us/library/dd207968(PROT.10).aspx for more information about the Windows implementation of TLS/SSL.
[6] See http://msdn.microsoft.com/en-us/library/cc227906(PROT.10).aspx for more information about the Windows implementation of Digest authentication.
[7] See http://msdn.microsoft.com/en-us/library/cc233219(PROT.10).aspx for more information about the Windows implementation of IKE and IPsec.
[8] See http://msdn.microsoft.com/en-us/library/cc223122(PROT.10).aspx for more information about the Windows implementation of LDAP.

- User data protection
  - Discretionary Access Control (DAC)
  - Mandatory Integrity Control (MIC)
  - IPsec information flow control
  - Windows firewall information flow control
- Cryptographic support
- Audit
- Identification and Authentication (I&A)
  - including trusted path/channel
- Security management
- Protection of the TOE Security Functions (TSF)
- TOE access/session control

The Windows security policies provide network-wide controlled access protection (access control for user data, web access and web content publishing, IPsec information flow, connection firewall information flow).

These policies enforce access limitations between individual users and data objects, as well as incoming and outgoing traffic channels through physically separated parts of the TOE. The TOE is capable of auditing security relevant events that occur within a Windows network. All these security controls require users to identify themselves and be authenticated prior to using any node on the network.

The Windows security target contains the following additional sections:

- TOE Description (Section 2): Provides an overview of the TSF and boundary.
- Security Problem Definition (Section 3): Describes the threats, organizational security policies and assumptions that pertain to the TOE.
- Security Objectives (Section 4): Identifies the security objectives that are satisfied by the TOE and the TOE operational environment.
- Security Requirements (Section 5): Presents the security functional and assurance requirements met by the TOE.
- TOE Summary Specification (TSS) (Section 6): Describes the security functions provided by the TOE to satisfy the security requirements and objectives.
- Protection Profile Conformance Claim (Section 7): Presents the rationale concerning compliance of the ST with the ***Operating System Protection Profile,*** the rationale for the security objectives, requirements, and TOE Summary Specification as to their consistency, completeness and suitability.

2 Product and TOE Description

The TOE includes the Windows 8 operating system, the Microsoft Windows RT operating system, supporting hardware, and those applications necessary to manage, support and configure the operating system.

2.1 Windows Editions

The TOE includes product variants of Windows 8 and Windows RT:

- Windows 8 [consumer edition]
- Windows RT

Windows 8 is suited for business desktops and notebook computers. It is the workstation product.

Windows RT is a new Windows-based operating system that is optimized for thin and light PCs that have extended battery life and are designed for mobile use. Windows RT only runs built-in apps or apps that are downloaded from the Windows Store. Windows Update automatically keeps your PC up to date. Windows RT client computers cannot be connected to a Windows domain.

Windows is a preemptive multitasking, multiprocessor, and multi-user operating systems. In general, operating systems provide users with a convenient interface to manage underlying hardware. They control the allocation and manage computing resources such as processors, memory, and Input/Output (I/O) devices. Windows 8 and Windows RT, collectively referred to as Windows, expand these basic operating system capabilities to controlling the allocation and managing higher level IT resources such as security principals (user or machine accounts), files, printing objects, services, window station, desktops, cryptographic keys, and network ports traffic. Multi-user operating systems such as Windows keep track of which user is using which resource, grant resource requests, account for resource usage, and mediate conflicting requests from different programs and users.

In terms of security, Windows product variants share the same security characteristics.

Windows provides an interactive User Interface (UI), as well as a network interface. The TOE includes a set of Windows 8systems that can be connected via their network.

In addition to core **operating system** capabilities Windows 8 and Windows RTcan also be categorized as the following types of Information Assurance (IA) or IA-enabled IT products:

- Windows is a **Single Sign-On** product (using password or certificate) for Windows networks to defend the computing environment.
- Windows is a **Firewall** product with the capability to filter network traffic based upon source and destination addresses, ports, applications, user or machine identity, and protocols.

Related operating system capabilities are covered in other evaluations:

- Windows 8 and Windows RT is a **VPN** product providing an IPsec service, known as Direct Access, and its associated Transport Driver Interface (TDI) as well as Windows Filtering Platform (WFP) based network support. Windows 8, Windows RT and Server 2012 were evaluated against the IPsec VPN Client protection profile, the evaluation report is published here.

2.1.1 Evaluation Scope

The security features addressed by this security target are only those provided by Windows as described by the OS PP.

The following table summarizes the Windows configurations included in the evaluation.

Table 2-1 Evaluated Configurations of Windows

	Windows 8	Windows RT
x86	Yes	Yes
x64	Yes	Yes
ARM v7	N/A	Yes
Single Core/Processor	Yes	Yes
Multiple Core/Processor	Yes	Yes

2.2 Product Features

Windows has many features that improve network security and scalability, several of which support simplifying the administration and management of a distributed computing environment. This section indicates which features implement security functions that are specified by the OS PP or related protection profiles, support a OS PP or related PP security function, and general infrastructure which Windows relies upon to operate.

With this approach the reader can better understand the context of what was tested, i.e., the TOE security functions, within the overall product.

Unless stated otherwise within the following subsections, the security feature being described apply to all of the versions and editions of Windows being evaluated.

2.2.1 Security Functions from Protection Profiles

2.2.1.1 OS Protection Profile

2.2.1.1.1 Access Control Lists

Windows permits only authenticated users to access system resources using the Discretionary Access Control policy. The security model includes components to control which users can access which objects (such as files, directories, and shared printers), what actions an individual can perform with respect to an object, and the events that are audited.

Every object has a unique Security Descriptor (SD) that includes an Access Control List (ACL). An ACL is a list of entries that grant or deny specific access rights to individuals or groups. The Windows object-

based security model lets administrators grant access rights to a user or group that govern who can access a specific object managed by the local computer.

2.2.1.1.2 Cryptographic API: Next Generation

Windows supplements the legacy CryptoAPI with the Cryptography API: Next Generation (CNG). CNG provides applications with access to cryptographic functions, public keys, credential management and certificate validation functions, and as well as support for the United States National Security Agency's Suite B cryptographic algorithms to user-mode and kernel mode applications. CNG also provides extensive auditing support, support for replaceable random number generators and a key isolation service to limit the exposure of secret and private keys.

2.2.1.1.3 Digest Authentication

Digest authentication operates much like Basic network authentication. However, unlike Basic authentication, Digest authentication transmits credentials across the network as a hash value, also known as a message digest. The user name and password cannot be deciphered from the hash value. Conversely, Basic authentication sends a Base 64 encoded password, essentially in clear text, across the network. Basic authentication is an unsupported scenario in the evaluation. Digest authentication does not have to use reversible password encryption.

2.2.1.1.4 Event Logging Infrastructure

Windows improvements to the event logging infrastructure make the platform easier to manage, monitor, and provide better information for troubleshooting. Many components that stored logging information in text files in previous versions are now able to add events to the event log. With event forwarding, administrators can centrally manage events from remote computers on the network, making it easier to identify problems and to correlate problems that affect multiple computers. Additionally, the Event Viewer application allows users to create custom views of audit data, to easily associate events with tasks, and to remotely view logs from other computers.

2.2.1.1.5 Integrated IPsec Support

Windows includes identical IPsec support for both IPv4 and IPv6. Full support for Internet Key Exchange (IKE) and data encryption is provided for both IP stacks. IPsec configuration is integrated with the Windows Firewall with Advanced Security MMC snap-in to improve manageability and reduce the likelihood of conflicting firewall and IPsec rules.

2.2.1.1.6 IPv6

Windows provides a dual IP stack in which IPv4 and IPv6 are implemented alongside each other and share a common IP transport (including TCP and UDP). IPv6 is enabled by default and supports numerous enhancements including a GUI based configuration, improvements to Teredo (an IPv6 transition technology), generation of interface IDs, a DHCPv6 client that support stateful address auto configuration, and for Windows Server, a DHCPv6 capable server.

2.2.1.1.7 Job Object

The Job Object API, with its ability to specify processor affinity, establish time limits, control process priorities, and limit memory utilization for a group of related processes, enables an application to

control system resources, is managed by the Discretionary Access Control policy. This additional level of control means the Job Object can prevent an application from negatively impacting overall system scalability.

2.2.1.1.8 Kerberos Authentication Support

Full support for Kerberos Version 5 (v5) protocol in Windows provides fast, single sign-on to Windows enterprise resources.

2.2.1.1.9 Kernel Debug Management

The Kernel Debugger subcomponent supports authorized users to debug running processes by allowing them to attach a debugger to a running process via a kernel object, the “Debug Object” that is managed by the Discretionary Access Control policy.

2.2.1.1.10 Kernel Transaction Manager

Windows includes a transaction engine that enables applications to use atomic transactions on resources to facilitate improved error recovery. This transaction engine allows transactional resource managers such as the NT File System (NTFS) and the Configuration Manager (i.e., the registry) to coordinate their updates for a specific set of changes made by an application which is managed by the Discretionary Access Control policy.

2.2.1.1.11 Mandatory Integrity Control

In addition to Discretionary Access Control (DAC), Windows provides Mandatory Integrity Control (MIC). MIC uses integrity levels and a mandatory policy to evaluate access. Processes and securable objects (e.g., files) are assigned integrity levels that determine their levels of protection or access.

As an integrity policy, a process with a lower integrity level (e.g., low) cannot write to an object with a higher integrity level (e.g., medium), even if that object's DAC policy allows write access. On the other hand, processes can access objects that have an integrity level lower than or equal to their own integrity level. In addition, the MIC policy addresses read and execute accesses, and can be configured to restrict a process with a lower integrity level from reading and/or executing objects with a higher integrity level.

The integrity labels defined in Windows are:

- Untrusted: Used by processes started by the Anonymous group;
- Low: Used by protected mode Internet Explorer; Low blocks write access to most objects (such as files and registry keys) on the system;
- Medium: Normal applications being launched while user account control (UAC) is enabled;
- High: Applications launched through administrator elevation when UAC is enabled, or normal applications if UAC is disabled; and
- System: Services and other system-level applications (such as WinLogon).

2.2.1.1.12 Microsoft Management Console

Microsoft Management Console (MMC) unifies and simplifies system management tasks through a central, customizable console that allows control, monitoring, and administration of widespread network resources. MMC 3.0 provides a new add or remove snap-ins dialog box, improved error handling, and an action pane that provides context sensitive access to features based on the currently selected items in the tree or results pane.

2.2.1.1.13 Network Address Translation

Network Address Translation (NAT) hides internally managed IP addresses from external networks by translating private internal addresses to public external addresses. This translation reduces IP address registration costs by using private IP addresses internally, which are translated to a small number of registered IP addresses externally. NAT also hides the internal network structure, which can reduce the risk of attacks against internal systems. The Windows TOE IPsec implementation works transparently with NAT without interoperability issues.

2.2.1.1.14 Secure Network Communications

Windows supports end-to-end encrypted communications across network using the IPsec standard. It protects sensitive internal communications from intentional or accidental viewing.

2.2.1.1.15 Support for Security Standards

Windows builds secure network sites using current standards, including 128-bit and 256-bit SSL/TLS, IPsec, and Kerberos v5 authentication.

2.2.1.1.16 User Account Control

User Account Control (UAC), alternately known as Least Privilege User Access (LUA) enables users to perform common tasks as non-administrators, called standard users, and as administrators without having to switch users, log off, or use the Run As command. A standard user account is synonymous with a user account in Windows. User accounts that are members of the local Administrators group will run most applications as a standard user.

When an administrator logs on to a computer running Windows, the user's full administrator access token is split into two access tokens: a full administrator access token and a standard user access token. During the logon process, authorization and access control components that identify an administrator are removed, resulting in a standard user access token. The standard user access token is then used to start the Widows desktop process. Because all applications inherit their access control data from the initial launch of the desktop, they all run as a standard user as well.

After an administrator logs on, the full administrator access token is not used until the administrator attempts to perform an administrative task at which point the user will be interactively prompted to confirm this access escalation.

2.2.1.1.17 Windows Firewall

Windows Firewall is a stateful firewall that drops unsolicited incoming traffic which does not correspond to either (1) traffic sent in response to a request of the computer (solicited traffic) or (2) unsolicited

traffic that has been specified as allowed (excepted traffic). Windows Firewall provides a level of protection from malicious users and programs that rely on unsolicited incoming traffic to attack computers. Windows Firewall supports IPv4 and IPv6. The firewall drivers (for IPv4 and for IPv6 respectively) have a static rule called a boot-time policy to perform stateful filtering. Once the firewall service is running, it will load and apply the runtime policy and remove the boot-time filters.

2.2.1.1.18 Window Manager

The Window Manager is implemented in kernel mode. It provides a machine independent graphical Application Programming Interface (API) for applications to control printing and window graphics, by providing a way to display information and receiving user input. Users interact with the application thorough graphical features. They can control applications by choosing menu commands. They can provide input using the mouse, keyboard, and other devices. They receive information from resources such as bitmaps, carets, cursors, and icons. The Window Manager exports two protected object types: Window Station objects and Desktop Objects. Each is an object with a Discretionary Access Control List (DACL) that is used to control access to it.

2.2.1.1.19 Windows Installer Service

The Windows Installer Service enables customers to better address corporate deployment and provide a standard format for component management. The installer supports advertisement of applications and features according to the operating system settings. It can install multiple updates with a single transaction that integrates installation progress, rollback, and reboots. It can apply patches in a constant order regardless of the order that the patches are provided to the system. Patches installed with the Windows Installer Service can be uninstalled in any order to leave the state of the product the same as if the patch was never installed. Patching using Windows Installer Service only updates files affected by the patch and can be significantly faster than earlier installer versions. Accounts with administrator privileges can use Windows Installer Service functions to query and inventory product, feature, component and patch information and to read, edit and replace installer source lists for network, URL and media sources.

2.2.1.1.20 "Winsock2" Installable File System Layer Driver

The "Winsock2" Installable File System (IFS) Layer Driver is a transport layer driver that emulates file handles for Windows Socket service providers for which a socket handle is not an IFS handle. As a result, Windows Sockets architecture accommodates service providers whose socket handles are not IFS objects. Applications can use Win32 file I/O calls with the handle without any knowledge about the network aspects.

2.2.1.2 Related Protection Profiles

This section summarizes essential security capabilities which were examined in evaluations of other protection profiles.

2.2.1.2.1 BitLocker Drive Encryption

BitLocker encrypts fixed and removable disk volumes (BitLocker to Go), including volumes that contain the operating system and user data. Access to the encrypted volume is protected by one or more protectors (authorization factors) that may include a Trusted Platform Module (TPM) which are

specified by the administrator for the computer. The BitLocker security target is at http://www.commoncriteriaportal.org/files/epfiles/st_vid10540-st.pdf.

2.2.1.2.2 Remote Access

Windows provides an integrated remote access solution that is easier to deploy and manage when compared to earlier versions that relied on multiple tools and consoles. Employees can access corporate network resources while they work remotely, and IT administrators can manage corporate computers in Active Directory that are located outside the internal network. The Windows IPsec VPN Client security target is at http://www.commoncriteriaportal.org/files/epfiles/st_vid10529-st.pdf.

2.2.1.2.3 Code Integrity Verification

Kernel-mode code signing (KMCS) prevents kernel-mode device drivers from loading unless they are published and digitally signed by developers who have been vetted by one of a limted approved and trusted certificate authorities (CAs).

Code Integrity was evaluated as part of the Windows 8 FIPS 140 validations in addition to the IPsec VPN Client and Software Disk Encryption evaluations.

2.2.1.2.4 Windows File Protection

The Windows File Protection technology prevents core system files from being overwritten by application installs. In the event a file is overwritten, Windows File Protection will replace that file with the correct version. Windows 8 identify device drivers that have passed the Windows Hardware Quality Labs test and warns users if they are about to install an uncertified driver. Windows RT does not allow uncertified drivers to be installed.

2.2.2 Security Functions Which Support the OS PP

This section describes security capabilities that were exercised during the OS PP evaluation which are not specifically associated with a OS PP functional requirement.

2.2.2.1 Client Side Caching of Off-line Files for SMB/Common Internet File System (CIFS)

When Windows caches a file to local storage from the network and the file server is available, the client within the SMB/CIFS Redirector checks with the file server to verify that the cached version of the file is up-to-date. If the file is up-to-date, then the client uses the cached copy of the file. Note that this check involves not only the content of the file, but also all of the file's attributes (e.g., its security descriptor). If the file server is not available, the client with the SMB/CIFS Redirector also has the cached copy to use.

2.2.2.2 Internet Connection Sharing

Internet Connection Sharing (ICS) is intended for use in a scenario where the ICS host computer directs network communication between two networks where one network is typically a more private LAN while the other is typically a wide area network. The ICS host computer needs two network connections. The LAN connection, automatically created by installing a network adapter, connects to the computers on the LAN. The other connection connects the LAN to the Wide Area Network (WAN). As a result, the shared connection connects computers on the LAN to the WAN.

2.2.3 General OS Infrastructure

This section describes general OS capabilities to provide context for the remainder of security target.

2.2.3.1 Background Intelligent Transfer Service (BITS)

Windows programmatically exposes a feature via the Component Object Model (COM) which will transfer data in a prioritized, throttled, and asynchronous manner between connected systems using idle network bandwidth. The Background Intelligent Transfer Service (BITS) protocol downloads content via HTTP and relies on HTTPS for data integrity. Windows uses BITS to download security updates for Windows from an update server.

2.2.3.2 COM Plus Component Service

The COM Plus Component Service extends the Component Object Model (COM) runtime environment with threading and security, object pooling, queued components, and application administration and packaging.

2.2.3.3 Credential Manager

The Credential Manager provides a secure store for usernames/passwords and also stores links to certificates and keys. This enables a consistent single sign-on experience for users, including roaming users. The combination of Credential Manager and Single sign-on makes it possible for users to access resources over the network without having to repeatedly supply their credentials.

2.2.3.4 Hardware Data Execution Prevention

64-bit hardware support adds a set of Data Execution Prevention (DEP) security checks to the TOE. These checks, known as hardware-enforced DEP, are designed to block malicious code that takes advantage of exception-handling mechanisms by intercepting attempts to execute code in memory that is marked for data only. This hardware protection feature is present in all 64-bit hardware architectures in the evaluated configuration.

While not available for 32-bit hardware architectures, due to hardware limitations, the only limitation is that application programs are not afforded additional protection from potential programming errors that might be exploitable by malicious users.[9]

2.2.3.5 Plug and Play

Plug and Play technology combines hardware and software support in such a way that Windows can recognize and adapt to hardware configuration changes automatically, without user intervention and or restarting the computer.

2.2.3.6 Windows Management Instrumentation

Windows Management Instrumentation (WMI) is a uniform model through which management data from any source can be managed in a standard way. WMI provides this for software, such as applications, while WMI extensions for the Windows Driver Model (WDM) provide this for hardware or hardware device drivers.

[9] Data Execution Prevention was tested in other Windows evaluations but not in the OS PP evaluation.

2.3 TOE Boundary and Security Environment

The TOE includes both physical and logical boundaries. Its operational environment is that of a networked environment.

2.3.1 Logical Boundaries

The diagram below depicts components and subcomponents of Windows. The components/subcomponents are large portions of the Windows operating system, and generally fall along process boundaries and a few major subdivisions of the kernel mode software.

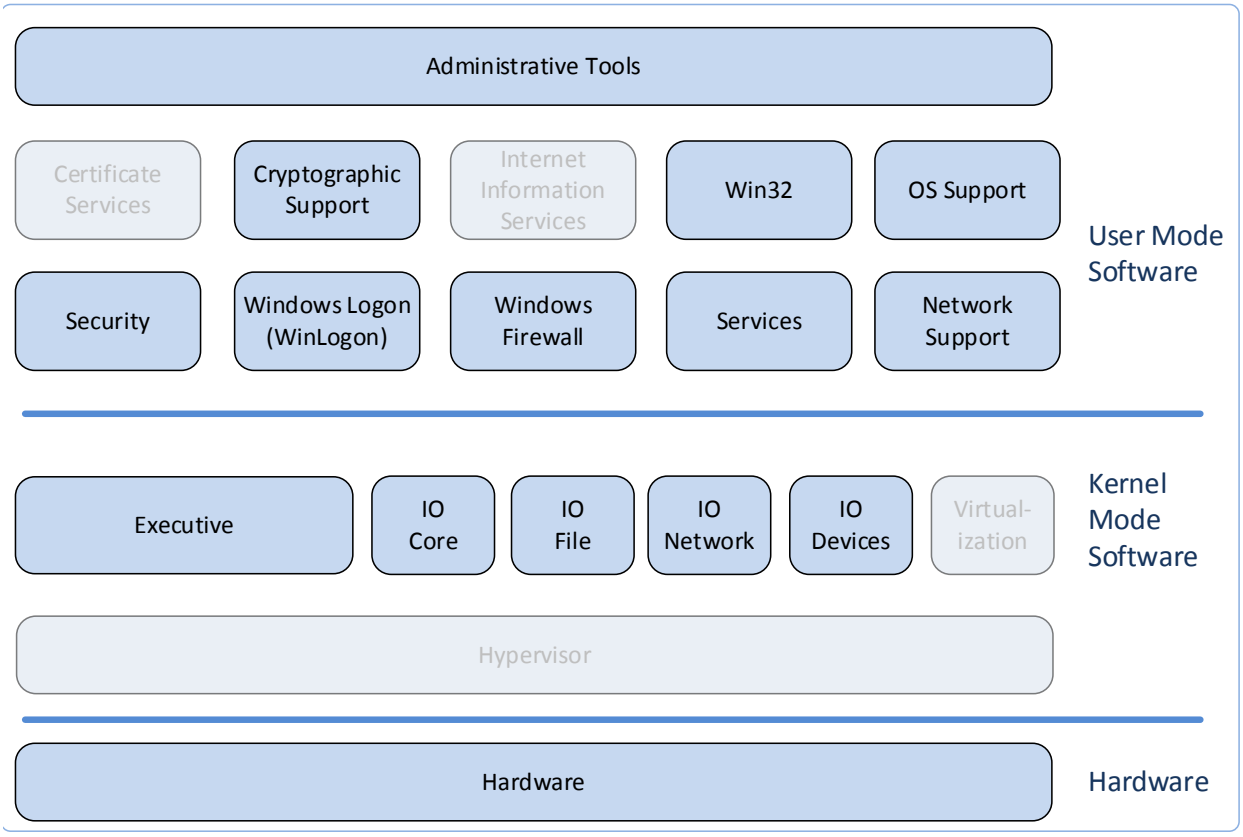


Figure 1 High-level Windows Architecture for Windows

The Windows RT and Windows 8 [consumer edition] architecture is the same as other Windows editions except that it does not include the Hypervisor, Virtualization, Certificate Services and Internet Information Services components.

- Administrative Tools Module
  - Administrative Tools Component: This component represents the range of tools available to manage the security properties of the TSF.
- Windows Firewall Module

  - Windows Firewall Component: This component provides services related to network information flow control.
- Hardware Module
  - Hardware Component: This component includes all hardware used by the TSF to include the processor(s), motherboard and associated chip sets, controllers, and I/O devices.
- Kernel Software Module
  - Executive Component: This is the kernel-mode software that provides core OS services including memory management, process management, and inter-process communication. This component implements all the non-I/O TSF interfaces for the kernel-mode.
  - I/O System: This is the kernel-mode software that implements all I/O related services, as well as all driver-related services. The I/O System is further divided into:
    - I/O Core Component
    - I/O File Component
    - I/O Network Component
    - I/O Devices Component
- [Miscellaneous] OS Support Module
  - OS Support Component: This component is a set of processes that provide various other OS support functions and services.
- Network Support Module
  - Network Support Component: This component contains various support services for RPC, COM, and other network services.
- Security Module
  - Security Component: This component includes all security management services and functions.
- Services Module
  - Services Component: This is the component that provides many system services as well as the service controller that manages win32 services.
- Win32 Module
  - Win32 Component: This component provides various support services for Win32 applications and the command console application.
- WinLogon Module
  - WinLogon Component: This component provides various interactive logon services to include interactive authentication, trusted path, session management and locking.
- Cryptographic Support Module
  - Cryptographic Support Component: This component provides cryptographic services for use by the kernel and other components in a manner that keeps them distinct from other components of the TOE.

2.3.2 Physical Boundaries

Physically, each TOE tablet, workstation, or server consists of an ARMv7 Thumb-2, x86, x64, architecture. The TOE executes on processors from Intel (x86 and x64), AMD (x86 and x64) Qualcomm

(ARM), or NVIDIA (ARM). Refer to section 1.1 for the specific list of hardware included in the evaluation.

A set of devices may be attached as part of the TOE:

- Display Monitors
- Fixed Disk Drives (including disk drives and solid state drives)
- Removable Disk Drives (including USB storage)
- Network Adaptor
- Keyboard
- Mouse
- Printer
- Audio Adaptor
- CD-ROM Drive
- Trusted Platform Module (TPM) version 1.2 or 2.0

The TOE does not include any network infrastructure components between the computers that comprise the distributed TOE. The security target assumes that any network connections, equipment, and cables are appropriately protected in the TOE security environment.

2.4 TOE Security Services

This section summarizes the security services provided by the TOE:

- **Security Audit:** Windows has the ability to collect audit data, review audit logs, protect audit logs from overflow, and restrict access to audit logs. Audit information generated by the system includes the date and time of the event, the user identity that caused the event to be generated, and other event specific data. Authorized administrators can review audit logs and have the ability to search and sort audit records. Authorized Administrators can also configure the audit system to include or exclude potentially auditable events to be audited based on a wide range of characteristics.
- **Identification and Authentication (I&A):** Each Windows user must be identified and authenticated based on administrator-defined policy (using password, network authentication token) prior to performing any TSF-mediated functions. An interactive user invokes a trusted path in order to protect his I&A information. Windows maintains databases of accounts including their identities, authentication information, group associations, and privilege and logon rights associations. Windows account policy functions include the ability to define the minimum password length, the number of failed logon attempts, the duration of lockout, and password age.
- **Security Management**: Windows includes several functions to manage security policies. Policy management is controlled through a combination of access control, membership in administrator groups, and privileges.
- **User Data Protection**: Windows protects user data by enforcing several access control policies (Discretionary Access Control, Mandatory Integrity Control) and several information flow

policies (IPsec filter information flow control, Windows Firewall), as well as object and subject residual information protection. Windows uses access control methods to allow or deny access to named objects, such as files, directory entries, and printers. Windows uses information flow control methods to control the flow of network traffic. Windows authorizes access to these resource objects through the use of security descriptors (an information set that identifies users and their specific access to resource objects), network filters, and port mapping rules. Windows also protects user data by ensuring that resources exported to user-mode processes do not have any residual information.

- **Cryptographic Protection:** Windows provides FIPS-140-2 validated cryptographic functions that support encryption/decryption, cryptographic signatures, cryptographic hashing, cryptographic key agreement, and random number generation. The TOE additionally provides support for public keys, credential management and certificate validation functions and provides support for the National Security Agency's Suite B cryptographic algorithms. Windows also provides extensive auditing support of cryptographic operations, the ability to replace cryptographic functions and random number generators with alternative implementations,[10] and a key isolation service designed to limit the potential exposure of secret and private keys. In addition to supporting its own security functions with cryptographic support, the TOE offers access to the cryptographic support functions for user application programs. Public key certificates generated and used by the TOE authenticate users and machines as well as user protect and system data in transit.
- **Protection of TOE Security Functions**: Windows provides a number of features to ensure the protection of TOE security functions. Windows protects against unauthorized data disclosure and modification by using a suite of Internet standard protocols including IPsec, IKE, and ISAKMP. Windows ensures process isolation security for all processes through private virtual address spaces, execution context, and security context. The Windows data structures defining process address space, execution context, memory protection, and security context are stored in protected kernel-mode memory. The Windows BitLocker features can be used to protect both fixed storage and removable USB storage volumes. Windows also includes some self-testing features that ensure the integrity executable TSF image and its cryptographic functions.
- **Session Locking**: Windows provides the ability for a user to lock their session either immediately or after a defined interval. Windows constantly monitors the mouse and keyboard for activity and locks the workstation after a set period of inactivity. Windows allows an authorized administrator to configure the system to display a logon banner before the logon dialogue.
- **Trusted Path**: Windows provides a trusted path for interactive session login as well as an IPsec trusted path when sending TSF data between machines that comprise a Windows deployment.

[10] This option is not included in the Windows Common Criteria evaluation.

3 Security Problem Definition

The security problem definition consists of the threats to security, organizational security policies, and usage assumptions as they relate to Windows. The assumptions, threats, and policies are primarily derived from the ***General Purpose Operating System Protection Profile*** and this security target.

3.1 Threats to Security

3.1.1 Threats to Security Covered by the OS PP

Table 3-1 presents known or presumed threats to protected resources that are addressed by Windows based on conformance to the ***Operating System Protection Profile***.

Table 3-1 OSPP Threats Addressed by Windows

Threat	Description
T.ACCESS.COMM	A threat agent may access cryptographically protected data transferred via a trusted channel between the TOE and another remote trusted IT system, modify such data during transfer in a way not detectable by the receiving party or masquerade as a remote trusted IT system.
T.ACCESS.TSFDATA	A threat agent may read or modify TSF data using functions of the TOE without the necessary authorization.
T.ACCESS.TSFFUNC	A threat agent may use or manage functionality of the TSF bypassing protection mechanisms of the TSF.
T.ACCESS.USERDATA	A threat agent may gain access to user data stored, processed or transmitted by the TOE without being appropriately authorized according to the TOE security policy by using functions provided by the TOE.
T.IA.MASQUERADE	A threat agent may masquerade as an authorized entity including the TOE itself or a part of the TOE in order to gain unauthorized access to user data, TSF data, or TOE resources.
T.IA.USER	A threat agent may gain access to user data, TSF data or TOE resources with the exception of public objects without being identified and authenticated by the TSF.
T.RESTRICT.NETTRAFFIC	A threat agent may send data packets to the recipient in the TOE via a network communication channel in violation of the information flow control policy.
T.UNATTENDED_SESSION	A threat agent may gain unauthorized access to an unattended session.

3.1.2 Additional Threats to Security

Table 3-2 presents known or presumed threats to protected resources that are addressed by Windows which are based on capabilities that surpass what is required to conform to the ***Operating System Protection Profile***.

Table 3-2 Additional Threats Addressed by Windows

Threat	Description	Source
T.CRYPTO_COMPROMISE	A malicious user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms.	Windows 8 security target[11]

3.2 Organizational Security Policies

3.2.1 Organizational Security Policies from the OSPP

An organizational security policy is a set of rules or procedures imposed by an organization upon its operations to protect its sensitive data and IT assets. **Table 3-3** describes organizational security policies that are addressed by Windows which are necessary for conformance to the OSPP.

Table 3-3 OSPP Organizational Security Policies

Security Policy	Description
P.ACCOUNTABILITY	The users of the TOE shall be held accountable for their security relevant actions within the TOE.
P.ROLES	Administrative authority to TSF functionality shall be given to trusted personnel and be as restricted as possible supporting only the administrative duties the person has.
P.USER	Authority shall only be given to users who are trusted to perform the actions correctly.

3.2.2 Additional Organizational Security Policies

Table 3-4 describes additional organizational security policies that are addressed by Windows which support the products' additional capabilities beyond the OSPP requirements.

Table 3-4 Additional Organizational Security Policies

Security Policy	Description	Source
P.CRYPTOGRAPHY	The TOE shall use standards-based cryptography as a baseline for key management (i.e., generation and destruction) and for cryptographic operations (i.e., encryption, decryption, signature, hashing, and random number generation).	Windows 8 security target[12]

[11] The original source was the U.S. Government Protection Profile for General-Purpose Operating Systems in a Networked Environment, Version 1.0.
[12] *Ibid.*

3.3 Secure Usage Assumptions

3.3.1 OSPP Assumptions

Table 3-5 describes the core security aspects of the environment in which Windows is intended to be used. It includes information about the physical, personnel, procedural, and connectivity aspects of the environment.

The following specific conditions are assumed to exist in an environment where the TOE is employed in order to conform to the OSPP:

Table 3-5 Secure Usage Assumptions

Assumption	Description
A.AUTHUSER	Authorized users possess the necessary authorization to access at least some of the information managed by the TOE and are expected to act in a cooperating manner in a benign environment.
A.TRAINEDUSER	Users are sufficiently trained and trusted to accomplish some task or group of tasks within a secure IT environment by exercising complete control over their user data.
A.CONNECT	All connections to and from remote trusted IT systems and between physically-separate parts of the TSF not protected by the TSF itself are physically or logically protected within the TOE environment to ensure the integrity and confidentiality of the data transmitted and to ensure the authenticity of the communication end points.
A.DETECT	Any modification or corruption of security-enforcing or security relevant files of the TOE, user or the underlying platform caused either intentionally or accidentally will be detected by an administrative user.
A.MANAGE	The TOE security functionality is managed by one or more competent individuals. The system administrative personnel are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the guidance documentation.
A.PEER.MGT	All remote trusted IT systems trusted by the TSF to provide TSF data or services to the TOE, or to support the TSF in the enforcement of security policy decisions are assumed to be under the same management control and operate under security policy constraints compatible with those of the TOE.
A.PEER.FUNC	All remote trusted IT systems trusted by the TSF to provide TSF data or services to the TOE, or to support the TSF in the enforcement of security policy decisions are assumed to correctly implement the functionality used by the TSF consistent with the assumptions defined for this functionality.
A.PHYSICAL	It is assumed that the IT environment provides the TOE with appropriate physical security, commensurate with the value of the IT assets protected by the TOE.

3.3.2 Assumptions Related to Additional Security Objectives

Table 3-6 describes additional security aspects of the environment in which Windows is intended to be used. It includes information about the physical, personnel, procedural, and connectivity aspects of the environment.

The following specific conditions are also assumed to exist in an environment where the TOE is employed in order to satisfy the additional security objectives:

Table 3-6 Secure Usage Assumptions

Assumption	Description	Source
None	There are no additional assumptions.	

4 Security Objectives

This section defines the security objectives of Windows and its supporting environment. Security objectives, categorized as either TOE security objectives or objectives by the supporting environment, reflect the stated intent to counter identified threats, comply with any organizational security policies identified, or address identified assumptions. All of the identified threats, organizational policies, and assumptions are addressed under one of the categories below.

4.1 TOE Security Objectives

4.1.1 OSPP Security Objectives

Table 4-1 describes the security objectives for Windows which are needed to comply with the OSPP.

Table 4-1 OSPP Security Objectives for the TOE

Security Objective	Source
O.AUDITING	The TSF must be able to record defined security-relevant events (which usually include security-critical actions of users of the TOE). The TSF must protect this information and present it to authorized users if the audit trail is stored on the local system. The information recorded for security-relevant events must contain the time and date the event happened and, if possible, the identification of the user that caused the event, and must be in sufficient detail to help the authorized user detect attempted security violations or potential misconfiguration of the TOE security features that would leave the IT assets open to compromise.
O.DISCRETIONARY.ACCESS	The TSF must control access of subjects and/or users to named resources based on identity of the object. The TSF must allow authorized users to specify for each access mode which users/subjects are allowed to access a specific named object in that access mode.
O.NETWORK.FLOW	The TOE shall mediate network communication between an entity outside of the TOE and a recipient within the TOE in accordance with its network information flow security policy.
O.SUBJECT.COM	The TOE shall mediate any possible sharing of objects or resources between subjects acting with different subject security attributes in accordance with its discretionary access control policy.
O.I&A	The TOE must ensure that users have been successfully authenticated before allowing any action the TOE has defined to be provided to authenticated users only.
O.MANAGE	The TSF must provide all the functions and facilities necessary to support the authorized users that are responsible for the management of TOE security mechanisms, must allow restricting such management actions to dedicated users, and must ensure that only such authorized users are able to access management functionality.
O.TRUSTED_CHANNEL	The TSF must allow authorized users to remotely access the TOE

	using a cryptographically-protected network protocol that ensures integrity and confidentiality of the transported data and is able to authenticate the end points of the communication. Note that the same protocols may also be used in the case where the TSF is physically separated into multiple parts that must communicate securely with each other over untrusted network connections. The protocol must also prevent masquerading of the remote trusted IT system.
O.UNATTENDED_SESSION	The TOE must allow for the temporary suspension of a user's session allowing the continuation of such a suspended session and user related input and output only after the user has resumed the session by re-authenticating himself to the TSF.

4.1.2 Additional Security Objectives

Table 4-2 describes the additional security objectives for Windows which surpass the objectives defined by the OSPP.

Table 4-2 Additional Security Objectives for the TOE

Security Objective	Description	Source
O.CRYPTOGRAPHIC_SERVICES	The TOE will make encryption services available to authorized users and/or user applications.	Windows 8 security target[13]

4.2 Security Objectives for the Operational Environment

4.2.1 OSPP Security Objectives for the Operational Environment

The TOE is assumed to be complete and self-contained and, as such, is not dependent upon any other products to perform properly. However, certain objectives with respect to the general operating environment must be met. **Table 4-3** describes the security objectives for the operational environment as specified in the OSPP.

Table 4-3 OSPP Security Objectives for the Operational Environment

Environment Objective	Description
OE.ADMIN	Those responsible for the TOE are competent and trustworthy individuals, capable of managing the TOE and the security of the information it contains.
OE.REMOTE	If the TOE relies on remote trusted IT systems to support the enforcement of its policy, those systems provide the functions

[13] The original source was the U.S. Government Protection Profile for General-Purpose Operating Systems in a Networked Environment, Version 1.0.

	required by the TOE and are sufficiently protected from any attack that may cause those functions to provide false results.
OE.INFO_PROTECT	Those responsible for the TOE must establish and implement procedures to ensure that information is protected in an appropriate manner. In particular: • All network and peripheral cabling must be approved for the transmittal of the most sensitive data held by the system. Such physical links are assumed to be adequately protected against threats to the confidentiality and integrity of the data transmitted. • DAC protections on security-relevant files (such as audit trails and authentication databases) shall always be set up correctly. • Users are authorized to access parts of the data managed by the TOE and are trained to exercise control over their own data.
OE.INSTALL	Those responsible for the TOE must establish and implement procedures to ensure that the hardware, software and firmware components that comprise the system are distributed, installed and configured in a secure manner supporting the security mechanisms provided by the TOE.
OE.MAINTENANCE	Authorized users of the TOE must ensure that the comprehensive diagnostics facilities provided by the product are invoked at every scheduled preventative maintenance period.
OE.PHYSICAL	Those responsible for the TOE must ensure that those parts of the TOE critical to enforcement of the security policy are protected from physical attack that might compromise IT security objectives. The protection must be commensurate with the value of the IT assets protected by the TOE.
OE.RECOVER	Those responsible for the TOE must ensure that procedures and/or mechanisms are provided to assure that after system failure or other discontinuity, recovery without a protection (security) compromise is achieved.
OE.TRUSTED.IT.SYSTEM	The remote trusted IT systems implement the protocols and mechanisms required by the TSF to support the enforcement of the security policy. These remote trusted IT systems are under the same management domain as the TOE, are managed based on the same rules and policies applicable to the TOE, and are physically and logically protected equivalent to the TOE.

4.2.2 Additional Security Environment Objectives for Additional Security Functions

Table 4-4 describes additional security objectives for the operational environment for capabilities which exceed the OSPP.

Table 4-4 Additional Security Objectives for the Operational Environment

Security Objective	Description	Source

None	There are no additional security objectives for the operational environment.

5 Security Requirements

The section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) for the TOE. The requirements in this section have been drawn from the ***General Purpose Operating System Protection Profile***, Version3.9, December 2012, the Common Criteria, or are defined in the following section.

Conventions:

Where requirements are drawn from the OSPP, the requirements are copied verbatim, except for some changes to required identifiers to match the iteration convention of this document, from that protection profile and only operations performed in this security target are identified.

Where general requirements are drawn from the Common Criteria, that is, not from the OSPP, the requirements are copied verbatim, except for some changes to required identifiers to match the iteration convention of this document, and the operations performed in this security target are identified.

Requirements defined within this security target do not have identified operations.

Where applicable the following conventions are used to identify operations:

- **Iteration**: Iterated requirements (components and elements) are identified with letter following the base component identifier. For example, iterations of FMT_MOF.1 are identified in a manner similar to FMT_MOF.1(Audit) (for the component) and FCS_COP.1(Audit).1 (for the elements).
- **Assignment**: Assignments are identified in brackets and bold (e.g., **[assigned value]**).
- **Selection**: Selections are identified in brackets, bold, and italics (e.g., ***[selected value]***).
    - Assignments within selections are identified using the previous conventions, except that the assigned value would also be italicized and extra brackets would occur (e.g., ***[selected value [assigned value]]***).
- **Refinement**: Refinements are identified using bold text (e.g., **added text**) for additions and strike-through text (e.g., ~~deleted text~~) for deletions.

5.1 Extended Components Definitions

5.1.1 OSPP Extended Components

Extended components which are used in this security target are listed in **Table 5-1**; these are defined in the OSPP:

Table 5-1 Extended Functional Components Defined in the OSPP

Short Name	Unique Name
FIA_PK_EXT.1	Public Key Based Authentication
FMT_SMF_RMT.1	Remote Management Capabilities

5.1.2 Additional Extended Components

The additional extended components which are used in this security target are listed in **Table 5-2** are defined in this security target.

Table 5-2 Additional Extended Functional Components

Short Name	Unique Name	Source
FCS_CKM_EXT.4	Cryptographic Key Zeroization	Windows 8 Security Target
FCS_SRV_EXT.1	Cryptographic Services	Windows 8 Security Target
FCS_RBG_EXT.1	Random Number Generation	Windows 8 Security Target

5.1.2.1 Extended: Cryptographic Key Zeroization (FCS_CKM_EXT.4)

Hierarchical to: No other components.

Dependencies: None.

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

Rationale: This component is necessary to specify a unique requirement for exporting cryptographic services to evaluations that is not addressed by the CC.

5.1.2.2 Extended: Cryptographic Services (FCS_SRV_EXT.1)

Hierarchical to: No other components.

Dependencies: None.

FCS_SRV_EXT.1.1 The TSF shall provide a mechanism for applications to request the TSF to perform the following cryptographic operations:

a) Symmetric Key Generation in FCS_CKM.1(SYM),
b) Asymmetric Key Generation in FCS_CKM.1(AUTH),
c) Encryption/Decryption in FCS_COP.1(AES),
d) Cryptographic Signature (Digital Signature) in FCS_COP.1(SIGN),
e) Hashing in FCS_COP.1(HASH),
f) Keyed Hashing in FCS_COP.1(HMAC) and
g) Random Number Generation in FCS_RBG_EXT.1.

Rationale: This component is necessary to specify a unique requirement for exporting cryptographic services to evaluations that is not addressed by the CC.

5.1.2.3 Extended: Random Number Generation (FCS_RBG_EXT.1)

Hierarchical to: No other components.

Dependencies: Extended: None.

FCS_RBG_EXT.1.1 **The TSF shall perform all random bit generation (RBG) services in**

	accordance with [selection: choose one of: NIST Special Publication 800-90, FIPS Pub 140-2 Annex C] implemented in a FIPS-validated cryptomodule operating in FIPS mode seeded by an entropy source that accumulates entropy from [selection: choose one of: one or more independent hardware-based noise sources, one or more independent software-based noise sources, a combination of hardware-based and software-based noise sources.]
FCS_RBG_EXT.1.2	**The deterministic RBG shall be seeded with a minimum of [selection, choose one of: 128 bits, 192 bits, 256 bits] of entropy at least equal to the greatest bit length of the keys that it will generate.**

Rationale: This component is necessary to specify a unique requirement for random number generation that is not addressed by the CC.

5.2 TOE Security Functional Requirements

This section specifies the SFRs for the TOE which are based on the OS PP and the **Additional Extended Components** mentioned above.

Table 5-3 TOE Security Functional Requirements

Requirement Class	Requirement Component
FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
	User Identity Association (FAU_GEN.2)
	Audit Review (FAU_SAR.1)
	Restricted Audit Review (FAU_SAR.2)
	Selective Audit (FAU_SEL.1)
	Protected Audit Trail Storage (FAU_STG.1)
	Action in Case of Possible Audit Data Loss (FAU_STG.3)
	Prevention of Audit Data Loss (FAU_STG.4(SL))
	Prevention of Audit Data Loss (FAU_STG.4(OL))
FCS: Cryptographic Support	Cryptographic Key Generation for Symmetric Keys (FCS_CKM.1(SYM))
	Cryptographic Key Generation for Asymmetric Keys Used for Key Establishment (FCS_CKM.1(ASYM))
	Cryptographic Key Generation for Asymmetric Keys Used for Authentication (FCS_CKM.1(AUTH))
	Cryptographic Key Zeroization (FCS_CKM_EXT.4)
	Cryptographic Services (FCS_SRV_EXT.1)
	Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))
	Cryptographic Operation for Cryptographic Signature (FCS_COP.1(SIGN))
	Cryptographic Operation for Cryptographic Hashing (FCS_COP.1(HASH))
	Cryptographic Operation for Keyed-Hash Message Authentication (FCS_COP.1(HMAC))
	Cryptographic Operation for DH Key Agreement (FCS_COP.1(DH KA))
	Cryptographic Operation for ECDH Key Agreement (FCS_COP.1(EC KA))
	Random Number Generation (FCS_RBG_EXT.1)

FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
	Subset Information Flow Control (FDP_IFC.1(OSPP))
	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
	Full Residual Information Protection (FDP_RIP.2)
FIA: Identification & Authentication	Authentication Failure Handling (FIA_AFL.1)
	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
	Timing of Authentication for OS Logon (FIA_UAU.1(RITE))
	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
	Multiple Authentication Mechanisms (FIA_UAU.5)
	Protected Authentication Feedback (FIA_UAU.7)
	Timing of Identification (FIA_UID.1)
	User-Subject Binding for Individual Users (FIA_USB.1(USR))
	Public Key Based Authentication (FIA_PK_EXT.1)
FMT: Security Management	Management of Security Functions Behavior for Password Management (FMT_MOF.1(Pass))
	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
	Management of Security Attributes for Object Ownership (FMT_MSA.1(OBJ))
	Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))
	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
	Static Attribute Initialization for Mandatory Integrity Control Policies (FMT_MSA.3(MIC))
	Static Attribute Initialization for Network Information Flow Control (FMT_MSA.3(OSPP))
	Static Attribute Value Inheritance (FMT_MSA.4)
	Management of TSF Data for Audit Selection (FMT_MTD.1(Audit Sel))
	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
	Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg)
	Management of TSF Data for Audit Log Failure (FMT_MTD.1(Audit Fail))
	Management of TSF Data for X.509 Certificates (FMT_MTD.1(X509))
	Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))
	Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Threshold))
	Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Re-enable))
	Management of TSF Data for Initialization of User Security Attributes (FMT_MTD.1(Init-Attr))

	Management of TSF Data for Modification of User Security Attributes Other Than Authentication Data (FMT_MTD.1(Mod-Attr))
	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Auth))
	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
	Revocation for Object Access (FMT_REV.1(OBJ))
	Revocation for Object Access for DAC (FMT_REV.1(DAC))
	Revocation for Authorized Administrators (FMT_REV.1(Admin))
	Remote Management Capabilities (FMT_SMF_RMT.1)
	Security Roles (FMT_SMR.1)
FPT: Protection of the TSF	Basic Internal TSF Data Transfer Protection (FPT_ITT.1)
	Reliable Time Stamps (FPT_STM.1)
FTA: TOE Access	TSF-initiated Session Locking (FTA_SSL.1)
	User-initiated Locking (FTA_SSL.2)
FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))

5.2.1 Security Audit (FAU)

5.2.1.1 Audit Data Generation (FAU_GEN.1(OSPP))

FAU_GEN.1(OSPP).1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions,
b) All auditable events for the not specified level of audit
c) All modifications to the set of events being audited,
d) All user authentication attempts,
e) All denied access to objects for which the access control policy defined in ~~the OSPP base~~ **FDP_ACF.1(DAC) and FDP_ACF.1(MIC)** applies,
f) Explicit modifications of access rights to objects covered by the access control policies,
g) Start-up and shutdown of the TOE,
h) Uses of special permissions that circumvent the access control policies,
i) All auditable events listed in ~~FAU_GEN.1.2~~ **Table 5-4**.

FAU_GEN.1(OSPP).2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b) For all management SFRs included in the Security Target:
    - The identity of the user that performed/attempted to perform the management operation

- An identification of what was managed and the indication what the administrative user has changed as part of the management operation.

c) For each audit event type, based on the auditable event definitions of the functional components included in the following table.

Table 5-4 Audit Events and Information

Security Functional Requirement	Audit Events Prompted by	Additional Information in the Audit Record
Audit Data Generation (FAU_GEN.1)	None	None
User Identity Association (FAU_GEN.2)	None	None
Audit Review (FAU_SAR.1)	Any attempt to access the audit records.	Identity of the user attempting to access the audit records Success or failure
Restricted Audit Review (FAU_SAR.2)	Unsuccessful attempts to read information from the audit records.	None
Selective Audit (FAU_SEL.1)	Any attempt to modify the events to be audited.	Identity of the user attempting to modify the events to be audited Success or failure If successful: the modification to the set of events to be audited.
Protected Audit Trail Storage (FAU_STG.1)	None	None
Action in Case of Possible Audit Data Loss (FAU_STG.3)	None	None
Prevention of Audit Data Loss (FAU_STG.4(SL))	Actions taken due to exceeding of a threshold.	Message sent to administrator
Prevention of Audit Data Loss (FAU_STG.4(OL))	None	None
Cryptographic Key Generation for Symmetric Keys (FCS_CKM.1(SYM))	None	None
Cryptographic Key Generation for Asymmetric Keys Used for Key Establishment (FCS_CKM.1(ASYM))	None	None
Cryptographic Key Generation for Asymmetric Keys Used for Authentication (FCS_CKM.1(AUTH))	None	None
Cryptographic Key Zeroization	None	None

(FCS_CKM_EXT.4)		
Cryptographic Services (FCS_SRV_EXT.1)	None	None
Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))	None	None
Cryptographic Operation For Cryptographic Signature (FCS_COP.1(SIGN))	None	None
Cryptographic Operation for Cryptographic Hashing (FCS_COP.1(HASH))	None	None
Cryptographic Operation for Keyed Hash Message Authentication (FCS_COP.1(HMAC))	None	None
Cryptographic Operation ECDH Key Agreement (FCS_COP.1(DH KA))	None	None
Cryptographic Operation for ECDSA Key Agreement (FCS_COP.1(EC KA))	None	None
Random Number Generation (FCS_RBG_EXT.1)	None	None
Discretionary Access (FDP_ACC.1(DAC))	None	None
Mandatory Integrity Control Policy (FDP_ACC.1(MIC))	None	None
Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))	Any attempt to access an object protected by the SFP. **Use of privilege to bypass the access control mechanism**.	Identity of the user attempting to access an object protected by the SFP. Identity of the object the user attempts to access. Attempted operation. Success or failure.
Mandatory Integrity Control Functions (FDP_ACF.1(MIC))	All requests to perform an operation on an object covered by the SFP.	None
Subset Information Flow Control (FDP_IFC.1(OSPP))	None	None
Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))	Denied information flow.	Identification of the network interface. Reason for denying the flow.
Full Residual Information Protection (FDP_RIP.2)	None	None
Authentication Failure Handling	The reaching of the threshold for	None

(FIA_AFL.1)	the unsuccessful authentication attempts. **The action taken (disable for non-administrators, delay for administrator).** **The re-enablement of disabled non-administrative accounts**.	
User Attribute Definition for Individual Users (FIA_ATD.1(USR))	None	None
Timing of Authentication for OS Logon (FIA_UAU.1.(RITE))	Verification that a user has been successfully authenticated. **All use of the authentication mechanism**.	User identity. Indicator that the user has been successfully authenticated. In the case the authentication is performed by the TOE, also the event of a failure authentication attempt need to be auditable: User identity provided Indicator that the authentication failed. **Origin of the attempt (e.g., terminal identifier, source IP address)**
Timing of Authentication for OS Logon (FIA_UAU.1.(OS))	Verification that a user has been successfully authenticated. **All use of the authentication mechanism**.	User identity. Indicator that the user has been successfully authenticated. In the case the authentication is performed by the TOE, also the event of a failure authentication attempt need to be auditable: User identity provided Indicator that the authentication failed.
Multiple Authentication Mechanisms (FIA_UAU.5)	None	None
Protected Authentication Feedback (FIA_UAU.7)	None	None
Timing of Identification (FIA_UID.1)	**All use of the user identification mechanism**	**Provided user identity, origin of the attempt (e.g., terminal identifier, source IP address)**
User-Subject Binding for Individual Users (FIA_USB.1(USR))	**Binding of user security attributes to a subject (e.g. creation of a subject).**	None
Public Key Based Authentication	None	None

(FIA_PK_EXT.1)		
Management of Security Functions Behavior for Password Management (FMT_MOF.1(Pass))	All modifications in the behavior of the functions in the TSF.	None.
Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))[14]	All modifications of the values of security attributes.	The name of the object, the old and new values of the attributes
Management of Security Attributes for Object Ownership (FMT_MSA.1(OBJ))	All modifications of the values of security attributes.	The name of the object, the old and new values of the attributes
Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))	None	None
Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))	Modifications of the default setting of permissive or restrictive rules. All modifications of the initial values of security attributes.	The old and new values of the attributes.
Static Attribute Initialization for Mandatory Integrity Control Policies (FMT_MSA.3(MIC))	None	None
Static Attribute Initialization for Network Information Flow Control (FMT_MSA.3(OSPP))	None	None
Static Attribute Value Inheritance (FMT_MSA.4)	None	None
Management of TSF Data for audit Selection (FMT_MTD.1(Audit Sel))[15]	None	None
Management of TSF Data for audit data (FMT_MTD.1(Audit))[16]	Actions taken with respect to the audit records.	The specific action that was performed.
Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg)	None	None
Management of TSF Data for Audit Log Failure) (FMT_MTD.1(Audit Fail))[17]	None	None
Management of TSF Data for X509 Certificates (FMT_MTD.1(X509))[18]	None	None
Management of TSF Data for	None	None

[14] This corresponds to FMT_MSA.1 in the OSPP.
[15] This corresponds to FMT_MTD.1(AT) in the OSPP.
[16] This corresponds to FMT_MTD.1(AS) in the OSPP.
[17] This corresponds to FMT_MTD.1(AF) in the OSPP.
[18] This corresponds to FMT_MTD.1(CM) in the OSPP.

Network Information Flow Control (FMT_MTD.1(OSPP))[19]		
Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Threshold))[20]	None	None
Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Re-enable))[21]	None	None
Management of TSF Data for initialization of user security attributes (FMT_MTD.1(Init-Attr))[22]	All initializations of the values of user security attributes.	The initial values for the user security attributes.
Management of TSF Data for modification of user security attributes, other than authentication data (FMT_MTD.1(Mod-Attr))[23]	All modifications of the values of user security attributes.	The new values of the attributes.
Management of TSF Data for modification of authentication data (FMT_MTD.1(Mod-Auth))[24]	All modifications of the values of user security attributes.	None
Revocation for Object Access (FMT_REV.1(Obj))[25]	All attempts to revoke security attributes.	The security attributes that are attempting to be revoked, the object with which the security attributes are associated.
Revocation for Object Access for DAC (FMT_REV.1(DAC))[26]	All attempts to revoke security attributes.	The security attributes that are attempting to be revoked, the object with which the security attributes are associated.
Revocation for Authorized Administrators (FMT_REV.1(Admin))[27]	All attempts to revoke security attributes.	The security attributes that are attempting to be revoked
Remote Management Capabilities (FMT_SMF_RMT.1)	None	None
Security Roles (FMT_SMR.1)	Modifications to the group of users that are part of a role.	The role the user is associated with or disassociated from.

[19] This corresponds to FMT_MTD.1(NI) in the OSPP.
[20] This corresponds to FMT_MTD.1(IAT) in the OSPP.
[21] This corresponds to FMT_MTD.1(IAF) in the OSPP.
[22] This corresponds to part of FMT_MTD.1(IAU) in the OSPP.
[23] This corresponds to part of FMT_MTD.1(IAU) in the OSPP.
[24] This corresponds to part of FMT_MTD.1(IAU) in the OSPP.
[25] This corresponds to part of FMT_REV.1(OBJ) in the OSPP.
[26] This corresponds to part of FMT_REV.1(OBJ) in the OSPP.
[27] This corresponds to FMT_REV.1(USR) in the OSPP.

Reliable Time Stamps (FPT_STM.1)	**Setting the time to a specific value.**	**The old and new values for the time.**
TSF-Initiated Session Locking (FTA_SSL.1)	**Locking of an interactive session by the session locking mechanism**. Any attempts at unlocking of an interactive session.	None
User-Initiated Locking (FTA_SSL.2)	**Locking of an interactive session by the session locking mechanism**. Any attempts at unlocking of an interactive session.	None
Inter-TSF Trusted Channel (FTP_ITC.1(OS))	Initialization of a trusted channel	Identity of the communication partner. Protocol used to establish the channel. Success or failure setting up the channel.

5.2.1.2 User Identity Association (FAU_GEN.2)

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.2.1.3 Audit Review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide **[authorized administrators]** with the capability to read **[all audit information]** from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for **the authorized administrator** ~~user~~ to interpret the information.

5.2.1.4 Restricted Audit Review (FAU_SAR.2)

FAU_SAR.2.1 The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

5.2.1.5 Selective Audit (FAU_SEL.1)

FAU_SEL.1.1 The TSF shall be able to ~~select the set of events to be audited~~ **include or exclude auditable events** from the set of audited events based on the following attributes:

a) Type of audit event
b) Subject or user identity,
c) Outcome (success or failure) of the audit event,
d) Named object identity,
e) **[host identity]**.

5.2.1.6 Protected Audit Trail Storage (FAU_STG.1)

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2 The TSF shall be able to ***[prevent]*** modifications to the **stored** audit records in the audit trail.

5.2.1.7 Action in Case of Possible Audit Data Loss (FAU_STG.3)

FAU_STG.3.1 The TSF shall **[notify an authorized administrator of the possible audit data loss in the security log, and overwrite the oldest stored audit events as needed in other administrative and other operational logs]** if the audit trail exceeds [**an authorized administrator selectable, pre-defined limit]** or if any of the following **[no other conditions]** is detected that may result in a loss of audit record.

5.2.1.8 Prevention of Audit Data Loss in Audit Log (FAU_STG.4(SL))

FAU_STG.4.(SL).1 The TSF shall ***[prevent audited events, except those taken by [the authorized user with special rights[28]]]*** and **[generate an alarm to the authorized administrator]** if the **security** audit trail is full.

5.2.1.9 Prevention of Audit Data Loss in Operational Log (FAU_STG.4(OL))

FAU_STG.4.(OL)1 The TSF shall ***["overwrite the oldest stored audit records"]*** and **[None]** ~~if the audit trail~~ **when an administrative or operational log** is full.

5.2.2 Cryptographic Support (FCS)

5.2.2.1 Cryptographic Key Generation for Symmetric Keys (FCS_CKM.1(SYM))

FCS_CKM.1(SYM).1 The TSF shall generate **symmetric** cryptographic keys using a Random Bit Generator as specified in FCS_RBG_EXT.1 **and** specified cryptographic key sizes [**128 bit, 256 bit**]**:** that meet the following **[No Standard]**.

5.2.2.2 Cryptographic Key Generation for Asymmetric Keys Used for Key Establishment (FCS_CKM.1(ASYM))

FCS_CKM.1(ASYM).1 The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with:

- NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for finite field-based key establishment schemes;
- NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for elliptic curve-based key establishment schemes and implementing "NIST curves" P- 256, P-384 and P-521 (as defined in FIPS PUB 186-4, "Digital Signature Standard")

[28] In this case the "authorized user with special rights" is the authorized administrator.

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.2.2.3 Cryptographic Key Generation for Asymmetric Keys Used for Authentication (FCS_CKM.1(AUTH))

FCS_CKM.1(AUTH).1 The TSF shall generate asymmetric cryptographic keys used for IKE peer authentication in accordance with a:

[

- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1 for FFC schemes;***
- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes;***
- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-256, P-384 and P-521;***

]

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.2.2.4 Cryptographic Key Zeroization (FCS_CKM_EXT.4)

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

5.2.2.5 Cryptographic Services (FCS_SRV_EXT.1)

FCS_SRV_EXT.1.1 The TSF shall provide a mechanism for applications to request the TSF to perform the following cryptographic operations:

a) Symmetric Key Generation in FCS_CKM.1(SYM),
b) Asymmetric Key Generation in FCS_CKM.1(AUTH),
c) Encryption/Decryption in FCS_COP.1(AES),
d) Cryptographic Signature (Digital Signature) in FCS_COP.1(SIGN),
e) Hashing in FCS_COP.1(HASH),
f) Keyed Hashing in FCS_COP.1(HMAC) and
g) Random Number Generation in FCS_RBG_EXT.1.

5.2.2.6 Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))

FCS_COP.1(AES).1 The TSF shall perform [encryption and decryption] in accordance with a specific cryptographic algorithm [AES operating in **[ECB, CBC, CFB8, CCM, and GCM modes]** and cryptographic key size of ***[128 bits, 192 bits, 256 bits]*** that meet the following:

- [FIPS PUB 197, "Advanced Encryption Standard (AES)
- NIST SP 800-38A, NIST SP 800-38C, NIST SP 800-38D].

5.2.2.7 Cryptographic Operation for Cryptographic Signature (FCS_COP.1(SIGN))

FCS_COP.1(SIGN).1 The TSF shall perform cryptographic signature services in accordance with a:

- ***[Digital Signature Algorithm (DSA) with a key size (modulus) of 2048 bits or greater that meets FIPS PUB 186-4,***
- ***RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater that meets FIPS PUB 186-4,***
- ***Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 256 bits, 384 bits, or 521 bits, that meets FIPS PUB 186-4, "Digital Signature Standard" with NIST curves P-256, P-384, P-521 as defined in FIPS PUB 186-4, 'Digital Signature Standard']***.

5.2.2.8 Cryptographic Operation for Cryptographic Hashing (FCS_COP.1(HASH))

FCS_COP.1(HASH).1 The TSF shall perform cryptographic hashing services in accordance with ***[SHA-1, SHA 256, SHA 384, SHA 512]*** and message digest sizes ***[160, 256, 384, and 512]*** bits that meet the following: FIPS 180-4, "Secure Hash Standard".

5.2.2.9 Cryptographic Operation for Keyed-Hash Message Authentication (FCS_COP.1(HMAC))

FCS_COP.1(HMAC).1 The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm HMAC- [***SHA-1, SHA-256, SHA-384, SHA-512***], key size [**160, 256, 384, 512**], and message digest size of [***160, 256, 384, 512***] bits that meet the following: FIPS PUB 198-1, "The Keyed-Hash Message Authentication Code", and FIPS PUB 180-4, "Secure Hash Standard".

5.2.2.10 Cryptographic Operation for DH Key Agreement (FCS_COP.1(DH KA))

FCS_COP.1(DH KA).1 The TSF shall perform **[key agreement]** in accordance with a specified cryptographic algorithm **[Diffie Hellman (DH) key agreement protocol]** and cryptographic key sizes [2048 and 4096 bits] that meet the following: **[NIST SP 800-56A]**.

5.2.2.11 Cryptographic Operation for ECDH Key Agreement (FCS_COP.1(EC KA))

FCS_COP.1(EC KA).1 The TSF shall perform **[key agreement]** in accordance with a specified cryptographic algorithm **[Elliptic Curve Diffie Hellman for key agreement with NIST P curves: P-256, P-384, and P-521]** and cryptographic key **sizes [256, 384, and 521, respectively]** that meet the following: **[NIST SP 800-56A]**.[29]

5.2.2.12 Random Number Generation (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1 The TSF shall perform all deterministic random bit generation (RBG) services in accordance with ***[NIST Special Publication 800-90A]*** using CTR_DBRG(AES) seeded by an entropy source that accumulates entropy from ***[a combination of hardware-based and software-based noise sources]***.

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of ***[256 bits]*** of entropy at least equal to the greatest bit length of the keys that it will generate.

[29] Note that these operations are performed within a FIPS 140-evaluated cryptographic module, See FIPS 140-2 CMVP certificates 1891 and 1892.

5.2.3 User Data Protection (FDP)

5.2.3.1 Discretionary Access Control (FDP_ACC.1(DAC))

FDP_ACC.1(DAC).1 The TSF shall enforce the **[Discretionary Access Control policy]** on

a) **[processes and threads running on behalf of a user]**;
b) **[desktop, event, event pair, I/O completion port, job, registry key, mutant, object directory, ALPC port, mailslot, named pipe, NTFS directory, NTFS file, printer, process, section, semaphore, symbolic link, thread, timer, security token, window station, debug, transaction enlistment, transaction, Resource Manager, and Transaction Manager objects]**;
c) **[and all operations among subjects and objects covered by the SFP including those operations identified by the following requirements:**
   i. **FMT_MSA.1(DAC),**
   ii. **FMT_MSA.3(DAC)]**.

5.2.3.2 Mandatory Integrity Control Policy (FDP_ACC.1(MIC))

FDP_ACC.1(MIC).1 The TSF shall enforce the **[Mandatory Integrity Control Policy] on**

a) **[subjects: processes acting on the behalf of users and]**;
b) **[objects: Event, Event Pair, I/O Completion Port, Job, Key, Mutant, Mailslot, Named Pipe, NTFS Directory, NTFS File, Object Directory, Process, Section, Semaphore, Symbolic Link, Thread, Timer, and Tokens]**;
c) **[and all operations among subjects and objects covered by the SFP including those operations identified by the following requirements:**
   i. FMT_MSA.1(MIC),
   ii. FMT_MSA.3(MIC)].

5.2.3.3 Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))

FDP_ACF.1(DAC).1 The TSF shall enforce the **[Discretionary Access Control policy]** to objects based on the following: **[**

a) the user identity, private keys, privileges and group membership(s) associated with subjects defined by FDP_ACC.1(DAC);
b) **the {user (or group) identity, access operations}[30] pairs and owner associated with objects defined by FDP_ACC.1(DAC);].**

FDP_ACF.1(DAC).2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[**

[30] The {set of user(or group) identity, access operation} pairs are referred to as a DACL (see section 6.2.2.1.2).

The Discretionary Access Control policy mechanism shall, either by explicit authorized user action or by default, provide that controlled objects are protected from unauthorized access during all operations according to the following ordered rules:

a) **If the requested mode of access is denied to that user, deny access.**
b) **If the requested mode of access is permitted to that user, permit access.**
c) **If the requested mode of access is denied to every group of which the user is a member, deny access.**
d) **If the requested mode of access is permitted to any group of which the user is a member, grant access.**
e) **Else deny access].**

FDP_ACF.1(DAC).3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: **[**

a) **Authorized administrators must follow the above-stated Discretionary Access Control policy, except after taking the following specific actions:**
    i. **Request to change the owner of an object,**
    ii. **Request to backup a file or registry key on the local system,**
    iii. **Request to restore a file or registry key onto the local system,**
b) **The enforcement mechanism (e.g., access control lists) shall allow users to specify and control sharing of objects by individual user identities and group identities if the user is the owner of the object or has the privilege to take ownership of the object, and**
c) **If an object has no access control list the object is not protected and any requested access is granted.**
].

FDP_ACF.1(DAC).4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **[**

a) **if an object has an assigned, but empty access control list no access is granted unless the subject is an authorized administrator receiving access by FDP_ACF.1(DAC).3 "a"]**.

5.2.3.4 Mandatory Integrity Control Functions (FDP_ACF.1(MIC))

FDP_ACF.1(MIC).1 The TSF shall enforce the **[Mandatory Integrity Control Policy]** to objects based on the following: **[**

a) **The integrity label and mandatory policy associated with subjects defined by FDP_ACC.1(MIC) and**
b) **The integrity label and mandatory policy associated with objects defined by FDP_ACC.1(MIC)]**.

FDP_ACF.1(MIC).2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed**: [**

a) Write access is allowed if the subject integrity label is greater than or equal to the object integrity label OR the object mandatory policy does not indicate "SYSTEM_MANDATORY_LABEL_NO_WRITE_UP".

b) Read access is allowed if the subject integrity label is greater than or equal to the object integrity label OR the object mandatory policy does not indicate "SYSTEM_MANDATORY_LABEL_NO_READ_UP".

c) **Execute access is allowed if the subject integrity label is greater than or equal to the object integrity label OR the object mandatory policy does not indicate "SYSTEM_MANDATORY_LABEL_NO_EXECUTE_UP"]**.

FDP_ACF.1(MIC).3 The TSF shall explicitly authorize access of subjects to objects based in the following additional rules**: [The mandatory policy associated with the subject does not indicate "TOKEN_MANDATORY_POLICY_NO_WRITE_UP"]**.

FDP_ACF.1(MIC).4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: **[no explicit denial rules]**.

5.2.3.5 Subset Information Flow Control (FDP_IFC.1(OSPP))

FDP_IFC.1(OSPP).1 The TSF shall enforce the Network Information Flow Control Policy on

a) Originating Entities:
   i. Unauthenticated external IT entities that send network data to a network interface of the TOE,
   ii. subjects within the TOE that send network data to unauthenticated external IT entities via a network interface of the TOE;
b) Information:
   i. Network data received by the TOE from an external IT entity,
   ii. Network data provided to the TOE by a subject executing on the TOE intended to be sent to an external IT entity via network interface controlled by the TOE,
   iii. ***[none]***;
c) Operations:
   i. Receiving network data from an unauthenticated external IT entity,
   ii. Sending network data to an unauthenticated IT entity by a subject within the TOE.

5.2.3.6 Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))

FDP_IFF.1(OSPP).1 The TSF shall enforce the Network Information Flow Control Policy based on the following types of subject and information security attributes:

Object security attribute: the logical or physical network interface through which the network data from an external IT entity entered the TOE or is intended to be sent out; ***[***

a) TCP/IP information security attributes:
   i. Source and destination IP address,

ii. ***Source and destination TCP port number,***
iii. ***Source and destination UDP port number,***
iv. ***Network protocol of IP, TCP, UDP, [ICMP, [IPv4 Encapsulation, IPv6, IPv6 Encapsulation]],***
v. ***[[Public, Private, or Domain network profile for the network interface; and***
vi. ***Pathname of an executable (.exe) program file]]].***

FDP_IFF.1(OSPP).2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

for both receiving network data from an external IT entity and sending network data by a subject within the TOE to an external IT entity**:**

a) If the set of rules defined in accordance with the security attributes defined in FDP_IFF.1**(OSPP)**.3 define that the network data is discarded the network data shall not be delivered by the TOE to the intended recipient;
b) If the set of rules defined in accordance with the security attributes FDP_IFF.1**(OSPP)**.3 define that the network data is to be delivered unaltered**,** the network data shall be delivered unaltered by the TOE to its intended recipient;
c) If the set of rules defined in accordance with the security attributes FDP_IFF.1**(OSPP)**.3 define another action to be taken than discarding the network data or delivering the data unaltered to the intended recipient, the TOE shall perform this action.

FDP_IFF.1(OSPP).3 The TSF shall enforce the following rules consisting of an identification when the rule fires and an action to be taken when the rule fires:

Identification of network data using one or more of the following concepts:

a) Information security attribute matching based on the following security attributes **[security attributes described in FDP_IFF.1(OSPP).1 and the Public, Private, or Domain network profile for the network interface]**,
b) ***[[the set of attribute-matching rules associated with a network profile are enforced only for network interfaces defined as part of that profile]], [[no other maching concepts]].***

Performing one or more of the following actions:

a) Discard the network data ***[without any further processing]***;
b) Allow the network data to be delivered unaltered by the TOE to the intended recipient;
c) ***[and perform no other action]***.

FDP_IFF.1(OSPP).4 The TSF shall explicitly authorize an information flow based on the following rules: **[no explicit authorization rules]**.

FDP_IFF.1(OSPP).5 The TSF shall explicitly deny an information flow based on the following rules: **[a network profile can be configured to block all connections regardless of rules which may explicitly allow the connection]**.

5.2.3.7 Full Residual Information Protection (FDP_RIP.2)

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [***allocation of the resource to***] all objects, subjects or subject/object related TSF data before the resource is assigned or made available to another subject or user.

5.2.4 Identification and Authentication (FIA)

5.2.4.1 Authentication Failure Handling (FIA_AFL.1)

FIA_AFL.1.1 The TSF shall detect when an **authorized** administrator configurable positive integer within a range of acceptable values of **consecutive** unsuccessful authentication attempts for the authentication method password-based authentication **[none]** occur related to **[any authorized user authentication process]**.

FIA_AFL.1.2 When the defined number of consecutive unsuccessful authentication attempts has been met **or surpassed**, the TSF shall: **[**

- **a) For all administrator accounts, limit to not more than ten authentication attempts per minute.**
- **b) For all other accounts, disable the user logon account until it is re-enabled by the authorized administrator.**
- **c) For all disabled accounts, any response to an authentication attempt given to the user shall not be based on the result of that authentication attempt.]**

5.2.4.2 User Attribute Definition for Individual Users (FIA_ATD.1(USR))

FIA_ATD.1(USR).1 The TSF shall maintain the following list of security attributes belonging to individual human users:

a) User identifier,
b) Group memberships,
c) ~~user password~~ **Authentication data**,
d) Security roles,[31]
e) **[Private/Public Keys**, and
f) **Privileges,**
g) **Logon rights on specific physically separated parts of the TOE and allowable time and day to logon,**
h) **None]**.

[31] See FMT_SMR.1.

5.2.4.3 Timing of Authentication for OS Logon (FIA_UAU.1(RITE))

FIA_UAU.1(RITE).1 The TSF shall allow

a) The information flow covered by the Network Information Flow Control Policy (for remote IT entities)
b) **[read access to public objects]**.

on behalf of the remote IT entity to be performed before the remote IT entity is authenticated.

FIA_UAU.1(RITE).2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application Note: A Public Object is any object covered by the Discretionary Access Control policy which the TSF unconditionally permits all entities "read" access under the Discretionary Access Control SFP.

5.2.4.4 Timing of Authentication for OS Logon (FIA_UAU.1(OS))

FIA_UAU.1(OS).1 The TSF shall allow **[**

a) read access to public objects]

on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1(OS).2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application Note: A Public Object is any object covered by the Discretionary Access Control policy which the TSF unconditionally permits all entities "read" access under the Discretionary Access Control SFP.

5.2.4.5 Multiple Authentication Mechanisms (FIA_UAU.5)

FIA_UAU.5.1 The TSF shall provide the following authentication mechanisms:

a) Authentication based on username and password (for human users),
b) ***[none]***

to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the following rules:

a) Authentication based on username and password is performed for TOE-originated requests and with credentials stored by the TSF by default unless another authentication method defined for human users in FIA_UAU.5.1(b) is selected;
b) Users with expired password are ***[required to create a new password after correctly entering the expired password* or *locked out until their password is reset by an administrator];***
c) **[none]**.

5.2.4.6 Protected Authentication Feedback (FIA_UAU.7)

FIA_UAU.7.1 The TSF shall provide only obscured feedback to the user while the authentication is in progress.

5.2.4.7 Timing of Identification (FIA_UID.1)

FIA_UID.1.1 The TSF shall allow **[read access to public objects]** on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Application Note: A Public Object is any object covered by the Discretionary Access Control policy which the TSF unconditionally permits all entities "read" access under the Discretionary Access Control SFP.

5.2.4.8 User-Subject Binding for Individual Users (FIA_USB.1(USR))

FIA_USB.1(USR).1 The TSF shall associate the following user security attributes with subjects acting on behalf of that human user:

a) The user identity,
b) **[The security attributed identified in FIA_ATD.1(USR).1 a, d, and FIA_ATD.1(USR).1 e when defined, privileges identified in FIA_ATD(USR).1f and logon rights identified in FIA_ATD.1(USR).1 g]**.

FIA_USB.1(USR).2 The TSF shall enforce the following rules on the initial association of **user** security attributes with subjects acting on the behalf of users: **[**

a) For administrative users, provide restrictive defaults for security attributes identified in FIA_ATD.1(USR),
b) **Restrict the ability to specify alternative initial user security attributes (that override the default attributes) to authorized administrators,**
c) Mandatory Integrity Control integrity labels and policies are assigned as follows:

i. Subjects associated with non-administrative users receive a medium integrity level by default.

ii. Subjects associated with administrative users receive a high integrity level by default.

iii. Subjects started by another subject are assigned the lower of the integrity level assigned to the subject or the integrity level assigned to the executable file associated with the subject if they have the TOKEN_MANDATORY_POLICY_NEW_PROCESS_MIN mandatory policy configured; otherwise they are assigned the integrity level assigned to the executable file associated with the subject.

iv. All subjects are assigned the Mandatory Integrity Control policies: "TOKEN_MANDATORY_POLICY_NO_WRITE_UP" and "TOKEN_MANDATORY_POLICY_NEW_PROCESS_MIN" by default.]

FIA_USB.1(USR).3 The TSF shall enforce the following rules governing changes to the **user** security attributes associated with subjects acting on the behalf of users: **[**

a) **User security attribute changes shall take effect at next user logon.**
b) **Subjects acting on behalf of users cannot add additional security attributes beyond those initially assigned, except when User Account Control is enabled in which case authorized administrators initially are assigned only access rights available to Standard Users and can subsequently escalate their access rights to their assigned (authorized administrator) level.]**

5.2.4.9 Public Key Based Authentication (FIA_PK_EXT.1)

FIA_PK_EXT.1.1 The TSF shall use ***[X.509v3 certificates]*** as defined by ***[RFC 5280]*** to support authentication for ***[IPsec, TLS]*** connections.

FIA_PK_EXT.1.2 The TSF shall store and protect certificate(s) from unauthorized deletion and modification.

5.2.5 Security Management (FMT)

5.2.5.1 Management of Security Functions Behavior for Password Management (FMT_MOF.1(Pass))

FMT_MOF.1(Pass).1 The TSF shall restrict the ability to modify the behavior of the functions password based user authentication to **[authorized administrators]** by allowing those users to specify rules for acceptable passwords that:

a) allow for uppercase characters, lowercase characters, digits, and special characters to be used in passwords
b) define a minimum password length of 8 characters or more (at least up to 15 characters)),
c) define that passwords must have at least one digit and one special character
d) reject passwords used by the same user before up to a history of at least six passwords.

5.2.5.2 Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))

FMT_MSA.1(DAC).1 The TSF shall enforce the **[Discretionary Access Control policy]** to restrict the ability to modify **the security attributes of the objects covered by the SFP except those that define ownership** and ***[query]*** the security attributes of the objects covered by the SFP to the owners of the object and ***[to authorized administrators]***.

5.2.5.3 Management of Security Attributes for Object Ownership (FMT_MSA.1(OBJ))

FMT_MSA.1(OBJ).1 The TSF shall enforce the **[Discretionary Access Control policy]** to restrict the ability to modify and ~~***[***~~***[no other operation]]*** the security attributes **that define ownership** of the objects covered by the SFP to ~~the owners of the object and ***[***~~***authorized administrators]***.

5.2.5.4 Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))

FMT_MSA.1(MIC).1 The TSF shall enforce the **[Mandatory Integrity Control Policy]** to restrict the ability to modify and ***[query]*** the security attributes of the objects covered by the SFP to ~~the owners of the object and~~ **[authorized administrators]**.

5.2.5.5 Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))

FMT_MSA.3(DAC).1 The TSF shall enforce the **[Discretionary Access Control policy]** to provide restrictive default values for security attributes that are used to enforce the SFP.

FMT_MSA.3(DAC).2 The TSF shall allow the **[authorized administrator]** to specify alternative initial values to override the default values when an object or information is created.

5.2.5.6 Static Attribute Initialization for Mandatory Integrity Control Policies (FMT_MSA.3(MIC))

FMT_MSA.3(MIC).1 The TSF shall enforce the **[Mandatory Integrity Control Policy]** to provide restrictive default values for security attributes that are used to enforce the SFP.

FMT_MSA.3(MIC).2 The TSF shall allow ~~the~~ **[no one]** to specify alternative initial values to override the default values when an object or information is created.

5.2.5.7 Static Attribute Initialization for Network Information Flow Control (FMT_MSA.3(OSPP))

FMT_MSA.3(OSPP).1 The TSF shall enforce the Network Information Flow Control Policy to provide ***[[restrictive for Windows 8 and Windows RT]]*** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3(OSPP).2 The TSF shall allow the **[authorized administrator**] to specify alternative initial values to override the default values when an object or information is created.

5.2.5.8 Static Attribute Value Inheritance (FMT_MSA.4)

FMT_MSA.4.1 The TSF shall use the following rules to set the value of security attributes for objects covered by an access control policy **[initial values for the security attributes and objects associated with SFP supported by the TOE are set as specified in Table 5-5]**.

Table 5-5 Attribute Initialization

SFP	Subject/Object	Attribute	Initial Value
DAC	all objects covered by FDP_ACC.1(DAC)	Owner	When an object is created it is assigned an owner based upon the owner SID in the token of the process creating the object.

	all objects covered by FDP_ACC.1(DAC)	DACL	When an object is created it is assigned a DACL from the first available option of the following possiblities • As specified by the creator, • As derived from parent object DACLs, • As defined by the TOE default for the object type, or • As copied from the default DACL in the creating subject's token.
	NTFS file	FEK	When an NTFS file is created, if it is to be encrypted, the TSF creates a randomly generated File Encryption Key (FEK) and encrypts it before storing it as an attribute of the NTFS file.
MIC	all objects covered by FDP_ACC.1(MIC) except process and thread	MIC label & MIC policy	When an object is created, it is assigned an integrity label equal to that of the creating process and a policy of SYSTEM_MANDATORY_LABEL_NO_EXECUTE_UP.
	process and thread objects	MIC label & MIC policy	When an object is created, it is assigned an integrity label equal to that of the creating process and a policy of SYSTEM_MANDATORY_LABEL_NO_READ_UP.

5.2.5.9 Management of TSF Data for Audit Selection (FMT_MTD.1(Audit Sel))

FMT_MTD.1(AuditSel).1 The TSF shall restrict the ability to query, modify the set of audited events to [**authorized administrators**].

Application Note: FMT_MTD.1(AuditSel) applies to FAU_SEL.1 and corresponds to FMT_MTF.1(AE) in the OS PP.

5.2.5.10 Management of TSF Data for Audit Data (FMT_MTD.1(Audit))

FMT_MTD.1(Audit).1 The TSF shall restrict the ability to clear, ***[delete, [and query]]*** the audit storage to **[authorized administrators]**.

Application Note: FMT_MTD.1(Audit) applies to FAU_STG.1 and corresponds to FMT_MTF.1(AS) in the OS PP.

5.2.5.11 Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg)

FMT_MTD.1(AuditStg).1 The TSF shall restrict the ability to modify, ~~[selection: add, delete]~~ the

a) Threshold of the audit trail when an action is performed;
b) Action when the threshold is reached

to **[authorized administrators]**.

Application Note: FMT_MTD.1(AuditStg) applies to FAU_STG.3 and corresponds to FMT_MTF.1(AT) in the OS PP.

5.2.5.12 Management of TSF Data for Audit Log Failure (FMT_MTD.1(Audit Fail))

FMT_MTD.1(AuditFail).1 The TSF shall restrict the ability to modify [***add, delete***]the actions to be taken in case of audit storage failure to [**authorized administrators**].

Application Note: FMT_MTD.1(AuditFail) applies to FAU_STG.4 and corresponds to FMT_MTF.1(AF) in the OS PP.

5.2.5.13 Management of TSF Data for X.509 Certificates (FMT_MTD.1(X509))

FMT_MTD.1(X509).1 The TSF shall restrict the ability to import, enable, disable the digital certificates used for remote entity authentication ***[no other security function]*** to **[authorized administrators]**.

5.2.5.14 Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))

FMT_MTD.1(OSPP).1 The TSF shall restrict the ability to define, ~~query,~~ modify delete, ***[[manage]]*** the security attributes for the rules governing the

a) identification and matching of network data;
b) actions performed on the identified network data

to **[authorized administrators]**.[32]

Application Note: FMT_MTD.1(OSPP) applies to FDP_IFF.1(OSPP).

5.2.5.15 Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Threshold))

FMT_MTD.1(Threshold).1 The TSF shall restrict the ability to modify the threshold for unsuccessful authentication attempts to **[authorized administrators]**.

Application Note: FMT_MTD.1(Threshold) applies to FIA_AFL.1.

5.2.5.16 Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Re-enable))

FMT_MTD.1(Re-enable).1 The TSF shall restrict the ability to re-enable the authentication to the account subject to authentication failure to **[authorized administrators]**.

Application Note: FMT_MTD.1(Re-enable) applies to FIA_AFL.1.

5.2.5.17 Management of TSF Data for Initialization of User Security Attributes (FMT_MTD.1(Init-Attr))

FMT_MTD.1(Init-Attr).1 The TSF shall restrict the ability to initialize~~, modify, delete~~ the user security attributes to **[authorized administrators]**.

[32] All authenticated users can query the status of the Windows firewall using the Get-NetFirewallRule Cmdlet.

5.2.5.18 Management of TSF Data for Modification of User Security Attributes Other Than Authentication Data (FMT_MTD.1(Mod-Attr))

FMT_MTD.1(Mod-Attr).1 The TSF shall restrict the ability to ~~initialize,~~ modify, delete the user security attributes**, other than authentication data,** to **[authorized administrators]**.

5.2.5.19 Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Auth))

FMT_MTD.1(Mod-Auth).1 The TSF shall restrict the ability to ~~initialize,~~ modify~~, delete~~ the user security attributes **of authentication data** to **[[authorized administrators [and users modifying their own authentication data]]**.

5.2.5.20 Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))

FMT_MTD.1(GEN).1 The TSF shall restrict the ability to [**manage**] the [**TSF data except for audit records, user security attributes, authentication data, and critical cryptographic security parameters**] to [**authorized administrators**].[33]

5.2.5.21 Revocation for Object Access (FMT_REV.1(OBJ))

FMT_REV.1(OBJ).1 The TSF shall restrict the ability to revoke object security attributes defined by SFPs**, excluding the Discretionary Access Control Policy**, associated with the corresponding object under the control of the TSF to [**authorized administrators**].

FMT_REV.1(OBJ).2 The TSF shall enforce the following rules:

a) The access rights associated with an object shall be enforced when an access check is made.
b) [**none**].

5.2.5.22 Revocation for Object Access for DAC (FMT_REV.1(DAC))

FMT_REV.1(DAC).1 The TSF shall restrict the ability to revoke object security attributes defined by ~~SFPs~~ **the Discretionary Access Control** Policy associated with the corresponding object under the control of the TSF **to [authorized administrators and [owners of the named object]]** .

FMT_REV.1(DAC).2 The TSF shall enforce the following rules:

a) The access rights associated with an object shall be enforced when an access check is made.
b) [**none**].

5.2.5.23 Revocation for Authorized Administrators (FMT_REV.1(Admin))

FMT_REV.1(Admin).1 The TSF shall restrict the ability to revoke user security attributes defined by the SFP**s** associated with the corresponding user under the control of the TSF to **[authorized administrators]**.

FMT_REV.1(Admin).2 The TSF shall enforce the following rules:

[33] This functional requirement is not part of the OS PP but was added in order to have a management requirement for session locking.

a) The enforcement of revocation of security-relevant authorizations ~~with the next user-subject binding process during the next authentication of the user~~ **at the next logon**.
b) [**None**].

5.2.5.24 Remote Management Capabilities (FMT_SMF_RMT.1)

FMT_SMF_RMT.1.1 The TSF shall allow management functions also to be performed from a remote IT entity using a trusted channel established in accordance with the requirements stated in FTP_ITC.1**(OS)**.

5.2.5.25 Security Roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles:

a) authorized administrator;
b) regular user;[34]
c) **[no other management role**].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

5.2.6 Protection of the TSF (FPT)

5.2.6.1 Basic Internal TSF Data Transfer Protection (FPT_ITT.1)

FPT_ITT.1.1 The TSF shall protect TSF data from [***disclosure***] when it is transmitted between separate parts of the TOE **through the use of the TSF-provided cryptographic services: encryption and decryption**.

5.2.6.2 Reliable Time Stamps (FPT_STM.1)

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

5.2.7 TOE Access (FTA)

5.2.7.1 TSF-initiated Session Locking (FTA_SSL.1)

FTA_SSL.1.1 The TSF shall lock an interactive session to a human user maintained by the TSF after **[an authorized administrator specified time interval of user inactivity]** by:

a) Clearing or overwriting TSF controlled display devices, making the current contents unreadable.
b) Disabling any activity of the user's data access / TSF controlled display devices other than unlocking the session.

FTA_SSL.1.2 The TSF shall require the following **user** events to occur prior to unlocking the session

[34] The OS PP mentions "regular users" however, see section 6.2.5.1 for a more precise description of user roles in Windows.

a) Successful re-authentication with the credentials of the user owning the session using **[the authentication methods described in FIA_UAU.5]**,
b) **[No other events]**.

5.2.7.2 User-initiated Locking (FTA_SSL.2)

FTA_SSL.2.1 The TSF shall allow user-initiated locking of the user's own interactive session maintained by the TSF by:

a) Clearing or overwriting TSF controlled display devices, making the current contents unreadable.
b) Disabling any activity of the user's data access / TSF controlled display devices other than unlocking the session.

FTA_SSL.2.2 The TSF shall require the following **user** events to occur prior to unlocking the session.

a) Successful re-authentication with the credentials of the user owning the session using **[the authentication methods described in FIA_UAU.5]**,
b) **[No other events]**.

5.2.8 Trusted Path/Channels (FTP)

5.2.8.1 Inter-TSF Trusted Channel (FTP_ITC.1 (OS))

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification and disclosure using the following mechanisms:

a) Cryptographically-protected communication channel using **[**
    i. ***TLS as defined in RFC 5246 using X.509 certificates and supporting the following cipher suites defined there:***
        - ***TLS_RSA_WITH_AES_128_CBC_SHA***
        - ***TLS_RSA_WITH_AES_256_CBC_SHA***

        [
        - ***TLS_RSA_WITH_AES_128_CBC_SHA256***
        - ***TLS_RSA_WITH_AES_256_CBC_SHA256***
        - ***TLS_DHE_DSS_WITH_AES_128_CBC_SHA***
        - ***TLS_DHE_DSS_WITH_AES_256_CBC_SHA***
        - ***TLS_DHE_DSS_WITH_AES_128_CBC_SHA256***
        - ***TLS_DHE_DSS_WITH_AES_256_CBC_SHA256***
        - ***TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256***
        - ***TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384***
        - ***TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256***
        - ***TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384]***

    ii. ***IPsec protocol ESP as defined in RFC 4303 using the cryptographic algorithms:***

- ***AES-CBC-128, AES-CBC-256 (both specified by RFC 3602), [AES-GCM-128 as specified in RFC 4106, as specified in RFC 4106, AES-GCM-256 as specified in RFC 4106] for ESP encryption;***[35]
- ***[HMAC-SHA1-96] for ESP authentication and authentication header protection;***[36]
- ***[IKEv1 as defined in RFCs 2407, 2408, 2409, RFC 4109, and [RFC 4868 for hash functions]; IKEv2 as defined in RFCs 5996, 4307, and [RFC 4868 for hash functions]] for key negotiation and SA establishment;***[37]
- ***DH Groups 14 (2048-bit MODP), and [24 (2048-bit MODP with 256-bit POS), 19 (256-bit Random ECP), 20 (384-bit Random ECP)] for use in IKE key establishment;***
- ***[RSA, ECDSA] algorithm for Peer Authentication;]***

FTP_ITC.1.2(OS) The TSF shall permit [***the TSF* or *another trusted IT product***] to initiate communication via the trusted channel.

FTP_ITC.1.3(OS) The TSF shall initiate communication via the trusted channel for all security functions specified in the ST that interact with remote trusted IT systems ~~and **[assignment: list of functions or other conditions which require a trusted channel]**~~.

5.3 OS PP Security Assurance Activities

This section copies the assurance activities from the protection profile in order to ease reading and comparisons between the protection profile and the security target.

5.3.1 Assurance Activities for Security Audit

5.3.1.1 Assurance Activities for FAU_GEN.1: Audit Data Generation[38]

5.3.1.1.1 Background

Operating Systems often have extensive auditing capabilities where not all events recorded are security related. It is therefore necessary to identify the event types and related audit records the operating system is capable to record that map to the generic event types defined in FAU_GEN.1 in the Protection Profile. This is usually one or more record types in the audit trail(s) maintained by the operating system. It is the task of the evaluator to confirm that the operating system is capable to correctly generate the audit records and that the audit records contain the information required by FAU_GEN.1.

[35] Windows also implements AES-CBC-192 and AES-GCM-192 which are not specified in the OS PP.
[36] Windows also implements SHA 256 hashing for ESP authentication which was examined in the IPsec VPN Client evaluation.
[37] RFC 5996 is an update to RFC 4306, which Windows implements.
[38] These activities apply to FAU_GEN.1(OSPP) in the Windows security target.

5.3.1.1.2 TOE Summary Specification (TSS)

5.3.1.1.2.1 Expectations

The TOE Summary Specification shall briefly describe the principle how the operating system generates audit records and name the audit mechanism used to generate the audit records required by FAU_GEN.1. Often this is a single system component and in this case it is just required to name the component and define where the component stores the audit records and how they are protected. The TSS should point to the developer documentation that defines the audit record format, either as they are stored or as they can be extracted (in the case they can only be extracted by a specific function of the TSF). It is important to describe how an administrative user (and the evaluator) can extract the audit records for further processing and analysis. The description in the TSS can be quite generic when it contains sufficient pointers to the developer documentation allowing the evaluator to generate test cases that analyze the audit records in the trail.

5.3.1.1.2.2 Evaluator Activities

The evaluator analyzes the TSS and the documentation the TSS points to in order to verify that this information allows him:

1. to identify the audit trail(s) that contain the audit records related to events defined by FAU_GEN.1
2. to identify the record types for each event defined in FAU_GEN.1
3. to verify that the description of the audit record contains the information required by FAU_GEN.1
4. to identify the interface(s) that can be used to extract and analyze the audit records

5.3.1.1.3 Functional Specification

5.3.1.1.3.1 Expectations

Audit records are usually related to specific events that happen when the operating system is executing. Many of the events defined are directly related to user actions and in those cases the TSFI that are related to the events need to be identified. This is important to allow the evaluator to trigger specific events by using those interfaces and then verifying that the audit record expected to be generated is actually stored in the audit trail. The evaluator therefore needs to ensure that he has obtained sufficient information to trigger the events defined in FAU_GEN.1 using the TSFI.

It is worth to note that some of the auditable events defined in FAU_GEN.1 may have several TSFI that will trigger them. In those cases assurance is needed that all of those interfaces actually also generate the related audit record. The evaluator may use design information provided by the developer that allows him to argue why there is no need to test all of the interfaces. If for example the design information clearly shows that different interfaces internally within the TSF use a common execution path and that the generation of the audit record is within this common execution path, the evaluator can justify performing tests only at one of those interfaces.

5.3.1.1.3.2 Evaluator Activities

The evaluator needs to ensure that all auditable events that can be directly linked to user actions can be mapped to TSFI where the event can be triggered. The evaluator analyzes those interfaces to the extent that he does not identify obvious problems with respect to the specification of the interface, ensuring that he knows how to use the interface for testing. A more detailed analysis will be performed when the interface is used for testing.

As a result of this activity the evaluator shall for every auditable events defined in FAU_GEN.1 have a mapping to the interface(s) that can be used to trigger the event. For events where no such interface exists, the evaluator shall provide his justification why such an interface cannot be expected (based on information provided by the developer) and will also indicate his view how those events may be triggered otherwise. This will be the basis for test cases that test the generation of audit records for those events.

5.3.1.1.4 Architectural Design

5.3.1.1.4.1 Expectations

The TOE design needs to provide an overview on the audit record generation functionality, accompanied by “assurance cases” addressing the potential problems of bypassing or otherwise disturbing audit functionality such that audit records are not generated when they should be, manipulating information to be included in audit records before and when it is collected by the audit record generation functionality, and the protection of the audit record generation functionality from being misused to generate audit records for events that did not happen. In addition the TOE design information needs to describe the format and content of the audit records required by FAU_GEN.1, mapping the details required by FAU_GEN.1 to the content of the records. The information may (and should) be presented by references to existing developer documentation.

5.3.1.1.4.2 Evaluator Activities

The TOE design information provided by the developer needs to be sufficient to address the following issues in the analysis of the functionality for FAU_GEN.1:

1. The evaluator needs to be able to identify a description of the format and structure of all the audit records that map to the auditable events required by FAU_GEN.1. The developer is free to describe the audit records as stored in the TOE internal audit trail or describe the content and format of the audit records extracted from the TOE internal audit trail by a specific tool provided by the developer as part of the TOE. The latter case requires the developer to have a description of the use of this tool sufficient to extract and analyze all the audit records required by FAU_GEN.1
2. The evaluator needs to be able to identify that the audit records are actually generated by the TSF and not by a part of the TOE. The developer needs to provide sufficient arguments that the audit record generation can be influenced or even bypassed by a user.
3. The evaluator needs to be able to identify where the TSF collects the information it stores in the audit record. The developer needs to provide sufficient arguments that this information may not be subject to manipulation.

4. The evaluator needs to be able to identify that the functionality used by the TOE to generate audit records cannot be invoked by an untrusted user such that it generates an audit record for an event that never happened by using the audit functionality to produce an audit record indistinguishable from an audit record generated by the TSF for an event defined in FAU_GEN.1.

5.3.1.1.5 User Guidance

5.3.1.1.5.1 Expectations

The user guidance related to FAU_GEN.1 needs to explain how a user authorized to extract the audit records can do this. It further needs to explain how individual information from the audit records can be presented or extracted in order to verify that all audit records expected have been generated and that the audit records contain the expected information.

5.3.1.1.5.2 Evaluator Activities

The evaluator needs to ensure that the user guidance contains information about the audit records that can be generated, how to extract the audit records and how to identify the information specified in FAU_GEN.1 in the individual audit records. This information is required to be able to test FAU_GEN.1 and to ensure that all the required information is included in the different audit records that map to the requirements in FAU_GEN.1.

5.3.1.1.6 Testing

5.3.1.1.6.1 Expectations

The developer should be able to present test results from his test suite demonstrating that:

1. audit records have been generated when they should be
2. audit records contain the expected information and correctly reflect the event

Usually there is little specific testing required since the generation of audit records is (in the case of the events described in FAU_GEN.1 in the base OSPP) related to the invocation of security functions provided by the TOE that need to be tested for their specific security functionality anyhow. In order to validate the generation of the audit records, the TOE should be tested generally with all auditable events specified in FAU_GEN.1 being turned on. As long as this is not done as part of stress testing, the timing overhead associated with this extensive auditing can be neglected. Stress tests or fuzz tests that are performed in addition to pure functional testing may well be performed with a configuration where no or only a few auditable events are actually being audited. The tests shall cover all audit events defined in FAU_GEN.1 in the Security Target to show that for each of the events defined in FAU_GEN.1.1 an audit record is created and contains the information defined for the audit records in FAU_GEN.1.2.

5.3.1.1.6.2 Evaluator Activities

The evaluator analyzes the test results presented by the developer and for completeness and correctness. Note that in the case where the developer has produced a massive amount of test results resulting in a very large number of audit records being generated, the developer and the evaluator should work together on a strategy to sample those results. The sample should include cases demonstrating the correct generation of audit records for all events defined in FAU_GEN.1.1.

For those audit records not found in the sample, the evaluator defines his own test cases that are expected to cause the events related to those audit records and therefore are expected to create those records. The evaluator verifies that those audit records have been generated correctly.

After the tests have been performed, the audit records need to be extracted as part of the test results and compared to the expected audit events and content of the audit records. The evaluator needs to ensure that for each event defined in FAU_GEN.1 the expected audit records have been generated and the audit records show the expected content.

5.3.1.2 Assurance Activities for FAU_GEN.2: User Identity Association

5.3.1.2.1 Background

In order to achieve the objective of user accountability it is required that the events recorded in the audit records can be traced to the user that caused the event, provided the event is directly related to the action of a user. This accountability has to be ensured even in cases where the subject operating on behalf of that user temporarily gets a different user ID assigned as one of its security attributes. Many operating systems allow a trusted subject's security attribute "user ID" to be changed under the control of the OS in order to perform actions the user would not be allowed to perform using an untrusted program.

FAU_GEN.2 requires that even in those cases the identity of the user that caused the event can be associated with the user. Note that this does not require that the ID of the user that caused the event is directly placed in the audit record.

If another audit record audits this change of ID that can be easily and unambiguously linked to the audit record of the event the ability to associate such auditable events with the identity of the user that caused the event is given.

In addition an operating system may allow a user to request a service from a trusted subject using some inter-process communication function. Also in this case it must be possible to associate the identity of the user the requested the service when an audit record is generated during the processing of the request.

5.3.1.2.2 TOE Summary Specification (TSS)

5.3.1.2.2.1 Expectations

The TSS shall identify and describe the possible ways where an audit record is created by a subject that – at the time of the creation of the audit record – is not "bound" to the user that caused the related event.

The TSS shall explain how the identity of the user that caused the event is associated with the audit record for the event also in those cases. If the identity of that user is not part of the audit record, the TSS shall describe how someone evaluating the audit records can easily and unambiguously establish the association between the audit record and the user that caused the event recorded in the audit record.

5.3.1.2.2.2 Evaluator Activities

The evaluator analyzes the TSS and the documentation the TSS points to in order to verify that this information allows him to establish an unambiguous link between the audit record and the user that caused the event.

5.3.1.2.3 Functional Specification

5.3.1.2.3.1 Expectations

The description of the audit records shall include all the information described as necessary to establish the association between the audit record and the user that caused the event leading to the creation of the audit record.

5.3.1.2.3.2 Evaluator Activities

The evaluator verifies that the information provided allows for unambiguous association between the user that caused the event recorded in the audit record and the audit record itself.

5.3.1.2.4 Architectural Design

5.3.1.2.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.1.2.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the association between the user that caused the event and the audit record is established.

5.3.1.2.5 User Guidance

5.3.1.2.5.1 Expectations

If a specific configuration is required to establish the association between the user that caused the event and the audit record, it is expected that the configuration and the steps to get to this configuration are correctly and completely described in the guidance.

5.3.1.2.5.2 Evaluator Activities

If a specific configuration is required to establish the association between the user that caused the event and the audit record, the evaluator follows this guidance to configure the TOE such that the association between the user that caused the event and the audit record can be established.

5.3.1.2.6 Testing

5.3.1.2.6.1 Expectations

The developer is expected to demonstrate in his testing that the association between the user that caused the event and the audit record can be established. Testing shall cover all cases identified in the TSS where an audit record is created by a subject that – at the time of the creation of the audit record –

is not “bound” to the user that caused the related event. The test cases must identify the user(s) that caused the events.

5.3.1.2.6.2 Evaluator Activities

The evaluator verifies that the test cases provided cover all cases identified in the TSS where an audit record is created by a subject that – at the time of the creation of the audit record – is not “bound” to the user that caused the related event. The evaluator extracts the audit records generated by those test cases and determines if he is able to establish the association of the event that caused the audit record to be created with the user that caused the event. The evaluator defines and executes his own test cases, collects the audit records generated and determines if he is able to establish the association of the event that caused the audit record to be created with the user that caused the event.

5.3.1.3 Assurance Activities for FAU_SAR.1: Audit Review and FAU_SAR.2: Restricted Audit Review

5.3.1.3.1 Background

Reading the audit records needs to be restricted to users authorized to do so. This authorization may be assigned to a role or a privilege or there may be more complex rules governing the reading of audit data. Documentation needs to be provided that describes the interface(s) that can be used to read the audit data and the format of the audit records when read using those interfaces.

5.3.1.3.2 TOE Summary Specification (TSS)

5.3.1.3.2.1 Expectations

The TSS shall describe when a user is allowed to read the audit data. The TSS or documentation pointed to by the TSS need to describe the interface(s) that can be used to read the audit data and the format of the audit records when read using those interfaces.

In the case a regular file interface is used to read the audit data where the file access control functionality is used to restrict the users able to read the audit data, the format of the audit data in the file needs to be described to the extent that it is possible to correctly identify and interpret the information in the audit record.

5.3.1.3.2.2 Evaluator Activities

The evaluator analyzes the TSS and the documentation the TSS points to in order to verify that this information allows him to identify the exact conditions that need to be met for a user to be allowed to read audit data. The evaluator analyzes also the information provided on how the audit records are provided to ensure that all information required by FAU_GEN.2 is provided and that the information is suitable for the intended purpose.

The intended purpose may be either reading the audit data directly (which requires them to be in printable form) or in a format suitable for post-processing by a program. In both cases the information required by FAU_GEN.2 needs to be identifiable and needs to be described such that they can be correctly interpreted,

5.3.1.3.3 Functional Specification

5.3.1.3.3.1 Expectations

The functional specification shall identify the interface(s) that can be used by appropriately authorized users to read the audit data. The functional specification or the guidance (or both) need to completely and correctly describe the conditions a user needs to meet in order to use those interfaces to read the audit data. The functional specification needs to describe how the audit data is presented in a way that allows extracting the information required by FAU_GEN.2 from the audit records.

5.3.1.3.3.2 Evaluator Activities

The evaluator verifies that the information provided for accessing the audit data completely describe the conditions that must be met to read the audit data and that this description is consistent with the specification provided in FAU_SAR.1.1 of the Security Target. The evaluator verifies that the description how the data is provided allows him to extract the information required by FAU_GEN.1.

Note: this requirement is also satisfied if the required information is provided in the guidance documentation. In this case the evaluator uses the guidance documentation for the activities described below.

5.3.1.3.4 Architectural Design

5.3.1.3.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.1.3.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the audit data can be read and what the format of the audit data presented is.

5.3.1.3.5 User Guidance

5.3.1.3.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to read the audit data. The guidance needs to explain the format the audit records are presented.

5.3.1.3.5.2 Evaluator Activities

See the evaluator activities for the functional specification.

5.3.1.3.6 Testing

5.3.1.3.6.1 Expectations

The developer is expected to demonstrate in his testing that the conditions for reading the audit data are enforced and how the audit records can be read.

5.3.1.3.6.2 Evaluator Activities

The evaluator activities for this SFR consist of two main aspects:

1. Verification that only properly authorized users can access the audit data.
2. Verification that the audit data contain the required information in a form suitable for the intended processing (reading directly or post-processing by some program)

For the first aspect, the evaluator treats the conditions that must be met for reading the audit data as an access control algorithm and requires testing to be performed in the same way as outlined in the testing for discretionary access control in FDP_ACF.1.

For the second aspect the evaluator obtains audit data via the described interface(s) and verifies that the information required by FAU_GEN.1 can be extracted in the form suitable for the intended processing. The test sample needs to include audit records for all events defined in FAU_GEN.1.

5.3.1.4 Assurance Activities for FAU_SEL.1: Selective Audit and FMT_MTD.1(AE): Management of TSF data: Audit Events[39]

5.3.1.4.1 Background

For performance reasons and in order to save disk space an installation will usually not always generate audit records for all events defined in FAU_GEN.1. Therefore the OSPP requires the possibility to limit the events that are actually audited using criteria defined in FAU_SEL.1. The SFR FMT_MTD.1(AE) defines the conditions a user must satisfy in order to select the set of events that are actually audited from the overall set of auditable events defined in FAU_GEN.1.

5.3.1.4.2 TOE Summary Specification (TSS)

5.3.1.4.2.1 Expectations

The TSS needs to explain how the set of events that are actually audited can be limited in compliance with the criteria defined in FAU_SEL.1. The also TSS needs to describe how the management of this set of auditable events, pointing to the interface(s) used for this management.

5.3.1.4.2.2 Evaluator Activities

The evaluator verifies that the explanation in the TSS and the documents pointed to by the TSS is consistent with the requirements defined in FAU_SEL.1 (i. e. allows restricting the set of audited events in accordance with the criteria defined in FAI_SEL.1) and is consistent with the conditions that must be met to perform this management operation as defined in FMT_MTD.1(AE).

5.3.1.4.3 Functional Specification

5.3.1.4.3.1 Expectations

The functional specification shall identify the interface(s) that can be used by appropriately authorized users to manage the set of events to be audited. The functional specification or the guidance (or both)

[39] The FMT_MTD.1(AE) assurance activities apply to FMT_MTD.1(AuditSel) in this security target.

need to completely and correctly describe the conditions a user needs to meet in order to use those interfaces to manage the event that are audited.

5.3.1.4.3.2 Evaluator Activities

The evaluator verifies that the information provided for managing the events to be audited completely describe the conditions that must be met to manage the audited events and that this description is consistent with the specification provided in FAU_SEL.1 and FMT_MTD.1(AE) of the Security Target.

Note: this requirement is also satisfied if the required information is provided in the guidance documentation. In this case the evaluator uses the guidance documentation for the activities described below.

5.3.1.4.4 Architectural Design

5.3.1.4.4.1 Expectations

There are no further expectations on the architectural design for those SFRs than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.1.4.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the audit events can be managed and what the possibilities for selecting the events to be audited are.

5.3.1.4.5 User Guidance

5.3.1.4.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to manage the set of auditable events.

5.3.1.4.5.2 Evaluator Activities

See the evaluator activities for the functional specification.

5.3.1.4.6 Testing

5.3.1.4.6.1 Expectations

The developer is expected to demonstrate in his testing that the conditions for managing the set of auditable events are enforced and how the auditable events can be restricted in accordance with the criteria defined in FAU_SEL.1.

5.3.1.4.6.2 Evaluator Activities

The evaluator activities for those SFRs consist of three main aspects:

1. Verification that only properly authorized users can manage the set of auditable events.
2. Verification that the set of auditable events can be restricted in accordance with the criteria defined in FAU_SEL.1.

3. Verification that the TOE audits exactly the events that are defined.

For the first aspect, the evaluator treats the conditions that must be met for managing the set of auditable events as an access control algorithm and requires testing to be performed in the same way as outlined in the testing for discretionary access control in FDP_ACF.1.

For the second and third aspect the evaluator identifies test cases for each criteria mentioned in FAU_SEL.1, sets the set of auditable events in accordance with those criteria, executes a test program that would generate the appropriate audit records and verifies that the audit records are created when the criteria are defined to create them and are not created if the criteria are defined to not create the audit records.

5.3.1.5 Assurance Activities for FAU_STG.1: Protected Audit Trail Storage

5.3.1.5.1 Background

Protection of the audit trail against unauthorized deletion of audit records is often achieved by using the file protection mechanism provided by the OS together with specific guidance on how to use this protection mechanism. If this is the case and no audit trail specific protection mechanisms have been implemented, the assessment of this SFR is covered by the assessment of the file protection mechanism and an assessment of the audit trail specific guidance. Only if the TOE implements audit trail specific functions for the protection of the audit records from unauthorized deletion the assurance activities for the functional specification, the architectural design, and the testing need to be performed.

5.3.1.5.2 TOE Summary Specification (TSS)

5.3.1.5.2.1 Expectations

The TSS shall describe how the audit records are protected from unauthorized deletion.

5.3.1.5.2.2 Evaluator Activities

The evaluator analyzes the TSS and the documentation the TSS points to and identifies if the TOE uses audit trail specific protection mechanisms. If this is the case, the evaluator needs to perform the complete set of assurance activities defined for FAU_STG.1. Otherwise the evaluation only verifies that the general protection mechanisms used are covered by other SFRs (usually those for access control to storage objects) and refers to the assurance activities defined there. In this case the evaluator only verifies that the guidance provided for the protection of the audit trail ensures that the protection mechanisms are used correctly.

5.3.1.5.3 Functional Specification

5.3.1.5.3.1 Expectations

The functional specification shall identify the audit trail specific interface(s) used for the protection of the audit trail if such interfaces exist e. g. for managing aspects of the protection.

The functional specification needs to identify if audit trail specific interfaces for deleting audit records from the audit trail or deleting all record from the audit trail exist. If they do, the functional specification

needs to describe how they can be used and how the authorization of the user of those interfaces is validated.

5.3.1.5.3.2 Evaluator Activities

The evaluator verifies that the information provided for deleting records from the audit trail or all records form the audit trail completely describe the conditions that must be met to delete records from the audit trail.

Note: this requirement is also satisfied if the required information is provided in the guidance documentation. In this case the evaluator uses the guidance documentation for the activities described below.

5.3.1.5.4 Architectural Design

5.3.1.5.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.1.5.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the audit data can be read and what the format of the audit data presented is.

5.3.1.5.5 User Guidance

5.3.1.5.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to delete records from the audit trail.

5.3.1.5.5.2 Evaluator Activities

See the evaluator activities for the functional specification.

5.3.1.5.6 Testing

5.3.1.5.6.1 Expectations

The developer is expected to demonstrate in his testing that the conditions for deleting records from the audit trail are enforced and that only the audit records selected are deleted.

5.3.1.5.6.2 Evaluator Activities

The evaluator activities for this SFR consist of two main aspects:

1. Verification that only properly authorized users can delete records from the audit trail.
2. Verification that only the records intended to be deleted are actually deleted.

For the first aspect, the evaluator treats the conditions that must be met for deleting records from the audit trail as an access control algorithm and requires testing to be performed in the same way as outlined in the testing for discretionary access control in FDP_ACF.1.

For the second aspect the evaluator deletes selected audit records or the complete audit trail and then verifies that only those audit records have been deleted that have been selected for deletion.

5.3.1.6 Assurance Activities for FAU_STG.3: Action in Case of Possible Audit Data Loss, FAU_STG.4: Prevention of Audit Data Loss, and FMT_MTD.1(AF) Management of TSF Data[40]

5.3.1.6.1 Background

There may be a number of conditions that potentially could lead to a loss of audit data; reaching a defined threshold is just one of them. Another problem is a critical situation detected by the TSF that causes the TSF to shut down the TOE. In cases where the audit data is automatically transferred to another trusted IT system, any problem in the communication link with this system could potentially lead to a loss of audit data.

5.3.1.6.2 TOE Summary Specification (TSS)

5.3.1.6.2.1 Expectations

FAU_STG.3.1 requires the author of an ST to list in the TSS the conditions of potential loss of audit data the TSF is able to detect and describe the reaction of the TSF when such a condition is detected. This reaction may consists of a notification of some

FAU_STG.4.1 is specific for the condition that the audit trail reaches its storage limits. The TSS needs to specify the actions the TSF take when the audit trail is full, explain which audit records may get lost and which options an authorized administrator has to configure the actions taken by the TSF when the audit trail is full.

FMT_MTD.1(AF) defines the management of the actions to be taken in case of an audit storage failure.

5.3.1.6.2.2 Evaluator Activities

The evaluator verifies that the explanation in the TSS and the documents pointed to by the TSS describe the reaction of the TOE to the situation described and that this is consistent with the specification in the FAU_STG.3.

The evaluator also verifies that the description of the actions taken in case the audit trail is full are consistent with the specification in FAU_STG.4.

The evaluator verifies that the TSS (and the documents pointed to by the TSS) define the possible actions taken by the TOE in case of an audit storage failure and those can be managed.

[40] The FMT_MTD.1(AF) assurance activities apply to FMT_MTD.1(AuditFail) in this security target.

5.3.1.6.3 Functional Specification

5.3.1.6.3.1 Expectations

The functional specification shall identify the interface(s) that can be used by appropriately authorized users to perform potential management activities related to FAU_STG.3 and FAU_STG.4. Note that the Protection Profile does not require such management functionality to exist, but leaves the option in FAU_STG.4 to specify a function to overwrite the default values for the action to be taken when the audit trail is full.

The functional specification shall identify the interfaces that allow the management of the actions to be taken in case of an audit storage failure (which include configuration interfaces that for example allow an automatic switch to another audit storage).

5.3.1.6.3.2 Evaluator Activities

The evaluator identifies from the description provided possible management actions that can be performed for FAU_STG.4. Those are mapped to the interfaces that have been identified for such management activities and analyzed for consistency with the specification in the ST.

The evaluator identifies the management interface(s) for managing the actions to be taken in case of an audit storage failure and verifies that they allow the type of management defined in FMT_MTD.1(AF) with the details mentioned in the TSS.

5.3.1.6.4 Architectural Design

5.3.1.6.4.1 Expectations

The architectural design needs to explain how the TSF detects that the audit storage exceeds the pre-defined limit or any other of the conditions specified in FAU_STG.3 and how the actions taken in this case are initiated by the TSF. The architectural design needs to explain how the TSF detects an audit storage failure and how it reacts to such a failure.

5.3.1.6.4.2 Evaluator Activities

The evaluator checks those descriptions for consistency with the specification in the ST.

5.3.1.6.5 User Guidance

5.3.1.6.5.1 Expectations

If there are management activities for FAU_STG.4, the guidance needs to explain those activities, the conditions that need to be met to perform those activities and the impact of those activities on the capability of the TOE to generate audit records. Especially if specific types of audit records get lost or if the TOE starts to overwrite old audit records, this needs to be explained in the guidance. The guidance also needs to provide advice how to avoid getting into a situation where audit records get lots (e. g, by automatically initiating backup procedures for the audit trail). The guidance needs to explain the options an administrator has for the actions to be taken in case of an audit storage failure and what the consequences of each of those options are.

5.3.1.6.5.2 Evaluator Activities

For the assessment of the management interfaces see the evaluator activities for the functional specification. The evaluator also analyzes the guidance given for preventing the loss of audit records and the management of actions in case of an audit storage failure and uses this in the development of test cases.

5.3.1.6.6 Testing

5.3.1.6.6.1 Expectations

The developer is expected to provide test cases and test results for the conditions defined in FAU_STG.3.1 showing that each of those conditions causes the TSF to take the actions described in FAU_STG.3.1. The developer is also expected to provide test cases showing the actions taken when the audit trail is full unless this condition cannot be reached in normal operation. In this case the developer needs to provide arguments based on the architectural design demonstrating that

a. Reaching the condition that the audit trail gets full is hard to test (even when configuring the minimum size of the audit trail allowed by the TOE)
b. If the audit trail gets full, the TSF will take the action described in FAU_STG.4.1 for this case.

The developer still has to present an estimate for the effort it would take to develop a test case for FAU_STG.4.1. The developer may choose to present a test case where the specific functionality of the TOE reacting to a full a audit trail is executed in a specific environment (e. g. using a debugger or a virtualized environment) that allows to simulate the condition of a full audit trail.

Note that in the case the audit records are sent to a remote system, the situation of a full audit trail is equivalent to the situation where the remote system is no longer capable of receiving audit records. In this case the situation of a “full” audit trail can be easily simulated by disrupting the connection to the remote system.

If possible there should also be tests simulating an audit storage failure. For example in cases where audit storage is on local disks and the TOE allows for easy removing of a disk (e. g. in case of a USB disk), the developer is expected to test the case where the audit storage is on such a removable disk and this disk is removed during operation.

5.3.1.6.6.2 Evaluator Activities

The evaluator verifies that all conditions listed in FAU_STG.3.1 are covered by test cases and also verifies that in each case the test results show that the actions defined inFAU_STG.3.1 have been taken.

If the developer has provided tests for FAU_STG.4.1, the evaluator will analyze the test results and determine if and why the test results show clearly that the actions specified in FAU_STG.4.1 have been taken by the TOE.

If the developer has not provided test cases for FAU_STG.4.1 with the arguments why reaching the situation where the audit trail is full is not possible without undue effort, the evaluator will provide his judgment of the arguments (including the arguments why this situation can not be tested in a specific

environment) and will then analyze the arguments presented by the developer showing that the TOE will take the actions defined in FAU_STG.4.1 for the case when the audit trail is full. The evaluator will provide his judgment for those arguments in the evaluation report. The final decision if the arguments presented by the developer are acceptable is with the Certification Body.

For testing FMT_MTD.1(AF) the evaluator checks if the audit storage can be configured to be on a device that can be easily removed or can be configured to be sent to a remote system where the network connection to this system can be easily disrupted. If this is the case the evaluator tests if the correct action in case of an audit storage failure is taken by removing the disk or disconnecting the network while the TOE is operating and produces audit records.

5.3.1.7 Assurance Activities for FMT_MTD.1(AS): Management of TSF Data: Audit Storage[41]

5.3.1.7.1 Background

This function is related to the management of the audit storage, which includes a possible selection and configuration of the audit storage location and parameter, a possible creation and deletion of such storage and the clearing of the full audit trail. Note that clearing of the full audit trail is equivalent to deleting all audit records and therefore the authority to clear the audit storage as a whole must be higher than the authority required to delete individual audit records.

Note: A TOE may implement a function in the audit subsystem that allows for deleting individual audit records while the clearing of the audit storage as a whole may be implemented by file system and just require the (file system specific) authority to delete the file assigned in the configuration to be the audit trail. In this case the authorizations for the two actions are usually independent from each other and in this case the guidance needs to give advise how to co-ordinate those authorizations.

5.3.1.7.2 TOE Summary Specification (TSS)

5.3.1.7.2.1 Expectations

The TSS needs to describe how the storage intended to contain the audit trail is initially set up and configured, needs to describe the operations that can be performed on the audit trail storage object and how those operations are controlled.

5.3.1.7.2.2 Evaluator Activities

The evaluator verifies that the description of the audit trail storage management covers the life-cycle of the audit trail storage object from its creation, assignment as the audit trail storage object and initial configuration, to its management (clearing, re-assignment of audit trail storage (if possible), pt o deleting the audit trail storage object (if possible).

For all those actions the authority required to perform the action needs to be specified. The evaluator verifies that this management model is consistent with the management of other audit trail functions.

[41] The FMT_MTD.1(AS) assurance activities apply to FMT_MTD.1(Audit) in this security target.

5.3.1.7.3 Functional Specification

5.3.1.7.3.1 Expectations

The functional specification shall identify the interface(s) that can be used by appropriately authorized users to perform management activities related to FMT_MTD.1(AS).

5.3.1.7.3.2 Evaluator Actions

The evaluator identifies from the description provided possible management actions that can be performed for FMT_MTD.1(AS). Those are mapped to the interfaces that have been identified for such management activities and analyzed for consistency with the specification in the ST.

5.3.1.7.4 Architectural Design

5.3.1.7.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.1.7.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the audit data storage object can be managed.

5.3.1.7.5 User Guidance

5.3.1.7.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to manage the audit trail storage object.

5.3.1.7.5.2 Evaluator Activities

For the assessment of the management interfaces see the evaluator activities for the functional specification.

5.3.1.7.6 Testing

5.3.1.7.6.1 Expectations

The developer is expected to provide test cases and test results for the individual management activities defined in FMT_MTD.1(AS), showing that the management activities can be performed and have specified effect when the user has the required authority to perform the activity. The developer is also expected to provide test cases showing the management activities defined in FMT_MTD.1(AS) cannot be performed when the user does not have the required authorization.

5.3.1.7.6.2 Evaluator Activities

The evaluator verifies that all management activities listed in FMT_MTD.1(AS) are covered by test cases and also verifies that in each case the test results show that the management activity has the specified effect when the user performing the management activity is sufficiently authorized. The evaluator also

verifies that the tests demonstrate that an attempt to perform a management operation specified in FMT_MTD.1(AS) without the required authorization fails.

5.3.1.8 Assurance Activities for FMT_MTD.1(AT): Management of TSF Data: Audit Threshold[42]

5.3.1.8.1 Background

This function is related to the setting of the threshold that triggers the actions defined in FAU_STG.3.1.

5.3.1.8.2 TOE Summary Specification (TSS)

5.3.1.8.2.1 Expectations

The TSS needs to describe how the threshold for the audit storage that triggers the actions defined in FAU_STG.3.1 can be managed.

5.3.1.8.2.2 Evaluator Activities

The evaluator verifies that the functionality described in the TSS specifies how the audit trail threshold can be managed and which interface(s) can be used for this action.

5.3.1.8.3 Functional Specification

5.3.1.8.3.1 Expectations

The functional specification shall identify the interface(s) that can be used by appropriately authorized users to manage the audit trail threshold use by FAU_STG.3.1.

5.3.1.8.3.2 Evaluator Activities

The evaluator identifies from the description how the audit trail threshold can be managed. This description needs to specify, which authorization is required to perform this management action and what the limits for the possible values of this threshold are.

The evaluator verifies that the possible values for the threshold make sense (e. g. are neither negative nor larger than 100% of the audit trail capacity).

5.3.1.8.4 Architectural Design

5.3.1.8.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.1.8.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the audit data storage object can be managed.

[42] The FMT_MTD.1(AT) assurance activities apply to FMT_MTD.1(AuditStg) in this security target.

5.3.1.8.5 User Guidance

5.3.1.8.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to manage the audit trail storage threshold.

5.3.1.8.5.2 Evaluator Activities

For the assessment of the management interfaces see the evaluator activities for the functional specification.

5.3.1.8.6 Testing

5.3.1.8.6.1 Expectations

The developer is expected to provide test cases and test results for the setting of the audit trail threshold. The developer is expected to execute the tests for FAU_STG.3.1 using different values for the audit trail threshold, showing that the actions defined in FAU_STG.3.1 are correctly taken when the threshold as defined is exceeded.

5.3.1.8.6.2 Evaluator Activities

The evaluator verifies that the test results show that the actions defined in FAU_STG.3.1 are taken when the threshold is exceeded independent how the value for this threshold has been set.

5.3.2 Assurance Activities for User Data Protection

5.3.2.1 Assurance Activities for FDP_ACC.1 "Subset Access Control", FDP_ACF.1 "Security Attribute Based Access Control"[43]

5.3.2.1.1 Background

Operating Systems need to control access to objects they define. The OSPP base requires that an access control policy exists for all objects that allow sharing of data between different users. An operating system may implement different access control policies for different types of objects and if this is the case, the Security Target needs to have multiple instances for FDP_ACC.1 and FDP_ACF.1. The OSPP further requires that at least one access control policy for one type of named objects provides the capability to define access down to granularity of a single user.

While the OSPP requires that objects that can be used for sharing data between different users are covered by an access control policy, an operating system may use access control policies also for controlling a user's access to specific operating system functions, use of specific privileges, or other type of "objects" not used for sharing data. A Security Target may well define also those access control policies.

The ST author needs to define in the SFRs for each access control policy:

[43] These activities apply to FDP_ACC.1(DAC) and FDP_ACF.1(DAC) in the Windows security target.

- The types of objects, type of subjects or users and the operations covered by the access control policy
- The exact rules used by the TOE to determine if a subject/user (of the type defined in the access control policy) is allowed to perform one of the operations covered by the access control policy on an object (of the type defined in the access control policy). If the access control policy allows the definition of conflicting access rights, the algorithm needs to define how those conflicts are resolved.

Note that the same type of object may appear in different access control policies if the rules differ for different types of subjects or users or for different operations.

Note also that there may be cases where the rules used by the access control policy themselves can be managed. In this case the Security Target needs to define a fixed rule set, the guidance needs to explain how to set up this rule set for the TOE, the management of the rule set needs to be restricted to trusted administrators (or deactivated) and the administrators need to be advised in an “Evaluated Configuration Guide” to not change this rule set.

There are strong dependencies between the assessment of the access control policies themselves and the management of (user and object) security attributes as well as other TSF data used in making an access control decision. This will result in overlap in the Assurance Activities for the access control policy and the management of TSF data used in the access control policy. The evaluator should not perform assessment related to management SFRs twice but refer to the assessment performed for management SFRs in his assessment of FDP_ACC and FDP_ACF where necessary.

5.3.2.1.2 TOE Summary Specification (TSS)

5.3.2.1.2.1 Expectations

The TOE Summary Specification (or public documentation pointed to by the TSS) shall briefly describe the mechanisms the TOE uses to implement the access control policies and the security attributes used in the policy. For example if the TOE uses a combination of “permission bits” and “access control lists” the TOE Summary Specification needs to explain this and needs to explain how they are managed. This applies also to any other security attribute mentioned in the access control policies. Concerning the management of those security attributes, the TOE Summary Specification needs to provide information about:

- How each security attribute is initialized, resp. what the default value of the security attribute is
- How the value of the security attribute can be modified (if at all) and what the rules are the TOE uses to determine if the modification is allowed
- In addition the TOE Summary Specification (or public documentation pointed to by the TSS) needs to describe:
- The conditions that need to be satisfied when a user/subject requests to create a new object (for all objects mentioned in one of the access control policies), - The rules that determine the default access rights assigned when a new object is created (for all objects mentioned in one of the access control policies),

- The conditions that need to be satisfied when a user/subject requests to delete an object (for all objects mentioned in one of the access control policies)

5.3.2.1.2.2 Evaluator Activities

The evaluator first analyzes the access control algorithm(s) defined in the SFRs (which may potentially be refined in the TSS) to validate that they are complete, providing a yes or no decision with all possible combinations of security attributes used in the rules defining the policy.

The evaluator analyzes the SFRs and the TSS for consistency. All access control policies listed in the SFRs should also be described in the TSS with the same types of objects, subjects and operations and for all security attributes mentioned in the policy the TSS needs to explain if and how they can be managed. The evaluator constructs for each access control policy a list of security attributes mentioned in the rules of the access control policy and verifies for each security attribute that the TSS either mentions it as either non-manageable (or managed internally by the TSS) or defines the rules governing the management of the security attribute. The evaluator then should have a complete model for all access control policies that define the types of subjects/users, the type of objects, and the operations covered by the access control policy as well as the full set of rules used by the TOE to determine if access is allowed by the policy. The evaluator also has the list of all security attributes used in the rules of the access control policy together with the rules that determine how those security attributes can be managed. In addition the evaluator has the rules that determine when a new object can be created together with the values of the object security attributes assigned at creation and the rules that determine when an object can be deleted.

The evaluator uses this model of the access control policies to check for completeness and for inconsistencies within this model. An example for an inconsistency would be a type of object that appears in more than one access control policy where the evaluator identifies an overlap also in the types of subjects/users and the operations and where the rules for the overlapping parts differ between the two policies. Another example of an inconsistency would be when the rules for an operation that implies another operation provide more access than the implied operation (e. g. the rules would allow a “read and write” operation in cases where it would not allow a “read” operation).

5.3.2.1.3 Functional Specification

5.3.2.1.3.1 Expectations

The functional specification (which is publically available) shall identify all the interfaces to the TSF where access control is enforced as well as all the interfaces used to manage the access control policy or the security attributes used in the access control policies. Each interface where access control is enforced needs to describe how the caller is informed in the case access is denied. All the interfaces need to be described such that they can be used in testing the access control policy or the management activity.

5.3.2.1.3.2 Evaluator Activities

The evaluator verifies with all the interfaces identified as one where access control is enforced that the types of objects and the operations of the access control policy (or policies) addressed by the interface

are identified and map to the description of the access control policy. If the description of the interface mentions more types of objects or more operations (as being subject to the access control policy) than defined in the access control policy description in the Security Target, the evaluator needs to flag this as an inconsistency. Unless the developer can provide an explanation accepted by the evaluation facility and the scheme that this is not an inconsistency, an update of the Security Target is required that removes this inconsistency.

In addition the evaluator verifies that for all security attributes that the Security Target claims are manageable, a management interface is identified in the functional specification that allows for the management action defined in the Security Target and that those interfaces are described such that they can be used for testing the management functionality.

Note: this assessment overlaps with assessment activities performed for SFRs in the management area and the evaluator ensures that the different aspects of the management activities are assessed only once. The evaluator may refer in the activities performed for FDP_ACC/FDP_ACF to the assessment performed when analyzing the SFRs related to management or vice versa.

5.3.2.1.4 Architectural Design

5.3.2.1.4.1 Expectations

The TOE design documentation (which consists of the TSS in the Security Target, the functional specification and any additional design related documentation provided for the evaluation) needs to explain the principles of the implementation of the access control policy (or policies), especially how and where the security attributes are stored and maintained by the TSF. In the cases where the internal representation of those security attributes is visible at external interfaces, also the internal representation of the security attributes needs to be described in the public documentation. The TOE design needs to describe how the security attributes are protected by the mechanisms of the TOE architecture.

The TOE design documentation needs to include a justification why the access control mechanisms cannot be bypassed.

Most operating systems for access to persistent storage objects perform access control when the object is “opened” and not for each access operation to the object. As long as the user/subject is able to maintain the “open” status for the object, the access operation may be performed for the access operation checked for during “open” even if the access has been revoked afterwards. This is acceptable as long as it is described in the TOE design, functional specification, or guidance.

Sometimes a single object can be accessed using different names or links to the object. The design needs to explain that the access control rules apply regardless how the object is addressed.

5.3.2.1.4.2 Evaluator Activities

The evaluator verifies that the principles of the implementation of the access control policies are consistent with the description of the policies in the Security Target, the functional specification and the guidance. The evaluator verifies that that the description of the storage and management of the security

attributes used in the access control policies is complete (with respect to the ones mentioned in the Security Target) and is consistent with the description in the Security Target and the guidance how they are used and managed. The best way to do this is by creating a table that maps each security attribute to its description in the Security Target, the functional specification, the design and the guidance and validating that those descriptions are consistent.

If objects can be addressed in different ways, the evaluator extracts those different ways from the TOE design, functional specification, and user guidance and determines if the TOE design provides sufficient information to ensure that regardless how the object is addressed, access control is enforced. Cases were the evaluator still is not certain may be addressed by additional test cases.

5.3.2.1.5 User Guidance (for Administrators as well as "Regular Users")

5.3.2.1.5.1 Expectations

The user guidance is expected to describe the different access control policies with their algorithms used to determine if access is allowed. The guidance also needs to explain the access control algorithm, allowing a user to understand what decision the algorithm will take based on the set of security attributes used in the rules of the algorithm.

The user guidance is expected to describe how the access control policy and the security attributes used by the access control rules can be managed, identifying the conditions that need to be satisfied to perform the individual management operations. Depending on how the developer has structured his public documentation, this information may be described together with the interfaces used for management, which is of course an acceptable way to provide this information.

The user guidance is expected to explain how to set up the policies securely and how a user responsible for managing access control or security attributes can query the current status of security attributes that are used in the access control rules. The guidance also needs to explain the access control algorithm, allowing a user to understand what decision the algorithm will take based on the set of security attributes used in the rules of the algorithm.

There may not always be the possibility for someone allowed to manage specific security attributes to query the status of other security attributes used in an access control policy. For example a user that is allowed to modify the access control list of objects he "owns", may not be allowed to query the list of members belonging to a group, although he is allowed to assign access rights to groups. This is not viewed as a security problem as long as this concept and how to use it securely is explained in the guidance.

5.3.2.1.5.2 Evaluator Activities

The evaluator verifies that the algorithms for the different access control policies are completely and correctly described in the user guidance.

The evaluator also verifies that for all security attributes that can be managed the user guidance describes:

- How they can be managed
- The rules that define when a user is allowed to perform the individual management operations
- The effect of the management operation on the access control policy behavior
- Potential side effects that may not be immediately obvious with warnings in cases those side effects may lead to security problems. An example would be a management operation that makes the object inaccessible to any user.
- (including the conditions that need to be satisfied to perform the query operation).

In addition the evaluator verifies that the guidance describes all the steps to initialize and configure each access control policy, including the steps to set default values for security attributes, assign the required privileges to perform management operations, and activate the access control policy.

The guidance is further required to explain situations where the TOE does not implement immediate revocation of access control related security attributes, providing guidance how to avoid situations where a user may access an object for a significant amount of time after the security attributes have been modified such that his access is revoked. For example the guidance could explain how to determine the users that currently have an active access path to the object together with possible actions an authorized administrator could take to force the access path to be closed.

The evaluator will use the guidance when configuring the access control policies, defining and modifying access rights and other security attributes used in the access control algorithms when defining the test cases he needs to perform.

5.3.2.1.6 Testing

5.3.2.1.6.1 Expectations

Test cases may either be provided by the developer to be executed by the evaluator or be developed by the evaluator.

The test cases are required to cover:

- All access control algorithms mentioned in the Security Target
- For each access control algorithm all paths through the algorithm (as defined in the Security Target), especially each leaf in the algorithm where the algorithm terminates with a “yes” or “no” decision
- A representative set of combinations of settings of the security attributes used in the access control algorithms

Test cases need to exist also for the management functions used to manage the security attributes used in the access control algorithms. Those test cases need to cover all security attributes, each with a representative set of values for the attribute. The test cases need to show:

- That the conditions for managing the security attributes are enforced (which includes test cases where the request for management is rejected)
- That the value of the security attribute has the effect described in the access control algorithm.

- That the values of security attributes can be queried (if the necessary conditions are satisfied)

Note: those test cases will overlap with test cases required for the assessment of some management SFRs. There is of course no need to execute those tests twice, but instead the test cases may be just mapped to both the SFRs for FDP_ACC/FDP_ACF and the management SFRs.

Additional test cases are required in cases where an object can be accessed using different ways. Test cases need to exist that demonstrate that access control is enforced for each possible way to access the object. Note that not all paths through the algorithm need to be tested for each possible way to access the object.

5.3.2.1.6.2 Evaluator Activities

The evaluator verifies that sufficient test cases have been provided (with their test results) showing that for each access control policy mentioned in the Security Target all paths through the access control algorithm are covered by at least one test case. He then maps the list of security attributes to test cases, showing that all security attributes are covered with a representative set of values. A representative set of values depends on the overall set of values for the security attribute and its expected effect on the access control policy.

For example an access control list is a security attribute where test cases need to exist for each possible type of access, but (of course) not for each possible user.

The evaluator also maps each management function to test cases, showing that all management functions are covered by test cases.

In most cases the developer will have significantly more test cases than required to show the coverage indicated above. When the evaluator has completed the mappings required in the description above using a subset of the test cases provided by the developer, there is no need for the evaluator to analyze the developer's test cases beyond this subset.

The evaluator will identify combinations of security attributes not found in the test cases he has analyzed and run a set of test using some of those combinations and validate that the results are consistent with the definition of the access control policy.

5.3.2.2 Assurance Activities for FDP_IFC.1 Subset Information Flow Control and FDP_IFF.1 Simple Security Attributes[44]

5.3.2.2.1 Background

An operating system compliant to the base OSPP is required to provide configurable functionality that allows to perform basic filtering on network traffic directed to the TOE as well as network traffic a subject generates to be sent to external IT entities. Filtering rules may be on layer 2 traffic, layer 3 traffic or both. At least the TOE needs to provide the possibility to define basic "matching" rules that allow an

[44] These activities apply to FDP_IFC.2(OSPP) and FDP_IFC.1(OSPP) in the Windows security target.

administrator that manages the filtering rules to prohibit traffic to and from specific unauthenticated external IT entities for layer 3 based on their IP address, TCP port number, UDP port number network protocol, and TCP header flags. For layer 2 an administrator needs to be able to define filtering rules based on MAC addresses and VLAN tags that allow or exclude traffic based on matching criteria for those attributes.

Related to those two SFRs is the SFR FMT_MSA.3(NI) which defines the conditions an administrator must meet to define the filtering rules.

5.3.2.2.2 TOE Summary Specification (TSS)

5.3.2.2.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to describe the type of filtering rules for network traffic the TOE implements with:

- The network protocol(s) for which the rules apply
- The network protocol data the filtering rules can be based upon
- The criteria that can be defines for the rule to “fire”
- The possible action(s) taken when the rule “fires”

The TSS (or public documentation pointed to by the TSS) also needs to describe the management interface used to define and/or activate the filtering rules

5.3.2.2.2.2 Evaluator Activities

The evaluator verifies that the network protocols, network protocol data, the criteria for the rules to “fire” and the possible action(s) as mentioned in the TSS are consistent with the definition in the SFRs FDP_IFC.1 and FDP_IFF.1, i. e. the criteria and rules defined in the SFRs can all be mapped to the description in the TSS or public documentation pointed to by the TSS. Note that the possibility for an administrator to define rules that match the capabilities defined in FDP_IFF.1 is verified in the assurance activities for FMT_MTD.1(NI).

5.3.2.2.3 Functional Specification

5.3.2.2.3.1 Expectations

The interfaces used for testing the effect of FDP_IFC.1 and FDP_IFF.1 are the external network interfaces, the interfaces a subject operating on the operating system can use to send and receive network traffic, and the interfaces an administrator can use to define and manage the filtering rules (which are analyzed and tested in the assurance activities for FMT_MTD.1(NI)). In order to verify the implementation of FDP_IFC.1 and FDP_IFF.1 the network interfaces need to be described with the specification of the network protocols they support (up to layer 3) and the interfaces a subject can use to send and receive network traffic need to be described with their parameter allowing to send and receive network data at a layer where the rules of the network information flow policy can be tested.

5.3.2.2.3.2 Evaluator Activities

The evaluator verifies that the interfaces are described to the extent that he can use them to test the effect of the filtering rules.

5.3.2.2.4 Architectural Design

5.3.2.2.4.1 Expectations

In the case not all effects of the filtering rules can not be tested directly at the TSFI, the architectural design needs to explain which TSF internal interfaces can be used for testing the effect of the filtering rules and how those interfaces can be used for testing.

5.3.2.2.4.2 Evaluator Activities

The evaluator verifies that in the case not all effects of the filtering rules can be tested at the TSFI, the sum of the TSFI described in the functional specification and the TSF internal interfaces are sufficient to test all effects of the filtering rules.

5.3.2.2.5 User Guidance (for Administrators as well a "Regular Users")

5.3.2.2.5.1 Expectations

There are no specific expectations on the user guidance.

5.3.2.2.5.2 Evaluator Activities

None.

5.3.2.2.6 Testing

5.3.2.2.6.1 Expectations

The developer is required to present test cases that test the following cases:

- Single filter rules based on each single security attribute showing that the defined action(s) are taken in each case the rule "fires" and are not taken if the rule does not "fire"
- A combination of two or more filter rules showing that the action(s) expected to be taken for the combination of filter rules are actually taken. Note that the Assurance Activities for FMT_MSA.1(NI) require that the evaluator verifies that for each possible combination of filter rules the developer documentation allows to identify unambiguously the action(s) taken by the TOE on packets inspected. The evaluator will take this specification from the developer's documentation to specify the expected result for the following cases:
  - A combination of filter rules that use different security attributes and define different actions
  - If possible, a combination of filter rules that use the same security attributes but define different actions
- All exceptions listed in FDP_IFF.1.4 and FDP_IFF.1.5

5.3.2.2.6.2 Evaluator Activities

The evaluator shall successively configure the TOE with different filter rules in accordance with the cases defined above. The evaluator then shall initiate network traffic to the TOE from one or more external IT entities and perform tests for each set of filter rules where traffic from the external IT entity to a subject in the TOE should be blocked and where traffic from the external IT entity to a subject in the TOE should be allowed and verify that the TOE operates in accordance with its specification. Similar the evaluator shall perform tests for network traffic from a subject in the TOE to an external IT entity using different rule sets in accordance with the cases defined above and verify that the TOE operates in accordance with its specification for network traffic from a subject within the TOE to an external IT entity. The evaluator may re-use test cases provided by the developer but should use those with modified rule sets and potentially modify those test cases to cover parameter combinations not addressed in the developer's test cases.

The test cases need to cover:

- All security attributes listed in FDP_IFF.1.3 and for each security attribute at least one test case for each possible action
- Rule sets that include multiple rules for different security attributes and test cases that test that the correct action is taken. Also in this case there needs to be at least one test case for each possible action

5.3.2.3 Assurance Activities for FDP_RIP.2 Residual Information Protection

5.3.2.3.1 Background

Residual information can potentially be present in a number of objects and resources when they are re-allocated to a different subject or user. The examples that need to be covered are:

- Residuals in persistent storage objects (file system objects) including object related TSF data (e. g. directory entries, object security attributes)
- Residuals in main memory objects
- Residuals in processor objects that can be read and written by untrusted subjects (e. g. general registers, floating point registers)

5.3.2.3.2 TOE Summary Specification (TSS)

5.3.2.3.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to identify the resources that may be subject to residuals and briefly describe for each of those resources the strategy implemented by the TOE to make information stored by a subject or user unavailable before the resource is made accessible to another user or subject. If the TOE needs specific initialization and/or configuration steps to enforce object re-use for all resources, this and the steps required need to be identified in the TSS (with pointers to additional public documentation were necessary).

5.3.2.3.2.2 Evaluator Activities

The evaluator verifies that at least all resources related to objects mentioned in the access control policy SFRs (including partial release of space occupied by the object), main memory and processor resources are addressed in the description of the object re-use related description in the TSS and that the description explains sufficiently the strategy used to ensure that information about the previous content of the resource is made unavailable.

5.3.2.3.3 Functional Specification

5.3.2.3.3.1 Expectations

There is no direct TSF interface for object re-use. Instead the interfaces where the effect of object re-use functionality can be observed need to be identified.

5.3.2.3.3.2 Evaluator Activities

For each resource identified as one that requires object re-use the evaluator identifies from the TSFI provided by the developer:

- Interfaces that can be used to release a resource
- Interfaces that can be used to re-allocate a resource
- Interfaces that can be used read the content of a resource after re-allocation

5.3.2.3.4 Architectural Design

5.3.2.3.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.2.3.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how object re-use is performed

5.3.2.3.5 User Guidance (for Administrators as well as “Regular Users”)

5.3.2.3.5.1 Expectations

There is no expectation on specific guidance related to object re-use unless there are management functions that can be used to specify details how object re-use is performed.

If this is the case and if the TOE allows for configuration where the object re-use requirement is not satisfied, the guidance needs to describe clearly the configuration steps that have to be taken to ensure that the object re-use functionality is active.

5.3.2.3.5.2 Evaluator Activities

Only in the case where the TOE needs to be specifically initialized and configured to provide object re-use capabilities for all resources the evaluator will follow the description in the TSS to identify the

interfaces and parameter required to initialize and/or configure the TOE to ensure that the object re-use functionality is active. This is required before using the TOE for testing of the object re-use functionality.

5.3.2.3.6 Testing

5.3.2.3.6.1 Expectations

Testing is expected to cover all interfaces that can be used to allocate resources that need to be subject to object re-use and then analyze if the resource potentially contains information from its previous use by a different subject. Testing is expected to cover all attempts to obtain information left from the previous use of the resource.

Testing needs to cover at least the following cases:

- Attempts to read from persistent storage objects from areas that have not been written to since the object was created.
- Reading from main storage areas that have been obtained using dynamic storage allocation but not yet written to by the subject.
- Reading user-accessible processor register after a content switch.
- Reading from other resources listed as being subject to object re-use and allocated to the subject before information has been placed in those resources by the subject.

5.3.2.3.6.2 Evaluator Activities

The evaluator verifies that all resources for which object re-use has been defined are covered by testing showing that the no access to previous information is possible. The evaluator verifies that all TSFI where newly allocated resourced can be read are included in the test suite and that in no case access to the previous information is possible.

5.3.2.4 Assurance Activities for FMT_MSA.1 Management of Object Security Attributes[45]

5.3.2.4.1 Background

Object security attributes include the all object security attributes used for the enforcement of the access control policy.

5.3.2.4.2 TOE Summary Specification (TSS)

5.3.2.4.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to list the object security attributes used for enforcing access control, management, and audit policies together with the rules that define when they can be managed.

[45] These activities apply to FMT_MSA.1(DAC) in the Windows security target.

5.3.2.4.2.2 Evaluator Activities

The evaluator compares the list of object security attributes mentioned in the SFRs in the rules for access control, object management and object related audit policies with the ones listed in the TSS as being manageable. For object security attributes that are mentioned in the TSS as being manageable but which are not used in any SFR, the evaluator needs to clarify their purpose. For object security attributes mentioned in SFRs but not defined as manageable in the TSS, the evaluator needs to verify that those object security attributes cannot be managed by the object owner or any other user.

5.3.2.4.3 Functional Specification

5.3.2.4.3.1 Expectations

The interfaces used to manage object security attributes need to be identified for all object security attributes listed in the TSS as being manageable.

5.3.2.4.3.2 Evaluator Activities

The evaluator verifies that for all object security attributes listed as manageable the management interfaces are identified and described and that all management actions listed in FMT_MSA.1 can be performed using those interfaces

5.3.2.4.4 Architectural Design

5.3.2.4.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.2.4.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the object security attributes can be managed.

5.3.2.4.5 User Guidance (for Administrators as well as “Regular Users”)

5.3.2.4.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to manage the object security attributes.

5.3.2.4.5.2 Evaluator Activities

The evaluator verifies that the conditions defined in the guidance for managing object security attributes match the conditions defined in FMT_MSA.1.

5.3.2.4.6 Testing

5.3.2.4.6.1 Expectations

The developer is expected to test the interfaces for the management of object security attributes as part of his functional testing. This is often done in conjunction with the testing of the SFRs where the object

security attributes are used like in testing of the access control policy where those interfaces are used to set the object security attributes for testing different aspects of the access control algorithm.

5.3.2.4.6.2 Evaluator Activities

The evaluator verifies that testing includes all interfaces defined for the management of object security attributes and all object security attributes, covering a sufficient set of values for the individual object security attributes. The evaluator verifies that the effect of the settings of the object security attributes are tested (often as part of the testing of the access control algorithm).

5.3.2.5 Assurance Activities for FMT_MSA.3(DAC) Static Attribute Initialization

5.3.2.5.1 Background

The default values for all security attributes used to enforce the discretionary access control policies need to be defined such that by default access is restricted to a defined set of users (usually the owner) when a new object is created. This applies to object security attributes which need to be initialized to such restrictive default values when a new object is created as well as to other security attributes used in the access control policy. Note that some object security attributes may be inherited from another object (as defined by FMT_MSA.4) and the default values in those cases are the inherited values. The rules how those inherited values are assigned are defined in FMT_MSA.4 and analyzed in the assurance activities for FMT_MSA.4.

5.3.2.5.2 TOE Summary Specification (TSS)

5.3.2.5.2.1 Expectations

For all object security attributes used in the discretionary access control policies the TSS (or public documentation pointed to by the TSS) needs to describe how they are initialized when a new object is created and how their initial default values are defined.

The TSS also needs to describe if and how those default values can be managed, what the interfaces used for those management activities are and which conditions need to be satisfied to perform those management activities.

5.3.2.5.2.2 Evaluator Activities

The evaluator verifies that the algorithm for initializing the security attributes used in the discretionary access control policies is defined for all security attributes. The evaluator verifies that the default values restrict access to only a defined set of users (e. g. the owner and the administrators). The evaluator verifies that the default values that can be managed and the conditions that need to be satisfied to perform those management activities are consistent with the specification in FMT_MSA.3(DAC).

5.3.2.5.3 Functional Specification

5.3.2.5.3.1 Expectations

The TSS identifies the interfaces that can be used to manage the default values for security attributes used to enforce the discretionary access control policies and points to the specification of those interfaces.

5.3.2.5.3.2 Evaluator Activities

The evaluator verifies that the management activities mentioned in the SFR and in the TSS can be performed using those interfaces and that the description is sufficient to use those interfaces in testing.

5.3.2.5.4 Architectural Design

5.3.2.5.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.2.5.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the default values of the security attributes used to enforce the discretionary access control policies can be managed.

5.3.2.5.5 User Guidance (for Administrators as well as “Regular Users”)

5.3.2.5.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to manage the default values for security attributes used to enforce the discretionary access control policies.

5.3.2.5.5.2 Evaluator Activities

The evaluator verifies that the conditions defined in the guidance for managing the default values of the security attributes match the conditions defined in FMT_MSA.3(DAC).

5.3.2.5.6 Testing

5.3.2.5.6.1 Expectations

The developer is expected to test the interfaces for the management of the default values for security attributes used to enforce the discretionary access control policies as part of his functional testing. This is often done in conjunction with the testing of the SFRs for the access control policies where those interfaces are used to set the default values for security attributes for testing different aspects of the access control algorithm.

5.3.2.5.6.2 Evaluator Activities

The evaluator verifies that testing includes all interfaces defined for the management of the default values for security attributes used to enforce the discretionary access control policies, covering a sufficient set of values for the individual security attributes. The evaluator verifies that the effect of the settings of the security attributes are tested (often as part of the testing of the access control algorithms).

5.3.2.6 Assurance Activities for FMT_MSA.3(NI) Static Attribute Initialization[46]

5.3.2.6.1 Background

The default values for all security attributes used to enforce the Network Information Flow Policy need to be defined by some set of default rules or no rules at all.

5.3.2.6.2 TOE Summary Specification (TSS)

5.3.2.6.2.1 Expectations

For all security attributes used in the Network Information Flow Policy the TSS (or public documentation pointed to by the TSS) needs to describe how they are initialized and how their initial default values are defined. The TSS also needs to describe if and how those default values can be managed, what the interfaces used for those management activities are and which conditions need to be satisfied to perform those management activities.

Note: most likely those management actions overlap significantly with those for FMT_MTD.1(NI) and will be covered by the assurance activities defined for FMT_MTD.1(NI).

5.3.2.6.2.2 Evaluator Activities

The evaluator verifies that the algorithm for initializing the security attributes used in the Network Information Flow Policy is defined for all security attributes. The evaluator verifies that the default values satisfy the specification in FMT_MSA.3(NI). The evaluator verifies that the default values that can be managed and the conditions that need to be satisfied to perform those management activities are consistent with the specification in FMT_MSA.3(NI).

5.3.2.6.3 Functional Specification

5.3.2.6.3.1 Expectations

The TSS identifies the interfaces that can be used to manage the default values for security attributes used to enforce the Network Information Flow Policy and points to the specification of those interfaces.

5.3.2.6.3.2 Evaluator Activities

The evaluator verifies that the management activities mentioned in the SFR and in the TSS can be performed using those interfaces and that the description is sufficient to use those interfaces in testing.

5.3.2.6.4 Architectural Design

5.3.2.6.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

[46] These activities apply to FMT_MSA.3(OSPP) in the Windows security target.

5.3.2.6.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the default values of the security attributes used to enforce the Network Information Flow Policy can be managed.

5.3.2.6.5 User Guidance (for Administrators as well as "Regular Users")

5.3.2.6.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to manage the default values for security attributes used to enforce the Network Information Flow Policy.

5.3.2.6.5.2 Evaluator Activities

The evaluator verifies that the conditions defined in the guidance for managing the default values for the security attributes match the conditions defined in FMT_MSA.3(NI).

5.3.2.6.6 Testing

5.3.2.6.6.1 Expectations

The developer is expected to test the interfaces for the management of the default values for security attributes used to enforce the Network Information Flow Policy as part of his functional testing. This is often done in conjunction with the testing of the SFRs for the Network Information Flow Policy where those interfaces are used to set the default values for security attributes for testing different aspects of the Network Information Flow Policy.

5.3.2.6.6.2 Evaluator Activities

The evaluator verifies that testing includes all interfaces defined for the management of the default values for security attributes used to enforce the Network Information Flow Policy, covering a sufficient set of values for the individual security attributes. The evaluator verifies that the effect of the settings of the security attributes are tested (often as part of the testing of the filtering rules).

5.3.2.7 Assurance Activities for FMT_MSA.4 Security Attribute Value Inheritance

5.3.2.7.1 Background

When creating a new object covered by a discretionary access control policy, the new object may inherit security attributes from an already existing object. This is often the case when objects are part of a hierarchical structure where new objects inherit security attributes from the next higher level of the hierarchy. Inheritance is not limited to hierarchical object structure but may also the cause where new objects become a member of some group and then inherit some security attributes from the group. Inheritance is a special case for the initialization of object security attributes for new objects.

5.3.2.7.2 TOE Summary Specification (TSS)

5.3.2.7.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to identify the object security attributes that are inherited when a new object is created, needs to describe what the rules for inheritance are and from where they are inherited.

5.3.2.7.2.2 Evaluator Activities

The evaluator verifies that the algorithm for inheriting security attributes used in the discretionary access control policies is defined for all security attributes that are inherited.

5.3.2.7.3 Functional Specification

5.3.2.7.3.1 Expectations

There are usually no interfaces related to this SFR except for the case where the inheritance rules can be managed. Inheritance is automatically performed when a new object is created. If the inheritance rules can be managed, the ST needs to define a SFR in the FMT_MTD family that describe the conditions that must be met by a user in order to be allowed to perform this management activity.

5.3.2.7.3.2 Evaluator Activities

None except when the inheritance rules can be managed. In this case the interfaces for the management of the inheritance rules need to be analyzed that they allow for the management actions defined.

5.3.2.7.4 Architectural Design

5.3.2.7.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.2.7.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the values of object security attributes are inherited.

5.3.2.7.5 User Guidance (for Administrators as well as “Regular Users”)

5.3.2.7.5.1 Expectations

None.

5.3.2.7.5.2 Evaluator Activities

None.

5.3.2.7.6 Testing

5.3.2.7.6.1 Expectations

The developer is expected to test the inheritance rules by creating new objects and then verify that the values of the object security attributes that are supposed to be inherited are correctly inherited.

5.3.2.7.6.2 Evaluator Activities

The evaluator verifies that testing covers all object security attributes that can be inherited with different values for those attributes.

5.3.2.8 Assurance Activities for FMT_MTD.1(NI) Management of TSF Data: Network Filtering Rules[47]

5.3.2.8.1 Background

Network data filtering rules need to be manageable, allowing a properly authorized administrator to define, query, modify, and delete the network data filtering rules. A TOE may well distinguish between the authority for the different operations, allowing for example specific users to query the filtering rules without giving them the right to modify or delete them. If such differentiations exist for different management actions, this needs to be expressed in the SFR.

5.3.2.8.2 TOE Summary Specification (TSS)

5.3.2.8.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to explain how network data filtering rules can be defined and how they can be viewed, activated, modified, and deleted. The TSS also needs to identify the interfaces that can be used for those activities and the conditions a user needs to meet when performing any of those management activities.

5.3.2.8.2.2 Evaluator Activities

The evaluator verifies that the management functions and interfaces described in the TSS allow for all the management actions defined in FMT_MTD.1(NI) and that the conditions a user needs to meet to perform those activities is consistent with the description in the SFR.

5.3.2.8.3 Functional Specification

5.3.2.8.3.1 Expectations

The functional specification needs to define the interfaces used for the management of the network data filtering rules, defining the syntax for the management of those rules, the exact semantic of each filtering rule, the functions to define, activate, query, modify, and delete network data filtering rules. Also the conditions a user must meet to perform each of the management actions need to be defined (either in the functional specification or in the guidance documentation).

[47] These activities apply to FMT_MTD.1(OSPP) in the Windows security target.

5.3.2.8.3.2 Evaluator Activities

The evaluator verifies that the description of the interfaces is sufficient to perform all the management activities defined in FMT_MTD.1(NI) allowing the definition of filtering rules that cover all aspects defined in FDP_IFC.1 and FDP_IFF.1.

5.3.2.8.4 Architectural Design

5.3.2.8.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.2.8.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the network data filtering rules can be managed.

5.3.2.8.5 User Guidance (for Administrators as well as “Regular Users”)

5.3.2.8.5.1 Expectations

Unless this is already covered in the assessment of the functional specification the guidance is expected to describe the conditions a user must satisfy to perform the different management activities for the network data filtering rules. In addition the guidance is expected to explain the semantics of the different rules and define how potential conflicts are addressed in a set of rules.

5.3.2.8.5.2 Evaluator Activities

The evaluator verifies that the guidance describe the semantics of the network data filtering rules including the aspect of potentially conflicting rules in a rule set allowing the evaluator to determine for each set of rules he defines to determine the expected effect on the network traffic.

5.3.2.8.6 Testing

5.3.2.8.6.1 Expectations

Testing of FMT_MTD.1(NI) is expected to be performed in conjunction with the testing defined for FDP_IFC.1 and FDP_IFF.1. The management interfaces are used to define the set of network filtering rules used for the testing of the Network Information Flow Policy.

In addition the developer is expected to test that the management interfaces enforce the conditions a user must satisfy to perform the different management activities. The test cases shall cover all branches of the algorithm that determines a user’s right to perform the management action, similar to testing an access control algorithm

5.3.2.8.6.2 Evaluator Activities

The evaluator shall verify that the test cases cover all combinations of management activities and all branches of the algorithm that determines a user’s right to perform the management action.

5.3.2.9 Assurance Activities for FMT_REV.1(OBJ) Revocation: Object Security Attributes

5.3.2.9.1 Background

Revocation of object security attributes is a special case of the management of object security attributes as addressed in FMT_MSA.1. Therefore the revocation of object security attributes is handled very similar to the assessment of FMT_MSA.1 and should be performed in combination with the assurance activities for FMT_MSA.1.

5.3.2.9.2 TOE Summary Specification (TSS)

5.3.2.9.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to list the object security attributes that can be revoked and needs to explain how revocation can be performed and what the conditions are that a user must satisfy to perform the revocation of an object security attribute. If those conditions are different for different objects security attributes, those differences need to be defined in the SFR.

5.3.2.9.2.2 Evaluator Activities

The evaluator compares the list of object security attributes mentioned in the SFRs with the ones listed in the TSS as being revocable and ensures that those lists are identical.

5.3.2.9.3 Functional Specification

5.3.2.9.3.1 Expectations

The interfaces used to revoke object security attributes need to be identified for all object security attributes listed in the TSS as being revocable.

5.3.2.9.3.2 Evaluator Activities

The evaluator verifies that for all object security attributes listed as revocable the management interfaces are identified and described and that they allow for the revocation of the object security attributes.

5.3.2.9.4 Architectural Design

5.3.2.9.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.2.9.4.2 Evaluator Activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the object security attributes can be revoked.

5.3.2.9.5 User Guidance (for Administrators as well as "Regular Users")

5.3.2.9.5.1 Expectations

The guidance (or the functional specification) needs to explain the conditions that must be met to allow a user to revoke the object security attributes.

5.3.2.9.5.2 Evaluator Activities

The evaluator verifies that the conditions defined in the guidance for revoking object security attributes match the conditions defined in FMT_REV.1(OBJ).

5.3.2.9.6 Testing

5.3.2.9.6.1 Expectations

The developer is expected to test the interfaces for the revocation of object security attributes as part of his functional testing. This is often done in conjunction with the testing of the SFRs where the object security attributes are used like in testing of the access control policy where those interfaces are used to revoke the object security attributes for testing different aspects of the access control algorithm.

5.3.2.9.6.2 Evaluator Activities

The evaluator verifies that testing includes all interfaces defined for the revocation of object security attributes and all object security attributes. The evaluator verifies that the effect of the revocation of the object security attributes are tested (often as part of the testing of the access control algorithm).

5.3.3 Assurance Activities for Identification and Authentication

5.3.3.1 Assurance Activities for FIA_AFL.1: Authentication Failure Handling

5.3.3.1.1 TOE Summary Specification (TSS)

The evaluator will find the details regarding how the TOE is expected to operate when handling authentication failures in the Security Functional Description provided in the TSS. The discussion pertaining to the I&A functionality present in the TOE will describe all the methods the TOE employs to perform authentication, including what happens when a failed attempt occurs. The evaluator examines the description to ensure that each authentication method identified in this SFR is fully described and it is clear what happens when the failed attempts reach either the met or surpassed threshold. At the very least, the password-based authentication must be covered. There may be instances where the TOE behaves differently for administrators and untrusted users, and this could be either captured in the SFR by refining the requirement, iterating the requirement, or attempting to capture it in the authentication events or list of actions assignments.

5.3.3.1.2 Functional Specification

With an understanding of how the I&A functions are intended to operate, the evaluator turns to the interface specification to see what interfaces support I&A. The developer is required to have provided a mapping of the interfaces to the I&A functions, including those that map to FIA_UAU.5, and the evaluator ensures that the description of the methods of I&A presented in the Security Functional Description are included in the provided interfaces and vice versa. There may be interfaces for

authentication that are not subject to the failure handling, and this is acceptable, as long as it is consistent with the authentication methods and authentication events listed in this SFR. For example, there may be interfaces to authenticate to the TOE that employ a smartcard, but authentication failures resulting in the use of a smartcard are not one of the authentication methods considered.

However, if during their analysis an evaluator discovers that an advertised interface whose description indicates an action will be taken relating to failed authentication attempts, and it employs the authentication method in the requirement, the evaluator must work with the developer to resolve the discrepancy. On the other hand, if the evaluator discovers an interface that employs an authentication method that is not specified in the requirement, no further action is required, since it is outside the scope of the product's claimed security functionality.

5.3.3.1.3 Operational User Guidance

The evaluator determines that the guidance for managing the threshold, and responding to potential actions required on their part, are consistent with the statement in the SFR.

5.3.3.1.4 Testing

The number of tests used to verify the TOE's behavior will, of course, depend upon the number of authentication methods that are subject to this requirement, the interfaces that invoke those methods, as well as the actions to be taken. It is suggested that the evaluator develop a matrix that contains the authentication methods to be considered, the potential authentication events that may be associated with each of the methods, and the actions that will be taken. Again, there may be various actions that are taken even given the same authentication method, and it is critical that all combinations are addressed in the testing activities. For example, when an untrusted user fails to enter the correct local password three consecutive times, their account may be disabled/locked until an administrator action is taken. On the other hand, when an administrative user fails to enter the correct local password three consecutive times their account may be disabled for 30 seconds. So the nature and number of the tests will vary due to the complexity of the TOE's failure handling mechanism.

Test 1: The evaluator, with the appropriate privilege, shall follow the operational guidance to configure the number of unsuccessful authentication attempts for each authentication method [password-based is minimally required; others may exist depending on the assignment.

Test 2: The evaluator shall attempt to authenticate successfully using the authentication method under test. After successfully authenticating, the evaluator will attempt X number of failed authentication attempts (number to be determined according to the "rules" specified in the list of authentication events. Upon satisfying the number of failed attempts, the evaluator shall observe that the TOE electrocutes the user with sufficient amperage to cause much harm.

Test 3: The evaluator shall attempt to modify the variable that enforces the limit on unsuccessful authentication attempts as an untrusted user. They shall be unsuccessful in modifying the controlling variable.

5.3.3.2 Assurance Activities for FIA_ATD.1: User Attribute Definition[48]

5.3.3.2.1 TOE Summary Specification (TSS)

The evaluator will find the user security attributes maintained for each defined user enumerated in the TSS. The list of user security attributes may differ from those identified in FIA_ATD.1 and also serve to extend the minimum set in the context of additional user security attributes assigned in FIA_ATD.1. When the list of user security attributes identified in the TSS differs from those in FIA_ATD.1, the TSS must provide a clear mapping showing the association and coverage of the required user security attributes. Any non-security related attributes associated with users need not be identified in the TSS.

5.3.3.2.2 Interface Specification

Given the user security attributes identified in the TSS, the evaluator turns to the interface specification to see what interfaces support FIA_ATD.1. The developer is required to have provided a mapping of the interfaces to the I&A functions, including those that map to FIA_ATD.1, and the evaluator ensures that the description of the methods available to create, view, modify, and delete the security attributes identified in the TSS are presented in the Security Functional Description.

Examples of applicable interfaces include those used to create and delete users, as well as any interfaces available to modify any of the security attributes of existing users (e.g., add/remove groups, change password).

However, if during their analysis an evaluator discovers that an advertised interface whose description indicates access to create or modify security attributes has not been mapped to FIA_ATD.1, the evaluator must work with the developer to resolve the discrepancy. On the other hand, if the evaluator discovers an interface that manipulates attributes not identified in FIA_ATD.1 (i.e., not security related), no further action is required, since it is outside the scope of the product's claimed security functionality.

5.3.3.2.3 Operational User Guidance

The evaluator determines that the administrative guidance for creating, viewing, modifying, and deleting user security attributes, in whole (e.g., create/delete users) or in part (e.g., change password), are consistent with FIA_ATD.1. The evaluator should, at a minimum, find instructions for creating and deleting users. Additional instructions may be available to manipulate one or more of the user security attributes individually and should be identified where available.

The description of the interface used to create or otherwise initially define users in the administrative guidance should serve to identify each of the required user attributes assignable upon creation. The description of any interface used to manipulate individual security attributes should clearly identify the applicable attribute(s).

5.3.3.2.4 Testing

See FMT_MTD.1(IAU) where the available interfaces are tested in conjunction with applicable restrictions.

[48] These activities apply to FIA_ATD.1(USR) in the Windows security target.

5.3.3.3 Assurance Activities for FIA_UAU.1(RITE): Timing of Authentication

5.3.3.3.1 TOE Summary Specification (TSS)

The evaluators shall examine the TSS to determine that it identifies the information flows that both support remote IT authentication as well as those that might be allowed prior to the remote IT entity being authenticated. The information in the TSS pertaining to how the FDP_IFC/FDP_IFF requirements are implemented will be a key part of this information, in that it will detail what might be allowed by the mechanism that is implemented to meet those requirements. When the TOE is configured for operational use the allowed protocols will be determined by the configuration of the parameters supported by functionality implementing FDP_IFC/FDP_IFF, so it may not be possible to provide an itemized list of what is and is not allowed prior to remote IT entity authentication. However, the evaluator shall determine that the information provided in the TSS allows a reader to understand the relationship between the configuration mechanisms supporting the FDP_IFC/FDP_IFF requirements and the resultant capabilities and functions available to remote IT entities prior to authentication.

5.3.3.3.2 Interface Specification

The interfaces used by remote IT entities are covered by the assurance activities for FDP_IFC.1, FDP_IFF.1, and FTP_ITC.1.

5.3.3.3.3 Operational User Guidance

The Operational Guidance should contain information describing the relationship between configuration the rules under which the TOE will allow information from remote IT entities and the implications of allowing flows that do not require endpoints to be authenticated. It should be possible for the administrator to determine—based on the guidance provided—what processing will be performed by the TOE (in terms of the service being allowed; for instance, allowing ICMP to pass will result in remote entities being able to “ping” the TOE without being authenticated) in response to the configuration of the rules implemented to meet the FDP_IFC/FDP_IFF requirements.

The administrative guidance will also cover the configuration of the TOE to support remote authentication; The evaluator shall examine the operational guidance to determine that any necessary preparatory steps (e.g., establishing credential material such as preshared keys, tunnels, certificates, etc.) to authentication are described. For each supported the authentication method, the evaluator shall ensure the operational guidance provides clear instructions for successfully performing the authentication. Some of all of these configuration activities are also addressed in the assurance activity for FTP_ITC.1.

5.3.3.3.4 Testing

The evaluator shall perform the following test for each remote authentication method supported:

Test 1: The evaluator shall use the operational guidance to configure the appropriate credential supported for the login method. For that credential/login method, the evaluator shall show that providing correct authentication-related information results in the ability to access the system, while providing incorrect information results in denial of access.

Tests for this capability are also addressed in the test activities for FDP_IFC.1, FDP_IFF.1, and FTP_ITC.1.

5.3.3.4 Assurance Activities for FIA_UAU.1(HU): Timing of Authentication[49]

5.3.3.4.1 TOE Summary Specification (TSS)

If the TOE implements a protocol used for remote authentication of users that provides a super-set of RFC-specified functionality—or if the protocol is not specified in an RFC or other published document—the TSS describes the portions of the protocol that are implemented that occur prior to the user being authenticated. For each action listed in the assignment that is allowed before a user logs on locally to the TOE, the TSS shall describe the functionality being provided by the TOE.

The evaluator shall examine the TSS to determine that it describes the logon process for each logon method (local, remote (HTTPS, SSH, etc.)) supported for the product. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a "successful authentication".

5.3.3.4.2 Interface Specification

The evaluator shall identify the TSFI used to authenticate to the TOE, both remotely and locally. The evaluator shall compare these interfaces to the information provided in the TSS, and determine that if the TSS describes an authorization method for a remote user (IT entity or human) or a local user, then there is an interface that corresponds to this method. If services (over and above those covered by the FDP_IFC/FDP_IFF requirements) are listed in the TSS as being available prior to user authorization, the evaluator ensures that the interfaces to these services are identified in the interface specification.

5.3.3.4.3 Operational User Guidance

For remote users, the operational guidance shall contain information pertaining to the configuration of the TOE to allow a user to authenticate remotely. This may involve establishing the credentials to be used by the user, as well as configuration of the TOE credentials depending on the protocol. For local authentication, the

5.3.3.4.4 Testing

The evaluator shall perform the following test for each local and remote authentication method supported:

Test 1: The evaluator shall use the operational guidance to configure the appropriate credential supported for the login method. For that credential/login method, the evaluator shall show that providing correct authentication-related information results in the ability to access the system, while providing incorrect information results in denial of access.

Test 2: For each specified service available to local users prior to authentication, the evaluation shall ensure that the service can be invoked without authentication being required.

[49] These activities apply to FIA_UAU.1(OS) in the Windows security target.

5.3.3.5 Assurance Activities for FIA_UAU.5: Multiple Authentication Mechanisms

5.3.3.5.1 TOE Summary Specification (TSS)

The evaluator will find the available authentication mechanisms identified in the TSS. At a minimum, the TSS will describe a username/password-based authentication mechanism as well as any other mechanisms that are assigned in FIA_UAU.5.

The evaluator will also find that the username/password-based mechanism description explains the behavior of the TOE when a password is expired in a manner consistent with that selected in FIA_UAU.5.2c.

If multiple authentication mechanisms are identified, the evaluator will also find that the description explains rules associated with the additional authentication mechanisms, including rules for determining which authentication mechanism will be used in each case.

5.3.3.5.2 Interface Specification

Given the list of available authentication mechanisms in the TSS, the evaluator turns to the interface specification to see what interfaces support FIA_UAU.5. The developer is required to have provided a mapping of the interfaces to the I&A functions, including those that map to FIA_UAU.5, and the evaluator ensures that the description of the methods available to authenticate user identities, along with rules associated with those methods, are presented in the Security Functional Description. The descriptions should address selecting authentication methods and results of both success (e.g., create a new process) and failure (e.g., password expired) conditions.

Note that when multiple authentication methods are available, it is possible that only some of those methods are applicable to specific interfaces and that should be clearly identified.

However, if during their analysis an evaluator discovers that an advertised interface whose description indicates authentication methods that have not been mapped to FIA_UAU.5, the evaluator must work with the developer to resolve the discrepancy.

Unlike some other functions, it is generally not acceptable that available authentication methods are ignored in the context of evaluation.

5.3.3.5.3 Operational User Guidance

The evaluator determines that the administrative guidance for functions requiring authentication is consistent with FIA_UAU.5. The evaluator should, at a minimum, find instructions for authenticating during initial login. Additional instructions may be available for additional functions requiring authentication such as changing passwords, activating privileges, etc.

If the TOE support for multiple authentication mechanisms is configurable (e.g., to enable or set up an authentication mechanism), the guidance may also have instructions for enabling/disabling mechanisms, configuring mechanisms, defining rules for the use of mechanisms, etc. The possibilities are extensive, so the activities here may need to be augmented during an evaluation to address additional variations.

5.3.3.5.4 Testing

For the most part the testing activities for FIA_UAU.5 should be accomplished in conjunction with those of FIA_UAU.1. While testing for FIA_UAU.1 necessarily addresses both successful and unsuccessfully attempts to authenticate, the evaluator shall further ensure that corresponding successful and unsuccessful attempts are made in the context of each available authentication mechanism. As such, the evaluator will need to configure all possible authentication mechanisms during the course of testing to ensure that the mechanism is invoked and can result in both successful and unsuccessful cases for each applicable interface.

At a minimum, the evaluator shall also test for each interface supporting username/password-based authentication that the authentication attempt will fail when the user password is expired. Presumably, the evaluator would have already tested that authentication attempts succeed when the password is not expired per the testing described above.

Given the possibility of assigning additional authentication mechanisms, this assurance activity may need to be augmented during an evaluation to address additional possibilities.

5.3.3.6 Assurance Activities for FIA_UAU.7: Protected Authentication Feedback

5.3.3.6.1 TOE Summary Specification (TSS)

For each authentication method where the TOE is in control of the feedback provided to the user, the TSS indicates that the feedback provided is obscured, and how it is obscured (not provided, masked, etc.). Each authentication method must be explicitly covered, and include not only login methods, but methods that require “re-authentication” such as changing a password, for example.

5.3.3.6.2 Interface Specification

This information is covered by the specification of interfaces used for authentication function.

5.3.3.6.3 Operational User Guidance

No additional information is required specific to this functionality.

5.3.3.6.4 Testing

The evaluator shall perform the following test for each method of local authentication described by the TSS:

Test 1: The evaluator shall locally authenticate to the TOE. While making this attempt, the evaluator shall verify that at most obscured feedback is provided while entering the authentication information.

5.3.3.7 Assurance Activities for FIA_UID.1 Timing of Identification

See FIA_UAU.1

5.3.3.8 Assurance Activities for FIA_USB.1 User-Subject Binding[50]

5.3.3.8.1 TOE Summary Specification (TSS)

The evaluator will find the user security attributes associated with each subject enumerated in the TSS. The list of user security attributes may differ from those identified in FIA_USB.1 and also serve to extend the minimum set in the context of additional user security attributes assigned in FIA_USB.1. When the list of user security attributes identified in the TSS differs from those in FIA_USB.1, the TSS must provide a clear mapping showing the association and coverage of the required user security attributes. Any non-security related attributes associated with subject need not be identified in the TSS.

While FIA_USB.1 supports assignment of user security attributes related to access control decisions, security management restrictions, and auditing, such attributes only need be identified in the SFR if they extend the minimum set of user security attributes (user identity, groups, and roles). Regardless, the evaluator will find that the TSS describes the association of all of the identified user security attributes in the context of other SFRs relate to access control, security management restrictions, and auditing. In other words, the use of each of the required user security attributes will be described in the TSS in association with at least one security function.

The Evaluator will find a description of how user security attributes are assigned to subjects. The description will describe how the user security attributes are initially assigned to a new subject, whether and how user security attributes can be changed, and how any additional security attributes might be associated with a subject. The description will serve to define all relationships between the user security attributes identified in FIA_ATD.1 and the security attributed identified in FIA_USB.1. The definition will also define all rules involved in the initial assignment and changes to security attributes associated with each subject.

If there are multiple types of subjects, potentially with different security attributes, the TSS will describe each case accordingly.

5.3.3.8.2 Interface Specification

Given the user security attributes identified in the TSS, the evaluator turns to the interface specification to see what interfaces support FIA_USB.1. The developer is required to have provided a mapping of the interfaces to the I&A functions, including those that map to FIA_USB.1, and the evaluator ensures that the description of the methods available to create subjects and modify security attributes associated with subjects are presented in the Security Functional Description.

Note that it is possible that interfaces may be indirect (e.g., a process created as a result of a user login) or direct (e.g., fork a new process), but they need to be identified and described in either case.

Note that it is also possible that security attributes associated with subjects cannot be changed, in which case no applicable interfaces should be identified or mapped.

[50] These activities apply to FIA_USB.1(USR) in the Windows security target.

The evaluator shall ensure that for each identified interface the rules for initial security attribute assignment and subsequent modifications, described in the TSS, are also described in the Security Functional Description and are consistent with the TSS.

Examples of applicable interfaces include those used to login, fork a process, as well as any interfaces available to modify any of the security attributes of existing subjects (e.g., enable/disable a privilege, change real or effective user or group identifiers).

However, if during their analysis an evaluator discovers that an advertised interface whose description indicates user-subject binding functions has not been mapped to FIA_USB.1, the evaluator must work with the developer to resolve the discrepancy. On the other hand, if the evaluator discovers an interface that manipulates subject security attributes not identified in FIA_USB.1 (i.e., not security related), no further action is required, since it is outside the scope of the product's claimed security functionality.

5.3.3.8.3 Operational User Guidance

The evaluator determines that the administrative guidance for creating subjects and changing security attributes associated with subjects are consistent with FIA_USB.2. The evaluator should, at a minimum, find instructions for logging in (to create a user process). Additional instructions may be available to manipulate one or more of the security attributes of subjects and should be identified where available.

The description of any interface used to manipulate security attributes of subjects should clearly identify the applicable attribute(s).

5.3.3.8.4 Testing

The number of tests used to verify the TOE's behavior will, of course, depend upon the number of user security attributes, the interfaces that provide access to them, and the complexity of associated restrictions. It is suggested that the evaluator develop a matrix that associates the user security attributes with interfaces available to initially assign and subsequently modify them. Note that in some cases user security attributes might be addressed collectively when an interface operates on a group of attributes simultaneously such as may be the case with functions like the UNIX 'setuid'.

The matrix should be further developed with mappings to specific rules, resulting in triples of user attribute(s), interface, and rules. Note that rules should be generally classified into two types: behavioral and restrictions. Behavioral rules serve to describe how assignment or changes occur but do not serve to limit, for example, which users or roles can perform the operation. Restrictive rules serve to describe limits for assignments and changes, such as the range of possible attributes or the roles that can make a change.

Given a list of attribute(s)/interface/rule triples, the evaluator shall perform the following tests in each case of a rule that is restrictive:

1. Perform the identified operation using instructions in the administrative guidance in order to assign or modify the identified security attribute(s) with the minimum necessary conditions to satisfy the identified rule to perform the operation. The operation should succeed. The

evaluator should use an alternate interface to verify that the operation did actually succeed and the applicable security attributes have been assigned or modified. In some cases, the evaluator should be able to either refer to or build on other tests (e.g., those associated with access control, security management restrictions, or auditing) to verify the resulting security attributes have changed as expected.

2. Perform the identified operation using instructions in the administrative guidance in order to assign or modify the identified security attribute(s) with the all but the minimum necessary conditions to satisfy the identified rule to perform the operation. The operation should fail with an appropriate error. The evaluator should use an alternate interface to verify that the operation did actually not succeed and the applicable security attributes have not been assigned or modified. In some cases, the evaluator should be able to either refer to or build on other tests (e.g., those associated with access control, security management restrictions, or auditing) to verify the resulting security attributes have changed as expected.
    a. This test should be repeated where multiple restrictive conditions are specified in a rule so that it is ensured that each condition is actually enforced. This is accomplished by testing with only one condition not satisfied, working through all the conditions.

Given a list of attribute(s)/interface/rule triples, the evaluator shall perform the following tests in each case of a rule that is behavioral:

1. Perform the identified operation using instructions in the administrative guidance in order to assign or modify the identified security attribute(s) in accordance with the behavioral rule. The operation should succeed. The evaluator should use an alternate interface to verify that the operation did actually succeed and the applicable security attributes have been assigned or modified. In some cases, the evaluator should be able to either refer to or build on other tests (e.g., those associated with access control, security management restrictions, or auditing) to verify the resulting security attributes have changed as expected.

    a. This test should be repeated where multiple behavioral rule components are specified in a rule so that it is ensured that each behavioral condition works as expected. This is accomplished by working through all the conditions using as few as possible in each case.
    b. Note that this test may be already addressed in the context of a test for a restive rule where one or more corresponding behavioral conditions is implied.

In general, it is expected that security attribute associated with a subject will be tested in the context of other requirements. However, the evaluator shall ensure that all security attributes are addressed in a combination of access control, security management enforcement, and audit tests. Additional tests may need to be developed in order to ensure coverage of all applicable security attributes.

Note that while the tests above should serve to verify that assignment and changes to security attributes occur as expected based on the TSS, Security Functional Description, and administrative guidance, it is not required or expected that the evaluator should comprehensively test every affected

security function (access control, security management enforcement, and audit) after every possible initial or changed security attribute assignment. The basic idea is that the use of the security attributes will be tested in the context of other applicable security functions, while the focus here is on whether the assignments and changes occur correctly and only when permitted.

5.3.3.9 Assurance Activities for FIA_PK_EXT.1 Public Key Based Authentication and FMT_MTD.1(CM) Management of TSF Data[51]

5.3.3.9.1 TOE Summary Specification (TSS)

In order to show that the TSF supports the use of X.509v3 certificates according to the RFC 5280, the evaluator shall ensure that the TSS describes the following information:

For each section of RFC 5280, any statement that is not "MUST" (for example, "MAY", "SHOULD", "SHOULD NOT", etc.) shall be described so that the reader can determine whether the TOE implements that specific part of the standard;

- For each section of RFC 5280, any non-conformance to "MUST" or “SHOULD" statements shall be described;

Any TOE-specific extensions or processing that is not included in the standard that may impact the security requirements the TOE is to enforce shall be described.

The TSS shall describe all certificate stores implemented that contain certificates used to meet the requirements of this PP. This description shall contain information pertaining to how certificates are loaded into the store, and how the store is protected from unauthorized access. That is to say that those certificates used to authorize remote IT entities can only be managed by administrative users, while untrusted users have the ability to manage certificates for their use.

5.3.3.9.2 Interface Specification

The collection of interfaces provided for the TOE will include those that specify how to load and manage certificates. If the TOE has the capability to import certificates from a Certificate Authority (CA), the included set of interfaces will describe how the TOE can be configured to import certificates from trusted authorities. This may also include how to set up a trusted channel to communicate with a CA.

5.3.3.9.3 Operational User Guidance

The operational guidance provides the administrator instruction as to how they configure the TOE to import certificates. The importation of certificates can be from a CA, and may require the configuration steps that ensure the CA is authenticated and the communication path is protected (e.g., trusted channel).

The guidance also instructs the administrator how they can load certificates manually (e.g., through portable media).

[51] The FMT_MTD.1(CM) activities apply to FMT_MTD.1(X509)in the Windows security target.

If the TOE comes preloaded with certificates, the guidance instructs the administrator to manage those certificates. This guidance will also most likely be relevant to certificates that are manually loaded, or imported from a CA as well. The guidance covers how to enable to disable the trust relationship of the certificates.

5.3.3.9.4 Testing

The evaluator shall devise tests that show that the TOE processes certificates that conform to the implementation described in the TSS; are able to form a certification path as specified in the standard and in the TSS; and are able to validate certificates as specified in the standard (certification path validation including CRL processing). The evaluator shall perform the following tests for each function in the system that requires the use of certificates:

Test 1: The evaluator shall demonstrate that using a certificate without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates needed to validate the certificate to be used in the function, and demonstrate that the function succeeds. The evaluator then shall delete one of the certificates, and show that the function fails.

Test 2: The evaluator shall attempt to use the operational guidance to load certificates from a network device and from portable/removable media (e.g., local CA, file server, USB stick, CD) that the TOE supports.

Test 3: The evaluator attempts to manage the certificates that are associated with a remote IT entity that supports the functions identified in FTP_ITC.1.3. For those entities, the evaluator ensures that with administrative rights, they are able to “trust” or “untrust” those certificates. Conversely, with the all but the needed rights, the evaluator attempts to modify the trust relationship; the result shall be a failed attempt.

5.3.4 Assurance Activities for Security Management

5.3.4.1 Assurance Activities for FMT_MOF.1 Management of Security Functions Behavior[52]

5.3.4.1.1 TOE Summary Specification (TSS)

The TSS shall describe the characteristics that are enforced for passwords, and describe the point at which the enforcement is performed.

5.3.4.1.2 Interface Specification

The interfaces that are used to configure the password enforcement capability are identified. The interfaces that are used to change passwords are also identified, and the evaluator ensures that these interfaces correspond to the one used for password-based authentication in the FIA_UAU requirements.

[52] These activities apply to FMT_MOF.1(Pass) in the Windows security target.

5.3.4.1.3 Operational User Guidance

The operational guidance shall describe the characteristics for passwords that are available; instructions for setting the enforcement mechanism; and a discussion of "strong" passwords and recommended minimum settings.

5.3.4.1.4 Testing

The evaluator shall also perform the following tests. Note that one or more of these tests can be performed with a single test case.

Test 1: The evaluator shall compose passwords that either meet the requirements, or fail to meet the requirements, in some way. For each password, the evaluator shall verify that the TOE supports the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that all characters, rule characteristics, and a minimum length listed in the requirement are supported, and justify the subset of those characters chosen for testing.

Test 2: The evaluator shall set rules that require the password be composed of specific combinations of password characteristics, and then attempt to use the password. The combinations of characteristics shall cover the breadth of characteristics, but not necessarily every combination. The evaluator shall include both valid (according to the rules) and invalid (do not conform to the rules) combinations, and observe that the valid passwords are accepted and the invalid passwords are rejected. In performing this test, the evaluator shall ensure that every interface that allows passwords to be changed is exercised, but not all cases need to be run on each interface.

Test 3: The evaluator shall attempt to configure the passwords while not a member of the group that is specified as allowed to change the passwords, and observe that they are unable to configure the password rules.

5.3.4.2 Assurance Activities for FMT_MTD.1(IAT) Management of TSF Data[53]

The assurance activity for this SFR is contained within the FIA_AFL.1 requirement as they are directly related.

5.3.4.3 Assurance Activities for FMT_MTD.1(IAF) Management of TSF Data[54]

The assurance activity for this SFR is contained within the FIA_AFL.1 requirement as they are directly related.

5.3.4.4 Assurance Activities for FMT_MTD.1(IAU) Management of TSF Data[55]

5.3.4.4.1 TOE Summary Specification (TSS)

The relevant user security attributes have been identified in the assurance activities for FIA_ATD.1.

[53] The FMT_MTD.1(IAT) assurance activities apply to FMT_MTD.1(Threshold) in this security target.
[54] The FMT_MTD.1(IAF) assurance activities apply to FMT_MTD.1(Re-enable) in this security target.
[55] The FMT_MTD.1(IAU) assurance activities apply to FMT_MTD.1(Init-Attr), FMT_MTD.1(Mod-Attr), and FMT_MTD.1(Mod-Auth) in this security target.

The evaluator shall further find that the TSS describes the restrictions that apply to creating (initializing), viewing, modifying, and deleting each of the identified user security attributes. If additional or alternate operations are available, the TSS must map them to the controlled operations identified in FMT_MTD.1(IAU). The restrictions shall identify the roles that can perform specific operations and/or rules that determine whether specific operations can be performed on each of the user security attributes. The description of restrictions must necessarily address both methods that manipulate user security attributes collectively, such as creating or deleting a user, and methods that manipulate user security attributes individual (or in groups), such as changing passwords and added/removing group memberships or roles.

5.3.4.4.2 Interface Specification

The relevant interfaces have been identified in the assurance activities for FIA_ATD.1. The evaluator shall further ensure that for each identified interface the restrictions, described in the TSS, are also described in the Security Functional Description and are consistent with the TSS.

5.3.4.4.3 Operational User Guidance

The relevant guidance has been identified in the assurance activities for FIA_ATD.1. However, if the rules above are subject to change, the guidance must provide any necessary instructions. Given the open ended nature of such a possibility, this activity would need to be revisited when the rules for access to the user security attributes can be changed in the context of a given TOE.

It is not necessarily expected that the administrative guidance should identify the applicable restrictions, but if it does and the evaluator finds a contradiction between the administrative guidance and TSS or Security Functional Description, the evaluator must work with the developer to resolve the discrepancy.

5.3.4.4.4 Testing

The number of tests used to verify the TOE's behavior will, of course, depend upon the number of user security attributes, the interfaces that provide access to them, and the complexity of associated restrictions. It is suggested that the evaluator develop a matrix that associates the user security attributes with interfaces available to create, view, modify, or delete them. Note that in some cases user security attributes might be addressed collectively when an interface operates on a group of attributes simultaneously such as amy be the case when creating or deleting a user. The matrix should be further developed with mappings to specific restrictions based on roles or rules, resulting in triples of user attribute(s), interface, and restriction.

Given a list of attribute(s)/interface/restriction triples, the evaluator shall perform the following tests in each case where the restriction is related to a role:

1. Perform the identified operation using instructions in the administrative guidance in order to manipulate the identified security attribute(s) in a role permitted to perform the operation. The operation should succeed. The evaluator should use an alternate interface to verify that the operation did actually succeed (e.g., a user was actually created or deleted).
2. Perform the identified operation using instructions in the administrative guidance in order to manipulate the identified security attribute(s) in a role not permitted to perform the operation.

The operation should fail with an appropriate error. The evaluator should use an alternate interface to verify that the operation did actually not succeed (e.g., a user was not actually created or deleted).

Given a list of attribute(s)/interface/restriction triples, the evaluator shall perform the following tests in each case where the restriction is related to a rule:

1. Perform the identified operation using instructions in the administrative guidance in order to manipulate the identified security attribute(s) with the minimum necessary conditions to satisfy the identified rule to perform the operation. The operation should succeed. The evaluator should use an alternate interface to verify that the operation did actually succeed (e.g., a user password was actually changed).
2. Perform the identified operation using instructions in the administrative guidance in order to manipulate the identified security attribute(s) with the all but the minimum necessary conditions to satisfy the identified rule to perform the operation. The operation should fail with an appropriate error. The evaluator should use an alternate interface to verify that the operation did actually not succeed (e.g., a user password was not actually changed).
    a. This test should be repeated where multiple conditions are specified in a rule so that it is ensured that each condition is actually enforced. This is accomplished by testing with only one condition not satisfied, working through all the conditions

5.3.5 Assurance Activity for Protection of the TSF

5.3.5.1 Assurance Activities for FPT_STM.1 Reliable time stamps

5.3.5.1.1 TOE Summary Specification (TSS)

The evaluator shall examine the TSS to ensure that it lists each function that makes use of time, including: recording of audit events, session timeout, and X.509 certificate revocation. The TSS provides a description of how the time is maintained and considered reliable in the context of each of the time related functions. This would include an indication of whether the function uses an internal interface to access the time or if it uses the externally visible interface.

The evaluator shall examine the TSS to gain an understanding of how system time is maintained to ensure it is reliable and monotonically increasing.

If the TOE is capable of receiving time from an external source, such as an NTP server, the TSS describes how this communication path is protected (e.g., IPsec, TLS) and ensures only authorized IT entities as defined by the administrator are able to modify the time.

5.3.5.1.2 Interface Specification

There should be an interface that allows all users/applications to obtain/read the system time. There will also be an interface that is used to set the local system clock. The evaluator ensures the interface specification describes how to use the interfaces to get and set time. The interface description for setting the time should specify what rights or privilege the caller must have in order to set the time.

If the TOE supports receiving time from an external entity, the interface specification describes the interface that is used to receive the time; this could be done as a manual activity, or there may be a capability that is configured that will request an update periodically.

When examining the interfaces associated with the time function, the evaluator ensures that the descriptions of the interfaces are consistent with what the TSS states about setting system time.

5.3.5.1.3 Operational User Guidance

The evaluator examines the operational guidance, which may reference the interface specification for the applicable interfaces, to ensure it instructs the administrator how to set the time.

If the TOE supports the use of an external entity to receive or update the time, the operational guidance provides the administrator guidance on how to setup the TOE in order to receive time from the authorized entity. The guidance should provide instructions on how to ensure the communication path is protected from attacks that could compromise the integrity of the time . For example, if the TOE is able to use an NTP server, the guidance would instruct the administrator how to configure the NTP client, and may instruct how to use a trusted channel to ensure the NTP server is authenticated and the integrity of the information transported across the channel is either maintained, or any changes are detected.

5.3.5.1.4 Testing

Test 1: The evaluator uses the operational guide to set the time. The evaluator shall then use an available interface to observe that the time was set correctly.

Test2: The evaluator attempts to use the available interfaces to set the time acting as an untrusted user. The evaluator shall not be able to modify the time.

Test3: [conditional] If the TOE supports the use of an NTP server and the assignment in FTP_ITC.1.3 is used to assign NTP as a function; the evaluator shall use the operational guidance to configure the NTP client on the TOE, and set up a protected communication path with the NTP server. The evaluator will observe that the NTP server has set the time to what is expected. If the TOE supports multiple cryptographic protocols for establishing a connection with the NTP server, the evaluator shall perform this test using each supported protocol.

5.3.6 Assurance Activities for TOE Access

5.3.6.1 Assurance Activities for FTA_SSL.1 TSF-initiated Session Locking and FTA_SSL.2 User-initiated Locking

5.3.6.1.1 TOE Summary Specification (TSS)

The evaluator shall examine the TSS to determine how the TOE determines when the period of inactivity has been reached (e.g., no activity on the keyboard or mouse, no active programs streaming video to the monitor, no dialog boxes being popped up on the screen). The TSS also describes what controls the ability to set the time period, and whether the time period is global (i.e., system wide) or is it configurable per user account. The evaluator also determines from the TSS description how the TOE

renders the display unreadable (e.g., a user defined screen saver is activated; administrators control what is displayed when the time period is reached, a system-defined screen is presented that cannot be modified).

The evaluator shall examine the TSS to ensure it identifies what activity the system responds to (e.g., depressing key on keyboard, moving the mouse, program interacting with display)and describes how the system responds to activity and what options are presented to a user (e.g., dialog box to enter authentication credentials to unlock the session, ability to login as another user, option to shutdown the machine).

Finally, the evaluator shall examine the TSS to make certain that it describes how the user initiates a locked session, and what happens when they initiate a session-lock. It may be the case where the TSS behaves the exactly the same way as when the time out occurs. If not, the TSS describes any differences in behavior.

5.3.6.1.2 Interface Specification

The evaluator shall examine the interface specification for the interfaces associated with these components to determine that the capabilities present in the system defined by the TSS are consistent with what the interfaces descriptions state. At the very least, there should be interfaces that provide the ability to set the time interval, lock the session, and unlock the session.

5.3.6.1.3 Operational User Guidance

The evaluator shall examine the operational user guidance to ensure it instructs the administrator how to configure the inactivity time period. If the TOE provides a means to specify what is displayed when the session is locked, the operational guide describes how this is done, and the evaluator shall ensure it is consistent with the description provided in the TSS.

The evaluator shall ensure the guide describes the options that are available when the system responds to activity, and how the user can invoke those options.

The evaluator shall determine that the guide describes how users can initiate a session lock.

5.3.6.1.4 Testing

The evaluator shall perform the following test:

Test 1: The evaluator follows the operational guidance to configure a few different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a local interactive session with the TOE. The evaluator then observes that the session is either locked after the configured time period and no remnants of data are visible on the display.

Test 2: The evaluator attempts to use the available interfaces to set the timeout period without having the proper authorizations (acting as an untrusted user). The evaluator shall fail in their attempts to modify the timeout period.

Test 3: [conditional] Variations of Test 1 and Test 2 may be necessary, depending on the complexity of the mechanism controlling the ability to set the timeout period. If the restriction is one needs to be an administrator than the test is straightforward and is as described. If there are privileges or an access control mechanism involved, the evaluator will have to determine the conditions under which to test the ability to change the timeout.

In such instances, the evaluator ensures the tester has the minimum set of privileges or access control settings to change the timeout, and does so successfully. The tester than has all but one of the necessary privileges or access control settings and attempts to change the timeout, this time failing.

Test 4: The evaluator attempts to initiate the session lock capability as specified in the operational guidance. The evaluator then observes that the session is either locked after the configured time period and no remnants of data are visible on the display.

Test 5: The evaluator then ensures that re-authentication for each authentication method allowed is needed when trying to unlock the session.

5.3.7 Assurance Activities for Trusted Path/Channels

5.3.7.1 Assurance Activities for FTP_ITC.1 Inter-TSF Trusted Channel[56]

5.3.7.1.1 Background

The capability to set up a trusted channel to another trusted IT product is required for an operating system compliant to the OSPP. The operating system needs to implement at least one of the protocols SSH, TLS, or IPsec compliant with the standards referred to by the SFR implementing at least the cipher suites listed as mandatory in the SFR. Note that those mandatory cipher suites may include additional cipher suites the related RFCs define as “REQUIRED”.

5.3.7.1.2 TOE Summary Specification (TSS)

5.3.7.1.2.1 Expectations

The TSS (or public documentation pointed to by the TSS) needs to list the protocols specified in the SFR and the standards implemented, including options taken where the standard allows for different options.

5.3.7.1.2.2 Evaluator Activities

The evaluator verifies that the standards are referenced correctly, that they describe the protocol completely, and that any options that the standard leaves open are defined in the TSS or the developer documentation pointed to by the TSS.

[56] These activities apply to FAU_GEN.1(OSPP) in the Windows security target.

5.3.7.1.3 Functional Specification

5.3.7.1.3.1 Expectations

For FTP_ITC.1 the interfaces are the network interfaces and the interfaces that can be used to set up a trusted channel. The interface specifications are the protocol specifications which are defined by references to the standards with a description of options taken (if the standard allows for different options). For interfaces a user can use to set up a trusted channel, the interface description needs to describe the options the user has for setting up the channel, and how to control the channel.

Note: for cases where the TSF (in accordance with the configuration defined by a trusted administrator) automatically and transparent for the user sets up a trusted channel, there may be no explicit user interface for initiating communication via a trusted channel. In this case there must be a management interface (which may be a configuration file) used by the TSF to decide when to initiate communication via a trusted channel and which options to use.

5.3.7.1.3.2 Evaluator Actions

The evaluator verifies that either user interfaces exist which allows a user to initiate communication with a remote IT product using a trusted channel, or communication via a trusted channel is initiated automatically by the TSF in accordance with the administrator defined configuration. In either case the evaluator verifies that he is able to get a communication link using the trusted channel protocols specified in FTP_ITC.1 with all the options defined in the SFR. He verifies that those options can either be selected when initiating the trusted channel or can be selected with an appropriate configuration defined via a management interface.

5.3.7.1.4 Architectural Design

5.3.7.1.4.1 Expectations

There are no further expectations on the architectural design for this SFR than the ones defined for the TSS. The developer may well point in the TSS to existing public design documentation for further detail of this functionality.

5.3.7.1.4.2 Evaluator activities

If additional design documentation is pointed to in the TSS, the evaluator verifies that this correctly refines the statements made in the TSS and correctly describes how the object security attributes can be revoked.

5.3.7.1.5 User Guidance (for Administrators as well as “Regular Users”)

5.3.7.1.5.1 Expectations

The guidance needs to explain how a trusted channel can be established and what the parameters for setting up a trusted channel are. The guidance needs to describe what options an administrator or a user may select and how those options affect the establishment and maintenance of the trusted channel. Especially options for selecting or excluding cipher suites that can be used as part of the protocol need to be documented, allowing an installation to restrict the cipher suites to those that are viewed as secure or are required to be used to comply with national or organizational policies.

5.3.7.1.5.2 Evaluator activities

The evaluator verifies that the guidance describes how to set up a trusted channel using the protocols defined in FTP_ITC.1 with all options defined there. Note that this activity overlaps significantly with the assessment of the functional specification and should therefore be performed together with the assessment of the interfaces.

5.3.7.1.6 Testing

5.3.7.1.6.1 Expectations

The developer is expected to test the protocols defined in FTP_ITC.1 with all options for the authentication of the remote IT system and all options for the cipher suites defined in FTP_ITC.1. Testing should be performed using a reference system that has a different implementation of the protocols and cipher suites to ensure that the TOE is able to set up and maintain the trusted channel to a product with an independent implementation of the protocol including the cryptographic algorithms used as part of the protocol.

5.3.7.1.6.2 Evaluator activity

The evaluator verifies that testing includes all protocols and protocol options defined. The evaluator will set up his own reference system and ensure that this system uses a different implementation of the protocols listed in FTP_ITC.1. The evaluator will perform his own tests by attempting to set up a trusted channel to an instance of the TOE.

The test shall cover the following cases:

- Attempts to use options (e. g. for remote system authentication) not supported by the TOE. Those attempts need to fail.
- Attempts to use options supported by the TOE but providing incorrect authentication credentials. Those attempts need to fail.
- Attempts to use correct authentication credentials and the correct protocol version, but cipher suites not supported by the TOE. Those attempts need to fail.
- Attempts to use protocol versions not supported by the TOE (e. g. older versions of a supported protocol). Those attempts need to fail.
- Attempts to use a protocol version supported by the TOE, an authentication method supported by the TOE, correct authentication credentials, and a cipher suite supported by the TOE. Those attempts need to pass (unless there are other conditions defined in the guidance or functional specification that cause the attempt to fail in an expected way).

5.4 Additional Assurance Activities

5.4.1 Assurance Activities for Cryptographic Key Generation for Symmetric Keys (FCS_CKM.1(SYM))

5.4.1.1 TOE Summary Specification (TSS)

The evaluator shall review the TSS to determine that it describes how the functionality described by **FCS_RBG_EXT.1** is invoked to generate symmetric cryptographic keys. The evaluator uses the description of the RBG functionality in **FCS_RBG_EXT.1** to determine that the key size being requested is identical to the key size and mode to be used for the encryption/decryption of the data.

5.4.1.2 Interface Specification

The Functional Specification shall identify interface(s) that generate audit events for symmetric key generation.

The functional specification shall identify interface(s) that can be used by applications or users to generate symmetric cryptographic keys.

For each such interface, the TOE documentation shall describe any authorization required to invoke the symmetric cryptographic key generation function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.1.3 Operational User Guidance

The guidance documentation shall describe audit events for symmetric key generation.

The operational guidance shall describe how the symmetric key generation is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.1.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for symmetric key generation as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

Test 2: The evaluator shall write, or the developer shall provide access to, an application that requests symmetric key generation operations by the TSF. The evaluator shall verify that the results from the validation match the expected results according to the API documentation.

5.4.2 Assurance Activities for Cryptographic Key Generation for Asymmetric Keys Used for Key Establishment (FCS_CKM.1(ASYM))

5.4.2.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_CKM.1(ASYM).

5.4.2.2 Interface Specification

The OS PP does not include separate expectations for this assurance activity.

5.4.2.3 Operational User Guidance

The OS PP does not include separate expectations for this assurance activity.

5.4.2.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for asymmetric key generation as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

This assurance activity will verify the key generation and key establishments schemes used on the TOE.
Key Generation:

The evaluator shall verify the implementation of the key generation routines of the supported schemes using the applicable tests below.

Key Generation for Finite-Field Cryptography (FFC) – Based 56A Schemes

FFC Domain Parameter

The evaluator shall verify the implementation of the Parameters Generation and the Key Generation for FFC by the TOE using the Parameter Generation and Key Generation test. This test verifies the ability of the TSF to correctly produce values for the field prime p, the cryptographic prime q (dividing p-1), the cryptographic group generator g, and the calculation of the private key x and public key y.

The Parameter generation specifies 2 ways (or methods) to generate the cryptographic prime q and the field prime p:

Cryptographic and Field Primes:

- Primes q and p shall both be provable primes
- Primes q and field prime p shall both be probable primes

and two ways to generate the cryptographic group generator g:

Cryptographic Group Generator:

- Generator g constructed through a verifiable process
- Generator g constructed through an unverifiable process.

The Key generation specifies 2 ways to generate the private key x:

Private Key:

- len(q) bit output of RBG where 1 <=x <= q-1
- len(q) + 64 bit output of RBG, followed by a mod q-1 operation where 1<= x<=q-1.

The security strength of the RBG must be at least that of the security offered by the FFC parameter set.
To test the cryptographic and field prime generation method for the provable primes method and/or the group generator g for a verifiable process, the evaluator must seed the TSF parameter generation routine with sufficient data to deterministically generate the parameter set.
For each key length supported, the evaluator shall have the TSF generate 25 parameter sets and key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation. Verification must also confirm

- g != 0,1
- q divides p-1
- g^q mod p = 1
- g^x mod p = y

for each FFC parameter set and key pair.

Key Generation for Elliptic Curve Cryptography (ECC) - Based 56A Schemes

ECC Key Generation Test

For each supported NIST curve, i.e., P-256, P-284 and P-521, the evaluator shall require the implementation under test (IUT) to generate 10 private/public key pairs. The private key shall be generated using an approved random bit generator (RBG). To determine correctness, the evaluator shall submit the generated key pairs to the public key verification (PKV) function of a known good implementation.

Key Establishment Schemes

The evaluator shall verify the implementation of the key establishment schemes of the supported by the TOE using the applicable tests below.

SP800-56A Key Establishment Schemes

The evaluator shall verify a TOE's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for each key agreement scheme verify that a TOE has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the DLC primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function (KDF). If key confirmation is supported, the evaluator shall also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MACdata and the calculation of MACtag.

Function Test

The Function test verifies the ability of the TOE to implement the key agreement schemes correctly. To conduct this test the evaluator shall generate or obtain test vectors from a known good implementation of the TOE supported schemes. For each supported key agreement scheme-key agreement role combination, KDF type, and, if supported, key confirmation role- key confirmation type combination, the tester shall generate 10 sets of test vectors. The data set consists of one set of domain parameter values (FFC) or the NIST approved curve (ECC) per 10 sets of public keys. These keys are static, ephemeral or both depending on the scheme being tested.

The evaluator shall obtain the DKM, the corresponding TOE's public keys (static and/or ephemeral), the MAC tag(s), and any inputs used in the KDF, such as the Other Information field OI and TOE id fields.

If the TOE does not use a KDF defined in SP 800-56A, the evaluator shall obtain only the public keys and the hashed value of the shared secret.

The evaluator shall verify the correctness of the TSF's implementation of a given scheme by using a known good implementation to calculate the shared secret value, derive the keying material DKM, and compare hashes or MAC tags generated from these values.

If key confirmation is supported, the TSF shall perform the above for each implemented approved MAC algorithm.

Validity Test

The Validity test verifies the ability of the TOE to recognize another party's valid and invalid key agreement results with or without key confirmation. To conduct this test, the evaluator shall obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to determine which errors the TOE should be able to recognize. The evaluator generates a set of 24 (FFC) or 30 (ECC) test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator's public keys, the TOE's public/private key pairs, MACTag, and any inputs used in the KDF, such as the other info and TOE id fields.

The evaluator shall inject an error in some of the test vectors to test that the TOE recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MACed, or the generated MACTag. If the TOE contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties' static public keys, both parties' ephemeral public keys and the TOE's static private key to assure the TOE detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass).

The TOE shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator shall compare the TOE's results with the results using a known good implementation verifying that the TOE detects these errors.

5.4.3 Assurance Activities for Cryptographic Key Generation for Asymmetric Keys Used for Peer Authentication (FCS_CKM.1(AUTH))

5.4.3.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_CKM.1(AUTH).

5.4.3.2 Interface Specification

The OS PP does not include separate expectations for this assurance activity.

5.4.3.3 Operational User Guidance

The OS PP does not include separate expectations for this assurance activity.

5.4.3.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for asymmetric key generation as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

This assurance activity will verify the key generation and key establishments schemes used on the TOE.
Key Generation:

The evaluator shall verify the implementation of the key generation routines of the supported schemes using the applicable tests below.

Key Generation for RSA-Based Key Establishment Schemes

The evaluator shall verify the implementation of RSA Key Generation by the TOE using the Key Generation test. This test verifies the ability of the TSF to correctly produce values for the key components including the public verification exponent e, the private prime factors p and q, the public modulus n and the calculation of the private signature exponent d.

Key Pair generation specifies 5 ways (or methods) to generate the primes p and q. These include:

1. Random Primes:

- Provable primes
- Probable primes

2. Primes with Conditions:

- Primes p1, p2, q1,q2, p and q shall all be provable primes
- Primes p1, p2, q1, and q2 shall be provable primes and p and q shall be probable primes
- Primes p1, p2, q1,q2, p and q shall all be probable primes

To test the key generation method for the Random Provable primes method and for all the Primes with Conditions methods, the evaluator must seed the TSF key generation routine with sufficient data to deterministically generate the RSA key pair. This includes the random seed(s), the public exponent of

the RSA key, and the desired key length. For each key length supported, the evaluator shall have the TSF generate 25 key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation.

Key Generation for Finite-Field Cryptography (FFC) Schemes

FFC Domain Parameter

The evaluator shall verify the implementation of the Parameters Generation and the Key Generation for FFC by the TOE using the Parameter Generation and Key Generation test. This test verifies the ability of the TSF to correctly produce values for the field prime p, the cryptographic prime q (dividing p-1), the cryptographic group generator g, and the calculation of the private key x and public key y.

The Parameter generation specifies 2 ways (or methods) to generate the cryptographic prime q and the field prime p:

Cryptographic and Field Primes:

- Primes q and p shall both be provable primes
- Primes q and field prime p shall both be probable primes

and two ways to generate the cryptographic group generator g:

Cryptographic Group Generator:

- Generator g constructed through a verifiable process
- Generator g constructed through an unverifiable process.

The Key generation specifies 2 ways to generate the private key x:

Private Key:

- len(q) bit output of RBG where 1 <=x <= q-1
- len(q) + 64 bit output of RBG, followed by a mod q-1 operation where 1<= x<=q-1.

The security strength of the RBG must be at least that of the security offered by the FFC parameter set.

To test the cryptographic and field prime generation method for the provable primes method and/or the group generator g for a verifiable process, the evaluator must seed the TSF parameter generation routine with sufficient data to deterministically generate the parameter set.

For each key length supported, the evaluator shall have the TSF generate 25 parameter sets and key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation.

Verification must also confirm

- g != 0,1
- q divides p-1
- g^q mod p = 1

- g^x mod p = y

for each FFC parameter set and key pair.

Key Generation for Elliptic Curve Cryptography (ECC) Schemes

ECC Key Generation Test

For each supported NIST curve, i.e., P-256, P-284 and P-521, the evaluator shall require the implementation under test (IUT) to generate 10 private/public key pairs. The private key shall be generated using an approved random bit generator (RBG). To determine correctness, the evaluator shall submit the generated key pairs to the public key verification (PKV) function of a known good implementation.

5.4.4 Assurance Activities for Cryptographic Key Zeroization (FCS_CKM_EXT.4)

5.4.4.1 TOE Summary Specification (TSS)

The evaluator shall check to ensure the TSS describes each of the secret keys (keys used for symmetric encryption), private keys, and CSPs used to generate key; when they are zeroized (for example, immediately after use, on system shutdown, etc.); and the type of zeroization procedure that is performed (overwrite with zeros, overwrite three times with random pattern, etc.). If different types of memory are used to store the materials to be protected, the evaluator shall check to ensure that the TSS describes the zeroization procedure in terms of the memory in which the data are stored (for example, "secret keys stored on flash are zeroized by overwriting once with zeros, while secret keys stored on the internal hard drive are zeroized by overwriting three times with a random pattern that is changed before each write"). If a read-back is done to verify the zeroization, this shall be described as well.

5.4.4.2 Interface Specification

The OS PP does not include separate expectations for this assurance activity.

5.4.4.3 Operational User Guidance

The guidance documentation shall describe audit events for key zeroization failure.

5.4.4.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for key zeroization failure as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

5.4.5 Assurance Activities for Cryptographic Services (FCS_SRV_EXT.1)

5.4.5.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_SRV_EXT.1).

5.4.5.2 Interface Specification

The evaluator shall verify that the API documentation provided according to Section 6.2.1 includes the security functions (cryptographic algorithms) described in these requirements.

The evaluator shall write, or the developer shall provide access to, an application that requests cryptographic operations by the TSF. The evaluator shall verify that the results from the validation match the expected results according to the API documentation. This application may be used to assist in verifying the cryptographic operation assurance activities for the other algorithm services requirements.

5.4.5.3 Operational User Guidance

The evaluator shall verify that the API documentation provided includes the security functions (cryptographic algorithms) identified in the requirements.

5.4.5.4 Testing

The evaluator shall write, or the developer shall provide access to, an application that requests cryptographic operations by the TSF. The evaluator shall verify that the results from the validation match the expected results according to the API documentation. This application may be used to assist in verifying the cryptographic operation assurance activities for the other algorithm services requirements.

5.4.6 Assurance Activities for Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))

5.4.6.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_COP.1(AES).

5.4.6.2 Interface Specification

If the TOE does not provide encryption or decryption as a service, then there are no applicable interfaces for FCS_COP.1(AES) and, consequently, no evaluator activities for the functional specification.

If the TOE provides encryption or decryption as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

The functional specification shall identify interface(s) that can be used by applications or users for encryption and decryption.

For each such interface, the TOE documentation shall describe any authorization required to invoke the encryption and decryption function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.6.3 Operational User Guidance

If the TOE does not provide encryption and decryption as a service, then there are no additional evaluator activities for FCS_COP.1(AES) guidance documentation.

If the TOE provides encryption and decryption as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.6.4 Testing

AES-ECB Tests

AES-ECB Known Answer Tests

There are four Known Answer Tests (KATs), described below. In all KATs, the plaintext and ciphertext, shall be 128-bit blocks. The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.

> **KAT-1.** To test the encrypt functionality of AES-ECB, the evaluator shall supply a set of 15 plaintext values and obtain the ciphertext value that results from AES-ECB encryption of the given plaintext using a key value of all zeros. Five plaintext values shall be encrypted with a 128-bit all-zeros key, five shall be encrypted with a 192-bit all-zeros key, and the remaining five shall be encrypted with a 256-bit all-zeros key.
>
> To test the decrypt functionality of AES-ECB, the evaluator shall perform the same test as for encrypt, using 15 ciphertext values as input and AES-ECB decryption.
>
> **KAT-2.** To test the encrypt functionality of AES-ECB, the evaluator shall supply a set of 15 key values and obtain the ciphertext value that results from AES-ECB encryption of an all-zeros plaintext using the given key value. Five of the keys shall be 128-bit keys, five shall be 192-bit keys, and the remaining five shall be 256-bit keys.
>
> To test the decrypt functionality of AES-ECB, the evaluator shall perform the same test as for encrypt, using an all-zero ciphertext value as input and AES-ECB decryption.
>
> **KAT-3.** To test the encrypt functionality of AES-ECB, the evaluator shall supply the three sets of key values described below and obtain the ciphertext value that results from AES encryption of an all-zeros plaintext using the given key value. The first set of keys shall have 128 128-bit keys, the second set of keys shall have 192 192-bit keys, and the third set shall have 256 256-bit keys. Key *i* in each set shall have the leftmost *i* bits be ones and the rightmost *N-i* bits be zeros, for *i* in [1,N].

To test the decrypt functionality of AES-EBC, the evaluator shall supply the three sets of key and ciphertext value pairs described below and obtain the plaintext value that results from AES-ECB decryption of the given ciphertext using the given key. The first set of key/ciphertext pairs shall have 128 128-bit key/ciphertext pairs, the second set of key/ciphertext pairs shall have 192 192-bit key/ciphertext pairs, and the third set of key/ciphertext pairs shall have 256 256-bit key/ciphertext pairs. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1,N]. The ciphertext value in each pair shall be the value that results in an all-zeros plaintext when decrypted with its corresponding key.

KAT-4. To test the encrypt functionality of AES-ECB, the evaluator shall supply the set of 128 plaintext values described below and obtain the three ciphertext values that result from AES-ECB encryption of the given plaintext using a 128-bit key value of all zeros, a 192-bit key value of all zeros, and a 256-bit key value of all zeros, respectively. Plaintext value i in each set shall have the leftmost i bits be ones and the rightmost 128-i bits be zeros, for i in [1,128].

To test the decrypt functionality of AES-ECB, the evaluator shall perform the same test as for encrypt, using ciphertext values of the same form as the plaintext in the encrypt test as input and AES-ECB decryption.

AES-ECB Multi-Block Message Test

The evaluator shall test the encrypt functionality by encrypting an *i*-block message where 1 < *i* <=10. The evaluator shall choose a key and plaintext message of length *i* blocks and encrypt the message, using the mode to be tested, with the chosen key. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key using a known good implementation.

The evaluator shall also test the decrypt functionality for each mode by decrypting an *i*-block message where 1 < *i* <=10. The evaluator shall choose a key and a ciphertext message of length *i* blocks and decrypt the message, using the mode to be tested, with the chosen key. The plaintext shall be compared to the result of decrypting the same ciphertext message with the same key and IV using a known good implementation.

AES-ECB Monte Carlo Tests

The evaluator shall test the encrypt functionality using a set of 300 plaintext and key pairs. 100 of these shall use 128 bit keys, 100 shall use 192-bit keys, and 100 shall use 256 bit keys. The plaintext values shall be 128-bit blocks. For each pair, 1000 iterations shall be run as follows:

```
# Input: PT, Key

for i = 1 to 1000:

        CT[1] = AES-EBC-Encrypt(Key, PT)

        PT = IV
```

The ciphertext computed in the 1000th iteration (i.e., CT[1000]) is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation.

The evaluator shall test the decrypt functionality using the same test as for encrypt, exchanging CT and PT and replacing AES-ECB-Encrypt with AES-ECB-Decrypt.

AES-CBC Tests

AES-CBC Known Answer Tests

There are four Known Answer Tests (KATs), described below. In all KATs, the plaintext, ciphertext, and IV values shall be 128-bit blocks. The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.

KAT-1. To test the encrypt functionality of AES-CBC, the evaluator shall supply a set of 15 plaintext values and obtain the ciphertext value that results from AES-CBC encryption of the given plaintext using a key value of all zeros and an IV of all zeros. Five plaintext values shall be encrypted with a 128-bit all-zeros key, five shall be encrypted with a 192-bit all-zeros key, and the remaining five shall be encrypted with a 256-bit all-zeros key.

To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using 15 ciphertext values as input and AES-CBC decryption.

KAT-2. To test the encrypt functionality of AES-CBC, the evaluator shall supply a set of 15 key values and obtain the ciphertext value that results from AES-CBC encryption of an all-zeros plaintext using the given key value and an IV of all zeros. Five of the keys shall be 128-bit keys, five shall be 192-bit keys, and the remaining five shall be 256-bit keys.

To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using an all-zero ciphertext value as input and AES-CBC decryption.

KAT-3. To test the encrypt functionality of AES-CBC, the evaluator shall supply the three sets of key values described below and obtain the ciphertext value that results from AES encryption of an all-zeros plaintext using the given key value and an IV of all zeros. The first set of keys shall have 128 128-bit keys, the second set shall have 192 192-bit keys, and the third set shall have 256 256-bit keys. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1,N].

To test the decrypt functionality of AES-CBC, the evaluator shall supply the two sets of key and ciphertext value pairs described below and obtain the plaintext value that results from AES-CBC decryption of the given ciphertext using the given key and an IV of all zeros. The first set of key/ciphertext pairs shall have 128 128-bit key/ciphertext pairs, the second set of

key/ciphertext pairs shall have 192 192-bit key/ciphertext pairs, and the third set of key/ciphertext pairs shall have 256 256-bit key/ciphertext pairs. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1,N]. The ciphertext value in each pair shall be the value that results in an all-zeros plaintext when decrypted with its corresponding key.

KAT-4. To test the encrypt functionality of AES-CBC, the evaluator shall supply the set of 128 plaintext values described below and obtain the three ciphertext values that result from AES-CBC encryption of the given plaintext using a 128-bit key value of all zeros with an IV of all zeros, a 192-bit key value of all zeros with an IV of all zeros, and a 256-bit key value of all zeros with an IV of all zeros, respectively. Plaintext value i in each set shall have the leftmost i bits be ones and the rightmost 128-i bits be zeros, for i in [1,128].

To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using ciphertext values of the same form as the plaintext in the encrypt test as input and AES-CBC decryption.

AES-CBC Multi-Block Message Test

The evaluator shall test the encrypt functionality by encrypting an i-block message where 1 < i <=10. The evaluator shall choose a key, an IV and plaintext message of length i blocks and encrypt the message, using the mode to be tested, with the chosen key and IV. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key and IV using a known good implementation.

The evaluator shall also test the decrypt functionality for each mode by decrypting an i-block message where 1 < i <=10. The evaluator shall choose a key, an IV and a ciphertext message of length i blocks and decrypt the message, using the mode to be tested, with the chosen key and IV. The plaintext shall be compared to the result of decrypting the same ciphertext message with the same key and IV using a known good implementation.

AES-CBC Monte Carlo Tests

The evaluator shall test the encrypt functionality using a set of 300 plaintext, IV, and key 3- tuples. 100 of these shall use 128 bit keys, 100 shall use 192-bit keys, and 100 shall use 256 bit keys. The plaintext and IV values shall be 128-bit blocks. For each 3-tuple, 1000 iterations shall be run as follows:

```
# Input: PT, IV, Key

for i = 1 to 1000:

        if i == 1:

                CT[1] = AES-CBC-Encrypt(Key, IV, PT)

                PT = IV
```

else:

CT[i] = AES-CBC-Encrypt(Key, PT)

PT = CT[i-1]

The ciphertext computed in the 1000th iteration (i.e., CT[1000]) is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation.

The evaluator shall test the decrypt functionality using the same test as for encrypt, exchanging CT and PT and replacing AES-CBC-Encrypt with AES-CBC-Decrypt.

AES-CFB8 Tests

AES-CFB8 Known Answer Tests

There are four Known Answer Tests (KATs), described below. In all KATs, the plaintext, ciphertext, and IV values shall be 128-bit blocks. The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.

KAT-1. To test the encrypt functionality of AES-CFB8, the evaluator shall supply a set of 15 plaintext values and obtain the ciphertext value that results from AES-CFB8 encryption of the given plaintext using a key value of all zeros and an IV of all zeros. Five plaintext values shall be encrypted with a 128-bit all-zeros key, five shall be encrypted with a 192-bit all-zeros key, and the remaining five shall be encrypted with a 256-bit all-zeros key.

To test the decrypt functionality of AES- CFB8, the evaluator shall perform the same test as for encrypt, using 15 ciphertext values as input and AES- CFB8 decryption.

KAT-2. To test the encrypt functionality of AES- CFB8, the evaluator shall supply a set of 15 key values and obtain the ciphertext value that results from AES- CFB8 encryption of an all-zeros plaintext using the given key value and an IV of all zeros. Five of the keys shall be 128-bit keys, five shall be 192-bit keys, and the remaining five shall be 256-bit keys.

To test the decrypt functionality of AES- CFB8, the evaluator shall perform the same test as for encrypt, using an all-zero ciphertext value as input and AES- CFB8 decryption.

KAT-3. To test the encrypt functionality of AES- CFB8, the evaluator shall supply the three sets of key values described below and obtain the ciphertext value that results from AES encryption of an all-zeros plaintext using the given key value and an IV of all zeros. The first set of keys shall have 128 128-bit keys, the second set shall have 192 192-bit keys, and the third set shall have 256 256-bit keys. Key *i* in each set shall have the leftmost *i* bits be ones and the rightmost *N-i* bits be zeros, for *i* in [1,N].

To test the decrypt functionality of AES- CFB8, the evaluator shall supply the three sets of key and ciphertext value pairs described below and obtain the plaintext value that results from AES-CFB8 decryption of the given ciphertext using the given key and an IV of all zeros. The first set of key/ciphertext pairs shall have 128 128-bit key/ciphertext pairs, the second set of key/ciphertext pairs shall have 192 192-bit key/ciphertext pairs, and the third set of key/ciphertext pairs shall have 256 256-bit key/ciphertext pairs. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1,N]. The ciphertext value in each pair shall be the value that results in an all-zeros plaintext when decrypted with its corresponding key.

KAT-4. To test the encrypt functionality of AES- CFB8, the evaluator shall supply the set of 128 plaintext values described below and obtain the three ciphertext values that result from AES-CFB8 encryption of the given plaintext using a 128-bit key value of all zeros with an IV of all zeros, a 192-bit key value of all zeros with an IV of all zeros, and a 256-bit key value of all zeros with an IV of all zeros, respectively. Plaintext value i in each set shall have the leftmost i bits be ones and the rightmost 128-i bits be zeros, for i in [1,128].

To test the decrypt functionality of AES- CFB8, the evaluator shall perform the same test as for encrypt, using ciphertext values of the same form as the plaintext in the encrypt test as input and AES- CFB8 decryption.

AES-CFB8 Multi-Block Message Test

The evaluator shall test the encrypt functionality by encrypting an *i*-block message where 1 < *i* <=10. The evaluator shall choose a key, an IV and plaintext message of length *i* blocks and encrypt the message, using the mode to be tested, with the chosen key and IV. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key and IV using a known good implementation.

The evaluator shall also test the decrypt functionality for each mode by decrypting an *i*-block message where 1 < *i* <=10. The evaluator shall choose a key, an IV and a ciphertext message of length *i* blocks and decrypt the message, using the mode to be tested, with the chosen key and IV. The plaintext shall be compared to the result of decrypting the same ciphertext message with the same key and IV using a known good implementation.

AES- CFB8 Monte Carlo Tests

The evaluator shall test the encrypt functionality using a set of 300 plaintext, IV, and key 3-tuples. 100 of these shall use 128 bit keys, 100 shall use 192-bit keys, and 100 shall use 256 bit keys. The plaintext and IV values shall be 128-bit blocks. For each 3-tuple, 1000 iterations shall be run as follows:

Input: PT, IV, Key

for i = 1 to 1000:

```
if i == 1:
        CT[1] = AES-CBC-Encrypt(Key, IV, PT)
        PT = ByteI(IV)
else:
        CT[i] = AES-CBC-Encrypt(Key, PT)
        If I <= 16
                PT = ByteI(IV)
        else
                PT = CT[i-16]
```

The ciphertext computed in the 1000th iteration (i.e., CT[1000]) is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation.

The evaluator shall test the decrypt functionality using the same test as for encrypt, exchanging CT and PT and replacing AES-CFB8-Encrypt with AES-CFB8-Decrypt.

AES-CCM Tests

The evaluator shall test the generation-encryption and decryption-verification functionality of AES-CCM for the following input parameter and tag lengths:

128 bit, 192 bit, and 256 bit keys

Two payload lengths. One payload length shall be the shortest supported payload length, greater than or equal to zero bytes. The other payload length shall be the longest supported payload length, less than or equal to 32 bytes (256 bits).

Two or three associated data lengths. One associated data length shall be 0, if supported. One associated data length shall be the shortest supported payload length, greater than or equal to zero bytes. One associated data length shall be the longest supported payload length, less than or equal to 32 bytes (256 bits). If the implementation supports an associated data length of 216 bytes, an associated data length of 216 bytes shall be tested.

Nonce lengths. All supported nonce lengths between 7 and 13 bytes, inclusive, shall be tested.

Tag lengths. All supported tag lengths of 4, 6, 8, 10, 12, 14 and 16 bytes shall be tested.

To test the generation-encryption functionality of AES-CCM, the evaluator shall perform the following four tests:

Test 1. For EACH supported key and associated data length and ANY supported payload, nonce and tag length, the evaluator shall supply one key value, one nonce value and 10 pairs of associated data and payload values and obtain the resulting ciphertext.

Test 2. For EACH supported key and payload length and ANY supported associated data, nonce and tag length, the evaluator shall supply one key value, one nonce value and 10 pairs of associated data and payload values and obtain the resulting ciphertext.

Test 3. For EACH supported key and nonce length and ANY supported associated data, payload and tag length, the evaluator shall supply one key value and 10 associated data, payload and nonce value 3-tuples and obtain the resulting ciphertext.

Test 4. For EACH supported key and tag length and ANY supported associated data, payload and nonce length, the evaluator shall supply one key value, one nonce value and 10 pairs of associated data and payload values and obtain the resulting ciphertext.

To determine correctness in each of the above tests, the evaluator shall compare the ciphertext with the result of generation-encryption of the same inputs with a known good implementation.

To test the decryption-verification functionality of AES-CCM, for EACH combination of supported associated data length, payload length, nonce length and tag length, the evaluator shall supply a key value and 15 nonce, associated data and ciphertext 3-tuples and obtain either a FAIL result or a PASS result with the decrypted payload. The evaluator shall supply 10 tuples that should FAIL and 5 that should PASS per set of 15.

AES-GCM Tests

The evaluator shall test the authenticated encrypt functionality of AES-GCM for each combination of the following input parameter lengths:

128 bit, 192 bit, and 256 bit keys

Two plaintext lengths. One of the plaintext lengths shall be a non-zero integer multiple of 128 bits, if supported. The other plaintext length shall not be an integer multiple of 128 bits, if supported.

Three AAD lengths. One AAD length shall be 0, if supported. One AAD length shall be a non-zero integer multiple of 128 bits, if supported. One AAD length shall not be an integer multiple of 128 bits, if supported.

Two IV lengths. If 96 bit IV is supported, 96 bits shall be one of the two IV lengths tested.

The evaluator shall test the encrypt functionality using a set of 10 key, plaintext, AAD, and IV tuples for each combination of parameter lengths above and obtain the ciphertext value and tag that results from

AES-GCM authenticated encrypt. Each supported tag length shall be tested at least once per set of 10. The IV value may be supplied by the evaluator or the implementation being tested, as long as it is known.

The evaluator shall test the decrypt functionality using a set of 10 key, ciphertext, tag, AAD, and IV 5-tuples for each combination of parameter lengths above and obtain a Pass/Fail result on authentication and the decrypted plaintext if Pass. The set shall include five tuples that Pass and five that Fail.

The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.

5.4.7 Assurance Activities for Cryptographic Operation for Cryptographic Signature (FCS_COP.1(SIGN))

5.4.7.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_COP.1(SIGN).

5.4.7.2 Interface Specification

The Functional Specification shall identify interface(s) that generate audit events for digital signature services.

If the TOE does not provide cryptographic signature generation or verification as a service, then there are no applicable interfaces for FCS_COP.1(SIGN) and, consequently, no evaluator activities for the functional specification.

If the TOE provides cryptographic signature generation or verification as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

The functional specification shall identify interface(s) that can be used by applications or users to cryptographic signature generation and verification.

For each such interface, the TOE documentation shall describe any authorization required to invoke the cryptographic signature generation and verification. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.7.3 Operational User Guidance

The guidance documentation shall describe audit events for digital signature services.

If the TOE does not provide cryptographic signature generation or verification as a service, then there are no additional evaluator activities for FCS_COP.1(SIGN) guidance documentation.

If the TOE provides cryptographic signature generation or verification as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.7.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for cryptographic signature failure as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

Key Generation:

Key Generation for DSA Scheme

See "Key Generation for Finite-Field Cryptography (FFC) – Based 56A Schemes"

Key Generation for RSA Signature Schemes

See "Key Generation for RSA-Based Key Establishment Schemes".

ECDSA Key Generation Tests

See "Key Generation for Elliptic Curve Cryptography (ECC) - Based 56A Schemes".

DSA Signature Algorithm Tests

Signature Generation Test

The evaluator shall verify the implementation of DSA Signature Generation by the TOE using the Signature Generation Test. To conduct this test the evaluator must generate or obtain 10 messages from a trusted reference implementation for each modulus size/SHA combination supported by the TSF. The evaluator shall have the TOE use their private key and modulus value to sign these messages.

The evaluator shall verify the correctness of the TSF's signature using a known good implementation and the associated public keys to verify the signatures.

Signature Verification Test

The evaluator shall perform the Signature Verification test to verify the ability of the TOE to recognize another party's valid and invalid signatures. The evaluator shall inject errors into the test vectors produced during the Signature Verification Test by introducing errors in some of the public keys e, messages, IR format, and/or signatures. The TOE attempts to verify the signatures and returns success or failure.

The evaluator shall use these test vectors to emulate the signature verification test using the corresponding parameters and verify that the TOE detects these errors.

RSA Signature Algorithm Tests

Signature Generation Test

The evaluator shall verify the implementation of RSA Signature Generation by the TOE using the Signature Generation Test. To conduct this test the evaluator must generate or obtain 10 messages from a trusted reference implementation for each modulus size/SHA combination supported by the TSF. The evaluator shall have the TOE use their private key and modulus value to sign these messages.

The evaluator shall verify the correctness of the TSF's signature using a known good implementation and the associated public keys to verify the signatures.

Signature Verification Test

The evaluator shall perform the Signature Verification test to verify the ability of the TOE to recognize another party's valid and invalid signatures. The evaluator shall inject errors into the test vectors produced during the Signature Verification Test by introducing errors in some of the public keys e, messages, IR format, and/or signatures. The TOE attempts to verify the signatures and returns success or failure.

The evaluator shall use these test vectors to emulate the signature verification test using the corresponding parameters and verify that the TOE detects these errors.

ECDSA Algorithm Tests

ECDSA FIPS 186-4 Signature Generation Test

For each supported NIST curve (i.e., P-256, P-284 and P-521) and SHA function pair, the evaluator shall generate 10 1024-bit long messages and obtain for each message a public key and the resulting signature values R and S. To determine correctness, the evaluator shall use the signature verification function of a known good implementation.

ECDSA FIPS 186-4 Signature Verification Test

For each supported NIST curve (i.e., P-256, P-284 and P-521) and SHA function pair, the evaluator shall generate a set of 10 1024-bit message, public key and signature tuples and modify one of the values (message, public key or signature) in five of the 10 tuples. The evaluator shall obtain in response a set of 10 PASS/FAIL values.

5.4.8 Assurance Activities for Cryptographic Operation for Cryptographic Hashing (FCS_COP.1(HASH))

5.4.8.1 TOE Summary Specification (TSS)

The evaluator shall check that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS.

5.4.8.2 Interface Specification

If the TOE does not provide cryptographic hashing as a service, then there are no applicable interfaces for FCS_COP.1(HASH) and, consequently, no evaluator activities for the functional specification.

If the TOE provides cryptographic hashing as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

The functional specification shall identify interface(s) that can be used by applications or users for cryptographic hashing.

For each such interface, the TOE documentation shall describe any authorization required to invoke the cryptographic hash function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.8.3 Operational User Guidance

The evaluator checks the guidance documentation to determine that any configuration that is required to be done to configure the functionality for the required hash sizes is present.

If the TOE does not provide cryptographic hash as a service, then there are no additional evaluator activities for FCS_COP.1(HASH) guidance documentation.

If the TOE provides cryptographic hash as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance

5.4.8.4 Testing

The TSF hashing functions can be implemented in one of two modes. The first mode is the byte-oriented mode. In this mode the TSF only hashes messages that are an integral number of bytes in length; i.e., the length (in bits) of the message to be hashed is divisible by 8. The second mode is the bit-oriented mode. In this mode the TSF hashes messages of arbitrary length. As there are different tests for each mode, an indication is given in the following sections for the bit-oriented vs. the byte-oriented testmacs.

The evaluator shall perform all of the following tests for each hash algorithm implemented by the TSF and used to satisfy the requirements of this PP.

SHS Algorithm Tests

Short Messages Test - Bit-oriented Mode

The evaluators devise an input set consisting of m+1 messages, where m is the block length of the hash algorithm. The length of the messages range sequentially from 0 to m bits. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.

Short Messages Test - Byte-oriented Mode

The evaluators devise an input set consisting of m/8+1 messages, where m is the block length of the hash algorithm. The length of the messages range sequentially from 0 to m/8 bytes, with each message being an integral number of bytes. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.

Selected Long Messages Test - Bit-oriented Mode

The evaluators devise an input set consisting of m messages, where m is the block length of the hash algorithm. The length of the ith message is 512 + 99*i, where 1 ≤ i ≤ m. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.

Selected Long Messages Test - Byte-oriented Mode

The evaluators devise an input set consisting of m/8 messages, where m is the block length of the hash algorithm. The length of the ith message is 512 + 8*99*i, where 1 ≤ i ≤ m/8. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF.

Pseudorandomly Generated Messages Test

This test is for byte-oriented implementations only. The evaluators randomly generate a seed that is n bits long, where n is the length of the message digest produced by the hash function to be tested. The evaluators then formulate a set of 100 messages and associated digests by following the algorithm provided in Figure 1 of [SHAVS]. The evaluators then ensure that the correct result is produced when the messages are provided to the TSF.

5.4.9 Assurance Activities for Keyed-Hash Message Authentication (FCS_COP.1(HMAC))

5.4.9.1 TOE Summary Specification (TSS)

The evaluator shall examine the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used.

5.4.9.2 Interface Specification

The Functional Specification shall identify interface(s) that generate audit events for keyed-hash message authentication.

If the TOE does not provide keyed-hash message authentication as a service, then there are no applicable interfaces for FCS_COP.1(HMAC) and, consequently, no evaluator activities for the functional specification.

If the TOE provides keyed-hash message authentication as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

For each such interface, the TOE documentation shall describe any authorization required to invoke the keyed-hash message authentication function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.9.3 Operational User Guidance

The guidance documentation shall describe audit events for keyed-hash message authentication failures.

If the TOE does not provide keyed-hash message authentication as a service, then there are no additional evaluator activities for FCS_COP.1(HMAC) guidance documentation.

If the TOE provides cryptographic signature generation or verification as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.9.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for keyed-hash message authentication failure as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

HMAC Algorithm Tests

Test 2: For each of the supported parameter sets, the evaluator shall compose 15 sets of test data. Each set shall consist of a key and message data. The evaluator shall have the TSF generate HMAC tags for these sets of test data. The resulting MAC tags shall be compared to the result of generating HMAC tags with the same key and IV using a known good implementation.

5.4.10 Assurance Activities for Cryptographic Operation for ECDH Key Agreement (FCS_COP.1(DH KA))

5.4.10.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_COP.1(DH KA).

5.4.10.2 Interface Specification

The Functional Specification shall identify interface(s) that generate audit events for DH key agreement failure.

If the TOE does not provide DH key agreement as a service, then there are no interfaces for DH key agreement and, consequently, no additional evaluator activities for FCS_COP.1(DH KA) functional specification.

If the TOE provides DH key agreement as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

The functional specification shall identify interface(s) that can be used by applications or users for DH key agreement.

For each such interface, the TOE documentation shall describe any authorization required to invoke the DH key agreement function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.10.3 Operational User Guidance

The guidance documentation shall describe audit events for DH key agreement failure.

If the TOE does not provide DH key agreement as a service, then there are no additional evaluator activities for FCS_COP.1(DH KA) guidance documentation.

If the TOE provides DH key agreement as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.10.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for key agreement failure as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

DH Key Agreement Tests

Key Establishment Schemes

The evaluator shall verify the implementation of the key establishment schemes supported by the TOE using the applicable tests below.

SP800-56A Key Establishment Schemes

The evaluator shall verify a TOE's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for each key agreement scheme verify that a TOE has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the DLC primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function

(KDF). If key confirmation is supported, the evaluator shall also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MACdata and the calculation of MACtag.

Function Test

The Function test verifies the ability of the TOE to implement the key agreement schemes correctly. To conduct this test the evaluator shall generate or obtain test vectors from a known good implementation of the TOE supported schemes. For each supported key agreement scheme-key agreement role combination, KDF type, and, if supported, key confirmation role- key confirmation type combination, the tester shall generate 10 sets of test vectors. The data set consists of one set of domain parameter values (FFC) or the NIST approved curve (ECC) per 10 sets of public keys. These keys are static, ephemeral or both depending on the scheme being tested.

The evaluator shall obtain the DKM, the corresponding TOE‘s public keys (static and/or ephemeral), the MAC tag(s), and any inputs used in the KDF, such as the Other Information field OI and TOE id fields.

If the TOE does not use a KDF defined in SP 800-56A, the evaluator shall obtain only the public keys and the hashed value of the shared secret.

The evaluator shall verify the correctness of the TSF‘s implementation of a given scheme by using a known good implementation to calculate the shared secret value, derive the keying material DKM, and compare hashes or MAC tags generated from these values.

If key confirmation is supported, the TSF shall perform the above for each implemented approved MAC algorithm.

Validity Test

The Validity test verifies the ability of the TOE to recognize another party‘s valid and invalid key agreement results with or without key confirmation. To conduct this test, the evaluator shall obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to determine which errors the TOE should be able to recognize. The evaluator generates a set of 24 (FFC) or 30 (ECC) test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator‘s public keys, the TOE‘s public/private key pairs, MACTag, and any inputs used in the KDF, such as the other info and TOE id fields.

The evaluator shall inject an error in some of the test vectors to test that the TOE recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MACed, or the generated MACTag. If the TOE contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties‘ static public keys, both parties‘ ephemeral public keys and the TOE‘s static private key to assure the TOE detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass).

The TOE shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator shall compare the TOE's results with the results using a known good implementation verifying that the TOE detects these errors.

5.4.11 Assurance Activities for Cryptographic Operation for ECDSA Key Agreement (FCS_COP.1(EC KA))

5.4.11.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_COP.1(EC KA).

5.4.11.2 Interface Specification

The Functional Specification shall identify interface(s) that generate audit events for elliptic curve DH key agreement failure.

If the TOE does not provide elliptic curve DH key agreement as a service, then there are no interfaces for elliptic curve DH key agreement and, consequently, no additional evaluator activities for FCS_COP.1(EC KA) functional specification.

If the TOE provides elliptic curve DH key agreement as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

The functional specification shall identify interface(s) that can be used by applications or users for elliptic curve DH key agreement.

For each such interface, the TOE documentation shall describe any authorization required to invoke the elliptic curve DH key agreement function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.11.3 Operational User Guidance

The guidance documentation shall describe audit events for elliptic curve DH key agreement failure.

If the TOE does not provide elliptic curve DH key agreement as a service, then there are no additional evaluator activities for FCS_COP.1(EC KA) guidance documentation.

If the TOE provides elliptic curve DH key agreement as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.11.4 Testing

Test 1: The evaluator confirms that TSF generates audit events for key agreement failure as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

ECDH Key Agreement Tests

Key Establishment Schemes

The evaluator shall verify the implementation of the key establishment schemes supported by the TOE using the applicable tests below.

SP800-56A Key Establishment Schemes

The evaluator shall verify a TOE's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for each key agreement scheme verify that a TOE has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the DLC primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function (KDF). If key confirmation is supported, the evaluator shall also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MACdata and the calculation of MACtag.

Function Test

The Function test verifies the ability of the TOE to implement the key agreement schemes correctly. To conduct this test the evaluator shall generate or obtain test vectors from a known good implementation of the TOE supported schemes. For each supported key agreement scheme-key agreement role combination, KDF type, and, if supported, key confirmation role- key confirmation type combination, the tester shall generate 10 sets of test vectors. The data set consists of one set of domain parameter values (FFC) or the NIST approved curve (ECC) per 10 sets of public keys. These keys are static, ephemeral or both depending on the scheme being tested.

The evaluator shall obtain the DKM, the corresponding TOE‘s public keys (static and/or ephemeral), the MAC tag(s), and any inputs used in the KDF, such as the Other Information field OI and TOE id fields.

If the TOE does not use a KDF defined in SP 800-56A, the evaluator shall obtain only the public keys and the hashed value of the shared secret.

The evaluator shall verify the correctness of the TSF‘s implementation of a given scheme by using a known good implementation to calculate the shared secret value, derive the keying material DKM, and compare hashes or MAC tags generated from these values.

If key confirmation is supported, the TSF shall perform the above for each implemented approved MAC algorithm.

Validity Test

The Validity test verifies the ability of the TOE to recognize another party‘s valid and invalid key agreement results with or without key confirmation. To conduct this test, the evaluator shall obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to

determine which errors the TOE should be able to recognize. The evaluator generates a set of 24 (FFC) or 30 (ECC) test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator's public keys, the TOE's public/private key pairs, MACTag, and any inputs used in the KDF, such as the other info and TOE id fields.

The evaluator shall inject an error in some of the test vectors to test that the TOE recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MACed, or the generated MACTag. If the TOE contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties' static public keys, both parties' ephemeral public keys and the TOE's static private key to assure the TOE detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass).

The TOE shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator shall compare the TOE's results with the results using a known good implementation verifying that the TOE detects these errors.

5.4.12 Assurance Activities for Random Number Generation (FCS_RBG_EXT.1)

5.4.12.1 TOE Summary Specification (TSS)

There are no TOE Summary Specification assurance activities for FCS_RBG_EXT.1.

5.4.12.2 Architecture Design

Entropy Documentation and Assessment

The documentation of the entropy source should be detailed enough that, after reading, the evaluator will thoroughly understand the entropy source and why it can be relied upon to provide entropy. This documentation should include multiple detailed sections: design description, entropy justification, operating conditions, and health testing. This documentation is not required to be part of the TSS.

Design Description

Documentation shall include the design of the entropy source as a whole, including the interaction of all entropy source components. It will describe the operation of the entropy source to include how it works, how entropy is produced, and how unprocessed (raw) data can be obtained from within the entropy source for testing purposes. The documentation should walk through the entropy source design indicating where the random comes from, where it is passed next, any post-processing of the raw outputs (hash, XOR, etc.), if/where it is stored, and finally, how it is output from the entropy source. Any conditions placed on the process (e.g., blocking) should also be described in the entropy source design. Diagrams and examples are encouraged.

This design must also include a description of the content of the security boundary of the entropy source and a description of how the security boundary ensures that an adversary outside the boundary cannot affect the entropy rate.

If implemented, the design description shall include a description of how third-party applications can add entropy to the RBG. A description of any RBG state saving between power-off and power-on shall be included.

Entropy Justification

There should be a technical argument for where the unpredictability in the source comes from and why there is confidence in the entropy source exhibiting probabilistic behavior (an explanation of the probability distribution and justification for that distribution given the particular source is one way to describe this). This argument will include a description of the expected entropy rate and explain how you ensure that sufficient entropy is going into the TOE randomizer seeding process. This discussion will be part of a justification for why the entropy source can be relied upon to produce bits with entropy.

The entropy justification shall not include any data added from any third-party application or from any state saving between restarts.

Operating Conditions

Documentation will also include the range of operating conditions under which the entropy source is expected to generate random data. It will clearly describe the measures that have been taken in the system design to ensure the entropy source continues to operate under those conditions. Similarly, documentation shall describe the conditions under which the entropy source is known to malfunction or become inconsistent. Methods used to detect failure or degradation of the source shall be included.

Health Testing

More specifically, all entropy source health tests and their rationale will be documented. This will include a description of the health tests, the rate and conditions under which each health test is performed (e.g., at startup, continuously, or on-demand), the expected results for each health test, and rationale indicating why each test is believed to be appropriate for detecting one or more failures in the entropy source.

5.4.12.3 Interface Specification

The Functional Specification shall identify interface(s) that generate audit events for randomization process failure.

If the TOE does not provide random bit generation as a service, then there are no applicable interfaces for FCS_RBG_EXT.1 and, consequently, no evaluator activities for the functional specification.

If the TOE provides random bit generation as a service (as specified in FCS_SRV_EXT.1), then the following activities apply for the TOE.

The functional specification shall identify interface(s) that can be used by applications or users to perform random bit generation.

For each such interface, the TOE documentation shall describe any authorization required to invoke the random bit generation function. The documentation shall include API information that is provided to application developers. The API documentation shall clearly indicate to which products and versions the function applies. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding guidance activity.

5.4.12.4 Operational User Guidance

The guidance documentation shall describe audit events for randomization process failure.

If the TOE does not provide random bit generation as a service, then there are no additional evaluator activities for FCS_RBG_EXT.1 guidance documentation.

If the TOE provides random bit as a service (as specified in FCS_SRV_EXT.1), then TOE documentation describes how the service is invoked. The required information is identified in functional specification assurance activity above. The evaluator confirms the information is in the functional specification or guidance documentation either as part of this activity or the corresponding functional specification activity.

5.4.12.5 Testing

Test 1: The evaluator confirms that TSF generates audit events for key zeroization failure as described in the TOE documentation for the applicable interfaces identified in the Functional Specification.

The evaluator shall perform the following tests, depending on the standard to which the RBG conforms.

Implementations Conforming to FIP 140-2 Annex C

The reference for the tests contained in this section is The Random Number Generator Validation System (RNGVS). The evaluators shall conduct the following two tests. Note that the "expected values" are produced by a reference implementation of the algorithm that is known to be correct. Proof of correctness is left to each Scheme.

The evaluators shall perform a Variable Seed Test. The evaluators shall provide a set of 128 (Seed, DT) pairs to the TSF RBG function, each 128 bits. The evaluators shall also provide a key (of the length appropriate to the AES algorithm) that is constant for all 128 (Seed, DT) pairs. The DT value is incremented by 1 for each set. The seed values shall have no repeats within the set. The evaluators ensure that the values returned by the TSF match the expected values.

The evaluators shall perform a Monte Carlo Test. For this test, they supply an initial Seed and DT value to the TSF RBG function; each of these is 128 bits. The evaluators shall also provide a key (of the length appropriate to the AES algorithm) that is constant throughout the test. The evaluators then invoke the TSF RBG 10,000 times, with the DT value being incremented by 1 on each iteration, and the new seed for the subsequent iteration produced as specified in NIST-Recommended Random Number Generator Based on ANSI X9.31 Appendix A.2.4 Using the 3-Key Triple DES and AES Algorithms, Section 3. The evaluators ensure that the 10,000th value produced matches the expected value.

Implementations Conforming to NIST Special Publication 800-90A

The evaluator shall perform 15 trials for the RNG implementation. If the RNG is configurable, the evaluator shall perform 15 trials for each configuration. The evaluator shall also confirm that the operational guidance contains appropriate instructions for configuring the RNG functionality.

If the RNG has prediction resistance enabled, each trial consists of (1) instantiate DRBG, (2) generate the first block of random bits (3) generate a second block of random bits (4) uninstantiate. The evaluator verifies that the second block of random bits is the expected value. The evaluator shall generate eight input values for each trial. The first is a count (0 – 14). The next three are entropy input, nonce, and personalization string for the instantiate operation. The next two are additional input and entropy input for the first call to generate. The final two are additional input and entropy input for the second call to generate. These values are randomly generated. —generate one block of random bits ‖ means to generate random bits with number of returned bits equal to the Output Block Length (as defined in NIST SP800-90A).

If the RNG does not have prediction resistance, each trial consists of (1) instantiate DRBG, (2) generate the first block of random bits (3) reseed, (4) generate a second block of random bits (5) uninstantiate. The evaluator verifies that the second block of random bits is the expected value. The evaluator shall generate eight input values for each trial. The first is a count (0 – 14). The next three are entropy input, nonce, and personalization string for the instantiate operation. The fifth value is additional input to the first call to generate. The sixth and seventh are additional input and entropy input to the call to reseed. The final value is additional input to the second generate call.

The following paragraphs contain more information on some of the input values to be generated/selected by the evaluator.

- **Entropy input**: the length of the entropy input value must equal the seed length.
- **Nonce**: If a nonce is supported (CTR_DRBG with no Derivation Function does not use a nonce), the nonce bit length is one-half the seed length.
- **Personalization string**: The length of the personalization string must be <= seed length. If the implementation only supports one personalization string length, then the same length can be used for both values. If more than one string length is support, the evaluator shall use personalization strings of two different lengths. If the implementation does not use a personalization string, no value needs to be supplied.
- **Additional input**: the additional input bit lengths have the same defaults and restrictions as the personalization string lengths.

5.5 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the requirements defined in the OS PP Assurance Package as specified in Part 3 of the Common Criteria. No operations are applied to the assurance components.

In addition, the assurance activities from the OSPP (version 3.9) are used to determine that Windows satisfies the OSPP security functional requirements. These OSPP assurance activities are described in section 5.4.

Table 5-6 TOE Security Assurance Requirements

Requirement Class	Requirement Component
ASE: Security Target	ASE_INT.1: ST introduction
	ASE_CCL.1: Conformance claims
	ASE_SPD.1 Security problem definition
	ASE_OBJ.2 Security objectives
	ASE_ECD.1 Extended components definition
	ASE_REQ.2 Stated security requirements
	ASE_TSS.1 TOE summary specification
ADV: Design	ADV_ARC.1 Security architecture description
	ADV_FSP.1: Basic functional specification
AGD: Guidance Documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle Support	ALC_CMC.3: Authorization controls
	ALC_CMS.3: Implementation representation CM coverage
	ALC_DEL.1: Delivery procedures
	ALC_FLR.3: Systematic Flaw Remediation
	ALC_LCD.1: Developer Defined Life-Cycle Model
ATE: Testing	ATE_COV.2: Analysis of coverage
	ATE_DPT.1: Testing: basic design
	ATE_FUN.1: Functional testing
	ATE_IND.2: Independent testing - sample
AVA: Vulnerability Assessment	AVA_VAN.2: Vulnerability analysis

6 TOE Summary Specification (TSS)

This chapter describes the Windows security functions which satisfy the security functional requirements of the ***General Purpose Operating System Protection profile***. The TOE also includes additional security functions that are relevant to Windows in the following sections, as well as a mapping to the security functional requirements satisfied by the TOE.

6.1 Product Architecture

The TSF provides a security domain for its own protection and provides process isolation. The security domains used within and by the TSF consists of the following:

- Hardware
- Kernel-mode software
- Trusted user-mode processes

- User-mode Administrative tools process

The TSF hardware is managed by the TSF kernel-mode software and is not modifiable by untrusted subjects. The TSF kernel-mode software is protected from modification by hardware execution state and protection for both physical memory and memory allocated to a partition; an operating system image runs within a first partition for x64 systems. The TSF hardware provides a software interrupt instruction that causes a state change from user mode to kernel mode. The TSF kernel-mode software is responsible for processing all interrupts, and determines whether or not a valid kernel-mode call is being made. In addition, the TSF memory protection features ensure that attempts to access kernel-mode memory from user mode results in a hardware exception, ensuring that kernel-mode memory cannot be directly accessed by software not executing in the kernel mode.

The TSF provides process isolation for all user-mode processes through private virtual address spaces (private per process page tables), execution context (registers, program counters), and security context (handle table and token). The data structures defining process address space, execution context and security context are all stored in protected kernel-mode memory. All security relevant privileges are considered to enforce TSF Protection.

User-mode administrator tools execute with the security context of the process running on behalf of the authorized administrator. Administrator processes are protected like other user-mode processes, by process isolation.

Like TSF processes, user processes also are provided a private address space and process context, and therefore are protected from each other. Additionally, on 64-bit based hardware platforms, the TSF has the added ability to protect memory pages using Hardware Data Execution Prevention (DEP). Hardware-enforced DEP marks all memory locations in a process as non-executable unless the location explicitly contains executable code. Hardware-enforced DEP relies on processor hardware to mark memory with an attribute that indicates that code should not be executed from that memory location. DEP functions on a per-virtual memory page basis, usually by changing a bit in the page table entry (PTE) to mark the memory page. Processors that support hardware-enforced DEP are capable of raising an exception when code is executed from a page marked with the appropriate attribute set.

The TSF implements cryptographic mechanisms within a distinct user-mode process, where its services can be accessed by both kernel- and user-mode components, in order to isolate those functions from the rest of the TSF to limit exposure to possible errors while protecting those functions from potential tampering attempts.

Furthermore, the TSF includes a Code Integrity Verification feature, also known as Kernel-mode code signing (KMCS), whereby device drivers will be loaded only if they are digitally signed by either Microsoft or from a trusted root certificate authority recognized by Microsoft. KMCS uses public-key cryptography technology to verify the digital signature of each driver as it is loaded. When a driver tries to load, the TSF decrypts the hash included with the driver using the public key stored in the certificate. It then verifies that the hash matches the one that it computes based on the driver code using the FIPS -certified cryptographic libraries in the TSF. The authenticity of the certificate is also checked in the same

way, but using the certificate authority's public key, which must be configured in and trusted by the TOE.

6.2 TOE Security Functions

This section presents the TOE Security Functions (TSFs) and a mapping of security functions to Security Functional Requirements (SFRs). The TOE performs the following security functions:

- Audit
- Cryptographic Protection
- Identification and Authentication
- Security Management
- TOE Access
- Trusted Path / Channels
- TSF Protection
- User Data Protection

6.2.1 Audit Function

The TOE Audit security function performs:

- Audit Collection
- Audit Log Review
- Selective Audit
- Audit Log Overflow Protection
- Audit Log Restricted Access Protection

6.2.1.1 Audit Collection

The Windows Event Log service creates the security event log, which contains security relevant audit records collected on a system, along with other event logs which are also registered by other audit entry providers. The Local Security Authority (LSA) server collects audit events from all other parts of the TSF and forwards them to the Windows Event Log service which will place the event into the log for the appropriate provider. For each audit event, the Windows Event Log service stores the following data in each audit entry:

Table 6-1 Standard Fields in a Windows Audit Entry

Field in Audit Entry	Description
Date	The date the event occurred.
Time	The time the event occurred.
User	The security identifier (SID) of that represents the user on whose behalf the event occurred that represents the user. SIDs are described in more detail in section **6.2.4 Identification and Authentication Function**.
Event ID	A unique numberwithin the audit category that identifies the specific audit event.

Source	The Windows component that generated the audit event.
Outcome	Indicates whether the security audit event recorded is the result of a successful or failed attempt to perform the action.
Category	The type of the event defined by the event source.

The LSA service defines the following categories for audit events in the security log:

- System,
- Logon / Logoff
- Object Access
- Directory Service Access
- Privilege Use
- Detailed Process Tracking
- Policy Change
- Account Management
- Account Logon

Each audit entry may also contain category-specific data that is contained in the body of the entry as described below:

- For the System Category, the audit entry includes information relating to the system such as the time the audit trail was cleared, start or shutdown of the audit function, and startup and shutdown of Windows. Furthermore, the specific cryptographic operation is identified when such operations are audited.
- For the Logon and Account Logon Category, the audit entry includes the reason the attempted logon failed.
- For the Object Access and the Directory Service Access Category, the audit entry includes the object name and the desired access requested.
- For the Privilege Use Category, the audit entry identifies the privilege.
- For the Detailed Process Tracking Category, the audit event includes the process identifier.
- For the Policy Change and Account Management Category, the audit event includes the new values of the policy or account attributes.
- For the Account Logon Category, the audit event includes the logon type that indicates the source of the logon attempt as one of the following types in the audit record:
    - Interactive (local logon)
    - Network (logon from the network)
    - Service (logon as a service)
    - Batch (logon as a batch job)
    - Unlock (for Unlock screen saver)
    -

There are two places within the TSF where security audit events are collected. Inside the kernel, the Security Reference Monitor (SRM), a part of the NT Executive, is responsible for generation of all audit entries for the object access, privilege use, and detailed process tracking event categories. Windows components can request the SRM to generate an audit record and supply all of the elements in the audit record except for the system time, which the Executive provides. With one exception, audit events for the other event categories are generated by various services that either co-exist in the LSA server or call, with the SeAuditPrivilege privilege, the Authz Report Audit interfaces implemented in the LSA Policy subcomponent. The exception is that the Event Log Service itself records an event record when the security log is cleared and when the security log exceeds the warning level configured by the authorized administrator.

The LSA server maintains an audit policy in its database that determines which categories of events are actually collected. Defining and modifying the audit policy is restricted to the authorized administrator. The authorized administrator can select events to be audited by selecting the category or categories to be audited. An authorized administrator can individually select each category. Those services in the security process determine the current audit policy via direct local function calls. The only other TSF component that uses the audit policy is the SRM in order to record object access, privilege use, and detailed tracking audit. LSA and the SRM share a private local connection port, which is used to pass the audit policy to the SRM. When an authorized administrator changes the audit policy, the LSA updates its database and notifies the SRM. The SRM receives a control flag indicating if auditing is enabled and a data structure indicating that the events in particular categories to audit.

In addition to the system-wide audit policy configuration, it is possible to define a per-user audit policy using auditpol.exe. This allows individual audit categories (of success or failure) to be enabled or disabled on a per user basis.[57] The per-user audit policy refines the system-wide audit policy with a more precise definition of the audit policy for which events will be audited for a specific user.

Within each category, auditing can be performed based on success, failure, or both. For object access events, auditing can be further controlled based on user/group identify and access rights using System Access Control Lists (SACLs). SACLs are associated with objects and indicate whether or not auditing for a specific object, or object attribute, is enabled.

The TSF is capable of generating the audit events associated with each audit category, as described in the Description column of **Table 6-2** (**Audit Event Categories**). The auditable events associated with each category capture the events listed in **Table 5-4**. For each category, the associated audit events (listed in **Table 5-4**) for each of the requirements in the FAU_GEN Required Events column of **Table 6-2** are listed and **Appendix B: Basic Functional Specification and Interfaces** lists each audit ID and subcategory.

[57] Windows will prevent a local administrator from disabling auditing for local administrator accounts. If an administrator can bypass auditing, they can avoid accountability for such actions as exfiltrating files without authorization.

Table 6-2 Audit Event Categories

Category	Description	FAU_GEN Required Events
System	Audit attempts that affect security of the entire system such as clearing the audit trail.	FAU_STG.3, FCS_CKM.1*, FCS_CKM_EXT.4, FCS_COP.1*, FCS_RBG_EXT.1, FMT_MTD.1(GEN), FPT_STM.1
Object Access	Audit attempts to access user objects, such as files.	FDP_ACF.1*, FDP_IFF.1*, FMT_MSA.1(DAC), FMT_MSA.1(OBJ), FMT_MSA.3(DAC)
Privilege Use	Audits attempts to use security relevant privileges. Security relevant privileges are those privileges that are related to the TSFs and can be assigned in the evaluated configuration.	FAU_SAR.1, FAU_SAR.2, FDP_ACF.1(DAC), FMT_SMR.1
Detailed Process Tracking	Audit subject-tracking events, including program activation, handle duplication, indirect access to an object, and process exit.	FIA_USB.1(USR)
Policy Change	Audit attempts to change security policy settings such as the audit policy and privilege assignment.	FAU_SEL.1, FMT_MOF.1*, FMT_MTD.1(GEN), FMT_MTD.1(Audit), FMT_MTD.1(AuditSel), FMT_MTD.1(AuditFail), FMT_MTD.1(AuditStg), FMT_REV.1(Admin), FMT_SMR.1, FPT_ITT.1,
Account Management	Audit attempts to create, delete, or change user or group accounts and changes to their attributes.	FIA_AFL.1, FMT_MTD.1(Init-Attr), FMT_MTD.1(Mod-Attr), FMT_MTD.1(Mod-Auth), FMT_REV1., FMT_SMR.1
Logon	Audit attempts to logon or logoff the system, attempts to make a network connection.	FIA_AFL.1, FIA_UAU.1, FIA_UID.1, FIA_USB.1(USR), FTA_SSL.1, FTA_SSL.2
Account Logon	Audit when a DC receives a logon request.	FIA_UAU.1(Logon), FIA_UID.1

6.2.1.2 Audit Log Review

The Event Viewer MMC snap-in provides a user interface to view, sort, and search the security log. The security log can be sorted and searched by user identity, event type (by category and event ID), date, time, source, and outcome (success and/or failure). The Get-EventLog PowerShell cmdlet provides similar capabilities as the Event Viewer. The security log can also be searched by free form text occurring in the audit records. For example, this enables searching based on object identifiers.

6.2.1.3 Selective Audit

The authorized administrator has the ability to select events to be audited based upon object identity, user identity, computer (host identity), type (category), and outcome (success or failure) of the event. Selecting the set of events that will be audited can be on a per-machine basis by using tools such as auditpol.exe and wevtutil.exe, or using group policies to audit sets of machines (i.e. auditing based on the host identity).

6.2.1.4 Audit Log Overflow Protection

The TSF protects against the loss of events through a combination of controls associated with audit queuing and event logging. As configured in the TOE, audit data is appended to the audit log until it is full. The TOE protects against lost audit data by allowing the authorized administrator to configure the system to generate an audit event when the security audit log reaches a specified capacity percentage (e.g., 90%). Additionally, the authorized administrator can configure the system not to overwrite events and to shutdown when the security audit log is full. When so configured, after the system shutdowns due to audit overflow, only the authorized administrator can restart the system to log on and manage the security log. When the security log is full, a message is written to the display of the authorized administrator indicating the audit log has overflowed.

As described above, the TSF collects security audit data in two ways, via the SRM and via the LSA server. Both components maintain audit in-memory event queues. The SRM puts audit records on an internal queue to be sent to the LSA server. The LSA maintains a second queue where it holds the audit data from SRM and the other services in the security process. Both audit queues detect when an audit event loss has occurred. The SRM service maintains a high water mark and a low water mark on its audit queue to determine when full. The LSA also maintains marks in its queue to indicate when it is full.

Windows also provides an eventing infrastructure that other system components can use to log events which are not managed by the SRM or the LSA. The maximum size for these administrative and operational event logs can either be limited to the maximum size for the log file (and then prevent generation of new audit events for that particular log) or overwrite the oldest audit event. The Windows security target selects the second option.

6.2.1.5 Audit Log Restricted Access Protection

The Windows Event Log service controls and protects the security audit log. Note that the underlying files are configured so that only the TSF can open the files and the Event Log service opens those files exclusively when it starts and keeps them open while it is running. To view the contents of the security audit log, the user must be an authorized administrator. The security audit log is a system resource, created during system startup. No interfaces exist to create, destroy, or modify an event within the event log. The LSA subsystem is the only service registered to enter events into the security log. The TOE only offers user interfaces to read and clear the security event log. In order to read the event log, the user must have a read ACE in the access control list for the **Event Log** service.

SFR Mapping:

The **Audit function** satisfies the following SFRs:

- **FAU_GEN.1(OSPP)**: The TOE audit collection is capable of generating audit events for items identified in **Table 6-1 Audit Event Categories**. For each audit event the TSF records the date, time, user Security Identifier (SID) or name, logon type (for logon audit records), event ID, source, type, and category.
- **FAU_GEN.2**: All audit records include the user SID, which uniquely represents each user.
- **FAU_SAR.1:** The event viewer provides authorized administrators with the ability to review audit data in a readable format.
- **FAU_SAR.2** and **FMT_MTD.1(Audit)**: Only authorized administrators have any access to the audit log.
- **FAU_SEL.1, FMT_MTD.1(Audit Sel)**: The TSF provides the ability for the authorized administrator to select the events to be audited based upon object identity, user identity, workstation (host identity), event type, and success or failure of the event.
- **FAU_STG.1, FMT_MTD.1(AuditStg)**: The interface to the audit logs is limited by the Event Log service. The interface to the log only allows for viewing the audit data and for clearing all the audit data. The interface to the logs are restricted to authorized administrators and does not allow for the modification of audit data within the security log.
- **FAU_STG.3**: The authorized administrator can configure the system such that an audit event (an alarm) is generated if the audit data exceeds a specified percentage of the security log.
- **FAU_STG.4(SL), FMT_MTD.1(Audit Fail)**: The TOE can be configured such that when the security audit log is full the system shuts down. At that point, only the authorized administrator can log on to the system to clear the security log and return the system to an operational state consistent with TOE guidance. Additionally, when the security log reaches a certain percentage, an audit event (alarm) is generated.
- **FAU_STG.4(OL) , FMT_MTD.1(Audit Fail)**: The TOE can be configured such that when any administrative operational logs are full the system will overwrite the oldest events in each log type.
- **FMT_MTD.1(GEN)**[58]: The TSF restricts the ability to specify the size of the security log to an authorized administrator. The audit function provides capabilities for selective auditing and review using the Event Viewer MMC snap-in and the Get-EventLog cmdlet. The TOE provides the capability to select events to be audited based on the success and/or failure at the category level. Additionally, for the object access category of events, events can be selected based on user identity. The TSF determines which audit events to record based on the current audit policy and the specific settings in the SACLs. The Event Viewer provides the capability to perform searches and sorting of audit data by date, time, user SID or name, computer, event ID, source, type, and category. Additionally, the Event Viewer provides the capability to perform searching based upon specified free form text substrings within the audit records (e.g., to search for specific object identifiers).

[58] This requirement is for general management of security functions, the above description is a specific instance.

6.2.2 User Data Protection Function

The User Data Protection security services provided by the TOE are:

- Discretionary Access Control
- Mandatory Integrity Control
- Information Flow Control and Protection
- Residual Data Protection

6.2.2.1 Discretionary Access Control (DAC)

The executive within Windows mediates access between subjects and user data objects, also known as named objects. Subjects consist of processes with one or more threads running on behalf of users. **Table 6-2** lists the specific user data objects under the control of the DAC policy for the TOE.

Table 6-3 Named Objects

Name	Description
Desktop	The primary object used for graphical displays.
Event	An object created for the interprocess communication mechanism.
Event Pair	An object created for the interprocess communication mechanism.
I/O Completion Port	An object that provides a means to synchronize I/O.
Job	An object that allows for the management of multiple processes as a unit.
Registry Key	Registry Keys are the objects that form the Registry.
Mutant	An object created for the interprocess communication mechanism (known as a mutex in the Windows API).
Object Directory	A directory in the object namespace.
ALPC Port	A connection-oriented local process communication mechanism object that supports client and server side communication end points such as message queues.
Mailslot	An I/O object that provides support for message passing IPC via the network.
Named Pipe	An I/O object used for IPC over the network.
NTFS Directory	NT file system file object.
NTFS File	A user data file object managed by NTFS.
Printer	Represents a particular print queue and its association with a print device.
Process	An execution context for threads that has associated address space and memory, token, and handle tables.
Section	A memory region.
Semaphore	An object created for interprocess communication mechanism.
Symbolic Link	A means for providing name aliasing in the object name space.
Thread	An execution context (registers, stacks). All user-mode threads are associated with a process.
Timer	A means for a thread to wait for a specified amount of time to pass.
[Security Access] Token	This object represents the security context of a process or thread.

Window Station	A container for desktop objects and related attributes.
Debug	A set of resources used for debugging a process.
[Transaction] Enlistment	An object representing a transactional enlistment. An enlistment is an association between a resource manager and a transaction.
Transaction	An object that defines a logical unit of work.
ResourceManager	An object used to manage the data that is associated with each transaction.
TransactionManager	An object used to track the state of each transaction and coordinates recovery operations after a system crash.

6.2.2.1.1 Subject DAC Attributes

Windows security access tokens contain the security attributes for a subject. Tokens are associated with processes and threads running on behalf of the user. Information in a security access token that is used by DAC includes:

- The Security Identifier (SID) for the user account
- SIDs representing groups for which the user is a member
- Privileges assigned to the user
- An owner SID that identifies the SID to assign as owner for newly created objects
- A default Discretionary Access Control List (DACL) for newly created objects
- Token type which is either a primary or an impersonation token
- The impersonation level (for impersonation tokens)
- The integrity label SID
- An optional list of restricting SIDs
- The logon SID that identifies the logon session.

An administrator can change all of these except for the user account SID and logon SID.

As described in 6.2.4.7 Impersonation, a thread can be assigned an impersonation token that would be used instead of the process' primary token when making an access check and generating audit data. Hence, that thread is impersonating the client that provided the impersonation token. Impersonation stops when the impersonation token is removed from the thread or when the thread terminates.

An access token may also include a list of restricting SIDs which are used to limit access to objects. Restricting SIDs are contained in restricted tokens, (which is a special form of a thread impersonation token), and when configured serve to limit the corresponding process access to no more than that available to the restricted SID.

Access decisions are made using the impersonation token of a thread if it exists, and otherwise the thread's process primary token (which always exists).

6.2.2.1.2 Object DAC Attributes

Security Descriptors (SDs) contain all of the security attributes associated with an object. All objects in Table 6-2 have an associated SD. The security attributes from a SD used for discretionary access control are the object owner SID which specifies the owner of the security descriptor, the DACL present flag, and the DACL itself, when present.

DACLs contain a list of Access Control Entries (ACEs). Each ACE specifies an ACE type, a SID representing a user or group, and an access mask containing a set of access rights. Each ACE has inheritance attributes associated with it that specify if the ACE applies to the associated object only, to its children objects only, or to both its children objects and the associated object.

There are two types of ACEs that apply to discretionary access control:

- ALLOW ACES
  - ACCESS_ALLOWED_ACE: used to grant access to a user or group of users.
- DENY ACES
  - ACCESS_DENIED_ACE: used to deny access to a user or group of users.

In the ACE, an access mask contains object access rights granted (or denied) to the SID, representing a user or group. An access mask is also used to specify the desired access to an object when accessing the object and to identify granted access associated with an opened object. Each bit in an access mask represents a particular access right. There are four categories of access rights: standard, specific, special, and generic. Standard access rights apply to all object types. Specific access rights have different semantic meanings depending on the type of object. Special access rights are used in desired access masks to request special access or to ask for all allowable rights. Generic access rights are convenient groupings of specific and standard access rights. Each object type provides its own mapping between generic access rights and the standard and specific access rights.

For most objects, a subject requests access to the object (e.g., opens it) and receives a pointer to a handle in return. The TSF associates a granted access mask with each opened handle. For kernel-mode objects, handles are maintained in a kernel-mode handle table. There is one handle table per process; each entry in the handle table identifies an opened object and the access rights granted to that object. For user-mode TSF servers, the handle is a server-controlled context pointer associated with the connection between the subject and the server. The server uses this context handle in the same manner as with the kernel mode (i.e., to locate an opened object and its associated granted access mask). In both cases (user and kernel-mode objects), the SRM makes all access control decisions.

For some objects, the TSF does not maintain an opened context (e.g., a handle) to the object. In these cases, access checks are performed on every reference to the object (in place of checking a handle's granted access mask).

The following table summarizes every DAC access right for each named object which were tested by the evaluation lab:

Table 6-4 DAC Access Rights and Named Objects

Named Object	Access Rights
Desktop	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER DESKTOP_READOBJECTS DESKTOP_CREATEWINDOW DESKTOP_CREATEMENU DESKTOP_HOOKCONTROL DESKTOP_JOURNALRECORD DESKTOP_JOURNALPLAYBACK DESKTOP_ENUMERATE DESKTOP_WRITEOBJECTS DESKTOP_SWITCHDESKTOP DESKTOP_READOBJECTS and DESKTOP_WRITEOBJECTS
Event	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE EVENT_QUERY_STATE EVENT_MODIFY_STATE
Event Pair	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER
I/O Completion Port	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE IO_COMPLETION_MODIFY_STATE
Job	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE JOB_OBJECT_ASSIGN_PROCESS JOB_OBJECT_SET_ATTRIBUTES JOB_OBJECT_SET_SECURITY_ATTRIBUTES JOB_OBJECT_QUERY JOB_OBJECT_TERMINATE
Registry Key	ACCESS_SYSTEM_SECURITY READ_CONTROL

Named Object	Access Rights
	WRITE_DAC WRITE_OWNER KEY_SET_VALUE KEY_QUERY_VALUE KEY_CREATE_SUB_KEY KEY_ENUMERATE_SUB_KEYS KEY_NOTIFY DELETE
Mutant	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE
Object Directory	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER DIRECTORY_TRAVERSE
ALPC Port	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER
Mailslot	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE FILE_WRITE_DATA FILE_READ_DATA FILE_APPEND_DATA FILE_WRITE_EA FILE_WRITE_ATTRIBUTES
Named Pipe	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE FILE_READ_DATA FILE_WRITE_EA FILE_WRITE_ATTRIBUTES FILE_WRITE_DATA DELETE
NTFS Directory	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER

Named Object	Access Rights
	SYNCHRONIZE FILE_LIST_DIRECTORY FILE_ADD_FILE FILE_ADD_SUBDIRECTORY FILE_DELETE_CHILD FILE_READ_ATTRIBUTES FILE_WRITE_ATTRIBUTES FILE_DELETE_CHILD|FILE_ADD_FILE DELETE
NTFS File	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE FILE_WRITE_DATA FILE_READ_DATA FILE_APPEND_DATA FILE_WRITE_EA FILE_EXECUTE FILE_READ_ATTRIBUTES FILE_WRITE_ATTRIBUTES FILE_WRITE_ATTRIBUTES. FILE_WRITE_DATA and FILE_WRITE_ATTRIBUTES. DELETE FILE_WRITE_DATA | FILE_READ_DATA FILE_READ_DATA | FILE_EXECUTE FILE_READ_DATA | FILE_EXECUTE | FILE_WRITE_DATA FILE_WRITE_DATA | FILE_WRITE_EA | FILE_WRITE_ATTRIBUTES
Printer	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER PRINTER_READ PRINTER_ACCESS_ADMINISTER PRINTER_ACCESS_USE DELETE
Process	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE PROCESS_DUP_HANDLE PROCESS_CREATE_PROCESS PROCESS_QUERY_INFORMATION PROCESS_QUERY_INFORMATION |PROCESS_VM_READ PROCESS_SET_PORT

Named Object	Access Rights
	PROCESS_CREATE_THREAD PROCESS_SET_QUOTA PROCESS_SET_INFORMATION PROCESS_TERMINATE PROCESS_SET_INFORMATION and PROCESS_VM_WRITE PROCESS_SET_INFORMATION | PROCESS_SET_SESSIONID
Section	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SECTION_MAP_WRITE SECTION_MAP_READ SECTION_MAP_EXECUTE SECTION_MAP_EXECUTE| SECTION_MAP_READ SECTION_MAP_EXECUTE| SECTION_MAP_WRITE SECTION_MAP_READ| SECTION_MAP_WRITE
Semaphore	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE SEMAPHORE_MODIFY_STATE
Symbolic Link	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYMBOLIC_LINK_QUERY
Thread	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE THREAD_TERMINATE THREAD_SUSPEND_RESUME THREAD_GET_CONTEXT THREAD_SET_CONTEXT THREAD_QUERY_INFORMATION THREAD_QUERY_LIMITED_INFORMATION THREAD_QUERY_LIMITED_INFORMATION|THREAD_QUERY_INFOR MATION THREAD_SET_INFORMATION THREAD_SET_THREAD_TOKEN THREAD_IMPERSONATE
Timer	ACCESS_SYSTEM_SECURITY READ_CONTROL

Named Object	Access Rights
	WRITE_DAC WRITE_OWNER SYNCHRONIZE TIMER_MODIFY_STATE
[Security Access] Token	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER TOKEN_ASSIGN_PRIMARY TOKEN_IMPERSONATE TOKEN_QUERY TOKEN_QUERY_SOURCE TOKEN_ADJUST_PRIVILEGES TOKEN_ADJUST_GROUPS TOKEN_ADJUST_DEFAULT TOKEN_ADJUST_DEFAULT | TOKEN_ADJUST_SESSIONID TOKEN_ADJUST_DEFAULT | TOKEN_QUERY TOKEN_QUERY | TOKEN_ADJUST_PRIVILEGES TOKEN_QUERY | TOKEN_ADJUST_GROUPS
Window Station	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER WINSTA_READATTRIBUTES WINTSTA_WRITEATTRIBUTES WINSTA_ACCESSCLIPBOARD WINSTA_READATTRIBUTES| WINTSTA_WRITEATTRIBUTES WINSTA_CREATEDESKTOP WINSTA_ENUMERATE WINSTA_ENUMDESKTOPS WINSTA_ACCESSGLOBALATOMS
Debug	ACCESS_SYSTEM_SECURITY READ_CONTROL WRITE_DAC WRITE_OWNER SYNCHRONIZE DEBUG_READ_EVENT DEBUG_PROCESS_ASSIGN DEBUG_SET_INFORMATION
[Transaction] Enlistment	ENLISTMENT_SUPERIOR_RIGHTS ENLISTMENT_QUERY_INFORMATION ENLISTMENT_RECOVER ENLISTMENT_SET_INFORMATION
Transaction	TRANSACTION_COMMIT SYNCHRONIZE TRANSACTION_QUERY_INFORMATION

Named Object	Access Rights
	TRANSACTION_ROLLBACK TRANSACTION_SET_INFORMATION
ResourceManager	RESOURCEMANAGER_ENLIST TRANSACTION_ENLIST RESOURCEMANAGER__GET_NOTIFICATION RESOURCEMANAGER_QUERY_INFORMATION RESOURCEMANAGER_RECOVER
TransactionManager	TRANSACTIONMANAGER_CREATE_RM TRANSACTIONMANAGER_QUERY_INFORMATION TRANSACTIONMANAGER _QUERY_INFORMATION TRANSACTIONMANAGER_RECOVER

6.2.2.1.3 DAC Enforcement Algorithm

The TSF enforces the DAC policy to objects based on SIDs and privileges in the requestor's token, the desired access mask requested, and the object's security descriptor.

Below is a summary of the algorithm used to determine whether a request to access a user data object is allowed. In order for access to be granted, all access rights specified in the desired access mask must be granted by one of the following steps. At the end of any step, if all of the requested access rights have been granted then access is allowed. At the end of the algorithm, if any requested access right has not been granted, then access is denied.

1. Privilege Check:
    a. Check for SeSecurity privilege: This is required if ACCESS_SYSTEM_SECURITY is in the desired access mask. If ACCESS_SYSTEM_SECURITY is requested and the requestor does not have this privilege, access is denied. Otherwise ACCESS_SYSTEM_SECURITY is granted.
    b. Check for SeTakeOwner privilege: If the desired mask has WRITE_OWNER access right, and the privilege is found in the requestor's token, then WRITE_OWNER access is granted.
    c. Check for SeBackupPrivilege: The Backup Files and Directories privilege allows a subject process to read files and registry objects for backup operations regardless of their ACE in the DACL. If the subject process has the SeBackupPrivilege privilege and the operation requires the privilege, no further checking is performed and access is allowed. Otherwise this check is irrelevant and the access check proceeds.
    d. Check for SeRestorePrivilege: The Restore Files and Directories privilege allows a subject process to write files and registry objects for restore operations regardless of their ACE in the DACL. If the subject process has the SeRestorePrivilege privilege and the operation requires the privilege no further checking is performed, and access is allowed. Otherwise this check is irrelevant and the access check proceeds.
2. Owner Check:

   a. If the DACL contains one or more ACEs with the OwnerRights SID, those entries, along with all other applicable ACEs for the user, are used to determine the owner's rights.
   b. Otherwise, check all the SIDs in the token to determine if there is a match with the object owner. If so, the READ_CONTROL and WRITE_DAC rights are granted if requested.
3. DACL not present:
   a. All further access rights requested are granted.
4. DACL present but empty:
   a. If any additional access rights are requested, access is denied.
5. Iteratively process each ACE in the order that they appear in the DACL as described below:
   a. If the inheritance attributes of the ACE indicate the ACE is applicable only to children objects of the associated object, the ACE is skipped.
   b. If the SID in the ACE does not match any SID in the requestor's access token, the ACE is skipped.
   c. If a SID match is found, and the access mask in the ACE matches an access in the desired access mask:
      i. Access Allowed ACE Types: The ACE grants access to the entire object.
      ii. Access Denied ACE Type: If a requested access is specifically denied by an ACE, then the entire access request fails.
6. If all accesses are granted but the requestor's token has at least one restricting SID, the complete access check is performed against the restricting SIDs. If this second access check does not grant the desired access, then the entire access request fails.

6.2.2.1.4 Default DAC Protection

The TSF provides a process ensuring a DACL is applied by default to all new objects. When new objects are created, the appropriate DACL is constructed.

The TOE uses the following rules to set the DACL in the SDs for new named kernel objects:

- The object's DACL is the DACL from the SD specified by the creating process. The TOE merges any inheritable ACEs into the DACL unless SE_DACL_PROTECTED is set in the SD control flags. The TOE then sets the SE_DACL_PRESENT SD control flag. Note that a creating process can explicitly provide a SD that includes no DACL. The result will be an object with no protections. This is distinct from providing no SD which is described below.
- If the creating process does not specify a SD, the TOE builds the object's DACL from inheritable ACEs in the parent object's DACL. The TOE then sets the SE_DACL_PRESENT SD control flag.
- If the parent object has no inheritable ACEs, the TOE uses its object manager subcomponent to provide a default DACL. The TOE then sets the SE_DACL_PRESENT and SE_DACL_DEFAULTED SD control flags.
- If the object manager does not provide a default DACL, the TOE uses the default DACL in the subject's access token. The TOE then sets the SE_DACL_PRESENT and SE_DACL_DEFAULTED SD control flags.

- The subject's access token always has a default DACL, which is set by the LSA subcomponent when the token is created.

All tokens are created with an appropriate default DACL, which can be applied to the new objects as appropriate. The default DACL is restrictive in that it only allows the SYSTEM SID and the user SID that created the object to have access. The SYSTEM SID is a special SID representing TSF trusted processes.

6.2.2.1.5 DAC Management

- The following are the four methods that DACL changes are controlled:
  - Object owner: Has implicit WRITE_DAC access.
  - Explicit DACL change access: A user granted explicit WRITE_DAC access on the DACL can change the DACL.
  - Take owner access: A user granted explicit WRITE_OWNER access on the DACL can take ownership of the object and then use the owner's implicit WRITE_DAC access.
  - Take owner privilege: A user with SeTakeOwner privilege can take ownership of the object and then user the owner's implicit WRITE_DAC access.

6.2.2.1.6 Reference Mediation

Access to objects on the system is generally predicated on obtaining a handle to the object. Handles are usually obtained as the result of opening or creating an object. In these cases, the TSF ensures that access validation occurs before creating a new handle for a subject. Handles may also be inherited from a parent process or directly copied (with appropriate access) from another subject. In all cases, before creating a handle, the TSF ensures that that the security policy allows the subject to have the handle (and thereby access) to the object. A handle always has a granted access mask associated with it. This mask indicates, based on the security policy, which access rights to the object that the subject was granted. On every attempt to use a handle, the TSF ensures that the action requested is allowed according to the handle's granted access mask. In a few cases, objects are directly accessed by name without the intermediate step of obtaining a handle first. In these cases, the TSF checks the request against the access policy directly (rather than checking for a granted access mask).

6.2.2.2 Mandatory Integrity Control

In addition to discretionary access control, the TSF provides mandatory integrity control (MIC). MIC uses integrity levels and mandatory policies to evaluate access. Processes (i.e., subjects) and most named kernel objects (see **Mandatory Integrity Control Policy (FDP_ACC.1(MIC))**) are assigned integrity labels that determinethe protection level for the object. For example, even when an object's DACL allows write access by the subject, a subject with a low integrity level cannot write to an object with a medium integrity level,.

Integrity labels specify the integrity levels of securable objects and processes. Integrity labels are represented by integrity SIDs. The integrity SID for a securable object is stored in its SACL, which can be

read by an authorized user.[59] The SACL contains a SYSTEM_MANDATORY_LABEL_ACE ACE that in turn contains the integrity SID. Any object without an integrity SID is treated as if it had medium integrity. The integrity SID for a process is stored in its access token.

The integrity labels implemented in Windows are:

- **Untrusted**: Used by processes started by the Anonymous group.
- **Low**: Used by protected mode processes (such as Internet Explorer), blocks write access to most objects, such as files and registry keys, on the system.
- **Medium:** Normal applications being launched when user account control (UAC) is enabled.
- **High:** Applications launched through administrator elevation when UAC is enabled, or normal applications if UAC is disabled.
- **System**: Services and other system-level applications (such as WinLogon).

Each process has a mandatory policy represented by its TOKEN_MANDATORY_POLICY which can have one of the following values:

- TOKEN_MANDATORY_POLICY_OFF: No mandatory policy is enforced for the access token.
- TOKEN_MANDATORY_POLICY_NO_WRITE_UP: The mandatory policy is enforced and the subject cannot write objects with higher integrity labels.
- TOKEN_MANDATORY_POLICY_NEW_PROCESS_MIN: A process that is created is assigned an integrity label that is the lesser of the parent-process and that of the executable file for the process.
- TOKEN_MANDATORY_POLICY_VALID_MASK: A combination of TOKEN_MANDATORY_POLICY_NO_WRITE_UP and TOKEN_MANDATORY_POLICY_NEW_PROCESS_MIN.

By default processes are assigned TOKEN_MANDATORY_POLICY_VALID_MASK.

Processes can access objects that have an integrity level lower than or equal to their own integrity level. The SYSTEM_MANDATORY_LABEL_ACE ACE in the SACL of a securable object contains an access mask that specifies the access that subjects with integrity levels lower than the object are granted (i.e., the mandatory policy for the object). The values defined for this access mask are:

- SYSTEM_MANDATORY_LABEL_NO_WRITE_UP: A subject with a lower integrity label cannot write an object with a higher integrity label.
- SYSTEM_MANDATORY_LABEL_NO_READ_UP: A subject with a lower integrity label cannot read an object with a higher integrity label.
- SYSTEM_MANDATORY_LABEL_NO_EXECUTE_UP: A subject with a lower integrity label cannot execute an object with a higher integrity label.

[59] By implication, when Windows creates the DACL and SACL for a new object the integrity label will be inherited from the parent object.

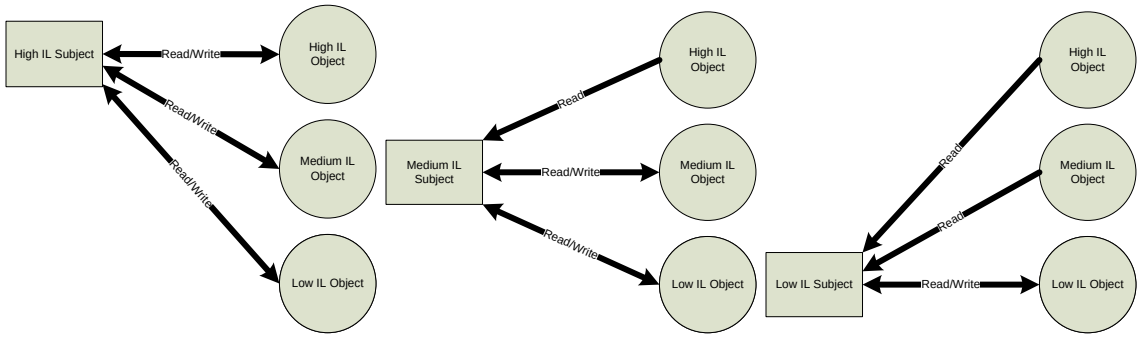
By default, every object, except processes and threads, has an access mask of SYSTEM_MANDATORY_LABEL_NO_EXECUTE_UP. Processes and threads have an access mask of SYSTEM_MANDATORY_LABEL_NO_READ_UP.

Note that both the process policy and the object policy are applied simultaneously whenever a subject attempts to access an object. The allowed access will effectively be the logical intersection of the respective policies. However, if a process does not have the TOKEN_MANDATORY_POLICY_NO_WRITE_UP value (i.e., either TOKEN_MANDATORY_POLICY_OFF or TOKEN_MANDATORY_POLICY_NEW_PROCESS_MIN, then the object label and policy are irrelevant.

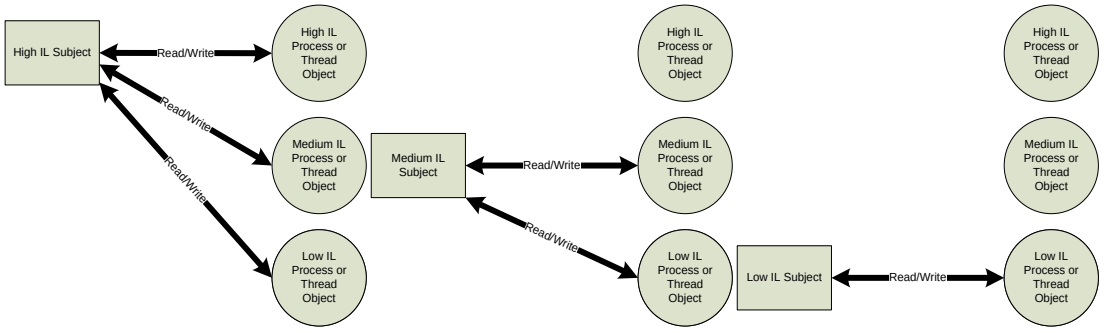
In the default cases, the MIC policy rules are twofold:

1. If the integrity label of the subject is greater than or equal to the integrity label of the object, then a write (the flow of information from the subject to the object) or execute (when applicable for the object) is permitted.
2. If the integrity label of the object is less than or equal to the integrity label of the subject, then a read (the flow of information from the object to the subject) is permitted.

The rules for hierarchical integrity attribute schemes as defined by the MIC rules above are reflected in the following three diagrams.



By default, process and thread objects are an exception to the integrity policy rules implemented by Windows. For these objects there is a stipulation of “no read up”. This is reflected in the following three diagrams.



When an object is created, it is assigned an integrity label equal to that of the creating process. Subsequently, only a process with the "modify an object label" privilege (i.e., SeRelabelPrivilege, assigned to an authorized administrator) can change the label of the object.

Processes associated with non-administrative users receive a medium integrity level by default (e.g., when they log in). Processes associated with administrative users receive a high integrity level by default. Processes started by another subject are assigned the lower of the integrity level assigned to the subject or the integrity level assigned to the executable file associated with the subject, unless the mandatory policy for the process does not indicate TOKEN_MANDATORY_POLICY_NEW_PROCESS_MIN in which case the integrity label of the executable file will be assigned.

6.2.2.3 Information Flow Control and Protection

6.2.2.3.1 Windows Firewall and IPsec

Windows implements a suite of Internet standard protocols including IPsec, Internet Key Exchange (IKE), and Internet Security Association and Key Management Protocol (ISAKMP). IPsec can be used to secure traffic using IP addresses or port number between two computers or between two computers. IKE and ISAKMP establish security associations within the IPsec protocol suite.

IPsec policies specify the functions that IPsec must perform for a given outbound or inbound packet and include a list of filters to be applied to IP packet traffic. Filters can be specified to control traffic flow based upon source IP address, destination IP address, protocol, source port, destination port or network interface. An action of permit or block can be specified within the filter for specific flows of traffic based upon source IP address, destination IP address, protocol, source port, destination port, or network interface.

During OS initialization, the Windows Firewall will read the IPsec configured policies to initialize the firewall. The TSF will start enforcing these filters before sending any outbound packets and before allowing any inbound packets to proceed.

The TSF also prevents the disclosure and modification of user data using IPsec policies and filters. IPsec policies and filters can be configured only by an authorized administrator and can be configured to apply actions to specify traffic flow characteristics such as encrypting or signing. IPsec uses the CNG algorithms to provide data confidentiality and integrity for IP packets.

See Section **6.2.6.3.3, Internal TOE Protection**, for more about IPsec.

The TSF allows for the authorized administrator to define a Windows Firewall policy that can specify which ports the TSF will allow connections. Using IPsec, this policy will then enforce the blocking of all other incoming connections and allows in only that which is a reply to a previous request that went out.

When the Windows Firewall is enabled by the authorized administrator, the TSF enforces the Windows Firewall policy that will block all unsolicited incoming packets except for packets destined for network ports and network profiles specified by the authorized administrator. To support this policy the TSF uses TCP/IP (IPv4 or IPv6).

When the Windows Firewall is enabled, it opens and closes the communications ports that are used by authorized applications (i.e., executable programs). Windows Firewall maintains a table of connections that are initiated on behalf of the other systems on the "protected" side of the local network, and inbound Internet traffic can reach the "protected" network only when the table holds a matching entry. The notion of a "protected" side of the network is generalized to concept of Public, Private, or Domain network profiles, in which the network profile is a group of security settings from the most restrictive (Public network profile) to the least restrictive (Domain network profile).The administrator configures which "services" will be permitted by Windows Firewall to access the network for each kind of profile. The administrator also configures Internet Control Message Protocol (ICMP) message handling. Service settings and ICMP options are per interface. The Windows Firewall implements both stateful packet filtering and port mapping.

Note that the Windows Firewall is enabled by default on all products. Windows 8 and Windows RT the settings are configured to be restrictive to prevent unsolicited incoming requests and allow outbound traffic.

6.2.2.4 Residual Data Protection Function

The TOE ensures that any previous information content is unavailable upon allocation to subjects and objects. The TSF ensures that resources exported to user-mode processes do not have residual information in the following ways:

- All objects are based on memory and disk storage. Memory allocated for objects is either overwritten with all zeros or overwritten with the provided data before being assigned to an object. Read/write pointers prevent reading beyond the space used by the object. Only the exact value of what is most recently written can be read and no more. For varying length objects, subsequent reads only return the exact value that was set, even though the actual allocated size of the object may be greater than this. Objects stored on disk are restricted to only disk space used for that object.
- Subjects have associated memory and an execution context. The TSF ensures that the memory associated with subjects is either overwritten with all zeros or overwritten with user data before allocation. In addition, the execution context (processor registers) is initialized when new threads within a process are created and restored when a thread context switch occurs.

SFR Mapping:

The **User Data Protection** function satisfies the following SFRs:

- **FDP_ACC.1(DAC):** The SRM mediates all access to named objects, including kernel-based objects and user-mode TSF server-based objects. All access to objects is predicated on the SRM validating the access request. In the case of most objects, this DAC validation is performed on initial access (e.g., "open") and subsequent use of the object is via a handle that includes a granted access mask. For some objects, every reference to the object requires a complete DAC validation to be performed.

- **FDP_ACF.1(DAC):** The TSF enforces access to user objects based on SIDs and privileges associated with subjects contained in tokens, and the security descriptors for objects. The rules governing access are defined as part of the DAC algorithm.
- **FDP_ACC.1(MIC)** and **FDP_ACF.1(MIC)**: The TSF enforces a Mandatory Integrity Control policy for process access to most objects covered by the DAC policy. The rules are enforced to ensure that process accesses to objects conform to rules that involve applicable attributes on the processes and objects as summarized earlier.
- **FDP_IFC.1(OSPP) and FDP_IFF.1(OSPP):** The TSF controls the flow of traffic from one Windows system's TSF to another using IPsec to enforce filters that can be configured to restrict the flow of traffic based upon source IP address, destination IP address, source port, destination port, and protocol.The TSF protects the flow of information by filtering unauthenticated IP traffic to prevent exploitation of resources on the internal network and gathering of unauthorized information. The TSF controls the flow of traffic into a Windows system's TSF by providing the capability to block all unsolicited traffic with the exceptions of traffic targeted to ports specified by the authorized administrator.
- **FDP_RIP.2**: The TSF ensures that previous information contents of resources used for new objects are not discernable in the new object via zeroing or overwriting of memory and tracking read/write pointers for disk storage. Every process is allocated new memory and an execution context. Memory is zeroed or overwritten before allocation.
- **FMT_MSA.1(DAC)**: The ability to change the DAC policy is controlled by the ability to change an object's DACL.
- **FMT_MSA.1(OBJ)**: Only the authorized administrator for a DAC object can change object ownership to another user or delete the object.
- **FMT_MSA.1(MIC)**: The ability to change Mandatory Integrity Control related security attributes is restricted to processes holding a specific privilege (i.e., SeRelabelPrivilege) allowing the modification of object labels.
- **FMT.MSA.3(DAC)**: The TSF provides restrictive default values for security attributes used to provide access control via the process's default DACLs which only allows access to the SYSTEM and the user creating the object. Users who create objects can specify a security descriptor with a DACL to override the default.
- **FMT_MSA.3(MIC)**: By default, objects and processes are assigned Mandatory Integrity labels and policies that prevent writing to higher integrity labels and read access to processes and threads at higher integrity labels. The defaults cannot be changed during process or object creation, though some attributes can be changed later per the FMT_MSA.1(MIC) requirement.
- **FMT_MSA.3(OSPP)**: By default, Windows has a very restrictive default firewall policy. Filters can be defined and assigned to restrict traffic flow from one TSF to another. However, by default, there are no filters assigned and traffic is allowed to flow in an unrestricted manner. Only the authorized administrator can define or modify the IPsec filters that specify the rules for traffic flow.
- **FMT_MSA.4**: The initial values for DACLs, tokens, ownership, and MIC labels are established as described throughout section the subsections of **6.2.2**.

- **FMT_MTD.1(OSPP):** Only an authorized administrator can define, modify, delete, and manage the firewall policies.
- **FMT_MTD.1(GEN)**:[60] Only an authorized administrator can modify the values for security attributes except for the additional restrictions described in these SFRs.
- **FMT_REV.1(DAC)**: The ability to revoke access to an object is controlled by the ability to change the DACL and is governed by the same conditions for FMT_MSA.1(DAC) above. The changed DACL is effective upon subsequent access checks against the object.
- **FMT_REV.1(OBJ)**: The ability to revoke security attributes for integrity control is restricted to only authorized administrators.

6.2.3 Cryptographic Protection

Cryptography API: Next Generation (CNG) API is designed to be extensible at many levels and agnostic to cryptographic algorithm suites. An important feature of CNG is its native implementation of the Suite B algorithms, including algorithms for AES (128, 192, 256 key sizes), the SHA-1 and SHA-2 family (SHA-256, SHA-384 and SHA-512) of hashing algorithms, elliptic curve Diffie Hellman (ECDH), and elliptical curve DSA (ECDSA) over the NIST-standard prime curves P-256, P-384, and P-521.

Protocols such as the Internet Key Exchange (IKE), and Transport Layer Security (TLS), make use of elliptic curve Diffie-Hellman (ECDH) included in Suite B.

Deterministic random bit generation (DRBG) is implemented in accordance with NIST Special Publication 800-90. Windows generates random bits by taking the output of a cascade of two SP800-90 AES-256 counter mode based DRBGs in kernel-mode and four cascaded SP800-90 AES-256 DRBGs in user-mode; all are seeded from the Windows entropy pool. The entropy pool is populated using the following values:

- An initial entropy value from a seed file provided to the Windows OS Loader at boot time (512 bits of entropy). [61]
- A calculated value based on the high-resolution CPU cycle counter which fires after every 1024 interrupts (a continuous source providing 16384 bits of entropy).
- Random values gathered periodically from the Trusted Platform Module (TPM), if one is available on the system (320 bits of entropy on boot, 384 bits thereafter).
- Random values gathered periodically by calling the RDRAND CPU instruction, if supported by the CPU (256 bits of entropy).

The main source of entropy in the system is the CPU cycle counter which tracks hardware interrupts. This is a sufficient health test; if the computer were not accumulating hardware and software interrupts

[60] This requirement is for general management of security functions, the above description is a specific instance
[61] The Windows OS Loader implements a SP 800-90 AES-CTR-DRBG and passes along 384 bits of entropy to the kernel for CNG to be use during initialization. This DBRG uses the same algorithms to obtain entropy from the CPU cycle counter, TPM, and RDRAND as described above.

it would not be running and therefore there would be no need for random bit generation. In the same manner, a failure of the TPM chip or processor would be a critical error that halts the computer. In addition, if the user chooses to operate Windows in the FIPS validated mode, it will run FIPS 140 AES-256 Counter Mode DBRG Known Answer Tests (instantiate, generate) and Dual-EC DRBG Known Answer Tests (instantiate, generate) on start-up. Windows always runs the SP 800-90-mandated self-tests for AES-CTR-DRBG during a reseed and runs the Dual-EC reseed self-test if the user chooses to operate Windows in the FIPS validated mode.

Each entropy source is independent of the other sources and does not depend on time. The CPU cycle counter inputs vary by environmental conditions such as data received on a network interface card, key presses on a keyboard, mouse movement and clicks, and touch input.

The TSF defends against tampering of the random number generation (RNG)/ pseudorandom number generation (PRNG) sources by encapsulating its use in Kernel Security Device Driver. The interface for the Windows random number generator is BCryptGenRandom. By default, the CNG provider for random number generation is the AES_CTR_DRBG, however CNG can be configured to use the Dual EC DRBG.

The encryption and decryption operations are performed by independent modules, known as Cryptographic Service Providers (CSPs) which are FIPS 140-2 Level 1 compliant. Windows generates symmetric keys (AES keys) using the FIPS Approved random number generator.

In addition to encryption and decryption services, the TSF provides other cryptographic operations such as hashing and digital signatures. Hashing is used by other FIPS Approved algorithms implemented in Windows (the hashed message authentication code, RSA, DSA, and EC DSA signature services, Diffie-Hellman and elliptic curve Diffie-Hellman key agreement, and the Dual EC random bit generator).

The hash-based message authentication code functions (HMAC) are based on SHA-1, SHA-256, SHA-384, and SHA-512, have the following characteristics:

Table 6-5 HMAC Characterisitics

HMAC Algorithm	Hash function Used	Block Size	Output MAC Length	Key Length / Key Size
HMAC-SHA-1	SHA-1	512 bits	20 bytes	The key size is 10-63 bytes when the key size is less than the block size and the key size is 65 to 1024 bytes when the key size is greater than the block size. The key size may also equal the block size. The key size is variable.
HMAC-SHA-256	SHA-256	512 bits	32 bytes	Same as HMAC-SHA-1
HMAC-SHA-384	SHA-384	1024 bits	48 bytes	The key size is 24-127 bytes when the key size is less than the block size and the key size is 129-1024 bytes when the key size is greater than the block

				size. The key size may also equal the block size. The key size is variable.
HMAC-SHA-512	SHA-512	1024 bits	64 bytes	The key size is 32-127 bytes when the key size is less than the block size and the key size is 129-1024 bytes when the key size is greater than the block size. The key size may also equal the block size. The key size is variable.

The compliance with FIPS Approved algorithms is:

Table 6-6 Cryptographic Algorithm Standards and Evaluation Methods

Cryptographic Operation	Standard	Evaluation Method
Encryption/Decryption	FIPS 197 AES For ECB, CBC, CFB8, CCM, and GCM modes	NIST CAVP #2197, #2216
Digital signature	FIPS 186-4 rDSA	NIST CAVP #1134, #1133
Digital signature	FIPS 186-4 DSA	NIST CAVP #687
Digital signature	FIPS 186-4 ECDSA	NIST CAVP #341
Hashing	FIPS 180-3 SHA-2	NIST CAVP #1903
Keyed-Hash Message Authentication Code	FIPS 198-2 HMAC	NIST CAVP #1345
Random number generation	NIST SP 800-90 CTR_DRBG	NIST CAVP #258 for CTR_DRBG
Key agreement	NIST SP 800-56A ECDH	NIST CAVP #36

Table 6-7 Cryptographic Modules in Windows

Cryptographic Module	OS Edition	NIST CMVP Certificate
Boot Manager	Windows 8 and Windows RT	1895
Code Integrity (CI.dll)	Windows 8 and Windows RT	1897
Cryptographic Primitives Library (BCryptPrimitives.dll)	Windows 8 and Windows RT	1892
Dump Filter (DumpFVE.sys)	Windows 8 and Windows RT	1899
Kernel-mode Cryptographic Primitives Library (cng.sys)	Windows 8 and Windows RT	1891
Enhanced DSS and Diffie-Hellman Cryptographic Services Provider (DSSENH.dll)	Windows 8 and Windows RT	1893
RSA Enhanced Cryptographic Services Provider (RSAENH.dll)	Windows 8 and Windows RT	1894
Windows OS Loader (WinLoad)	Windows 8 and Windows RT	1896

Windows Resume (WinResume)	Windows 8	1898

The TSF includes a key isolation service designed specifically to host secret and private keys in a protected process to mitigate tampering or access to sensitive key materials. The TSF performs key entry and output in accordance with the FIPS 140-2 standard. The TSF performs a key error detection check on each transfer of key (internal and intermediate transfers). The TSF prevents archiving of expired (private) signature keys. The TSF destroys non-persistent cryptographic keys – note that all keys which are subject to destruction are stored within the cryptomodule that was subject to FIPS 140-2 certification – after a cryptographic administrator-defined period of time of inactivity. The TSF overwrites each intermediate storage area for plaintext key/critical cryptographic security parameter (i.e., any storage, such as memory buffers, that is included in the path of such data). This overwriting is performed as follows:

- For non-volatile memories other than EEPROM and Flash, the overwrite is executed three or more times using a different alternating data pattern each time upon the transfer of the key/critical cryptographic security parameter to another location.
- For volatile memory and non-volatile EEPROM and Flash memories, the overwrite is a single direct overwrite consisting of a pseudo random pattern, followed by a read-verify upon the transfer of the key/critical cryptographic security parameter to another location.

SFR Mapping:

The **Cryptographic Protection** function satisfies the following SFRs:

- **FCS_COP.1(AES)**: The TSF uses the AES (128-bit and higher key sizes) algorithm to encrypt user data and only allows the user who encrypted the data to decrypt the data by ensuring that the SID of the subject requesting decryption is the same as the SID of the subject that requested encryption of the data.
- **FCS_COP.1(AES), FCS_COP.1 (SIGN), FCS_COP.1 (HASH), FCS_COP.1(HMAC) FCS_COP.1 (DSA), FCS_COP.1 (DH KA), FCS_COP.1 (EC KA):** See **Table 6-5 Cryptographic Algorithm Standards and Evaluation Methods**.
- **FCS_CKM.1(SYM), FCS_CKM.1 (ASYM), FCS_CKM.1(AUTH):** See **Table 6-5 Cryptographic Algorithm Standards and Evaluation Methods**.
- **FCS_CKM_EXT.4:** See **Table 6-5 Cryptographic Algorithm Standards and Evaluation Methods**.
- **FCS_SRV_EXT.1:** See **Table 6-5 Cryptographic Algorithm Standards and Evaluation Methods.**
- **FCS_RBG_EXT.1:** See **Table 6-5 Cryptographic Algorithm Standards and Evaluation Methods**.

6.2.4 Identification and Authentication Function

The TOE requires each user to be identified and authenticated prior to performing TSF-mediated functions on behalf of that user, with one exception, regardless of whether the user is logging on interactively or is accessing the system via a network connection. The exception is the function allowing a user to shut the system down; however, an authorized administrator may disable even that function if it is not appropriate for a given environment.

6.2.4.1 User Attribute Database

Windows maintains account databases (collectively referred to as user attribute database) that fully define user and group accounts. These definitions include:

- Account name: used to represent the account in human-readable form
- Security Identifier (SID): a user identifier or group identifier used to represent the user or group account within the TOE.
- Groups: used to associate group memberships with the account
- Privileges: used to associate TSF privileges with the account
- Logon rights: used to control the logon methods available to the account (e.g. the “logon locally” right allows a user to interactively logon to a given system)
- Password (only for user accounts): used to authenticate a userwhen they log onto Windows or need to unlock a workstation.
-
- X509 certificates that represent human users and machines which are used for network authentication
- Miscellaneous control information: to keep track of additional security relevant account attributes such as allowable periods of usage, whether the account has been locked, whether the password has expired, password history, and time since the password was last changed
- Other non-security relevant information: used to complete the definition with other useful information such a user’s real name and the purpose of the account.

Note that security relevant roles are associated with users by virtue of group assignments (which in turn have privilege assignments) and are not otherwise specifically identified.

6.2.4.2 Logon Type

Windows supports the following types of user logon:

Table 6-8 Logon Types in Windows

Logon Type	Description	Purpose
Interactive	Logon locally	This logon type is intended for users who will be interactively using the computer, such as a user being logged on by a terminal server, remote shell, or similar process. This logon type has the additional expense of caching logon information for disconnected operations; therefore, it is

		inappropriate for some client/server applications, such as a mail server.
Network	Access this computer from the network	This logon type is intended for high performance servers. The LogonUser function does not cache credentials for this logon type
Service	Logon as a service	Indicates a service-type logon. The account provided must have the service privilege enabled.
Batch	Logon as a batch job	This logon type is intended for batch servers, where processes may be executing on behalf of a user without their direct intervention. This type is also for higher performance servers, such as mail or web servers. The LogonUser function does not cache credentials for this logon type
Unlock	Unlock screen saver	This logon type is for WinLogon extension DLLs that log on users who will be interactively using the computer. This logon type can generate a unique audit record that shows when the workstation was unlocked.
New Credentials	Clone and create new security token	This logon type allows the caller to clone its current security token and specify new credentials for outbound connections. The new logon session has the same local identifier but uses different credentials for other network connections.

Each of the logon types has a corresponding user logon right that can be assigned to user and group accounts to control the logon methods available to users associated with those accounts.

6.2.4.3 Trusted Path and Re-authentication

For initial interactive logon, a user invokes a trusted path in order to ensure the protection of identification and authentication information. The trusted path is invoked by using the **Ctrl+Alt+Del** key sequence, which is always captured by the TSF (i.e., it cannot be intercepted by an untrusted process), and the result will be a logon dialog that is under the control of the TSF. Once the logon dialog is displayed, the user can enter their identity (username and domain) and authenticator. Additionally, the TSF uses IPsec, among other techniques, to provide a trusted path between TSFs to ensure the protection of the I&A information transferred between TSFs.

A user can change their password either during the initial interactive log or while logged on. To change a user's password, the user must invoke the trusted path by using the **Ctrl+Alt+Del** key sequence. The screen allows the user to select an option to change their password. If selected, a second screen is displayed which requires the user to enter their current password and a new password. The TSF will change the password only if the TSF can successfully authenticate the user using the current password that is entered (see section Logon Process for a description of the authentication process) and if the new password conforms to the password policy defined by the administrator.

Another action that requires the user to invoke the trusted path by using the **Ctrl+Alt+Del** key sequence and re-authenticate themselves is session locking and unlocking (see the **Session Locking Function** section).

6.2.4.4 Logon Banner

An authorized administrator can configure the interactive logon screen to display a logon banner with a title and warning. This logon banner will be displayed immediately before the interactive logon dialog (see above) and the user must select "OK" to exit the banner and access the logon dialog.

Furthermore, when a user logs onto an interactive session, they are presented with the date and time of their last successful login along with the number of unsuccessful attempts that may have occurred since then. This information persists in a screen that can be dismissed using the "OK" button.

6.2.4.5 Account Policies

Every Windows computer contains a user account policy database. The account policy is controlled by an authorized administrator and allows the definition of a password policy and an account lockout policy with respect to interactive logons.

The password policy includes:

- The number of historical passwords to maintain in order to restrict changing passwords back to a previous value
- The maximum password age before the user is forced to change their password
- The minimum password age before the user is allowed to changed their password
- The minimum password length when changing to a new password (0 or higher)
- Pre-defined password complexity requirements that can be enabled or disabled.

The account lockout policy includes:

- Duration (including an option for an indefinite lockout requiring an administrator to enable the account) of the account lockout once it occurs
- Number of failed logon attempts before the account will be locked out
- The amount of time after which the failed logon count will be reset.

These policies allow Windows to make appropriate decisions and change user security attributes in the absence of an authorized administrator. For example, Windows will expire a password automatically when the maximum password age has been reached. Similarly, it will lock an account once the predefined number of failed logon attempts have occurred and will subsequently only unlock the account as the policy dictates. There are also related policies to restrict features available to authorized users (e.g., frequency of password change, size of password, reuse of passwords).

After the password has expired, the user must reset their password as part of re-authentication, see the **Session Locking Function** section for how the user initiates the password change.

6.2.4.6 Logon Process

All logons are treated essentially in the same manner regardless of their source (e.g., interactive logon, network interface, internally initiated service logon) and start with an account name, domain name (which may be NULL; indicating the local system), and credentials that must be provided to the TSF.

The domain name parameter indicates where the account is defined. If the local machine name (or NULL) is selected for the domain name, the local SAM user account database is used.

Note that if the User Account Control feature is enabled, the process of any user with authorized administrator privileges are initially assigned only those privileges available to standard users. Subsequently, if that process attempts to perform an operation requiring the privileges of an authorized administrator, the user will be prompted to confirm whether the additional privileges should be granted. If acknowledged, the full set of privileges are enabled in the process' token.

When a web site or another computer requests authentication through NTLM or Kerberos, an Update Default Credentials or Save Password check box appears in the Net Logon UI dialog box. If the user selects the check box, the Credential Manager keeps track of the user's name, password, and related information for the authentication service to use.

The next time that service is used, the Credential Manager automatically supplies the stored credential. If it is not accepted, the user is prompted for the correct access information. If access is granted, the Credential Manager overwrites the previous credential with the new one.

6.2.4.6.1 Network Logon Support

Public key certificate network logon is supported by the TLS/SSL Security Provider that implements the Security Protocol Provider security package. This package provides support for several network security protocols, and in particular SSL version 3.0, TLS versions 1.0, 1.1 and 1.2. In the TOE, security package APIs are not directly accessible, rather they are accessed via LSA Authentication APIs. The TLS/SSL Security Provider authenticates connections, and/or encrypts messages between clients and servers. When an application needs to use a network resource on an authenticated channel, the LSA accesses the TLS/SSL Security Service Provider (SSP) via the SSP interfaces. Windows implements TLS/SSL in accordance with RFCs 5246 and 2246 with extensions specified in RFCs 4366, 3546, and 4681 and additional supported cipher suites as specified in RFCs 3268, 4492, and 5289. For more information regarding the Windows implementation of TLS/SSL refer to http://msdn.microsoft.com/en-us/library/dd207968(PROT.10).aspx. Furthermore, the Windows TLS/SSL feature is designed to employ the applicable cryptographic protection mechanisms described earlier.

Digest network logon is supported by the Microsoft Digest Access Authentication Package. Digest performs user authentication for LSA Authentication in support of network logon attempts. Interactive logons cannot be performed using Digest Access. Digest implements a network security protocol, in this case digest challenge/response authentication that supports remote network logon user authentication and other network security services according to RFCs 2617 and 2831. For more information regarding the Windows implementation of Digest Access Authentication refer to http://msdn.microsoft.com/en-

us/library/cc227906(PROT.10).aspx. Furthermore, the Windows Digest feature is designed to employ the applicable Cryptographic Protection mechanisms described earlier.

6.2.4.7 Impersonation

In some cases, specifically for server processes, it is necessary to impersonate another user in order to ensure that access control and accountability are performed in an appropriate context. To support this, the TSF has an internal mechanism for a server process to impersonate the identity of a client process. As described above, each process has a token that primarily includes account SIDs, privileges, logon rights, and a default DACL. Normally, each thread within a process uses the process' token for its security context. However, a thread can be assigned an impersonation token that would be used instead of the processes token when making access checks and generating audit data. Hence, that thread is impersonating the client that provided the impersonation token. Impersonation stops when the impersonation token is removed from the thread or when the thread terminates.

When communicating with a server, the client can select an impersonation level that constrains whether and how a server may impersonate the client. The client can select one of four available impersonation levels: anonymous, identify, impersonate, and delegate:

- Anonymous allows the server to impersonate the client, but the impersonation token does not contain any client information.
- Identify allows the server to get the identity and privileges of the client, but can not impersonate the client.
- Impersonate enables the server to impersonate, i.e., perform access checks as the client's security context on the local system to access resources local to the server's TSF.
- Delegate enables server can impersonate the client's security context on local and remote systems.

6.2.4.8 Restricted Tokens

Whenever a process is created or a thread is assigned an impersonation token, Windows allows the caller to restrict the token that will be used in the new process or impersonation thread. Specifically, the caller can remove privileges from the token, assign a deny-only attribute to SIDs, and specify a list of restricting SIDs. That is:

- Removed privileges are simply not present in the resulting token.
- SIDs with the deny-only attribute are used only to identify access denied settings when checking for access, but ignore any access allowed settings.
- When a list of restricting SIDs is assigned to a token, access is checked twice once using the tokens enabled SIDs and again using the restricting SIDs. Access is granted only if both checks allow the desired access.

6.2.4.9 Strength of Authentication

As indicated above, Windows provides a set of functions that allow the account policy to be managed. These functions include the ability to define account policy parameters, including minimum password length. The minimum password length can be configured to require up to 16 characters. The

administrator can also configure the number of passwords for Windows to remember so that a user cannot reuse a previous password until the password has changed the configure number of times.

During authentication, the Logon UI will not provide feedback that will reduce the probability of guessing a password beyond eliminating that one choice. However, if an account becomes locked, Windows will report that the account is disabled. Furthermore, the TSF forces a delay between attempts, such that there can be no more than ten attempts per minute.

For each subsequent failed logon following five consecutive failed logon occurrences in the last sixty seconds, the logon component sleeps for 30 seconds before showing a new logon dialog. It therefore supports the I&A function that no more than ten interactive logon attempts are possible in any sixty second period.

When Kerberos is used, the password requirements are the same as those described above. However, there are both Ticket Granting Tickets and Service Tickets that are used to store, protect, and represent user credentials and are effectively used in identifying and authenticating the user. Session keys are initially exchanged using a hash of the user's password for a key.

6.2.4.10 Certificates Used in IPsec and TLS

IPsec and TLS use X.509 certificates in order to authenticate computers within the enterprise network (IPsec) or for general purpose web traffic (TLS). Apart from any ephermal key negotiation that is part of the networking protocol, keying material can not be loaded or handled directly by a user.

Certificates are loaded into separate stores for each user, the service account, and an account that represents the computer's identity. A certificate can be added manually by a user with either the Certificates MMC snap-in. Access to the certificate store is controlled by discretionary access control as described above.

SFR Mapping:

The **Identification and Authentication** function satisfies the following SFRs:

- **FIA_AFL.1, FMT_MTD.1(Threshold), FMT_MTD.1(Re-enable)**: The TSF locks the account after the administrator-defined threshold of unsuccessful logon attempts has occurred. The account will remain locked until an authorized administrator unlocks it. Note that the limit of 10 attempts per minute is enforced regardless of the threshold. While locked, responses to the user will not reflect whether the authentication attempt was successful.
- **FIA_ATD.1(USR)**: Each Windows machine has a user attribute database for local machine accounts. Each user attribute database describes accounts, including identity, group memberships, password (e.g., authentication data), privileges, logon rights, allowable time periods of usage, as well as other security-relevant control information. Security-relevant roles are associated with users via group memberships and privileges.
- **FIA_UAU.1(OS):** An authorized administrator can configure the TSF to allow no TSF-mediated functions prior to authentication.

- **FIA_UAU.5**: In the evaluated configuration, Windows can authenticate human users based on password.
- **FIA_UAU.7**: During an interactive logon, the TSF echoes the users password with "*" characters to prevent disclosure of the user's password.
- **FIA_UID.1**: An authorized administrator can configure the TSF to allow no TSF-mediated functions prior to identification.
- **FIA_USB.1(USR):** Each process and thread has an associated token that identifies the responsible user (used for audit and access), associated groups (used for access), privileges, Mandatory Integrity Control integrity labels and policies, and logon rights held by that process or thread on behalf of the user. Normally the security attributes assigned to a process and its threads remain unchanged, but when User Account Control is enabled, processes belonging to an authorized administrators are initially assigned an access token limited to access rights available to other standard users and must interactively acknowledge the escalation before the process can use the full authorized administrator access rights. Note that any changes to user security attributes are applied when the user next logs in and a new subject is created to act on the user's behalf.
- **FIA_PK_EXT.1, FMT_MTD.1(X509)**: Windows uses X.509v3 certificates for IPsec and TLS, Windows will generate most certificates automatically but an authorized administrator can generate a certificate.

6.2.5 Security Management Function

The TOE supports the definition of roles as well as providing a number of functions to manage the various security policies and features provided by the TOE.

6.2.5.1 Roles

The notion of a role within the TOE is generally realized by assigning group accounts and privileges to a given user account. Whenever that user account is used to logon, the user will be assuming the role that corresponds with the combination of groups and privileges that it holds. While additional roles could be defined, this ST defines the authorized administrator role as being special.

The Administrator role is defined as any user account that is assigned one of the security-relevant privileges (e.g., Take Owner privilege) or is made a member of one or more of the several pre-defined administrative groups (e.g., Administrators, Cryptographic Operators, and Backup Operators local groups). The Administrator Guide fully identifies all security-related privileges and administrative groups, and provides advice on how and when to assign them to user accounts. A user assumes an administrator role by logging on using a user account assigned one of these privileges or group membership.

Any user that can successfully logon is considered to be in an authorized user, though this is not specifically identified as a security management role per se. Of the functions all users can perform, creating objects, modifying DAC permissions of their objects, and managing their own passwords are particularly notable.

6.2.5.2 Security Management Functions

The TOE supports a number of policies and features that require appropriate management. With few exceptions, the security management functions are restricted to an authorized administrator. This constraint is generally accomplished by privilege or access control (e.g., SD), and occasionally by a specific SID requirement (e.g., "Administrators"). The TOE supports security management functions for the following security policies and features:

- **Audit Policy**: The audit policy management functions allow an authorized administrator the ability to enable and disable auditing, to configure which categories of events will be audited for success and/or failure, and to manage (e.g., clear) and access the security event log. An authorized administrator can also define specifically which user and access mode combinations will be audited for specific objects in the TOE.
- **Account Policy**: The account policy management functions allow only an authorized administrator to define constraints for passwords (password complexity requirements), account lockout (due to failed logon attempts) parameters, and Kerberos key usage parameters. The constraints for passwords restrict changes by including minimum password length, password history, and the minimum and maximum allowable password age. If the maximum password age is exceeded, the corresponding user cannot logon until the password is changed. The account lockout parameters include the number of failed logon attempts (in a selected interval) before locking the account and duration of the lockout. The Kerberos key usage parameters primarily specify how long various keys remain valid. While an authorized administrator can change passwords and a user can change their own password, the TSF does not allow any user (including the authorized administrator) to read passwords. Additionally, the authorized administrator can define the advisory warning message displayed before access to the TOE is granted.
- **Account Database Policy**: The account database management functions allow an authorized administrator to define, assign, and remove security attributes to and from both user and group accounts, both locally and for a domain, if applicable. The set of attributes includes account names, SIDs, passwords, group memberships, and other security-relevant and non-security relevant information. Of the set of user information, only the password can be modified by a user that is not an authorized administrator. Specifically, an authorized administrator assigns an initial password when an account is created and may also change the password like any other account attribute. However, a user may change their password. This is enforced by requiring the user to enter their old password in order to change the password to a new value.
- **User Rights Policy**: The user rights management functions allow an authorized administrator to assign or remove user and group accounts to and from specific logon rights and privileges.
- **Local [Group] Policy**: The policy management functions allow an authorized administrator to define accounts, user right assignments, and TOE machine/computer security settings, etc. for the computer. The policies effectively modify the policies (e.g., machine security settings, and user rights policy) defined for the corresponding TSFs or users.

- **IPsec Policy**: The IPsec management functions allow an authorized administrator to define whether and how (e.g., protocols and ports to be protected, outbound and/or inbound traffic, with what cryptographic algorithms) IPsec will be used to protect traffic among distributed TSFs.
- **Disk Quota Policy**: The disk quota management functions allow an authorized administrator to manage disk quotas for NTFS volumes. More specifically, the functions allow an authorized administrator to enable or disable disk quotas, define default disk quotas, and define actions to take when disk quotas are exceeded.
- **DAC Policy**: The DAC functions allow authorized users to modify access control attributes associated with a named object.
- **Other**: The TSF also allows the administrator the ability to modify the time and modify object integrity labels.

6.2.5.3 Valid Attributes

The TSF ensures that only valid values are accepted as security attributes for the password. Valid values are values that meet the password complexity restrictions as defined by the administrator. For example, the minimum password length should be set to greater than or equal to eight characters by the administrator. Subsequently, attempts to create passwords shorter than eight characters will not be accepted by the TSF.

Beyond this, the TSF generally checks parameters provided for security management and other functions in order to ensure that only valid values are accepted in order to avoid the potential to get into unknown or bad states of operation.

6.2.5.4 Remote Management

Management applications that are built as Microsoft Management Console (MMC) snap-ins provide native support for remote management of computers within the enterprise network using remote procedure calls. In addition,deploy IPsec will protect network traffic used for remote management..

SFR Mapping;

The **Security Management** function satisfies the following SFRs:

- **FMT_MOF.1(Pass)**: Only an authorized administrator can configure the settings that serve to constrain acceptability of authentication data (length, history, complexity, etc.). The TSF ensures that values for password security attributes meet the password complexity and other restrictions, as defined by the administrator. Furthermore, each security management function is generally designed to ensure that values offered by administrators are valid before being accepted.
- **FMT_MTD.1(GEN)**: As a rules security management functions are limited to authorized administrators as indicated above. Manipulation of a user's own authentication data is a notable exception.
- **FMT_MTD.1(Audit)**: Only an authorized administrator can view or clear the security event log. Furthermore, only authorized administrators can manipulate the security event log to cause applicable files to be archived or deleted.

- **FMT_MTD.1(Init-Attr), FMT_MTD.1(Mod-Auth)**: Only an authorized administrator can initially assign a password to a user account. Subsequently, both an authorized administrator and the user corresponding to the password can change a password.
- **FMT_MTD.1(Mod-Attr)**: Only an authorized administrator can define user accounts and group accounts, define user/group associations (e.g., group memberships), assign privileges and user rights to accounts, as well as define other security-relevant and non-security relevant user attributes, with the exception of passwords (which are addressed above) and private/public key pairs.
- **FMT_REV.1(Admin)**: Only an authorized administrator can remove security attributes from users and group accounts. By default such changes take effect the next time the user attempts to log in.
- **FMT_SMR.1**: The TOE supports the definition of an authorized administrator through the association of specific privileges and group memberships with user accounts. As described in the User Data Protection section, users are generally allowed to control the security attributes of objects depending upon the access that they have to those objects. Users can also modify their own authentication data (e.g., passwords) by providing their old password for authorization. Additionally, upon the creation of an object, the user creating the object (object creator) can define initial values for its security attributes that override the default values (e.g. DACL).
- **FMT_SMF_RMT.1**: Windows provides remote administration using MMC snap-in applications, network traffic is protected by IPsec. The Internet Information Servcies web server can be supports remote administration through a web interfaces, network traffic is protected by TLS.

6.2.6 TSF Protection Function

6.2.6.1 Time Service

Each hardware platform supported by the TOE includes a real-time clock. The real-time clock is a device that can only be accessed using functions provided by the TSF and serves as the reference clock. Specifically, the TSF provides functions that allow users, including the TSF itself, to query and set the clock, as well as functions to synchronize clocks within a domain. The ability to query the clock is unrestricted, while the ability to set the clock requires the SeSystemtimePrivilege. This privilege is only granted to authorized administrators to protect the integrity of the time service.

Accuracy (which the OS PP describes as “reliable and monotonically increasing”) is described in How the Windows Time Service Works. In addition this communications path can be protected using IPsec.

Windows capabilities that are included in the OS protection profile evaluation which use the time service are:

- Audit record generation
- Network expirations for authentication and data access
- Session timeout and screen locking
- X.509 certificate generation, revocation, and expiration

These capabilities use the interfaces described at http://msdn.microsoft.com/en-us/library/ms725473(v=vs.85).aspx. Public documentation about time functions in Windows is located at http://msdn.microsoft.com/en-us/library/ms724962(v=vs.85).aspx. This describes the different types of time services offered to developers.

SFR Mapping:

The **TSF Protection** function satisfies the following SFRs:

- **FPT_STM.1**: The real-time clock in each Windows platform, in conjunction with periodic domain synchronization and restricting the ability to change the clock to authorized administrators, provides a reliable source of time stamps for the TSF.

6.2.6.2 Architecture and Self-Protection

6.2.6.2.1 Internal TOE Protection

The TOE protects against unauthorized disclosure and modification of data when it is transferred between physically separated parts of the TOE using a suite of Internet standard protocols including IPsec and ISAKMP. IPsec can be used to secure traffic using IP addresses or port number between two computers. IPsec does not apply to broadcast or multicast traffic. IPsec services are configurable on the system to allow for a variety of security services including data origin authentication, message integrity, and data confidentiality. The TOE implements IPsec with a set of kernel subsystems and user-mode trusted servers. IPsec allows for the application of a set of security services to be applied to IP data based on predefined IPsec policies. IPsec policies specify the functions that IPsec must perform for a given outbound or inbound packet. IPsec policies identify the local host algorithms and associated attributes, mode of communication (transport is the only mode included in the evaluation configuration), and a list of filters to be applied to IP packet traffic. Filters are used to associate inbound and outbound packets with a specific IPsec policy. They specify the source and destination IP addresses, ports, and protocol. IPsec uses the elliptic curve Diffie-Hellman (ECDH) to provide data confidentiality and integrity for IP packets.

Keys are exchanged between computers within the TOE before secured data can be exchanged by the establishment of a security agreement between the two computers. In this security agreement, called a Security Association (SA), both agree on how to exchange and protect information. To build this agreement between the two computers, the Internet Engineering Task Force (IETF) has established a standard method of security association and key exchange resolution named IKE which is applied in the TOE. A SA is the combination of a negotiated key, security protocol, and Security Parameters Index (SPI), which collectively define the security used to protect the communication from sender to receiver. The SPI is a unique identifying value in the SA that is used to distinguish among multiple SAs that exist at the receiving computer.

In order to ensure successful and secure communication, IKE performs a two-phase operation. Confidentiality and authentication are ensured during each phase by the use of encryption (i.e., AES per FCS_COP.1(AES)) and authentication algorithms that are agreed upon by the two computers during security negotiations.

The IPsec management functions allow an authorized administrator to define the IPsec Policy including whether and how (i.e., protocols and ports to be protected, outbound and/or inbound traffic, with what cryptographic algorithms) IPsec will be used to protect traffic among distributed TSFs.

The evaluated configurations support the use of Kerberos and the use of public key certificate for machine authentication in the IKE processing. IKE processing includes the validation of the peer's certificate (including path validation) and signature payload verification.

The IPsec policy MMC snap-in allows an administrator to select the authentication method based on public key certificate. To use a public key certificate for authentication services, the CA associated with the public key certificate and the associated root CA can be chosen.

The IKE processing also processes ISAKMP payload messages to allow IKE processing to obtain each other's public key value. IPsec policies and filters may be configured to reject the packet or audit the event if the results of a service applied to a packet challenges the integrity of the packet (modification, insertion of data, replay of data).

6.2.6.2.2 TSF Failure Recovery

When a failure occurs within the TSF, the TSF will immediately halt and produce a memory dump to a location on the system volume that is readable only be an authorized administrator. The machine will remain in a halted state until user intervention occurs. A user can then reset the system in order to reboot the operating system. During the subsequent boot, the user will be presented the option of booting into a limited mode (e.g., where only some device drivers and services are loaded or started) in order to attempt any necessary recovery functions (after logging in).

6.2.6.3 TSF Code Integrity

The TSF Boot Manager is an Authenticode-signed image file, based on the Portable Executable (PE) image file format. A SHA hash based signature and a public key certificate chain are embedded in the boot manager Authenticode signed image file under the "Certificate" IMAGE_DATA_DIRECTORY of the IMAGE_OPTIONAL_HEADER of the file. This public key certificate chain ends in a root public key. The boot manager uses the embedded SHA hash based signature and public key certificate chain to validate its own integrity. A SHA hash of the boot manager image file is calculated for the whole file, with the exception of the following three elements which are excluded from the hash calculation: the CheckSum field in the IMAGE_OPTIONAL_HEADER, the IMAGE_DIRECTORY_ENTRY_SECURITY IMAGE_DATA_DIRECTORY, and the public key certificate table, which always resides at the end of the image file.

If the boot manager is validated, then the root public key of the embedded public key certificate chain must match one of the Microsoft root public keys which indicate that Microsoft is the publisher of the

boot manager. These root public keys are necessarily hardcoded in the boot manager. If the boot manager cannot validate its own integrity, then the boot manager does not continue to load other modules and displays an error message.

After the boot manager determines its integrity, it attempts to load one application from the following list of boot applications:

- Winload.exe or Winload.efi, the boot application used to load the Windows kernel
- ntoskrnl.exe, the Windows kernel
- winresume.exe or winresume.efi, the boot application used for resuming from the hibernation file “hiberfil.sys”
- memtest.exe, a memory testing application.

These boot applications are also Authenticode signed image files. For each of the Windows boot applications, the boot manager uses the embedded trusted SHA hash based signature and public key certificate chain within the boot application’s IMAGE_OPTIONAL_HEADER to validate the integrity of the boot application before attempting to load it. Except for the following three elements which are excluded from the hash calculation, a SHA hash of a boot application image file is calculated for the whole file: the CheckSum field in the IMAGE_OPTIONAL_HEADER, the IMAGE_DIRECTORY_ENTRY_SECURITY IMAGE_DATA_DIRECTORY, and the public key certificate table, which always resides at the end of the image file.

If the boot application is validated, then the root public key of the embedded public key certificate chain must match one of the hardcoded Microsoft’s root public keys. If the boot manager cannot validate the integrity of the boot application, then the boot manager does not continue to load Windows modules, instead displaying an error message below along with the full name of the boot application that failed the integrity check.

After the boot application’s integrity has been determined, the boot manager attempts to load the boot application. When configured, the full volume encryption (FVE) facility within the Windows boot manager also conducts its own independent SHA-256 hash based validation of the boot applications as identified above. If the boot application is successfully loaded, the boot manager then transfers execution to the loaded application.

After the Winload boot application is loaded, it receives the transfer of execution from the boot manager. During its execution, Winload attempts to load the Windows kernel (ntoskrnl.exe) together with a number of critical drivers. Among the modules that Winload must validate in the Portable Executable (PE) image file format, are the cryptography related modules listed below. These modules are listed in a hardcoded list.

- The Windows kernel;
- The Windows kernel security device driver;
- The Windows code integrity library module; and
- The BitLocker™ drive encryption filter driver.

The four image files above have their trusted SHA hashes stored in catalog files that reside in the local machine catalog directory.

Because they are PKCS #7 SignedData messages, catalog files are signed. The root public key of the certificate chain used to verify the signature of a Microsoft's catalog file must match one of the Microsoft's root public keys indicating that Microsoft is the publisher of the Windows image files. These Microsoft's root public keys are hardcoded in the Winload boot application.

If the image files are validated, their SHA hashes, as calculated by the Winload boot application, must match their trusted SHA hashes in a Microsoft's catalog file, which has been verified by the Winload boot application. A SHA hash of an image file is calculated for the whole file, with the exception of the following three elements which are excluded from the hash calculation: the CheckSum field in the IMAGE_OPTIONAL_HEADER, the IMAGE_DIRECTORY_ENTRY_SECURITY IMAGE_DATA_DIRECTORY, and the public key certificate table, which always resides at the end of the image file.

Should the Winload boot application be unable to validate the integrity of one of the Windows image files, the Winload boot application does not continue to load other Windows image files. Rather it displays an error message, along with the full name of the Windows image file which does not have the validated integrity.

In addition, Windows File Protection maintains a set of protected files that are stored in a cache along with cryptographic hashes of each of those files. Once the system is initialized, Windows File Protection is loaded and will scan the protected files to ensure they have valid cryptographic hashes. Windows File Protection also registers itself to be notified should any of the protected files be modified so that it can recheck the cryptographic checksum at any point while the system is operational. Should the any of the cryptographic hash checks fail, the applicable file will be restored from the cache.

SFR Mapping:

The **TSF Protection** function satisfies the following SFRs:

- **FPT_ITT.1**: The TSF provides internet-based standard protocols for IP security and Key management. IPsec with AH and ESP implementations protect transferred TSF data from disclosure and modification. AH provides data signature functionality to protect against modification; ESP provides encryption to protect against disclosure as well as modification. The TSF implements IP AH. AH provides integrity, authentication and anti-replay. AH uses a hashing algorithm, such as SHA, to compute a keyed message hash for each IP packet. Additionally, IPsec policies and filters may be configured to reject the packet or audit the event if the results of a service applied to a packet challenges the integrity of the packet (modification, insertion of data, replay of data). Any packets rejected as a result of an integrity error are rejected and the event is audited.

6.2.7 Session Locking Function

Windows provides the ability for a user to lock their interactive logon session at their own volition or after a user-defined inactivity timeout. Windows also provides the ability for the administrator to specify the interval of inactivity after which the session will be locked. This policy will be applied to either the local machine. If both the administrator and a standard user specify an inactivity timeout period, Windows will lock the session when the shortest time period expires.

Once a user has a desktop session, they can invoke the session locking function by using the same key sequence used to invoke the trusted path (**Ctrl+Alt+Del**). This key sequence is captured by the TSF and cannot be intercepted or altered by any user process. The result of that key sequence is a menu of functions, one of which is to lock the workstation. The user can also lock their desktop session by going to the Start screen, selecting their logon name, and then choosing the "Lock" option.

Windows constantly monitors the mouse, keyboard, touch display, and the orientiation sensor for inactivity in order to determine if they are inactive for the specified time period. After which, Windows will lock the workstation and execute the screen saver unless the user is streaming video such as a movie. Note that if the workstation was not locked manually, the TSF will lock the display and start the screen saver program if and when the inactivity period is exceeded.

After the computer was locked, in order to unlock their session, the user either presses a key or swipes the display, or the user must provide either the **Ctrl+Alt+Del** key combination on a system with a physical keyboard, or press the Windows button and power buttons simulataneously on tablet systems, if the **Interactive Logon: Do not required CTRL+ALT+DEL** policy is set to disabled.[62] The result is that Windows will present an authentication dialogon the secure desktop. The user must then re-enter their authentication data, which has been cached by the local system from the initial logon, after which the user's display will be restored and the session will resume.

SFR Mapping:

The **Session Locking** function satisfies the following SFR:

- **FTA_SSL.1**: Windows allows users and the authorized administrator to define an inactivity interval, after which their session will be locked. The locked display has only the user's default background, instructions to unlock, and optionally the output from a user-selected screen saver program. The user must re-enter their password to unlock the workstation.
- **FTA_SSL.2**: Windows also allows a user to directly invoke the session lock as described above.
- **FMT_MTD.1(GEN)**:[63] The TSF allows an authorized user to define and modify the time interval of inactivity before the session associated with that user will be locked.

[62] This policy is defined under Local Policies / Security Options.
[63] This requirement, which is not a OS PP functional requirement, is for general management of security functions, the above description is a specific instance.

6.2.8 Trusted Paths / Channels Function

6.2.8.1 TSS Description

6.2.8.1.1 IPsec

The Windows IPsec implementation conforms to RFC 4301, Security Architecture for the Internet Protocol. This is documented publicly in the Windows protocol documentation at section 7.5.1 IPsec (DirectAccess) Overview and covers Windows 8, Windows RT, and Server 2012.[64]

Windows implements both RFCS 2409, Internet Key Exchange (IKEv1), and RFC 4306, Internet Key Exchange version 2, (IKEv2).[65] Windows IPsec supports both tunnel mode and transport mode and provides an option for NAT transversal (reference: section 7.5.5, IPsec Encapsulations).[66] The RAS VPN interface uses tunnel mode only.

The Windows IPsec implementation includes a security policy database (SPD), which states how Windows should process network packets. The SPD uses the traffic source, destination and transport protocol to determine if a packet should be transmitted or received, blocked, or protected with IPsec. (reference: 7.5.3, Security Policy Database Structure).[67] An authorized administrator does not need to define a final catch-all rule which will discard a network packet when no other rules in the SPD apply because Windows will discard the packet. The security policy database includes configuration settings to limit the time and number of sessions before a new key needs to be generated.

Windows implements the encryption algorithms described above in section 5.2.7.1, **Inter-TSF Trusted Channel (FTP_ITC.1 (OS)),** (reference: section 6, Appendix A, Product Behavior).[68] Windows implements HMAC-SHA1, AES-GMAC, and SHA-256 as authentication algorithms as well as Diffie-Hellman Groups 14, 19, 20, and 24 (reference: section 6, Appendix A, Product Behavior), which were evaluated as part of the OS PP evaluation.[69] This applies to both the encapsulating security payload (ESP) and the encrypted payload in IKEv1 and IKEv2. The resulting potential strength of the symmetric key will be 128 or 256 bits of security depending on whether the two computers agreed to using a 128 or 256 AES symmetric key to protect the network traffic.

Windows constructs nonces as specified in RFC 2408, Internet Security Association and Key Management Protocol (ISAKMP) section 3.13.[70] When a random number is needed for either a nonce or for key agreement, Windows uses a FIPS 140-validated random bit generator. When requested, the Windows random bit generator can generate 256 or 512 bits for the caller, the probability of guessing a 256 bit value is 1 in 2^{256} and a 512 bit value is 1 in 2^{512}. When generating the security value *x* used in the

[64] Also available as [MS-WSO], *Windows System Overview*, page 43 for offline reading.
[65] [MS-IKEE], *Internet Key Exchange Protocol Extensions*, page 8.
[66] [MS-WSO], page 45.
[67] [MS-WDO], page 44.
[68] [MS-IKEE], pages 74 – 75.
[69] *Ibid*.
[70] [MS-IKEE], page 51.

IKE Diffie-Hellman key exchange, g^x mod p, Windows uses a FIPS validated key agreement function.[71] See the TSS section Cryptographic Protection for the NIST CAVP validation numbers.

Windows implements peer authentication using 2048 bit RSA certificates,[72] or ECDSA certificates using the P-256 and P-384 curves for both IKEv1 and IKEv2.[73]

While Windows supports pre-shared IPsec keys, it is not recommended due to the potential use of weak pre-shared keys.[74] Windows simply uses the pre-shared key that was entered by the authorized administrator, there is no additional processing on the input data.

Windows operating systems do not implement the IKEv1 aggressive mode option during a Phase 1 key exchange.

The following table summarizes the use of RFCs by Windows:

RFC #	Name	How Used
2407	The Internet IP Security Domain of Interpretation for ISAKMP	Integral part of the Windows Internet Key Exchange (IKE) implementation.
2408	Internet Security Association and Key Management Protocol (ISAKMP)	Integral part of the Windows Internet Key Exchange (IKE) implementation.
2409	The Internet Key Exchange (IKE)	Integral part of the Windows Internet Key Exchange (IKE) implementation.
2986	PKCS #10: Certification Request Syntax Specification; Version 1.7	Public key certification requests issued by Windows.
4106	The Use of Galois/Counter Mode (GCM) in IPsec Encapsulating Security Payload (ESP)	Certain IPsec cryptosuites implemented by Windows.
4109	Algorithms for Internet Key Exchange version 1 (IKEv1)	Certain IPsec cryptosuites implemented by Windows.
4301	Security Architecture for the Internet Protocol	Description of the general security architecture for IPsec.
4303	IP Encapsulating Security Payload (ESP)	Specifies the IP Encapsulating Security Payload (ESP) implemented by Windows.
4304	Extended Sequence Number (ESN) Addendum to IPsec Domain of Interpretation (DOI) for Internet Security Association and Key Management Protocol (ISAKMP)	Specifies a sequence number high-order extension that is implemented by Windows.
4306	Internet Key Exchange (IKEv2) Protocol	Integral part of the Windows Internet Key Exchange (IKE) implementation.
4307	Cryptographic Algorithms for Use in the	Certain IPsec cryptosuites implemented

[71] http://technet.microsoft.com/en-us/library/cc962035.aspx.
[72] [MS-IKEE], page 73.
[73] http://technet.microsoft.com/en-us/library/905aa96a-4af7-44b0-8e8f-d2b6854a91e6.
[74] http://technet.microsoft.com/en-us/library/cc782582(v=WS.10).aspx.

	Internet Key Exchange Version 2 (IKEv2)	by Windows.
4868	Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec	Certain IPsec cryptosuites implemented by Windows.
4945	The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2, and PKIX	Integral part of the Windows Internet Key Exchange (IKE) implementation.
5280	Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	Specifies PKI support implemented by Windows.
5282	Using Authenticated Encryption Algorithms with the Encrypted Payload of the Internet Key Exchange version 2 (IKEv2) Protocol	Certain IPsec cryptosuites implemented by Windows.
5996	Internet Key Exchange Protocol Version 2 (IKEv2)	Integral part of the Windows Internet Key Exchange (IKE) implementation.
6379	Suite B Cryptographic Suites for IPsec	Certain IPsec cryptosuites implemented by Windows.

Table 9 IPsec RFCs Implemented by Windows

Exceptions from the protocols are described in [MS-IKEE].

6.2.8.1.2 TLS

Windows implements TLS to enable a trusted network path.

The following table summarizes the use of RFCs by Windows:

RFC #	Name	How Used
2246	The TLS Protocol Version 1.0	Specifies requirements for TLS 1.0.
3268	Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)	Specifies additional ciphersuites implemented by Windows.
3546	Transport Layer Security (TLS) Extensions	Updates RFC 2246 with TLS 1.0 extensions implemented by Windows.
4366	Transport Layer Security (TLS) Extensions	Obsoletes RFC 3546 Requirements for TLS 1.0 extensions implemented by Windows.
4492	Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)	Specifies additional ciphersuites implemented by Windows.
4681	TLS User Mapping Extension	Extends TLS to include a User Prinicpal Name during the TLS handshake.
5246	The Transport Layer Security (TLS) Protocol Version 1.2	Oboletes RFCs 3268, 4346, and 4366. Specifies requirements for TLS 1.2.
5289	TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)	Specifies additional ciphersuites implemented by Windows.
SSL3	The SSL Protocol Version 3	Specifies requirements for SSL3.

Exceptions from the protocols are described in these documents:

- MS-TLSP Transport Layer Security (TLS) Profile.docx
- RFC 2246 - The TLS Protocol Version 1.0.docx
- RFC 3268 - AES Ciphersuites for TLS.docx
- RFC 3546 Transport Layer Security (TLS) Extensions.docx
- RFC 4366 Transport Layer Security (TLS) Extensions.docx
- RFC 4492 - ECC Cipher Suites for TLS.docx
- RFC 4681 - TLS User Mapping Extension.docx
- RFC 5246 - The Transport Layer Security (TLS) Protocol, Version 1.2.docx
- RFC 5289 - TLS ECC Suites with SHA-256384 and AES GCM.docx
- Internet Draft - SSL3 SSL 3.0 Specification.docx

The Cipher Suites in Schannel article describes the set of TLS cipher suites implemented in Windows (reference: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374757(v=vs.85).aspx).

6.2.8.2 SFR Mapping:

The **Trusted Path / Channels** function satisfies the following SFR:

- **FTP_ITC.1(OS)**: Windows uses both IPsec and TLS to provide a trusted path to other computers; IPsec is used to protect administrative communications.

7 Protection Profile Conformance Claim

This section provides the protection profile conformance claim and supporting justifications and rationale.

7.1 Rationale for Conformance to Protection Profile

This Security Target is in compliance with the ***General Purpose Operating System Protection Profile***, version 3.9, December 2012 (OS PP).

For all of the content incorporated from the OS PP, the corresponding rationale in that protection profile remains applicable; refer to that OS PP for the rationale.

7.2 Security Problem Definition

The core of the security problem definition is formed by the statements of threats, policies, and assumptions that have been copied from the OSPP.

Since the OS PP security problem definition is complete and consistent; the security problem definition for OS PP functional requirements in this security target is defined there; the reader should refer to the OS PP for SPD the conformance claim. The only addition to the security problem definition is to address the threat of cryptographic functionality to be inappropriately accessed which would result in the compromise of the cryptographic mechanisms and the data protected by those mechanisms.

7.3 Security Objectives

The statements of objectives for the TOE and its operational environment have been copied verbatim from the OSPP into section 4.1.1 **OSPP Security Objectives** in the security target and section 4.1.2 **Additional Security Objectives**.

The OS PP provides a mapping for threats defined in the OS PP to security objectives. The following tables maps the additional threat defined in this security target to a security objective and associates the security objective to a policy.

Table 7-1 Mapping Threats to Security Objectives

Threat	Objective for the TOE or the Environment
T.CRYPTO_COMPROMISE	O.CRYPTOGRAPHIC_SERVICES

Table 7-2 Mapping Policies to Security Objectives

Security Policy	Objective for the TOE or the Environment
P.CRYPTOGRAPHY “The TOE shall use standards-based cryptography as a baseline for key management (i.e., generation and destruction) and for cryptographic operations (i.e., encryption, decryption, signature, hashing, and random number generation)”	O.CRYPTOGRAPHIC_SERVICES The TOE will make cryptographic services available to authorized users and/or user applications.

By building upon NIST FIPS-validated cryptography, Windows not only provides, but also augments the cryptographic support offered solely by baseline NIST FIPS-validated cryptography. Windows cryptography supports key management (i.e., generation and destruction of keys) and cryptographic operations (i.e., encryption, decryption, signature, hashing, and random number generation). O.CRYPTOGRAPHIC_SERVICES provides these cryptographic services to authorized users and/or user applications.

7.4 Security Requirements

This section will provide a rationale for the supplemental functional requirements which are not part of the OS PP and also describe how this Security Target reproduced the requirements from the OSPP.

The security target includes several requirements for cryptographic support (FCS_CKM.1(SYM), FCS_CKM.1(ASYM), FCS_CKM.1(AUTH), FCS_CKM_EXT.4, FCS_SRV_EXT.1, FCS_COP.1(AES), FCS_COP.1(SIGN), FCS_COP.1(HASH), FCS_COP.1(HMAC), FCS_COP.1(DH KA), FCS_COP.1(EC KA), FCS_RBG_EXT.1) which contribute to the security objective that "[t]he TOE will make encryption services available to authorized users and/or user applications" (O.CRYPTOGRAPIC_SERVICES).

The requirement to provide basic protection while transferring TSF data (FPT_ITT.1) helps to contribute to the network trusted channel between Windows computers in the domain (O.TRUSTED_CHANNEL), and the requirement to manage any TSF data that is not covered by any other requirement (FMT_MTD.1(GEN)) contributes to the security management objective (O.MANAGE).

Finally, the Security Assurance Requirements within this Security Target are considered.

7.4.1 SFRs from the OSPP, CC Part 2, and the ST

This Security Target includes security functional requirements (SFRs) that can be mapped to SFRs found in the OSPP along with SFRs that describe additional features and capabilities. The mapping from OSPP SFRs to Security Target SFRs along with rationale for operations is presented in **Table 7-1 Rationale for Operations**. SFR operations left incomplete in the OSPP have been completed in this ST and are identified within each SFR in section 5.2 TOE Security Functional Requirements.

Table 7-3 Rationale for Operations

OSPP Requirement	ST Requirement	Operation & Rationale
FAU_GEN.1	FAU_GEN.1(OSPP)	The requirement in the OS PP does not offer assignments or selections. The following refinements were made to better reflect TOE functionality: • A reference to 'the OSPP base' was replaced by a more explicit reference to the FDP_ACF.1(DAC) requirement. • Two additional audit events (g) and (h) were added because the TOE supports these audit events which have historically been deemed important to operating system security. • The reference to the table listing audit events was

OSPP Requirement	ST Requirement	Operation & Rationale
		updated to refer to the table within the ST instead of referring into the OS PP.
FAU_GEN.2	FAU_GEN.2	Reproduced exactly as found in the OS PP with no operations performed.
FAU_SAR.1	FAU_SAR.1	The operations offered by the OS PP were completed in this requirement. The term 'user' was refined to 'authorized administrator' as it is more consistent with the FMT requirements.
FAU_SAR.2	FAU_SAR.2	Reproduced exactly as found in the OS PP with no operations performed.
FAU_SEL.1	FAU_SEL.1	The operation offered by the OS PP was completed in this requirement. A refinement was to clarify text of the SFR in order to utilize a version of the FAU_SEL.1 requirement that this product has satisfied in previous evaluations..
FAU_STG.1	FAU_STG.1	The operation offered by the OS PP was completed in this requirement. The term "audit records" was change to "stored audit records" in the 2nd element in order to use the same terminology as is used in the 1st element. This should prevent a reader from inferring that there are two different sets of audit records being referenced by the two different elements of this SFR.
FAU_STG.3	FAU_STG.3	The operations offered by the OS PP were completed in this requirement. No refinements were performed.
FAU_STG.4	FAU_STG.4(SL)	The operations offered by the OS PP were completed in this requirement. Added a refinement to diffentiate between the two different kind of audit logs that contain audit records relevant to the OS PP requirements.
	FAU_STG.4(OL)	The operations offered by the OS PP were completed in this requirement. Added a refinement to diffentiate between the two different kind of audit logs that contain audit records relevant to the OS PP requirements.
[CC Part 2]	FCS_CKM.1(SYM)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_CKM.1(ASYM)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.

OSPP Requirement	ST Requirement	Operation & Rationale
[CC Part 2]	FCS_CKM.1(AUTH)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_CKM_EXT.4	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_SRV_EXT.1	The operation offered by the extended component definition was completed in this requirement. No refinements have been made.
[CC Part 2]	FCS_COP.1(AES)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_COP.1(SIGN)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_COP.1(HASH)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_COP.1(HMAC)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_COP.1(DH KA)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[CC Part 2]	FCS_COP.1(EC KA)	This requirement from CC Part 2 was added to cover cryptographic functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
[Defined in the ST]	FCS_RBG_EXT.1	The operation offered by the extended component definition was completed in this requirement. No refinements have been made.

OSPP Requirement	ST Requirement	Operation & Rationale
FDP_ACC.1[75]	FDP_ACC.1(DAC)	The operations offered by the OS PP were completed in this requirement.
	FDP_ACC.1(MIC)	The operations offered by the OS PP were completed in this requirement.
FDP_ACF.1	FDP_ACF.1(DAC)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
	FDP_ACF.1(MIC)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FDP_IFC.1	FDP_IFC.1(OSPP)	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FDP_IFF.1	FDP_IFF.1(OSPP)	The operations offered by the OS PP were completed in this requirement. The only refinement made to this requirement is done to resolve the difference in names between this ST and the corresponding requirement in the OS PP.
FDP_RIP.2	FDP_RIP.2	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FIA_AFL.1	FIA_AFL.1	The operations offered by the OS PP were completed in this requirement. Two refinements have been made to this SFR. The first adds “authorized” to the requirement’s use of “administrator”, this corresponds with FMT_SMR.1 better. The second refinement clarifies that only consecutive unsuccessful attempts are counted. This would ensure that a successful authentication would reset the count.
FIA_ATD.1	FIA_ATD.1(USR)	The operation offered by the OS PP was completed in this requirement.
FIA_UAU.1(RITE)	FIA_UAU.1(RITE)	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FIA_UAU.1 (HU)	FIA_UAU.1(OS)	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FIA_UAU.5	FIA_UAU.5	The operations offered by the OS PP were completed in

[75] The application notes within the OS PP surrounding FDP_ACC.1 address the issue of iterating the requirement for multiple policies. It was the intention of the protection profile authors that FDP_ACC.1 and FDP_ACF.1 would be iterated to describe policies supported by the TOE. Thus, the ST iterates these requirments to describe supported policies.

OSPP Requirement	ST Requirement	Operation & Rationale
		this requirement.
FIA_UAU.7	FIA_UAU.7	The requirement is reproduced exactly from the OS PP without operations.
FIA_UID.1	FIA_UID.1	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FIA_USB.1	FIA_USB.1(USR)	The operations offered by the OS PP were completed in this requirement. The term “security attributes” in the 2nd and 3rd element has been refined to “user security attributes”. This refinement is used for clarity to properly scope the requirement.
FIA_PK_EXT.1	FIA_PK_EXT.1	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FMT_MOF.1	FMT_MOF.1(Pass)	The operation offered by the OS PP was completed in this requirement. No refinements have been made. This iteration fully reproduces the required functionality of the corresponding OS PP requirement (FMT_MOF.1). However, it addresses only password-based user authentication.
FMT_MSA.1 [76]	FMT_MSA.1(DAC)	The operations offered by the OS PP were completed in this requirement. A refinement was made to scope this iteration of FMT_MSA.1 to the DAC policy and to cover the management function to change object ownership.
	FMT_MSA.1(OBJ)	FMT_MOF.1 from the OS PP, allows the ST author to specify operations in its first assignment. In order to properly scope the use of this assignment for the "change" operation, it is necessary to also 'refine' the SFR's scope and reduce the set of users permitted to perform the operation.
	FMT_MSA.1(MIC)	The operations offered by the OS PP were completed in this requirement. While the MIC policy operates upon objects that have 'owners', it is intended to be a stricter policy and less discretionary in nature (i.e., it is a policy that cannot be overridden by the request of the end users). Thus, the requirement was refined to eliminate the owner permissions.

[76] The TOE enforces multiple security functional policies. Despite the similarity of the TOE enforced restrictions, the FMT_MSA.1 requirement has been iterated for each SFP.

OSPP Requirement	ST Requirement	Operation & Rationale
FMT_MSA.3(DAC)[77]	FMT_MSA.3 (DAC)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
	FMT_MSA.3 (MIC)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FMT_MSA.3(NI)	FMT_MSA.3 (OSPP)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FMT_MSA.4	FMT_MSA.4	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FMT_MTD.1(AE)	FMT_MTD.1 (AuditSel)	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FMT_MTD.1(AS)	FMT_MTD.1 (Audit)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FMT_MTD.1(AT)	FMT_MTD.1 (AuditStg)	The operations offered by the OS PP were completed in this requirement. A refinements was needed because the TOE does not support any notion of 'add' or 'delete' in the context of the audit threshold.
FMT_MTD.1(AF)	FMT_MTD.1 (AuditFail)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FMT_MTD.1(CM)	FMT_MTD.1 (X509)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FMT_MTD.1(NI)	FMT_MTD.1 (OSPP)	The operations offered by the OS PP were completed in this requirement. No refinements have been made.
FMT_MTD.1(IAT)	FMT_MTD.1 (Threshold)	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FMT_MTD.1(IAF)	FMT_MTD.1(Re-enable)	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
FMT_MTD.1(IAU)[78]	FMT_MTD.1(Init-	The operation offered by the OS PP was completed in this

[77] The TOE enforces multiple security functional policies. Despite the similarity of the TOE enforced restrictions, the FMT_MSA.3 requirement has been iterated for each SFP.

[78] The ST iterated the FMT_MTD.1(IAU) requirement from the OS PP to more clearly define the TOE capabilities regarding the initialization, modification and deletion of security attributes. This approach leads to an observation that the OS PP is asking for a TOE capability which deletes authentication data. Since FIA_ATD.1 requires

OSPP Requirement	ST Requirement	Operation & Rationale
	Attr)	requirement. Refinements are used in this Iteration to reproduce the "initialize" capability from the OS PP requirement.
	FMT_MTD.1(Mod-Attr)	The operation offered by the OS PP was completed in this requirement. Refinements are used in this Iteration to reproduce the "modify" and "delete" capability of the OS PP for all security data except "authentication data" which is covered by the (Mod-Auth) iteration.
	FMT_MTD.1(Mod-Auth)	The operation offered by the OS PP was completed in this requirement. Refinements are used in this Iteration to reproduce the "modify" capability of the OS PP for “authentication data" which is the excluded data that is not covered by the (Mod-Attr) iteration.
[CC Part 2]	FMT_MTD.1(GEN)	This requirement from CC Part 2 was added to cover management functionality that was not included in the OS PP requirements. The operations offered by the Part 2 SFR were completed in this requirement.
FMT_REV.1(OBJ)[79]	FMT_REV.1(OBJ)	The operations offered by the OS PP were completed in this requirement. The OS PP version of the requirement was refined to scope the requirement to all policies except DAC. This was done because the TOE implementation authorizes a different set of users to revoke security attributes under the DAC policy than are authorized under the other policies.
	FMT_REV.1(DAC)	The operations offered by the OS PP were completed in this requirement. The OS PP version of the requirement was refined to scope the requirement to the DAC policy. This was done because the TOE implementation authorizes a different set of users to revoke security attributes under the DAC policy than are authorized under the other policies.
FMT_REV.1(USR)	FMT_REV.1(Admin)	The operations offered by the OS PP were completed in this requirement. Part 'a)' of FMT_REV.1(Admin).2 is refined to provide a more natural description of how Windows works. Conceptually the “user / subject binding” in the OSPP is more like a Unix operating system in which binding during authentication and binding to create a new process are

authentication data for users, this appears to be a conflict in OS PP requirements. This ST attempts to resolve the conflict by iterating and refining the FMT_MTD.1(IAU) requirement.

[79] The ST iterates the FMT_REV.1(OBJ) requirement because restrictions enforced by the TOE is different for the DAC policy than for other policies.

OSPP Requirement	ST Requirement	Operation & Rationale
		essentially the same. In Windows, the binding between a user and their logon session is significantly different than binding when creating a new process.
FMT_SMF_RMT.1	FMT_SMF_RMT.1	This ST maps FTP_ITC.1(OS) to the FTP_ITC.1 requirement of the OS PP. Thus, this SFR is refined to ensure that it reference the proper requirement within the ST as was intended by the PP authors.
FMT_SMR.1	FMT_SMR.1	The operation offered by the OS PP was completed in this requirement. No refinements have been made.
[CC Part 2]	FPT_ITT.1	The operation offered by CC Part 2 was completed in this requirement. The refinement describes that the TOE provides the cryptographic services that protect TSF data from disclosure.
FPT_STM.1	FPT_STM.1	This SFR is reproduced exactly from the OSPP.
FTA_SSL.1	FTA_SSL.1	The operations offered by the OS PP were completed in this requirement. A refinement adds 'user' to 'events', thus scoping the actions identified within the requirement to 'user events'.
FTA_SSL.2	FTA_SSL.2	The operations offered by the OS PP were completed in this requirement. A refinement adds 'user' to 'events', thus scoping the actions identified within the requirement to 'user events'.
FTP_ITC.1	FTP_ITC.1(OS)	The operations offered by the OS PP were completed in this requirement. Refinements were made to remove and add ciphersuites to fully document those provided by the TOE.

7.4.2 Security Assurance Requirements

The statement of security assurance requirements (SARs) found in section 5.5 TOE Security Assurance Requirements, is in conformance with the "Mapping to the Assurance Components of the CC", found in the ***Operating System Protection Profile***. This ST has added the ALC_FLR.3 requirement, to satisfy customer requirements to have more assurance in the security functions of products that are being deployed in security solutions.

7.5 TOE Summary Specification Rationale

This section, in conjunction with section 6, the TOE Summary Specification (TSS), provides evidence that the security functions are suitable to meet the TOE security requirements.

Each subsection in section 6 describes a Security Function (SF) for Windows. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the functional requirements. Furthermore, all the security functions are necessary in order for the TSF to provide the required security functionality.

The set of security functions work together to provide all of the security requirements as indicated in **Table 7-2**. The security functions described in the TOE Summary Specification and listed in the tables below are all necessary for the required security functionality in the TSF.

Table 7-4 Requirement to Security Function Correspondence

Requirement	Audit	User Data Protection	Cryptographic Protection	I & A	Security Management	TSF Protection	Resource Utilization	Session Locking	Trusted Path / Channel
FAU_GEN.1(OSPP)	X								
FAU_GEN.2	X								
FAU_SAR.1	X								
FAU_SAR.2	X								
FAU_SEL.1	X								
FAU_STG.1	X								
FAU_STG.3	X								
FAU_STG.4(SL)	X								
FAU_STG.4(OL)	X								
FCS_CKM.1(SYM)			X						
FCS_CKM.1(ASYM)			X						
FCS_CKM.1(AUTH)			X						
FCS_CKM_EXT.4			X						
FCS_SRV_EXT.1			X						
FCS_COP.1(AES)			X						
FCS_COP.1(SIGN)			X						
FCS_COP.1(HASH)			X						
FCS_COP.1(HMAC)			X						
FCS_COP.1(DH KA)			X						

FCS_COP.1(EC KA)			X						
FCS_RBG_EXT.1			X						
FDP_ACC.1(DAC)		X							
FDP_ACC.1(MIC)		X							
FDP_ACF.1(DAC)		X							
FDP_ACF.1(MIC)		X							
FDP_IFC.1(OSPP)		X							
FDP_IFF.1(OSPP)		X							
FDP_RIP.2		X							
FIA_AFL.1				X					
FIA_ATD.1(USR)				X					
FIA_UAU.1(RITE)				X					
FIA_UAU.1(OS)				X					
FIA_UAU.5				X					
FIA_UAU.7				X					
FIA_UID.1				X					
FIA_USB.1(USR)				X					
FMT_MOF.1(Pass)					X				
FMT_MSA.1(DAC)		X			X				
FMT_MSA.1(OBJ)		X			X				
FMT_MSA.1(MIC)		X			X				
FMT_MSA.3(DAC)		X			X				
FMT_MSA.3(MIC)		X			X				
FMT_MSA.3(OSPP)		X			X				
FMT_MSA.4		X			X				
FMT_MTD.1(AuditSel)	X				X				
FMT_MTD.1(Audit)	X				X				
FMT_MTD.1(AuditStg)	X				X				
FMT_MTD.1(AuditFail)	X				X				
FMT_MTD.1(OSPP)	X				X				
FMT_MTD.1(Threshold)	X				X				
FMT_MTD.1(Re-enable)	X				X				
FMT_MTD.1(Init-Attr)					X				
FMT_MTD.1(Mod-Attr)					X				

FMT_MTD.1(Mod-Auth)					X				
FMT_MTD.1(GEN)	X	X		X	X	X		X	
FMT_REV.1(DAC)					X				
FMT_REV.1(Admin)					X				
FMT_SMF_RMT.1					X				
FMT_SMR.1					X				
FPT_ITT.1						X			
FPT_STM.1						X			
FTA_SSL.1								X	
FTA_SSL.2								X	
FTP_ITC.1(OS)									X

8 Appendix A: List of Abbreviations

Abbreviation	Meaning
3DES	Triple DES
ACE	Access Control Entry
ACL	Access Control List
ACP	Access Control Policy
AD	Active Directory
ADAM	Active Directory Application Mode
AES	Advanced Encryption Standard
AGD	Administrator Guidance Document
AH	Authentication Header
ALPC	Advanced Local Process Communication
ANSI	American National Standards Institute
API	Application Programming Interface
APIC	Advanced Programmable Interrupt Controller
BTG	BitLocker To Go
CA	Certificate Authority
CBAC	Claims Basic Access Control, see DYN
CBC	Cipher Block Chaining
CC	Common Criteria
CD-ROM	Compact Disk Read Only Memory
CIFS	Common Internet File System
CIMC	Certificate Issuing and Management Components
CM	Configuration Management; Control Management
COM	Component Object Model
CP	Content Provider
CPU	Central Processing Unit
CRL	Certificate Revocation List
CryptoAPI	Cryptographic API
CSP	Cryptographic Service Provider
DAC	Discretionary Access Control
DACL	Discretionary Access Control List
DC	Domain Controller
DEP	Data Execution Prevention
DES	Data Encryption Standard
DH	Diffie-Hellman
DHCP	Dynamic Host Configuration Protocol
DFS	Distributed File System
DMA	Direct Memory Access
DNS	Domain Name System
DS	Directory Service
DSA	Digital Signature Algorithm

DYN	Dynamic Access Control
EAL	Evaluation Assurance Level
ECB	Electronic Code Book
EFS	Encrypting File System
ESP	Encapsulating Security Protocol
FEK	File Encryption Key
FIPS	Federal Information Processing Standard
FRS	File Replication Service
FSMO	Flexible Single Master Operation
FTP	File Transfer Protocol
FVE	Full Volume Encryption
GB	Gigabyte
GC	Global Catalog
GHz	Gigahertz
GPC	Group Policy Container
GPO	Group Policy Object
GPOSPP	US Government Protection Profile for General-Purpose Operating System in a Networked Environment
GPT	Group Policy Template
GPT	GUID Partition Table
GUI	Graphical User Interface
GUID	Globally Unique Identifiers
HTTP	Hypertext Transfer Protocol
HTTPS	Secure HTTP
I/O	Input / Output
I&A	Identification and Authentication
IA	Information Assurance
ICF	Internet Connection Firewall
ICMP	Internet Control Message Protocol
ICS	Internet Connection Sharing
ID	Identification
IDE	Integrated Drive Electronics
IETF	Internet Engineering Task Force
IFS	Installable File System
IIS	Internet Information Services
IKE	Internet Key Exchange
IP	Internet Protocol
IPv4	IP Version 4
IPv6	IP Version 6
IPC	Inter-process Communication
IPI	Inter-process Interrupt
IPsec	IP Security
ISAPI	Internet Server API
IT	Information Technology

KDC	Key Distribution Center
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LPC	Local Procedure Call
LSA	Local Security Authority
LSASS	LSA Subsystem Service
LUA	Least-privilege User Account
MAC	Message Authentication Code
MB	Megabyte
MMC	Microsoft Management Console
MSR	Model Specific Register
NAC	(Cisco) Network Admission Control
NAP	Network Access Protection
NAT	Network Address Translation
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
NLB	Network Load Balancing
NMI	Non-maskable Interrupt
NTFS	New Technology File System
NTLM	New Technology LAN Manager
NUMA	Non-Uniform Memory Access
OS	Operating System
PAE	Physical Address Extension
PC/SC	Personal Computer/Smart Card
PIN	Personal Identification Number
PKCS	Public Key Certificate Standard
PKI	Public Key Infrastructure
PP	Protection Profile
RADIUS	Remote Authentication Dial In Service
RAID	Redundant Array of Independent Disks
RAM	Random Access Memory
RAS	Remote Access Service
RC4	Rivest's Cipher 4
RID	Relative Identifier
RNG	Random Number Generator
RPC	Remote Procedure Call
RSA	Rivest, Shamir and Adleman
RSASSA	RSA Signature Scheme with Appendix
SA	Security Association
SACL	System Access Control List
SAM	Security Assurance Measure
SAML	Security Assertion Markup Language
SAR	Security Assurance Requirement
SAS	Secure Attention Sequence

SD	Security Descriptor
SHA	Secure Hash Algorithm
SID	Security Identifier
SIP	Session Initiation Protocol
SIPI	Startup IPI
SF	Security Functions
SFP	Security Functional Policy
SFR	Security Functional Requirement
SMB	Server Message Block
SMI	System Management Interrupt
SMTP	Simple Mail Transport Protocol
SP	Service Pack
SPI	Security Parameters Index
SPI	Stateful Packet Inspection
SRM	Security Reference Monitor
SSL	Secure Sockets Layer
SSP	Security Support Providers
SSPI	Security Support Provider Interface
ST	Security Target
SYSVOL	System Volume
TCP	Transmission Control Protocol
TDI	Transport Driver Interface
TLS	Transport Layer Security
TOE	Target of Evaluation
TPM	Trusted Platform Module
TSC	TOE Scope of Control
TSF	TOE Security Functions
TSS	TOE Summary Specification
UART	Universal Asynchronous Receiver / Transmitter
UI	User Interface
UID	User Identifier
UNC	Universal Naming Convention
US	United States
UPN	User Principal Name
URL	Uniform Resource Locator
USB	Universal Serial Bus
USN	Update Sequence Number
v5	Version 5
VDS	Virtual Disk Service
VPN	Virtual Private Network
VSS	Volume Shadow Copy Service
WAN	Wide Area Network
WCF	Windows Communications Framework
WebDAV	Web Document Authoring and Versioning

WebSSO	Web Single Sign On
WDM	Windows Driver Model
WIF	Windows Identity Framework
WMI	Windows Management Instrumentation
WSC	Windows Security Center
WU	Windows Update
WSDL	Web Service Description Language
WWW	World-Wide Web
X64	A 64-bit instruction set architecture
X86	A 32-bit instruction set architecture

9 Appendix B: Basic Functional Specification and Interfaces

This appendix is a list of the interfaces which were used to satisfy the CC assurance requirement for a basic functional specification (ADV_FSP.1); the API reference for Windows is at http://msdn.microsoft.com.

9.1 Functional Specification – Interfaces Table Legend

The following is a legend for the interfaces table used in each of the Functional Specification sections:

Legend:
SCE Id – Numeric identifier for each security check or effect associated with each interface
Interface Name – Name of the interface
Search Term – Public name of the interface
Design Information – URL to the documentation for the interface
Secuirty Functional Class – The SFR class that applies to the interface
Security Functional Requirement – The SFR that applies to the interface

9.2 User Data Protection (FDP)

9.2.1 Discretionary Access Control Policy

Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
(FMT_MSA.1(DAC))
Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
Static Attribute Value Inheritance for Discretionary Access (FMT_MSA.4)
Revocation for Object Access for DAC (FMT_REV.1(DAC))

The interfaces to the TSF where access control is enforced for the DAC policy are identified in the table below by the FDP_ACF.1(DAC) and FDP_ACC.1(DAC) security functional requirement pairs. The interfaces utilized to modify security descriptors are indicated by the FMT_MSA.1(DAC) security functional requirement. The interfaces used to manage security descriptor default values are indicated by the FMT_MSA.3(DAC) security functional requirement.

9.2.1.1 Interfaces

The functional specification evidence associated with the interfaces for FDP_ACF.1(DAC) and FDP_ACC.1(DAC), and related audits and management operations are indicated in the table below (the legend for the below table is in section 1).

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15345	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15346	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15355	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy(FMT_MSA.3(DAC))
15349	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15781	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15350	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
15351	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15357	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy(FMT_MSA.3(DAC))
15354	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15782	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15373	RegOpenKey	RegOpenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724895(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15374	RegOpenKey	RegOpenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724895(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15375	RegOpenKey	RegOpenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724895(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15783	RegOpenKey	RegOpenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724895(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)

15376	RegOpenKeyEx	RegOpenKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724897(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15377	RegOpenKeyEx	RegOpenKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724897(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15378	RegOpenKeyEx	RegOpenKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724897(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15784	RegOpenKeyEx	RegOpenKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724897(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15472	CreateEvent	CreateEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682396(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15478	CreateEventEx	CreateEventEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682400(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15510	OpenEvent	OpenEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15511	OpenEvent	OpenEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.	FDP: User Data	Complete Access Control for Discretionary Access

		85).aspx	Protection	(FDP_ACC.1(DAC))
15512	OpenEvent	OpenEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15483	CreateMutex	CreateMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682411(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15489	CreateMutexEx	CreateMutexEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682418(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15504	OpenMutex	OpenMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684315(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15505	OpenMutex	OpenMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684315(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15506	OpenMutex	OpenMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684315(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15495	CreateSemaphore	CreateSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682438(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15501	CreateSemaphoreEx	CreateSemaphoreEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682446(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))

15507	OpenSemaphore	OpenSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684326(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15508	OpenSemaphore	OpenSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684326(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15509	OpenSemaphore	OpenSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684326(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16024	CreateSymbolicLink	CreateSymbolicLink: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363866(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15341	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15340	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15339	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15343	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))

15785	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16378	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16379	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15372	DeleteFile	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15371	DeleteFile	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15370	DeleteFile	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15787	DeleteFile	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15368	OpenFile	OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15365	OpenFile	OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15362	OpenFile	OpenFile: http://msdn.microsoft.com/en-	FDP: User	Security Attribute Based

		us/library/windows/desktop/aa365430(v=vs.85).aspx	Data Protection	Access Control for Discretionary Access (FDP_ACF.1(DAC))
15366	OpenFile	OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15786	OpenFile	OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15942	CreateNamedPipe	CreateNamedPipe: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365150(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15515	CreateTransactionManager	CreateTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366014(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15520	OpenTransactionManager	OpenTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366316(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15521	OpenTransactionManager	OpenTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366316(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15519	OpenTransactionManager	OpenTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366316(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

15788	OpenTransactionManager	OpenTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366316(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15513	CreateTransaction	CreateTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366011(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15522	OpenTransaction	OpenTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366315(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15523	OpenTransaction	OpenTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366315(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15524	OpenTransaction	OpenTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366315(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15789	OpenTransaction	OpenTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366315(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15525	CreateResourceManager	CreateResourceManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366009(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15527	OpenResourceManager	OpenResourceManage:r http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/windows/desktop/aa366311(v=vs.85).aspx	Protection	(FDP_ACC.1(DAC))
15528	OpenResourceManager	OpenResourceManage:r http://msdn.microsoft.com/en-us/library/windows/desktop/aa366311(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15529	OpenResourceManager	OpenResourceManage:r http://msdn.microsoft.com/en-us/library/windows/desktop/aa366311(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15790	OpenResourceManager	OpenResourceManage:r http://msdn.microsoft.com/en-us/library/windows/desktop/aa366311(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15530	CreateEnlistment	CreateEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366006(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15532	OpenEnlistment	OpenEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366305(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15533	OpenEnlistment	OpenEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366305(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15534	OpenEnlistment	OpenEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366305(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

15791	OpenEnlistment	OpenEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366305(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15539	CreateFileMapping	CreateFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366537(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15541	CreateFileMappingFromApp	CreateFileMappingFromApp: http://msdn.microsoft.com/en-us/windows/desktop/hh994453(v=vs.85)	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15543	CreateFileMappingNuma	CreateFileMappingNuma: http://msdn.microsoft.com/en-us/windows/desktop/aa366539(v=vs.85)	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15550	OpenFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15551	OpenFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15552	OpenFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15792	OpenFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15556	CreateDesktop	CreateDesktop:	FAU: Security	Audit Data Generation

		http://msdn.microsoft.com/en-us/library/windows/desktop/ms682124(v=vs.85).aspx	Audit	(FAU_GEN.1(OSPP))
15555	CreateDesktop	CreateDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682124(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15554	CreateDesktop	CreateDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682124(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15557	CreateDesktop	CreateDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682124(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15793	CreateDesktop	CreateDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682124(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15553	CreateDesktopEx	CreateDesktopEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682127(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15536	CreateDesktopEx	CreateDesktopEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682127(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15535	CreateDesktopEx	CreateDesktopEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682127(v=vs.	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access

		85).aspx		(FDP_ACF.1(DAC))
15537	CreateDesktopEx	CreateDesktopEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682127(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15794	CreateDesktopEx	CreateDesktopEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682127(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15547	CreateWindowStation	CreateWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682496(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15546	CreateWindowStation	CreateWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682496(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15545	CreateWindowStation	CreateWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682496(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15548	CreateWindowStation	CreateWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682496(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15795	CreateWindowStation	CreateWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682496(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15561	OpenDesktop	OpenDesktop:	FAU: Security	Audit Data Generation

		http://msdn.microsoft.com/en-us/library/windows/desktop/ms684303(v=vs.85).aspx	Audit	(FAU_GEN.1(OSPP))
15560	OpenDesktop	OpenDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684303(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15559	OpenDesktop	OpenDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684303(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15797	OpenDesktop	OpenDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684303(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15564	OpenInputDesktop	OpenInputDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684309(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15562	OpenInputDesktop	OpenInputDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684309(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15563	OpenInputDesktop	OpenInputDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684309(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15798	OpenInputDesktop	OpenInputDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684309(v=vs.	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		85).aspx		
35	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
8608	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
33	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
8609	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
196	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
8619	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
195	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16389	NtOpenDirectoryObject	NtOpenDirectoryObject:	FAU: Security	Audit Data Generation

		http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	Audit	(FAU_GEN.1(OSPP))
15567	OpenWindowStation	OpenWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684339(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15566	OpenWindowStation	OpenWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684339(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15565	OpenWindowStation	OpenWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684339(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15796	OpenWindowStation	OpenWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684339(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16305	GetSecurityInfo[80]	GetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446654(v=vs.85).aspx	FMT: Security Management	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
15568	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.	FMT: Security Management	Management of Security Attributes for Discretionary Access Control

[80] The required audit record for DAC policy enforcement is generated by the corresponding open interface that returns the handle granting the requested query access for the given object, e.g. for files this occurs in the CreateFile API or for Desktop object in the OpenDesktop API. These audits are tested in the corresponding “Open interface” in the Access Control test variations.

		85).aspx		(FMT_MSA.1(DAC))
15569	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.85).aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16468	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.85).aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
15799	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15800	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15613	CreateJobObject	CreateJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682409(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15625	OpenJobObject	OpenJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684312(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15626	OpenJobObject	OpenJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684312(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15627	OpenJobObject	OpenJobObject:	FAU: Security	Audit Data Generation

		http://msdn.microsoft.com/en-us/library/windows/desktop/ms684312(v=vs.85).aspx	Audit	(FAU_GEN.1(OSPP))
15816	OpenJobObject	OpenJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684312(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15615	CreateThread	CreateThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682453(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15628	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15629	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15630	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15817	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15617	CreateProcess	CreateProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682425(v=vs.	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy

		85).aspx		(FMT_MSA.3(DAC))
15619	CreateProcessAsUser	CreateProcessAsUser: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682429(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15621	CreateProcessWithLogonW	CreateProcessWithLogonW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682431(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15623	CreateProcessWithTokenW	CreateProcessWithTokenW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682434(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15631	OpenProcess	OpenProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684320(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15632	OpenProcess	OpenProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684320(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15633	OpenProcess	OpenProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684320(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15818	OpenProcess	OpenProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684320(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15653	LsaLogonUser	LsaLogonUser:	FMT:	Static Attribute Initialization

		http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	Security Management	for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15645	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15649	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15651	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15647	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15634	OpenProcessToken	OpenProcessToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379295(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15635	OpenProcessToken	OpenProcessToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379295(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15636	OpenProcessToken	OpenProcessToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379295(v=vs.	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		85).aspx		
15821	OpenProcessToken	OpenProcessToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379295(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15655	OpenThreadToken	OpenThreadToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379296(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
15656	OpenThreadToken	OpenThreadToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379296(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15657	OpenThreadToken	OpenThreadToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379296(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15822	OpenThreadToken	OpenThreadToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379296(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15637	DuplicateToken	DuplicateToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446616(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15639	DuplicateTokenEx	DuplicateTokenEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446617(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15641	RpcAddPrinter	RpcAddPrinter:	FMT:	Static Attribute Initialization

		http://msdn.microsoft.com/en-us/library/cc244763(v=prot.20).aspx	Security Management	for Discretionary Access Control Policy (FMT_MSA.3(DAC))
15643	RpcAddPrinterEx	RpcAddPrinterEx: http://msdn.microsoft.com/en-us/library/cc244766.aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
8674	RpcOpenPrinter	RpcOpenPrinter: http://msdn.microsoft.com/en-us/library/cc244808.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15823	RpcOpenPrinter	RpcOpenPrinter: http://msdn.microsoft.com/en-us/library/cc244808.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4630	RpcOpenPrinter	RpcOpenPrinter: http://msdn.microsoft.com/en-us/library/cc244808.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4631	RpcOpenPrinter	RpcOpenPrinter: http://msdn.microsoft.com/en-us/library/cc244808.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
8675	RpcOpenPrinterEx	RpcOpenPrinterEx: http://msdn.microsoft.com/en-us/library/cc244809(v=prot.20).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15824	RpcOpenPrinterEx	RpcOpenPrinterEx: http://msdn.microsoft.com/en-us/library/cc244809(v=prot.20).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4721	RpcOpenPrinterEx	RpcOpenPrinterEx: http://msdn.microsoft.com/en-us/library/cc244809(v=prot.20).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4722	RpcOpenPrinterEx	RpcOpenPrinterEx:	FDP: User	Security Attribute Based

		http://msdn.microsoft.com/en-us/library/cc244809(v=prot.20).aspx	Data Protection	Access Control for Discretionary Access (FDP_ACF.1(DAC))
4647	RpcSetPort	RpcSetPort: http://msdn.microsoft.com/en-us/library/cc244824.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4648	RpcSetPort	RpcSetPort: http://msdn.microsoft.com/en-us/library/cc244824.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15700	RpcSetPrinter	RpcSetPrinter: http://msdn.microsoft.com/en-us/library/cc244825.aspx	FMT: Security Management	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
16475	RpcSetPrinter	RpcSetPrinter: http://msdn.microsoft.com/en-us/library/cc244825.aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
15702	RpcSetPrinter	RpcSetPrinter: http://msdn.microsoft.com/en-us/library/cc244825.aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
4681	RpcUploadPrinterDriverPackage	UploadPrinterDriverPackage: http://msdn.microsoft.com/en-us/library/windows/desktop/dd145168(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4682	RpcUploadPrinterDriverPackage	UploadPrinterDriverPackage: http://msdn.microsoft.com/en-us/library/windows/desktop/dd145168(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4479	RpcDeleteMonitor	RpcDeleteMonitor: http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/cc244771.aspx	Protection	(FDP_ACC.1(DAC))
4480	RpcDeleteMonitor	RpcDeleteMonitor: http://msdn.microsoft.com/en-us/library/cc244771.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4481	RpcDeletePerMachineConnection	RpcDeletePerMachineConnection: http://msdn.microsoft.com/en-us/library/cc244772.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4482	RpcDeletePerMachineConnection	RpcDeletePerMachineConnection: http://msdn.microsoft.com/en-us/library/cc244772.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4487	RpcDeletePort	RpcDeletePort: http://msdn.microsoft.com/en-us/library/cc244773.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4488	RpcDeletePort	RpcDeletePort: http://msdn.microsoft.com/en-us/library/cc244773.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4519	RpcDeletePrinterDriver	RpcDeletePrinterDriver: http://msdn.microsoft.com/en-us/library/cc244778.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4520	RpcDeletePrinterDriver	RpcDeletePrinterDriver: http://msdn.microsoft.com/en-us/library/cc244778.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4493	RpcDeletePrinterDriverEx	RpcDeletePrinterDriverEx: http://msdn.microsoft.com/en-us/library/cc244779.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4494	RpcDeletePrinterDriverEx	RpcDeletePrinterDriverEx:	FDP: User	Security Attribute Based

		http://msdn.microsoft.com/en-us/library/cc244779.aspx	Data Protection	Access Control for Discretionary Access (FDP_ACF.1(DAC))
4694	RpcDeletePrinterDriverPackage	DeletePrinterDriverPackage: http://msdn.microsoft.com/en-us/library/windows/desktop/dd183547(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4695	RpcDeletePrinterDriverPackage	DeletePrinterDriverPackage: http://msdn.microsoft.com/en-us/library/windows/desktop/dd183547(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4530	RpcDeletePrintProcessor	RpcDeletePrintProcessor: http://msdn.microsoft.com/en-us/library/cc244782.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4531	RpcDeletePrintProcessor	RpcDeletePrintProcessor: http://msdn.microsoft.com/en-us/library/cc244782.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4536	RpcEnumMonitors	RpcEnumMonitors: http://msdn.microsoft.com/en-us/library/cc244787.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4537	RpcEnumMonitors	RpcEnumMonitors: http://msdn.microsoft.com/en-us/library/cc244787.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4550	RpcEnumPorts	RpcEnumPorts: http://msdn.microsoft.com/en-us/library/cc244789.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4551	RpcEnumPorts	RpcEnumPorts: http://msdn.microsoft.com/en-	FDP: User Data	Security Attribute Based Access Control for

		us/library/cc244789.aspx	Protection	Discretionary Access (FDP_ACF.1(DAC))
4553	RpcEnumPrinterDrivers	RpcEnumPrinterDrivers: http://msdn.microsoft.com/en-us/library/cc244792(v=prot.20).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4554	RpcEnumPrinterDrivers	RpcEnumPrinterDrivers: http://msdn.microsoft.com/en-us/library/cc244792(v=prot.20).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4557	RpcEnumPrinters	RpcEnumPrinters: http://msdn.microsoft.com/en-us/library/cc244794.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4558	RpcEnumPrinters	RpcEnumPrinters: http://msdn.microsoft.com/en-us/library/cc244794.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4560	RpcEnumPrintProcessorDatatypes	RpcEnumPrintProcessorDatatypes: http://msdn.microsoft.com/en-us/library/cc244795.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4561	RpcEnumPrintProcessorDatatypes	RpcEnumPrintProcessorDatatypes: http://msdn.microsoft.com/en-us/library/cc244795.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4563	RpcEnumPrintProcessors	RpcEnumPrintProcessors: http://msdn.microsoft.com/en-us/library/cc244796.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4564	RpcEnumPrintProcessors	RpcEnumPrintProcessors: http://msdn.microsoft.com/en-us/library/cc244796.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))

4685	RpcGetCorePrinterDrivers	RpcGetCorePrinterDrivers: http://msdn.microsoft.com/en-us/library/dd891419(v=prot.20).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4686	RpcGetCorePrinterDrivers	RpcGetCorePrinterDrivers: http://msdn.microsoft.com/en-us/library/dd891419(v=prot.20).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4691	RpcGetPrinterDriverPackagePath	RpcGetPrinterDriverPackagePath: http://msdn.microsoft.com/en-us/library/dd871495(v=prot.20).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4692	RpcGetPrinterDriverPackagePath	RpcGetPrinterDriverPackagePath: http://msdn.microsoft.com/en-us/library/dd871495(v=prot.20).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4582	RpcGetPrintProcessorDirectory	RpcGetPrintProcessorDirectory: http://msdn.microsoft.com/en-us/library/cc244807.aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4583	RpcGetPrintProcessorDirectory	RpcGetPrintProcessorDirectory: http://msdn.microsoft.com/en-us/library/cc244807.aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
4678	RpcInstallPrinterDriverFromPackage	InstallPrinterDriverFromPackage: http://msdn.microsoft.com/en-us/library/windows/desktop/dd144997(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
4679	RpcInstallPrinterDriverFromPackage	InstallPrinterDriverFromPackage: http://msdn.microsoft.com/en-us/library/windows/desktop/dd144997(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))

14559	Explorer - Security Tab (DACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
15664	Explorer - Security Tab (DACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16469	Explorer - Security Tab (DACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
14560	Explorer - Security Tab (DACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15765	Explorer - Security Tab (DACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
14562	Explorer - Advanced Security Settings Audit Tab (SACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Management of Security Attributes for Object Ownership (FMT_MSA.1(OBJ))
15666	Explorer - Advanced Security Settings Audit Tab (SACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16470	Explorer - Advanced Security Settings Audit Tab (SACL)	Set, view, change, or remove special permissions:	FMT: Security	Management of TSF Data for General TSF Data

		http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	Management	(FMT_MTD.1(GEN))
14563	Explorer - Advanced Security Settings Audit Tab (SACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15766	Explorer - Advanced Security Settings Audit Tab (SACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
14566	Explorer - Advanced Security Settings, Change Owner Link	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
15667	Explorer - Advanced Security Settings, Change Owner Link	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16471	Explorer - Advanced Security Settings, Change Owner Link	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
15677	Explorer - Advanced Security Settings, Change Owner Link	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15767	Explorer - Advanced Security Settings, Change Owner Link	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)

16309	Explorer – Verify the Backup/Restore privleges (DAC)	SE_BACKUP_NAME/SE_RESTORE_NAME: http://msdn.microsoft.com/en-us/library/windows/desktop/bb530716(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16310	Explorer – Verify the Backup/Restore privleges (DAC)	SE_BACKUP_NAME/SE_RESTORE_NAME: http://msdn.microsoft.com/en-us/library/windows/desktop/bb530716(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
16372	Explorer – Verify the Backup/Restore privleges (DAC)	SE_BACKUP_NAME/SE_RESTORE_NAME: http://msdn.microsoft.com/en-us/library/windows/desktop/bb530716(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16373	Explorer – Verify the Backup/Restore privleges (DAC)	SE_BACKUP_NAME/SE_RESTORE_NAME: http://msdn.microsoft.com/en-us/library/windows/desktop/bb530716(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16311	Explorer – Verify Restricted SIDs (DAC)	Restricted Tokens: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379316(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16312	Explorer – Verify Restricted SIDs (DAC)	Restricted Tokens: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379316(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
16370	Explorer – Verify Restricted SIDs (DAC)	Restricted Tokens: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379316(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16371	Explorer – Verify Restricted SIDs (DAC)	Restricted Tokens: http://msdn.microsoft.com/en-	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		us/library/windows/desktop/aa379316(v=vs.85).aspx		
16376	Explorer –Manage Inheritance Rules	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FMT: Security Management	Static Attribute Value Inheritance (FMT_MSA.4)
16550	CreateIoCompletionPort	CreateIoCompletionPort: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363862(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16551	CreateIoCompletionPort	CreateIoCompletionPort: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363862(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
15946	CreateIoCompletionPort	CreateIoCompletionPort: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363862(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
16351	iCacls – DACL	icacls: http://technet.microsoft.com/en-us/library/cc753525.aspx	FMT: Security Management	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
16352	iCacls – DACL	icacls: http://technet.microsoft.com/en-us/library/cc753525.aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16472	iCacls – DACL	icacls: http://technet.microsoft.com/en-us/library/cc753525.aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
16353	iCacls – DACL	icacls: http://technet.microsoft.com/en-us/library/cc753525.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

16354	iCacls – DACL	icacls: http://technet.microsoft.com/en-us/library/cc753525.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16360	Registry - Security Tab (DACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Management of Security Attributes for Discretionary Access Control (FMT_MSA.1(DAC))
16361	Registry - Security Tab (DACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16473	Registry - Security Tab (DACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
16362	Registry - Security Tab (DACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16363	Registry - Security Tab (DACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16364	Registry - Advanced Security Settings, Audit Tab (SACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Management of Security Attributes for Object Ownership (FMT_MSA.1(OBJ))
16365	Registry - Advanced Security Settings, Audit Tab (SACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Revocation for Object Access for DAC (FMT_REV.1(DAC))
16474	Registry - Advanced Security Settings, Audit Tab (SACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
16366	Registry - Advanced Security Settings, Audit Tab (SACL)	Registry Editor: http://technet.microsoft.com/en-	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		us/library/cc755256.aspx		
16367	Registry - Advanced Security Settings, Audit Tab (SACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16377	Registry – Manage Inheritance Rules	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FMT: Security Management	Static Attribute Value Inheritance (FMT_MSA.4)
16552	CreateWaitableTimerEx	CreateWaitableTimerEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682494(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
16553	OpenWaitableTimer	OpenWaitableTimer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684337(v=vs.85).aspx	FDP: User Data Protection	Security Attribute Based Access Control for Discretionary Access (FDP_ACF.1(DAC))
16554	OpenWaitableTimer	OpenWaitableTimer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684337(v=vs.85).aspx	FMT: Security Management	Static Attribute Initialization for Discretionary Access Control Policy (FMT_MSA.3(DAC))
16555	OpenWaitableTimer	OpenWaitableTimer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684337(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16556	OpenWaitableTimer	OpenWaitableTimer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684337(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)

16557	GetUserObjectInformation	GetUserObjectInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms683238(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16558	CreateDialogIndirectParam	CreateDialogIndirectParam: http://msdn.microsoft.com/en-us/library/windows/desktop/ms645441(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16559	CreateMenu	CreateMenu: http://msdn.microsoft.com/en-us/library/windows/desktop/ms647624(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16560	SetWindowsHookEx	SetWindowsHookEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms644990(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16561	EnumWindowStations	EnumWindowStations: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682644(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16562	SetUserObjectInformation	SetUserObjectInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686287(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16563	SwitchDesktop	SwitchDesktop: http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/windows/desktop/ms686347(v=vs.85).aspx	Protection	(FDP_ACC.1(DAC))
16564	WaitForSingleObject	WaitForSingleObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms687032(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16565	WaitForMultipleObjects	WaitForMultipleObjects: http://msdn.microsoft.com/en-us/library/windows/desktop/ms687025(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16696	SetEvent	SetEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686211(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16697	PulseEvent	PulseEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684914(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16698	ResetEvent	ResetEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms685081(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16566	SignalObjectAndWait	SignalObjectAndWait: http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/windows/desktop/ms686293(v=vs.85).aspx	Protection	(FDP_ACC.1(DAC))
16567	LockFile	LockFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365202(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16568	ReadFile	ReadFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365467(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16569	ReadFileScatter	http://msdn.microsoft.com/en-us/library/windows/desktop/aa365469(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16570	WriteFile	WriteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365747(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16571	WriteFileGather	WriteFileGather: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365749(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16572	DeviceIoControl	DeviceIoControl: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363216(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16573	ReadDirectoryChanges	ReadDirectoryChanges: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365465(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16574	UnlockFile	UnlockFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365715(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16575	SetFileAttributes	http://msdn.microsoft.com/en-us/library/windows/desktop/aa365535(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16576	GetFileAttributesEx	http://msdn.microsoft.com/en-us/library/windows/desktop/aa364946(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16577	GetFileAttributes	http://msdn.microsoft.com/en-us/library/windows/desktop/aa364944(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16578	FlushFileBuffers	FlushFileBuffers: http://msdn.microsoft.com/en-us/library/windows/desktop/aa364439(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16579	GetQueuedCompletionStatus	GetQueuedCompletionStatus: http://msdn.microsoft.com/en-us/library/windows/desktop/aa364986(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16580	GetQueuedCompletionStatusEx	GetQueuedCompletionStatusEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa364988(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

16581	PostQueuedCompletionStatus	PostQueuedCompletionStatus: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365458(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16582	AssignProcessToJobObject	AssignProcessToJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms681949(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16583	SetInformationJobObject	SetInformationJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686216(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16584	UserHandleGrantAccess	UserHandleGrantAccess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686884(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16585	IsProcessInJob	IsProcessInJob: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684127(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16586	QueryInformationJobObject	QueryInformationJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684925(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16587	TerminateJobObject	TerminateJobObject: http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/windows/desktop/ms686709(v=vs.85).aspx	Protection	(FDP_ACC.1(DAC))
16588	ContinueDebugEvent	ContinueDebugEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms679285(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16589	WaitForDebugEvent	WaitForDebugEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms681423(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16590	DebugActiveProcess	DebugActiveProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms679295(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16591	DebugActiveProcessStop	DebugActiveProcessStop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms679296(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16699	CreateRemoteThreadEx	CreateRemoteThreadEx: http://msdn.microsoft.com/en-us/library/windows/desktop/dd405484(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16592	DebugSetProcessKillOnExit	DebugSetProcessKillOnExit: http://msdn.microsoft.com/en-us/library/windows/desktop/ms679307(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16593	RegSetValueEx	RegSetValueEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724923(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16594	RegDeleteValue	RegDeleteValue: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724851(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16595	RegQueryInfoKey	RegQueryInfoKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724902(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16596	RegQueryValueEx	RegQueryValueEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724911(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16597	RegQueryMultipleValues	RegQueryMultipleValues: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724905(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16598	RegEnumValue	RegEnumValue: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724865(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

16599	RegEnumKey	RegEnumKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724861(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16600	RegNotifyChangeKeyValue	RegNotifyChangeKeyValue: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724892(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16601	RegDeleteKeyEx	RegDeleteKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724847(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16602	GetVolumeInformation	GetVolumeInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa364993(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16603	ReleaseMutex	ReleaseMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms685066(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16604	GetPrinter	GetPrinter: http://msdn.microsoft.com/en-us/library/windows/desktop/dd144911(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16605	SetPrinter	SetPrinter: http://msdn.microsoft.com/en-us/library/windows/desktop/dd145082(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16606	AddPrinterConnection	AddPrinterConnection: http://msdn.microsoft.com/en-us/library/windows/desktop/dd183344(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16607	ReadPrinter	ReadPrinter: http://msdn.microsoft.com/en-us/library /windows/desktop/dd162895(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16608	GetPrinterDriver	GetPrinterDriver: http://msdn.microsoft.com/en-us/library/windows/desktop/dd144914(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16609	SetPrinterData	SetPrinterData: http://msdn.microsoft.com/en-us/library/windows/desktop/dd145083(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16610	SetPrinterDataEx	SetPrinterDataEx: http://msdn.microsoft.com/en-us/library/windows/desktop/dd145084(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16611	DeletePrinterData	DeletePrinterData: http://msdn.microsoft.com/en-us/library/windows/desktop/dd183543(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16612	DeletePrinterDataEx	DeletePrinterDataEx:	FDP: User	Complete Access Control for

		http://msdn.microsoft.com/en-us/library/windows/desktop/dd183544(v=vs.85).aspx	Data Protection	Discretionary Access (FDP_ACC.1(DAC))
16613	DeletePrinterKey	DeletePrinterKey: http://msdn.microsoft.com/en-us/library/windows/desktop/dd183548(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16614	DeletePrinter	DeletePrinter: http://msdn.microsoft.com/en-us/library/windows/desktop/dd183541(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16615	SetJob	SetJob: http://msdn.microsoft.com/en-us/library/windows/desktop/dd162978(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16616	DuplicateHandle	DuplicateHandle: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724251(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16617	VirtualQueryEx	VirtualQueryEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366907(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16618	GetGuiResources	GetGuiResources: http://msdn.microsoft.com/en-us/library/windows/desktop/ms683192(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16619	NtQueryInformationProcess	NtQueryInformationProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684280(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16620	IsProcessInJob	IsProcessInJob: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684127(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16621	SetProcessMitigationPolicy	SetProcessMitigationPolicy: http://msdn.microsoft.com/en-us/library/windows/desktop/hh769088(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16622	TerminateProcess	TerminateProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686714(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16623	MapViewOfFile	MapViewOfFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366761(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16624	ReleaseSemaphore	ReleaseSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms685071(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

16625	QueryDosDevice	QueryDosDevice: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365461(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16626	TerminateThread	TerminateThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686717(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16627	SuspendThread	SuspendThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686345(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16628	ResumeThread	ResumeThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms685086(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16629	GetThreadContext	GetThreadContext: http://msdn.microsoft.com/en-us/library/windows/desktop/ms679362(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16630	SetThreadContext	SetThreadContext: http://msdn.microsoft.com/en-us/library/windows/desktop/ms680632(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16631	QueueUserAPC	QueueUserAPC: http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/windows/desktop/ms684954(v=vs.85).aspx	Protection	(FDP_ACC.1(DAC))
16632	GetThreadPriority	GetThreadPriority: http://msdn.microsoft.com/en-us/library/windows/desktop/ms683235(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16633	GetThreadPriority	GetThreadPriority: http://msdn.microsoft.com/en-us/library/windows/desktop/ms683235(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16634	SetThreadPriority	SetThreadPriority: http://msdn.microsoft.com/en-us/library/windows/desktop/ms686277(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16635	RevertToSelf	RevertToSelf: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379317(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16636	ImpersonateAnonymousToken	ImpersonateAnonymousToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378610(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16637	SetWaitableTimerEx	SetWaitableTimerEx: http://msdn.microsoft.com/en-us/library/windows/desktop/dd405521(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16638	CancelWaitableTimer	CancelWaitableTimer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms681985(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16639	GetTokenInformation	GetTokenInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446671(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16640	SetTokenInformation	SetTokenInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379591(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16641	PrivilegeCheck	PrivilegeCheck: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379304(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16642	AdjustTokenPrivileges	AdjustTokenPrivileges: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375202(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16643	AdjustTokenGroups	AdjustTokenGroups: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375199(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

16644	ObjectOpenAuditAlarm	ObjectOpenAuditAlarm: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379289(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16645	ObjectPrivilegeAuditAlarm	ObjectPrivilegeAuditAlarm: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379290(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16646	PrivilegedServiceAuditAlarm	PrivilegedServiceAuditAlarm: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379305(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16647	AccessCheck	AccessCheck: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374815(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16648	AccessCheckByType	AccessCheckByType: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374826(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16649	AccessCheckByTypeResultList	AccessCheckByTypeResultList: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374836(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16650	AccessCheckByTypeResultListAndAuditAlarmByHandle	AccessCheckByTypeResultListAndAuditAlarmByHandle: http://msdn.microsoft.com/en-	FDP: User Data	Complete Access Control for Discretionary Access

		us/library/windows/desktop/aa374843(v=vs.85).aspx	Protection	(FDP_ACC.1(DAC))
16651	GetCaretBlinkTime	GetCaretBlinkTime: http://msdn.microsoft.com/en-us/library/windows/desktop/ms648401(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16652	GetClipCursor	GetClipCursor: http://msdn.microsoft.com/en-us/library/windows/desktop/ms648387(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16653	GetDoubleClickTime	GetDoubleClickTime: http://msdn.microsoft.com/en-us/library/windows/desktop/ms646258(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16654	GetCaretBlinkTime	GetCaretBlinkTime: http://msdn.microsoft.com/en-us/library/windows/desktop/ms648401(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16655	ClipCursor	ClipCursor: http://msdn.microsoft.com/en-us/library/windows/desktop/ms648383(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16656	RegisterHotKey	RegisterHotKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms646309(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

16657	SetSysColors	SetSysColors: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724940(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16700	SetSystemCursor	SetSystemCursor: http://msdn.microsoft.com/en-us/library/windows/desktop/ms648395(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16658	OpenClipboard	OpenClipboard: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649048(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16659	CloseClipboard	CloseClipboard: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649035(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16660	EmptyClipboard	EmptyClipboard: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649037(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16661	GetClipboardData	GetClipboardData: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649039(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16662	GetClipboardSequenceNumb	GetClipboardSequenceNumber:	FDP: User	Complete Access Control for

	er	http://msdn.microsoft.com/en-us/library/windows/desktop/ms649042(v=vs.85).aspx	Data Protection	Discretionary Access (FDP_ACC.1(DAC))
16663	IsClipboardFormatAvailable	IsClipboardFormatAvailable: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649047(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16664	CountClipboardFormats	CountClipboardFormats: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649036(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16665	GetPriorityClipboardFormat	GetPriorityClipboardFormat: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649045(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16666	GetClipboardOwner	GetClipboardOwner: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649041(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16667	GetClipboardViewer	GetClipboardViewer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649043(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16668	SetClipboardData	SetClipboardData: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649051(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16701	SetClipboardViewer	SetClipboardViewer: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649052(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16669	ChangeClipboardChain	ChangeClipboardChain: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649034(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16670	GetOpenClipboardWindow	GetOpenClipboardWindow: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649044(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16701	AddAtom	AddAtom: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649056(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16671	FindAtom	FindAtom: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649058(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16672	GlobalGetAtomName	GlobalGetAtomName: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649063(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16673	DeleteAtom	DeleteAtom: http://msdn.microsoft.com/en-us/library/windows/desktop/ms649057(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16674	CommitComplete	CommitComplete: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365996(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16675	CommitEnlistment	CommitEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/bb613465(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16676	PrePrepareComplete	PrePrepareComplete: http://msdn.microsoft.com/en-us/library/windows/desktop/bb613467(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16677	ReadOnlyEnlistment	ReadOnlyEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366346(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16678	RollbackComplete	RollbackComplete: http://msdn.microsoft.com/en-us/library/windows/desktop/aa965197(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16679	RollbackEnlistment	RollbackEnlistment:	FDP: User	Complete Access Control for

		http://msdn.microsoft.com/en-us/library/windows/desktop/aa366361(v=vs.85).aspx	Data Protection	Discretionary Access (FDP_ACC.1(DAC))
16680	SinglePhaseReject	SinglePhaseReject: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366379(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16681	PrepareComplete	PrepareComplete: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366318(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16682	PrePrepareEnlistment	PrePrepareEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/bb613467(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16683	PrepareEnlistment	PrepareEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/bb613466(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16684	GetEnlistmentRecoveryInformation	GetEnlistmentRecoveryInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366193(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16685	RecoverEnlistment	RecoverEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa965195(v=vs.	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

		85).aspx		
16686	SetEnlistmentRecoveryInformation	SetEnlistmentRecoveryInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366375(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16687	CommitTransaction	CommitTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366001(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16688	GetTransactionInformation	GetTransactionInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366204(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16689	RollbackTransaction	RollbackTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366366(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16690	SetTransactionInformation	SetTransactionInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366377(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16691	GetNotificationResourceManager	GetNotificationResourceManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366196(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

16692	RecoverResourceManager	RecoverResourceManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa965196(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16693	GetTransactionManagerId	GetTransactionManagerId: http://msdn.microsoft.com/en-us/library/windows/desktop/aa965192(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16694	RecoverTransactionManager	RecoverTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366350(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))
16695	RollforwardTransactionManager	RollforwardTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/bb613469(v=vs.85).aspx	FDP: User Data Protection	Complete Access Control for Discretionary Access (FDP_ACC.1(DAC))

In addition to the interfaces listed above, there are two named kernel objects subject to the Discretionary Access Control policy that do not have directly accessible programming interfaces:

- Filter Communication Port – The Filter Communication Port is a DAC object for use as a communication mechanism between clients and a mini-filter driver. Filter Communication Ports are created by the Filter Manager.
- Filter Connection Port – The Filter Connection Port is a DAC object used as a communication mechanism between clients and the NTFS Filter Manager for sending messages to manage mini-filter drivers. Filter Connection Ports are created by the Filter Manager.

All DAC named kernel objects, including the security descriptor, can also managed by opening the Windows kernel object directory, using NtOpenDirectoryObject, enumerating the object names in the kernel object directory, using NtQueryDirectoryObject, and then open the object and use NtSetSecurityObject to manage the security descriptor.

9.2.1.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

- For objects other than Directory Services objects

  To enable audit policy subcategories for Object Access operations, run the following command at an elevated command prompt:

  - auditpol /set /subcategory:”Handle Manipulation” /success:enable /failure:enable

  In addition for each of the object types the following command must be run at an elevated command prompt:

  - auditpol /set /subcategory:<Object Subcategory Type> /success:enable /failure:enable

  For the Object Subcategory Type in the command above the appropriate subcategory name must be used to turn on auditing for that object type.

- For Directory Services objects

  To enable audit policy subcategories for Object Access operations, run the following command at an elevated command prompt:

  - auditpol /set /subcategory:”Directory Service Access” /success:enable /failure:enable

  In addition for each of the object types the following command must be run at an elevated command prompt:

  - auditpol /set /subcategory:”Directory Service Changes” /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4670	File System Or Registry	Permissions on an object were changed.	Logged: <Date and time of event> Security ID: <SID of account making the change> Object Name: <Name of the object changed>

	Or Kernel Object Or File Share Or Other Object Access Events		Permissions Change: <Old and new security descriptor>
4656	File System Or Registry Or Kernel Object Or File Share Or Other Object Access Events	A handle to an object was requested.	Logged: <Date and time of event> Security ID: <SID of locked account> Object Name: <Name of the object changed> Accesses: <Access granted> Access Mask: <Access requested>

9.2.2 Mandatory Intregrity Control Policy

Mandatory Integrity Control Functions (FDP_ACC.1(MIC))
Mandatory Integrity Control Functions (FDP_ACF.1(MIC)
Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))
Static Attribute Initialization for Mandatory Integrity Control Policies (FMT_MSA.3(MIC))
Revocation for Object Access (FMT_REV.1(OBJ))

9.2.2.1 Interfaces

The functional specification evidence associated with the interfaces for FDP_ACC.1(MIC), FDP_ACF.1(MIC) and related audits and management operations are indicated in the table below (the legend for the below table is in section 1).

There are no interfaces defined for FMT_MSA.3(MIC) because there are no alternative initial values to override the MIC default values when an object is created.

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15876	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15877	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15880	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15881	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15884	RegOpenKey	RegOpenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724895(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))

15885	RegOpenKey	RegOpenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724895(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15886	RegOpenKeyEx	RegOpenKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724897(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15887	RegOpenKeyEx	RegOpenKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724897(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
16027	OpenEvent	OpenEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
16028	OpenEvent	OpenEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684305(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15892	OpenMutex	OpenMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684315(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15893	OpenMutex	OpenMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684315(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15928	OpenSemaphore	OpenSemaphore: http://msdn.microsoft.com/en-	FDP: User Data	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))

		us/library/windows/desktop/ms684326(v=vs.85).aspx	Protection	
15929	OpenSemaphore	OpenSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684326(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15894	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15895	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15898	OpenFile	OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15899	OpenFile	OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15900	DeleteFile	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15901	DeleteFile	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))

15922	OpenFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15923	OpenFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
16388	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
16387	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
33	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
8609	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16391	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))

		85).aspx		
16390	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
195	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16389	NtOpenDirectoryObject	NtOpenDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556557(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15950	OpenJobObject	OpenJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684312(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15951	OpenJobObject	OpenJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684312(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15912	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))

15913	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15914	OpenProcess	OpenProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684320(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15915	OpenProcess	OpenProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684320(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15952	OpenProcessToken	OpenProcessToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379295(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15953	OpenProcessToken	OpenProcessToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379295(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
15954	OpenThreadToken	OpenThreadToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379296(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15955	OpenThreadToken	OpenThreadToken: http://msdn.microsoft.com/en-	FDP: User Data	Mandatory Integrity Control

		us/library/windows/desktop/aa379296(v=vs.85).aspx	Protection	Policy (FDP_ACC.1(MIC))
15912	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Functions (FDP_ACF.1(MIC))
15913	OpenThread	OpenThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms684335(v=vs.85).aspx	FDP: User Data Protection	Mandatory Integrity Control Policy (FDP_ACC.1(MIC))
16306	GetSecurityInfo	GetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446654(v=vs.85).aspx	FMT: Security Management	Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))
15962	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.85).aspx	FMT: Security Management	Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))
15963	SetSecurityInfo	SetSecurityInfo: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379588(v=vs.85).aspx	FMT: Security Management	Revocation for Object Access (FMT_REV.1(OBJ))
15603	Verify Mandatory Integrity Control (MIC) Audit	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

15604	Verify Mandatory Integrity Control (MIC) Audit	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16386	Icacls /setintegritylevel	Icacls: http://technet.microsoft.com/en-us/library/cc753525.aspx	FMT: Security Management	Management of Security Attributes for Mandatory Integrity Control (FMT_MSA.1(MIC))

9.2.2.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

To enable audit policy for MIC access checks enable the Object Access category, run the following command at an elevated command prompt:

- auditpol /set /subcategory:"Object Access" /success:enable /failure:enable

To enable audit policy for MIC access checks by object type, conduct the following steps at an elevated command prompt to open the Local Security Policy editor:

- secpol.msc

and then in the Local Security Policy editor do the following:

- Navigate to **Security Settings\Advanced Audit Policy Configuration\System Audit Policies – Local Group Policy Object\Object Access** node in the left pane, and then in the right pane open the **Audit File System** properties dialog and check the **Configure**, **Success** and **Failure** checkboxes and press the **OK** button.

Event Id	Policy Subcategory	Message	Fields
4656	Object Access	A handle to an object was requested	Logged: <Date and time of event> Keywords: <Outcome as Success> Access Request Information: <Requested access

			rights with DAC and MIC access check decisions>

9.2.3 Network Information Flow Control Policy

Subset Information Flow Control (FDP_IFC.1(OSPP))
Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
Static Attribute Initialization for Network Information Flow Control (FMT_MSA.3(OSPP))
Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))

9.2.3.1 Interfaces

The functional specification evidence associated with the interfaces for FDP_IFC.2(OSPP), FDP_IFF.1(OSPP), FMT_MSA.3(OSPP), and FMT_MTD.1(OSPP) and related audits and management operations are indicated in the table below (the legend for the below table is in section 1). The Interface Documentation column includes information about the parameters that are used to indicate the network protocol for the interface and also describe the protocols that are supported by the interface for that parameter.

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15970	WSASocket	WSASocket: http://msdn.microsoft.com/en-us/library/windows/desktop/ms742212(v=vs.85).aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
15971	WSASocket	WSASocket: http://msdn.microsoft.com/en-us/library/windows/desktop/ms742212(v=vs.85).aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15975	WSASend	WSASend: http://msdn.microsoft.com/en-us/library/windows/desktop/ms742203(v=vs.85).aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))

15976	WSASend	WSASend: http://msdn.microsoft.com/en-us/library/windows/desktop/ms742203(v=vs.85).aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15969	WSARecvEx	WSARecvEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms741684(v=vs.85).aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
15973	WSARecvEx	WSARecvEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms741684(v=vs.85).aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15983	new-netfirewallrule (block IPv4 address)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
15984	new-netfirewallrule (block IPv4 address)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15987	new-netfirewallrule (block IPv4 address)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15988	new-netfirewallrule (block executable pathname)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
15989	new-netfirewallrule (block	new-netfirewallrule:	FDP: User	Simple Security Attributes for

	executable pathname)	http://technet.microsoft.com/en-us/library/jj554908.aspx	Data Protection	Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15991	new-netfirewallrule (block executable pathname)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15992	new-netfirewallrule (block IP address and protocol)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
15993	new-netfirewallrule (block IP address and protocol)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15995	new-netfirewallrule (block IP address and protocol)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16410	new-netfirewallrule (block all connections)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
15996	new-netfirewallrule (block all connections)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
15997	new-netfirewallrule (block all connections)	new-netfirewallrule: http://technet.microsoft.com/en-	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		us/library/jj554908.aspx		
15999	Get-NetFirewallProfile (query default restrictive/permissive Windows firewall security attributes)	Get-NetFirewallProfile: http://technet.microsoft.com/en-us/library/jj573830.aspx	FMT: Security Management	Static Attribute Initialization for Network Information Flow Control (FMT_MSA.3(OSPP))
16001	Set-NetFirewallProfile (enable and disable Windows firewall)	Set-NetFirewallProfile: http://technet.microsoft.com/en-us/library/jj554896.aspx	FMT: Security Management	Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))
16002	Set-NetFirewallProfile (enable and disable Windows firewall)	Set-NetFirewallProfile: http://technet.microsoft.com/en-us/library/jj554896.aspx	FMT: Security Management	Static Attribute Initialization for Network Information Flow Control (FMT_MSA.3(OSPP))
16404	new-netfirewallrule (block TCP port)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
16405	new-netfirewallrule (block TCP port)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
16406	new-netfirewallrule (block TCP port)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16407	new-netfirewallrule (block UDP port)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))

16408	new-netfirewallrule (block UDP port)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
16409	new-netfirewallrule (block UDP port)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16444	new-netfirewallrule (block IPv6 address)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Subset Complete Information Flow Control (FDP_IFC.1(OSPP))
16443	new-netfirewallrule (block IPv6 address)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FDP: User Data Protection	Simple Security Attributes for Network Information Flow Control Policy (FDP_IFF.1(OSPP))
16445	new-netfirewallrule (block IPv6 address)	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15978	new-netfirewallrule	new-netfirewallrule: http://technet.microsoft.com/en-us/library/jj554908.aspx	FMT: Security Management	Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))
15985	INetFwRules::Add	INetFwRules::Add http://msdn.microsoft.com/en-us/library/windows/desktop/aa365346(v=vs.85).aspx	FMT: Security Management	Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))

15979	Remove-NetFirewallRule	Remove-NetFirewallRule: http://technet.microsoft.com/en-us/library/jj554893.aspx	FMT: Security Management	Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))
15986	INetFwRules::Remove	INetFwRules::Remove http://msdn.microsoft.com/en-us/library/windows/desktop/aa365349(v=vs.85).aspx	FMT: Security Management	Management of TSF Data for Network Information Flow Control (FMT_MTD.1(OSPP))

9.2.3.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

To enable audit policy subcategories for Object Access operations, run the following command at an elevated command prompt:

- auditpol /set /subcategory:"Filtering Platform Packet Drop" /success:enable /failure:enable

The audit Id 5152 identifies the IP address for the network interface on which the network flow was denied as the Source Address or Destination Address field (depending upon the direction of flow) and the reason for denial is then indicated via one or more of the field values that are included for the audit. The actual fields that triggered the flow denial are dependent upon the firewall rule that was triggered. The firewall rule that was triggered the flow denial can be correlated by the "Filter Run-Time ID". The following TechhNet topic explains how to produce an xml file that identifies all the inbound and outbound firewall rules and associated attributes (in produced xml file the <name> tag can be correlated with the Windows Firewall with Advanced Security utility and the <filterId> tag canbe correlated with the Filter Run-Time ID in the Event Viewer utility):

- Netsh Commands for Windows Filtering Platform (WFP): http://technet.microsoft.com/en-us/library/dd735538(v=ws.10).aspx#bkmk_show3

Event Id	Policy Subcategory	Message	Fields
5152	Filtering Platform Packet Drop	The Windows Filtering Platform has blocked a packet.	Logged: <Date and time of event> Process ID: <process ID holding the network connection> Account Name: <name of the process holding

			the network connection > Direction: <Inbound or Outbound> Source Address: <source IP address of source> Source Port: <source port number> Destination Address: <destination IP address> Destination Port: <destination port number> Protocol: <protocol number> Filter Run-Time ID: <Filter ID associated with firewall rule triggering flow denial>

9.2.4 Full Residual Information Protection (FDP_RIP.2)

9.2.4.1 Interfaces

The functional specification evidence associated with the interfaces for FDP_RIP.2 are indicated in the table below (the legend for the below table is in section 1.1). These are the interfaces that perform all the necessary operations to ensure any previous information content is unavailable when the resource is re-allocated.

The interfaces that release a resource are not security-relevant with respect to FDP_RIP.2 because they do not perform any operations ensuring previous information content of a resource is made unavailable upon re-allocation but are indicated in the table below for completeness. Event pair, keyed event, ALPC port, timer, debug, filter connection port and filter communication port objects have no public interfaces to release their content.

Object	**Interface**
Directory, event, IO completion port, job, mutex, process, section, semaphore, thread, token, transaction enlistment, transaction, resource manager, transaction manager	CloseHandle: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724211(v=vs.85).aspx
Registry key	RegDeleteKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724845(v=vs.85).aspx

NtfsDirectory, NtfsFile, mailslot, symbolic link	DeleteFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363915(v=vs.85).aspx
Desktop	CloseDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682024(v=vs.85).aspx
WindowStation	CloseWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682047(v=vs.85).aspx
Printer	RpcDeletePrinter: http://msdn.microsoft.com/en-us/library/cc244774.aspx

The interfaces that read the content of a resource after re-allocation are effectively those interfaces that return a handle for the given object with READ access rights or equivalent. These interfaces are indicated in the interface table below. Event pair, keyed event, ALPC port, timer, debug, filter connection port and filter communication port objects have no public interface to read their content.

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15471	CreateEvent	CreateEvent: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682396(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
15477	CreateEventEx	CreateEventEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682400(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
15485	CreateMutex	CreateMutex: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682411(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
15491	CreateMutexEx	CreateMutexEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682418(v=vs.	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)

		85).aspx		
15497	CreateSemaphore	CreateSemaphore: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682438(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
15503	CreateSemaphoreEx	CreateSemaphoreEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682446(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16026	CreateSymbolicLink	CreateSymbolicLink: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363866(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16003	CreateFile	CreateFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363858(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16004	CreateProcess	CreateProcess: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682425(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16005	CreateProcessAsUser	CreateProcessAsUser: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682429(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16006	CreateProcessWithLogonW	CreateProcessWithLogonW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682431(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16007	CreateProcessWithTokenW	CreateProcessWithTokenW: http://msdn.microsoft.com/en-	FDP: User Data	Full Residual Information Protection (FDP_RIP.2)

		us/library/windows/desktop/ms682434(v=vs.85).aspx	Protection	
16008	CreateFileMapping	OpenFileMapping: http://msdn.microsoft.com/en-us/windows/desktop/aa366791(v=vs.85)	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16009	CreateFileMappingFromApp	CreateFileMappingFromApp: http://msdn.microsoft.com/en-us/windows/desktop/hh994453(v=vs.85)	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16010	CreateFileMappingNuma	CreateFileMappingNuma: http://msdn.microsoft.com/en-us/windows/desktop/aa366539(v=vs.85)	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16011	CreateThread	CreateThread: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682453(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16012	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16013	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16014	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16015	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16016	AcceptSecurityContext	AcceptSecurityContext :	FDP: User	Full Residual Information

		http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	Data Protection	Protection (FDP_RIP.2)
16017	CreateIoCompletionPort	CreateIoCompletionPort: http://msdn.microsoft.com/en-us/library/windows/desktop/aa363862(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16018	RegCreateKey	RegCreateKey: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724842(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16019	RegCreateKeyEx	RegCreateKeyEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms724844(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16401	CreateDesktop	CreateDesktop: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682124(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16402	CreateDesktopEx	CreateDesktopEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682127(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16400	CreateWindowStation	CreateWindowStation: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682496(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
34	NtCreateDirectoryObject	NtCreateDirectoryObject: http://msdn.microsoft.com/en-us/library/windows/hardware/ff556456(v=vs.	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)

		85).aspx		
16448	CreateJobObject	CreateJobObject: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682409(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16449	CreateNamedPipe	CreateNamedPipe: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365150(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16450	RpcAddPrinter	RpcAddPrinter: http://msdn.microsoft.com/en-us/library/cc244763(v=prot.20).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16451	RpcAddPrinterEx	RpcAddPrinterEx: http://msdn.microsoft.com/en-us/library/cc244766.aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16456	CreateEnlistment	CreateEnlistment: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366006(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16457	CreateResourceManager	CreateResourceManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366009(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16458	CreateTransactionManager	CreateTransactionManager: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366014(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16459	CreateTransaction	CreateTransaction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa366011(v=vs.	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)

16460	OpenFile	85).aspx OpenFile: http://msdn.microsoft.com/en-us/library/windows/desktop/aa365430(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)
16557	CreateWaitableTimerEx	CreateWaitableTimerEx: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682494(v=vs.85).aspx	FDP: User Data Protection	Full Residual Information Protection (FDP_RIP.2)

9.2.4.2 Audit Policy

<Not applicable.>

9.3 Identification and Authentication (FIA)

9.3.1 Authentication Failure Handling

Authentication Failure Handling (FIA_AFL.1)

Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Threshold)), Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Re-enable))

9.3.1.1 Interfaces

The functional specification evidence associated with the interfaces for FIA_AFL.1 and related audits and management operations are indicated in the table below (the legend for the below table is in section 1). All interfaces for authentication are subject to failure handling.

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15202	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FIA: Identification & Authenticatio	Authentication Failure Handling (FIA_AFL.1)

			n	
15203	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15714	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15261	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FIA: Identification & Authenticatio n	Authentication Failure Handling (FIA_AFL.1)
15262	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15717	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15263	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FIA: Identification & Authenticatio n	Authentication Failure Handling (FIA_AFL.1)
15264	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15719	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		85).aspx		
15265	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FIA: Identification & Authentication	Authentication Failure Handling (FIA_AFL.1)
15266	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15721	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15198	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FIA: Identification & Authentication	Authentication Failure Handling (FIA_AFL.1)
15199	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15723	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15200	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.	FIA: Identification &	Authentication Failure Handling (FIA_AFL.1)

		85).aspx	Authentication	
15201	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15725	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
14441	Local account lockout policy	The following TechNet topic explains the net accounts command line utility for standalone computers (options for managing account lockout policy are included below the link): Net Accounts: http://technet.microsoft.com/en-us/library/bb490698.aspx **/lockoutthreshold: *number* :** Sets the number of times a bad password may be entered until the account is locked out. If set to 0 then the account is never locked out. **/lockoutwindow: *minutes* :** Sets the number of minutes of the lockout window. **/lockoutduration: *minutes* :** Sets the number of minutes the account will be	FMT: Security Management	Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Threshold))

		locked out for.		
16308	Local Administrator Account Logon Delay	The following TechNet topic explains the net accounts command line utility (options for managing account lockout policy are included below the link): Net Accounts: http://technet.microsoft.com/en-us/library/bb490698.aspx **/lockoutthreshold: *number*** **:** Sets the number of times a bad password may be entered until the account is locked out. If set to 0 then the account is never locked out. **/lockoutwindow: *minutes*** **:** Sets the number of minutes of the lockout window. **/lockoutduration: *minutes*** **:** Sets the number of minutes the account will be locked out for.	FMT: Security Management	Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Threshold))
15079	Unlock local user account	Disable or activate a local user account: http://technet.microsoft.com/en-us/library/cc781924(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Authentication Failure Handling (FMT_MTD.1(Re-enable))
15080	Unlock local user account	Disable or activate a local user account: http://technet.microsoft.com/en-us/library/cc781924(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

15730	Unlock local user account	Disable or activate a local user account: http://technet.microsoft.com/en-us/library/cc781924(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)

9.3.1.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. Audit Id 4670 indicates the threshold of unsuccessful authentication attempts has been reached and the action taken to disable the user account. Audit Id 4767 indicates the action taken to unlock a non-administrator user account. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) in the Security event log by a user with administrator credentials on the local computer.

To enable audit policy subcategories for Account Management of User operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:"Account Lockout" /success:enable /failure:enable
- auditpol /set /subcategory:"User Account Management" /success:enable /failure:enable
- auditpol /set /subcategory:"User Account Management" /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4625	Account Lockout	An account failed to logon	Logged: <Date and time of event> Security ID: <SID of locked account> Account Name: <name of locked account> Account Domain: <domain of locked account>
4740	User Account Management	A user account was locked out.	Logged: <Date and time of event> Security ID: <SID of locked account> Account Name: <name of locked account> Account Domain: <domain of locked account>
4767	User Account Management	A user account was unlocked.	Logged: <Date and time of event> Security ID: <SID of user account> Account Name: <name of unlocked account> Account Domain: <domain of unlocked account>

9.3.2 User Security Attributes

User Attribute Definition for Individual Users (FIA_ATD.1(USR)), Revocation for Authorized Administrators (FMT_REV.1(Admin)
Management of TSF Data for Initialization of User Security Attributes (FMT_MTD.1(Init-Attr)), Management of TSF Data for Modification of User Security Attributes Other Than Authentication Data (FMT_MTD.1(Mod-Attr))
Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Auth)), Security Roles (FMT_SMR.1)

9.3.2.1 Interfaces

The functional specification evidence associated with the interfaces for FIA_ATD.1 and related audits and management operations are indicated in the table below (the legend for the below table is in section 1):

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15121	Create local machine group	Create a local group: http://technet.microsoft.com/en-us/library/cc737998(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Init-Attr))
14339	Create local machine group	Create a local group: http://technet.microsoft.com/en-us/library/cc737998(v=ws.10).aspx	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
14340	Create local machine group	Create a local group: http://technet.microsoft.com/en-us/library/cc737998(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15746	Create local machine group	Create a local group: http://technet.microsoft.com/en-	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		us/library/cc737998(v=ws.10).aspx		
15123	Add a member to a local machine group	Add a member to a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Attr))
15124	Add a member to a local machine group	Add a member to a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
15125	Add a member to a local machine group	Add a member to a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15754	Add a member to a local machine group	Add a member to a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15304	Add a member to a local machine group	Add a member to a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FMT: Security Management	Security Roles (FMT_SMR.1)
15126	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Attr))

		Notice the “Additional considerations” heading modifies the instructions to accommodate removing a member from a local group in the user interface method. For the command-line method the same command is used as for adding a member with the exception of replacing the “/add” parameter with “/delete” (see the following TechNet topic for the syntax for the command line option: Net localgroup: http://technet.microsoft.com/en-us/library/bb490706.aspx).		
15127	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx Notice the “Additional considerations” heading modifies the instructions to accommodate removing a member from a local group in the user interface method. For the command-line method the same command is used as for adding a member with the exception of replacing the “/add” parameter with “/delete” (see the following TechNet topic for the syntax for the command line option: Net localgroup: http://technet.microsoft.com/en-	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))

		us/library/bb490706.aspx).		
15128	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx Notice the “Additional considerations” heading modifies the instructions to accommodate removing a member from a local group in the user interface method. For the command-line method the same command is used as for adding a member with the exception of replacing the “/add” parameter with “/delete” (see the following TechNet topic for the syntax for the command line option: Net localgroup: http://technet.microsoft.com/en-us/library/bb490706.aspx).	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15755	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx Notice the “Additional considerations” heading modifies the instructions to accommodate removing a member from a local group in the user interface method. For the command-line method the same command is used as for adding a member with the exception of replacing the “/add” parameter with “/delete” (see the following	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		TechNet topic for the syntax for the command line option: Net localgroup: http://technet.microsoft.com/en-us/library/bb490706.aspx).		
15303	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx Notice the "Additional considerations" heading modifies the instructions to accommodate removing a member from a local group in the user interface method. For the command-line method the same command is used as for adding a member with the exception of replacing the "/add" parameter with "/delete" (see the following TechNet topic for the syntax for the command line option: Net localgroup: http://technet.microsoft.com/en-us/library/bb490706.aspx).	FMT: Security Management	Security Roles (FMT_SMR.1)
15302	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx Notice the "Additional considerations" heading modifies the instructions to	FMT: Security Management	Revocation for Authorized Administrators (FMT_REV.1(Admin))

		accommodate removing a member from a local group in the user interface method. For the command-line method the same command is used as for adding a member with the exception of replacing the "/add" parameter with "/delete" (see the following TechNet topic for the syntax for the command line option: Net localgroup: http://technet.microsoft.com/en-us/library/bb490706.aspx).		
15122	Delete local machine group	Delete a local group: http://technet.microsoft.com/en-us/library/cc778278(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Attr))
14355	Delete local machine group	Delete a local group: http://technet.microsoft.com/en-us/library/cc778278(v=ws.10).aspx	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
14357	Delete local machine group	Delete a local group: http://technet.microsoft.com/en-us/library/cc778278(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15747	Delete local machine group	Delete a local group: http://technet.microsoft.com/en-us/library/cc778278(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15137	Create local user	Create a local user account: http://technet.microsoft.com/en-	FMT: Security	Management of TSF Data for Initialization of User Security

		us/library/cc778832(v=ws.10).aspx	Management	Attributes (FMT_MTD.1(Init-Attr))
14362	Create local user	Create a local user account: http://technet.microsoft.com/en-us/library/cc778832(v=ws.10).aspx	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
14364	Create local user	Create a local user account: http://technet.microsoft.com/en-us/library/cc778832(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15748	Create local user	Create a local user account: http://technet.microsoft.com/en-us/library/cc778832(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15307	Create local user	Create a local user account: http://technet.microsoft.com/en-us/library/cc778832(v=ws.10).aspx	FMT: Security Management	Security Roles (FMT_SMR.1)
15138	Delete Local User	Delete a local user account: http://technet.microsoft.com/en-us/library/cc739627(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Attr))
15090	Delete Local User	Delete a local user account: http://technet.microsoft.com/en-us/library/cc739627(v=ws.10).aspx	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))

15091	Delete Local User	Delete a local user account: http://technet.microsoft.com/en-us/library/cc739627(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15751	Delete Local User	Delete a local user account: http://technet.microsoft.com/en-us/library/cc739627(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15308	Delete Local User	Delete a local user account: http://technet.microsoft.com/en-us/library/cc739627(v=ws.10).aspx	FMT: Security Management	Security Roles (FMT_SMR.1)
15151	Change Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Auth))
15152	Change Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
15153	Change Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15760	Change Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15154	Reset Local Account Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Auth))

15155	Reset Local Account Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
15156	Reset Local Account Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15762	Reset Local Account Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15320	Change local user name	Change local user name: http://technet.microsoft.com/en-us/library/cc738626(v=ws.10).aspx	FMT: Security Management	Management of TSF Data for Modification of Authentication Data (FMT_MTD.1(Mod-Attr))
15321	Change local user name	Change local user name: http://technet.microsoft.com/en-us/library/cc738626(v=ws.10).aspx	FIA: Identification & Authentication	User Attribute Definition for Individual Users (FIA_ATD.1(USR))
15322	Change local user name	Change local user name: http://technet.microsoft.com/en-us/library/cc738626(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15758	Change local user name	Change local user name: http://technet.microsoft.com/en-us/library/cc738626(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)

9.3.2.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

To enable audit policy subcategories for Account Management of User operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:”User Account Management” /success:enable /failure:enable
- auditpol /set /subcategory:”Security Group Management” /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4720	User Account Management	A user account was created.	Logged: <Date and time of event> Security ID: <SID of new user account> Account Name: <new user account name> Account Domain: <domain of new account (or computer name for standalone)> SAM Account Name: <new SAM account name> Display Name: <display name of new account> User Principal Name < UPN of new account> Primary Group ID: <group membership for new account> Logon Hours <time and day to logon for new account> Privileges: <list of privileges for new account>
4723	User Account Management	An attempt was made to change an account's password.	Logged: <Date and time of event> Security ID: <SID of user account> Account Name: <name of account> Account Domain: <domain of account if applicable, otherwise computer>
4724	User Account Management	An attempt was made to reset an account's password.	Logged: <Date and time of event> Security ID: <SID of user account> Account Name: <name of account> Account Domain: <domain of account if applicable, otherwise computer>
4726	User Account Management	A user account was deleted.	Logged: <Date and time of event> Security ID: <SID of deleted user account> Account Name: <name of deleted account> Account Domain: <domain of deleted account if

			applicable, otherwise computer>
4738	User Account Management	A user account was changed.	Logged: <Date and time of event> Security ID: <SID of changed user account> Account Name: <name of changed account> Account Domain: <domain of changed account if applicable, otherwise computer> <List of changed attributes and their corresponding values – see audit Id 4720 for set of all possible attributes that may be changed>
4781	User Account Management	The name of an account was changed	Logged: <Date and time of event> Security ID: <SID of renamed user account> Old Account Name: <old name of account> New Account Name: <new name of account> Account Domain: <domain of deleted account if applicable, otherwise computer>
4731	Security Group Management	A security-enabled local group was created.	Logged: <Date and time of event> Security ID: <SID of new security group> Group Name: <name of new security group> Group Domain: <local computer name >
4732	Security Group Management	A member was added to a security-enabled local group.	Logged: <Date and time of event> Security ID: <SID of added user account> Account Name: <name of added account> Security ID: <SID of security group> Group Name: <name of security group> Group Domain: <domain of security group>
4733	Security Group Management	A member was removed from a security-enabled local group.	Logged: <Date and time of event> Security ID: <SID of removed user account> Account Name: <name of removed account> Security ID: <SID of security group> Group Name: <name of security group> Group Domain: <domain of security group>
4734	Security Group Management	A security-enabled local group was deleted.	Logged: <Date and time of event> Security ID: <SID of deleted security group> Group Name: <name of deleted security group>

			Group Domain: <local computer name>

9.3.3 Timing of OS Logon for Remote IT Entities

Timing of Authentication for OS Logon (FIA_UAU.1(RITE))

The functional specification evidence associated with the interfaces for FIA_UAU.1(RITE) and related audits and management operations are covered by the FDP_IFC.1(OSPP), FDP_IFF.1(OSPP), and FTP_ITC.1(OS).

9.3.4 Timing of OS Logon for Users

Timing of Authentication for OS Logon (FIA_UAU.1(OS))
Timing of Identification (FIA_UID.1)
Multiple Authentication Mechanisms (FIA_UAU.5), Management of Security Functions Behavior for Password Management (FMT_MOF.1(Pass)), Protected Authentication Feedback (FIA_UAU.7)

9.3.4.1 Interfaces

The functional specification evidence associated with the interfaces for FIA_UAU.1(OS), FIA_UID.1, FIA_UAU.5, FIA_UAU.7 and related audits and management operations are indicated in the table below (the legend for the below table is in section 1).

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15184	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15185	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.	FIA: Identification &	Timing of Identification (FIA_UID.1)

		85).aspx	Authentication	
15186	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15187	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15715	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15249	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15252	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)

15255	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15258	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15716	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15250	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15253	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)
15256	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)

15259	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15718	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15251	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15254	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)
15257	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15260	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

15720	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15188	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15189	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)
15190	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15191	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15722	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		85).aspx		
15192	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15193	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)
15194	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15195	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15724	InitiailizeSecurityContext	InitializeSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
14537	scwcmd.exe configure, analyze and rollback	scwcmd.exe: http://technet.microsoft.com/en-	FIA: Identification	Timing of Authentication for

	commands	us/library/ff807358(v=ws.10).aspx	& Authenticatio n	OS Logon (FIA_UAU.1(OS))
14536	scwcmd.exe configure, analyze and rollback commands	scwcmd.exe: http://technet.microsoft.com/en-us/library/ff807358(v=ws.10).aspx	FIA: Identification & Authenticatio n	Timing of Identification (FIA_UID.1)
15168	scwcmd.exe configure, analyze and rollback commands	scwcmd.exe: http://technet.microsoft.com/en-us/library/ff807358(v=ws.10).aspx	FIA: Identification & Authenticatio n	Multiple Authentication Mechanisms (FIA_UAU.5)
15169	scwcmd.exe configure, analyze and rollback commands	scwcmd.exe: http://technet.microsoft.com/en-us/library/ff807358(v=ws.10).aspx	FIA: Identification & Authenticatio n	Protected Authentication Feedback (FIA_UAU.7)
15160	scwcmd.exe configure, analyze and rollback commands (success audit)	scwcmd.exe: http://technet.microsoft.com/en-us/library/ff807358(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15161	scwcmd.exe configure, analyze and rollback commands	scwcmd.exe: http://technet.microsoft.com/en-us/library/ff807358(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

	(failure audit)			
15736	scwcmd.exe configure, analyze and rollback commands	scwcmd.exe: http://technet.microsoft.com/en-us/library/ff807358(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
14426	Log On Tab of <Name of Service> Properties Window	Services Snap-in: http://technet.microsoft.com/en-us/library/cc757797(v=WS.10).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
14425	Log On Tab of <Name of Service> Properties Window	Services Snap-in: http://technet.microsoft.com/en-us/library/cc757797(v=WS.10).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)
15177	Log On Tab of <Name of Service> Properties Window	Services Snap-in: http://technet.microsoft.com/en-us/library/cc757797(v=WS.10).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15178	Log On Tab of <Name of Service> Properties Window	Services Snap-in: http://technet.microsoft.com/en-us/library/cc757797(v=WS.10).aspx	FIA: Identification & Authentication	Protected Authentication Feedback (FIA_UAU.7)
14427	Log On Tab of <Name of	Services Snap-in: http://technet.microsoft.com/en-	FAU: Security	Audit Data Generation

	Service> Properties Window (success audit)	us/library/cc757797(v=WS.10).aspx	Audit	(FAU_GEN.1(OSPP))
15175	Log On Tab of <Name of Service> Properties Window (failure audit)	Services Snap-in: http://technet.microsoft.com/en-us/library/cc757797(v=WS.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15738	Log On Tab of <Name of Service> Properties Window	Services Snap-in: http://technet.microsoft.com/en-us/library/cc757797(v=WS.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15216	SMB Commands (SMB_COM_SESSION_SETUP_ANDX, SMB_COM_NEGOTIATE)	SMB_COM_NEGOTIATE: http://msdn.microsoft.com/en-us/library/ee441913(v=prot.20).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15217	SMB Commands (SMB_COM_SESSION_SETUP_ANDX, SMB_COM_NEGOTIATE)	SMB_COM_NEGOTIATE: http://msdn.microsoft.com/en-us/library/ee441913(v=prot.20).aspx	FIA: Identification & Authentication	Timing of Identification (FIA_UID.1)
15218	SMB Commands (SMB_COM_SESSION_SETUP_ANDX, SMB_COM_NEGOTIATE)	SMB_COM_NEGOTIATE: http://msdn.microsoft.com/en-us/library/ee441913(v=prot.20).aspx	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
14414	Batch Logon	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-	FIA: Identification	Timing of Authentication for

		us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	& Authentication	OS Logon (FIA_UAU.1(OS))
15173	Batch Logon	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15737	Batch Logon	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16411	NewCredentials Logon	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))

		85).aspx		
16412	NewCredentials Logon	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16413	NewCredentials Logon	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16414	User initiated locking	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15281	User initiated locking	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx		
15711	User initiated locking	SECURITY_LOGON_TYPE: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380129(v=vs.85).aspx LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15149	Password expiration with username and password	Change your password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15141	User logon with username and password	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FIA: Identification & Authentication	Timing of Authentication for OS Logon (FIA_UAU.1(OS))
15179	User logon with username and password	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FIA: Identification & Authenticatio	Timing of Identification (FIA_UID.1)

			n	
15221	User logon with username and password	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows Passwords in Windows 8: FAQ: http://windows.microsoft.com/en-us/windows-8/passwords-in-windows-8-faq	FIA: Identification & Authentication	Multiple Authentication Mechanisms (FIA_UAU.5)
15180	User logon with username and password	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FIA: Identification & Authentication	Protected Authentication Feedback (FIA_UAU.7)
15142	User logon with username and password (failure audit)	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15144	User logon with username and password (success audit)	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15740	User logon with username and password	Sign in to or out of Windows: http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15312	Password Management	Password Policy: http://technet.microsoft.com/en-us/library/cc783512(v=ws.10).aspx	FMT: Security Management	Management of Security Functions Behavior for Password Management

				(FMT_MOF.1(Pass))
15313	Password Management	Password Policy: http://technet.microsoft.com/en-us/library/cc783512(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15735	Password Management	Password Policy: http://technet.microsoft.com/en-us/library/cc783512(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15316	Change Password	Change Password: http://windows.microsoft.com/en-us/windows-8/change-your-password	FMT: Security Management	Management of Security Functions Behavior for Password Management (FMT_MOF.1(Pass))

9.3.4.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

To enable audit policy subcategories for operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:"Logon" /success:enable /failure:enable
- auditpol /set /subcategory:"Authentication Policy Change" /success:enable /failure:enable
- auditpol /set /subcategory:"User Account Management" /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4624	Logon	An account was successfully logged on.	Logged: <Date and time of event> Security ID: <SID of enabled user account> Account Name: <name of enabled account> Account Domain: <domain of enabled account if applicable, otherwise computer> Workstation Name: <name of computer user logged on> Source Network Address: <IP address of computer logged on>

4625	Logon	An account failed to log on.	Logged: <Date and time of event> Security ID: <SID of enabled user account> Account Name: <name of enabled account> Account Domain: <domain of enabled account if applicable, otherwise computer> Workstation Name: <name of computer user logged on> Source Network Address: <IP address of computer logged on>
4723	User Account Management	An attempt was made to change an account's password.	Logged: <Date and time of event> Security ID: <SID of user account> Account Name: <name of account> Account Domain: <domain of account if applicable, otherwise computer>
4724	User Account Management	An attempt was made to reset an account's password.	Logged: <Date and time of event> Security ID: <SID of user account> Account Name: <name of account> Account Domain: <domain of account if applicable, otherwise computer>

To enable audit policy subcategories for operations for picuture password, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:"Credential Validation" /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4776	Credential Validation	The computer attempted to validate the credentials for an account.	Logged: <Date and time of event> Logon Account: <name of enabled account> Source Workstation: <name of computer user logged on>

9.3.5 User-Subject Binding for Individual Users (FIA_USB.1(USR))

Only interfaces that create new subjects (processes) are indicated for security audit generation.

9.3.5.1 Interfaces

The functional specification evidence associated with the interfaces for FIA_USB.1 and related audits and management operations are indicated in the table below (the legend for the below table is in section 1):

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15227	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15276	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15277	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15278	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-	FIA: Identification	User-Subject Binding for Individual Users

		us/library/windows/desktop/bb540756(v=vs.85).aspx	& Authentication	(FIA_USB.1(USR))
15222	AcceptSecurityContext	AcceptSecurityContext: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15229	AdjustTokenGroups	AdjustTokenGroups: http://msdn.microsoft.com/en-us/library/aa375199(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15230	AdjustTokenPrivileges	AdjustTokenPrivileges:: http://msdn.microsoft.com/en-us/library/aa375202(VS.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15245	CreateProcess	CreateProcess:: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682425(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15282	CreateProcess	CreateProcess:: http://msdn.microsoft.com/en-	FAU: Security	Audit Data Generation

		us/library/windows/desktop/ms682425(v=vs.85).aspx	Audit	(FAU_GEN.1(OSPP))
15741	CreateProcess	CreateProcess:: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682425(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15246	CreateProcessAsUser	CreateProcessAsUser: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682429(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15283	CreateProcessAsUser	CreateProcessAsUser: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682429(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15742	CreateProcessAsUser	CreateProcessAsUser: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682429(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15247	CreateProcessWithLogonW	CreateProcessWithLogonW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682431(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15284	CreateProcessWithLogonW	CreateProcessWithLogonW: http://msdn.microsoft.com/en-	FAU: Security	Audit Data Generation

		us/library/windows/desktop/ms682431(v=vs.85).aspx	Audit	(FAU_GEN.1(OSPP))
15743	CreateProcessWithLogonW	CreateProcessWithLogonW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682431(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15248	CreateProcessWithTokenW	CreateProcess:WithTokenW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682434(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15285	CreateProcessWithTokenW	CreateProcess:WithTokenW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682434(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15744	CreateProcessWithTokenW	CreateProcess:WithTokenW: http://msdn.microsoft.com/en-us/library/windows/desktop/ms682434(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15231	CreateRestrictedToken	CreateRestrictedToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446583(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15242	CreateToken	CreateToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374780(v=vs.	FIA: Identification	User-Subject Binding for Individual Users

		85).aspx	& Authentication	(FIA_USB.1(USR))
15243	CreateTokenEx	CreateTokenEx : http://msdn.microsoft.com/en-us/library/windows/desktop/ff714497(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15244	DuplicateToken	DuplicateToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa446616(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15232	DuplicateTokenEx	DuplicateTokenEx: http://msdn.microsoft.com/en-us/library/aa446617(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15233	ImpersonateLoggedOnUser	ImpersonateLoggedOnUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378612(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15234	ImpersonateSelf	ImpresonateSelf: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378729(v=vs.	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))

		85).aspx	n	
15237	RevertToSelf	RevertToSelf: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379317(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15235	SetThreadToken	SetThreadToken: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379590(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15236	SetTokenInformation	SetTokenInformation: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379591(v=vs.85).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15239	Run as administrator	How do I run an application once with a full administrator access token? http://windows.microsoft.com/en-us/windows7/how-do-i-run-an-application-once-with-a-full-administrator-access-token	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
15286	Run as administrator	How do I run an application once with a full administrator access token? http://windows.microsoft.com/en-us/windows7/how-do-i-run-an-application-once-with-a-full-administrator-access-token	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

15745	Run as administrator	How do I run an application once with a full administrator access token? http://windows.microsoft.com/en-us/windows7/how-do-i-run-an-application-once-with-a-full-administrator-access-token	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15238	User logon with username and password	Sign in to or out of Windows : http://windows.microsoft.com/en-us/windows-8/sign-in-out-of-windows	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
16328	Add a member to a local machine group	Add a member to a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))
16327	Remove member from a local machine group	Remove a member from a local group: http://technet.microsoft.com/en-us/library/cc739265(v=ws.10).aspx	FIA: Identification & Authentication	User-Subject Binding for Individual Users (FIA_USB.1(USR))

The following table maps interfaces to user security attributes:

SCE Id	Interface Name	Security Attribute
16328	Add a member to a local machine group	This interface modifies group membership, security roles and privileges.
16327	Remove member from a local machine group	This interface modifies group membership, security roles and privileges.

9.3.5.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

To enable audit policy subcategories for Account Management of User operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:"Process Creation" /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4688	Process Creation	A new process has been created.	Logged: <Date and time of event> Security ID: <SID of enabled user account> Account Name: <name of enabled account> Account Domain: <domain of enabled account if applicable, otherwise computer> New Process ID: <unique Id of new process> New Process Name: <name of new process> Token Elevation Type: <UAC type (full, elevated or limited)>

9.3.6 Public Key Based Authentication (FIA_PK_EXT.1)

9.3.6.1 Interfaces

The interfaces for setting up a trusted channel to communicate with a CA are described in FTP_ITC.1.

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement

15297	Viewing a certificate with a good chain	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FIA: Identification & Authentication	Public Key Based Authentication (FIA_PK_EXT.1)
15298	Viewing a certificate chain with no trusted root	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FIA: Identification & Authentication	Public Key Based Authentication (FIA_PK_EXT.1)
15300	Viewing a revoked certificate	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FIA: Identification & Authentication	Public Key Based Authentication (FIA_PK_EXT.1)
15329	Viewing a revoked certificate	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FMT: Security Management	Management of TSF Data for X.509 Certificates (FMT_MTD.1(X509))
15299	Import a trusted root	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FIA: Identification & Authentication	Public Key Based Authentication (FIA_PK_EXT.1)
15330	Import a trusted root	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FMT: Security Management	Management of TSF Data for X.509 Certificates (FMT_MTD.1(X509))
15301	Delete a trusted root	Manage Certificates: http://technet.microsoft.com/en-	FIA: Identification	Public Key Based Authentication

		us/library/cc771377.aspx	& Authenticatio n	(FIA_PK_EXT.1)
15331	Delete a trusted root	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FMT: Security Management	Management of TSF Data for X.509 Certificates (FMT_MTD.1(X509))
15311	Enrolling for a certificate on a standalone machine	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FIA: Identification & Authenticatio n	Public Key Based Authentication (FIA_PK_EXT.1)
15333	Enrolling for a certificate on a standalone machine	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FMT: Security Management	Management of TSF Data for X.509 Certificates (FMT_MTD.1(X509))
15338	Remotely administering certificates	Manage Certificates: http://technet.microsoft.com/en-us/library/cc771377.aspx	FIA: Identification & Authenticatio n	Public Key Based Authentication (FIA_PK_EXT.1)

9.3.6.2 Audit Policy

There are no audits defined for FIA_PK_EXT.1.

9.4 Protection of the TSF (FPT)

9.4.1 Timestamps

Reliable Time Stamps (FPT_STM.1)

9.4.1.1 Interfaces

The functional specification evidence associated with the interfaces for FPT_STM.1 and related audits is indicated in the following table[81](the legend for the below table is in section 1.1):

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15053	Set-Date	Set-Date: http://technet.microsoft.com/en-us/library/ee176960.aspx	FPT: Protection of the TSF	Reliable Time Stamps (FPT_STM.1)
14774	Set-Date	Set-Date: http://technet.microsoft.com/en-us/library/ee176960.aspx	FMT: Security Management	Management of TSF Data for General TSF Data (FMT_MTD.1(GEN))
14776	Set-Date	Set-Date: http://technet.microsoft.com/en-us/library/ee176960.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15764	Set-Date	Set-Date: http://technet.microsoft.com/en-us/library/ee176960.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
8848	s_W32TimeSync	W32TimeSync: http://msdn.microsoft.com/en-us/library/cc249685(v=prot.20).aspx	FPT: Protection of the TSF	Reliable Time Stamps (FPT_STM.1)
4933	s_W32TimeSync	W32TimeSync: http://msdn.microsoft.com/en-	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

[81] This table will be in the final Security Target as a list of interfaces that were examined in the evaluation.

		us/library/cc249685(v=prot.20).aspx		
15709	s_W32TimeSync	W32TimeSync: http://msdn.microsoft.com/en-us/library/cc249685(v=prot.20).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15050	Get-Date	Get-Date: http://technet.microsoft.com/en-us/library/ee176845.aspx	FPT: Protection of the TSF	Reliable Time Stamps (FPT_STM.1)

9.4.1.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the System event log.

To enable the audit policy subcategory for syncing time, run the following command at an elevated command prompt:

- auditpol /set /subcategory:"Security State Change" /success:enable /failure:enable

Event Id	Event Source	Message	Fields
37	Time Service In the System Log	The time provider NtpClient is currently receiving valid time data from <time source>.	Logged: <Date and time of event> Time Source: <source of provided time>
4616	Security State Change In the Security Log	The system time was changed.	Logged: <Date and time of event> Previous Time: <old system time> New Time: <new system time>

9.5 Trusted Path / Channels (FTP)

Inter-TSF Trusted Channel (FTP_ITC.1 (OS)) – IPSEC, Basic Internal TSF Data Transfer Protection (FPT_ITT.1), Remote Management Capabilities (FMT_SMF_RMT.1)

9.5.1 IPsec

9.5.1.1 Interfaces

The functional specification evidence associated with the interfaces for FTP_ITC.1 (OS), FPT_ITT.1, and related audit SFRs is indicated in the table below (the legend for the below table is in section 1).

SCE ID	Interface Name	Search Term: Interface Design	Security Functional Class	Security Functional Requirement
1499	ISA_HASH	RFC 4306: http://tools.ietf.org/html/rfc4306	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
16350	ISA_HASH	RFC 4306: http://tools.ietf.org/html/rfc4306	FMT: Security Management	Remote Management Capabilities (FMT_SMF_RMT.1)
3678	ISA_HASH	RFC 4306: http://tools.ietf.org/html/rfc4306	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15712	ISA_HASH	RFC 4306: http://tools.ietf.org/html/rfc4306	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15062	New-NetIPsecPhase1AuthSet	New-NetIPsecAuthProposal: http://technet.microsoft.com/en-us/library/jj554847.aspx FTP_ITC.1.F2	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))

16543	New-NetIPsecPhase1AuthSet	New-NetIPsecAuthProposal: http://technet.microsoft.com/en-us/library/jj554847.aspx	FPT: Protection of the TSF	Basic Internal TSF Data Transfer Protection (FPT_ITT.1)
15063	New-NetIPsecMainModeCryptoSet	New-NetIPsecMainModeCryptoSet: http://technet.microsoft.com/en-us/library/jj554882.aspx FTP_ITC.1.F2	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
16544	New-NetIPsecMainModeCryptoSet	New-NetIPsecMainModeCryptoSet: http://technet.microsoft.com/en-us/library/jj554882.aspx	FPT: Protection of the TSF	Basic Internal TSF Data Transfer Protection (FPT_ITT.1)
15064	New-NetIpsecQuickModeCryptoSet	New-NetIpsecQuickModeCryptoSet: http://technet.microsoft.com/en-us/library/jj573823.aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
16545	New-NetIpsecQuickModeCryptoSet	New-NetIpsecQuickModeCryptoSet: http://technet.microsoft.com/en-us/library/jj573823.aspx	FPT: Protection of the TSF	Basic Internal TSF Data Transfer Protection (FPT_ITT.1)
15061	Network Interfaces	[MS-IKEE]: Internet Key Exchange Protocol Extensions (Appendix A): http://msdn.microsoft.com/en-us/library/cc233476.aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
16546	Network Interfaces	[MS-IKEE]: Internet Key Exchange Protocol Extensions (Appendix A): http://msdn.microsoft.com/en-us/library/cc233476.aspx	FPT: Protection of the TSF	Basic Internal TSF Data Transfer Protection (FPT_ITT.1)

9.5.1.2 Audit Policy

Audits for IPsec operations are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer.

To enable audit policy subcategories for IPsec operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:"IPsec Main Mode" /success:enable /failure:enable
- auditpol /set /subcategory: "IPsec Quick Mode" /success:enable /failure:enable

Id	Policy Subcategory	Message	Fields
4650, 4651	IPsec Main Mode	Ipsec main mode security association was established. A certificate was used for authentication.	Logged: <Date and time of event> Task category: <type of event> Local Endpoint: <Subject identity as IP address> Remote Endpoint: <Subject identity as IP address of non-TOE endpoint of connection > Keying Module Name: <Transport layer protocol as IKEv1 or IKEv2> Local Certificate: <The entry in the SPD that applied to the decision as certificate SHA Thumbprint> Remote Certificate: <The entry in the SPD that applied to the decision as certificate SHA Thumbprint> Cryptographic Information: <The entry in the SPD that applied to the decision as MM SA Id and cryptographic parameters established in the SA> Keywords: <Outcome as Success>
5451	IPsec Quick Mode	IPsec quick mode security association was established	Logged: <Date and time of event> Task category: <type of event> Local Endpoint: <Subject identity as IP address/port> Remote Endpoint: <Subject identity as IP address/port of non-TOE endpoint of connection >

			Keying Module Name: <Transport layer protocol as IKEv1 or IKEv2> Cryptographic Information: <The entry in the SPD that applied to the decision as MM SA Id, QM SA Id, Inbound SPI, Outbound SPI and cryptographic parameters established in the SA > Keywords: <Outcome as Success>
4652	IPsec Main Mode	IPsec main mode negotiation failed	Logged: <Date and time of event> Local Endpoint: <Subject identity as IP address> Remote Endpoint: <Subject identity as IP address of non-TOE endpoint of connection/channel> Keying Module Name: <Transport layer protocol as IKEv1 or IKEv2> Failure Information: <Outcome as Failure; Reason for failure as the entry in the SPD that applied to the decision> Cryptographic Information: <The entry in the SPD that applied to the decision as cryptographic parameters attempted to establish in the SA>
4654	IPsec Quick Mode	IPsec quick mode negotiation failed	Logged: <Date and time of event> Local Endpoint: <Subject identity as IP address/port> Remote Endpoint: <Subject identity as IP address/port of non-TOE endpoint of connection/channel > Keying Module Name: <Transport layer protocol as IKEv1 or IKEv2>Failure Information: <Outcome as Failure; Reason for failure as the entry in the SPD that applied to the decision as the MA SA Id, QM Filter Id, Tunnel Id, Traffic Selector Id >

9.5.2 TLS

Inter-TSF Trusted Channel (FTP_ITC.1 (OS)) – TLS, Remote Management Capabilities (FMT_SMF_RMT.1)

9.5.2.1 Interfaces

The functional specification evidence associated with the interfaces for FTP_ITC.1 (OS) and related audit SFRs is indicated in the table below (the legend for the below table is in section 1).

SCE ID	Interface Name	Search Term: Interface Design	Security Functional Class	Security Functional Requirement
15869	AcquireCredentialsHandle	AcquireCredentialsHandle: http://msdn.microsoft.com/en-us/library/windows/desktop/aa374712(v=vs.85).aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
15205	AcceptSecurityContext	AcceptSecurityContext : http://msdn.microsoft.com/en-us/library/windows/desktop/aa374703(v=vs.85).aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
15294	IniitializeSecurityContext	InitializeSecurityContext: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375506(v=vs.85).aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
15870	BCryptAddContextFunction	BCryptAddContextFunction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375360(v=vs.85).aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
15871	BCryptRemoveContextFunction	BCryptRemoveContextFunction: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375492(v=vs.85).aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
15057	TLS-SSL Security Provider	RFC 5246, http://tools.ietf.org/html/rfc5246	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))
15058	TLS-SSL Security Provider	RFC 5246, http://tools.ietf.org/html/rfc5246	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15713	TLS-SSL Security Provider	RFC 5246, http://tools.ietf.org/html/rfc5246	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15059	SSL Cipher Suite Order	Prioritizing Schannel Cipher Suites: http://msdn.microsoft.com/en-	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))

		us/library/windows/desktop/bb870930(v=vs.85).aspx		
15060	7.4.9. Finished	[MS-TLSP]: Transport Layer Security (TLS) Profile (Appendix A): http://msdn.microsoft.com/en-us/library/dd208005.aspx	FTP: Trusted Path/Channels	Inter-TSF Trusted Channel (FTP_ITC.1 (OS))

9.5.2.2 Audit Policy

Audits for TLS operations are outlined in the tables below with details for each event Id. The indicated events may be viewed in the Event Viewer application (eventvwr.exe).

To enable TLS event logging in the System Event Log, see the following link:

- http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q260729

To enable CAPI2 logging in the Operational log, see the following link:

- http://technet.microsoft.com/en-us/library/cc749296(v=WS.10).aspx
-

Event Id	Event Source	Message	Fields
36880	Schannel in the System Event Log	An SSL server handshake completed successfully. The negotiated cryptographic parameters are as follows:.	Logged: <Date and time of event> Protocol: <protocol designator> CipherSuite: <hexadecimal designator for cipher suite> Exchange strength: <key length of exchange key in bits> In the Details view of the event: System -> TimeCreated -> SystemTime: <Date and time of event> System -> Execution -> ProcessID: <process ID of the process that created the event> System -> Execution -> ThreadID: <thread ID of the thread that created the event>
36874	Schannel in the System	The SSL connection request has failed.	Logged: <Date and time of event>

	Event Log		Reason: <reason for failure> In the Details view of the event: System -> TimeCreated -> SystemTime: <Date and time of event> System -> Execution -> ProcessID: <process ID of the process that created the event> System -> Execution -> ThreadID: <thread ID of the thread that created the event>
11	CAPI2 Operational log in the Microsoft Windows section of the Applications and Services Logs This event is relevant on the server side of the channel when client authentication is performed. For successful connections this event provides the subject name of the client's certificate.	Build Chain	In the Details view of the event: System -> TimeCreated -> SystemTime: <Date and time of event> System -> Execution -> ProcessID: <process ID of the process that created the event> System -> Execution -> ThreadID: <thread ID of the thread that created the event> UserData -> CertGetCertificateChain -> Certificate -> subjectName : <name in client certificate>
81	CAPI2 Operational log in the Microsoft Windows section of the Applications and Services Logs This event is relevant on the client side of the channel. This provides the servers	Verify Trust	In the Details view of the event: System -> TimeCreated -> SystemTime: <Date and time of event> System -> Execution -> ProcessID: <process ID of the process that created the event> System -> Execution -> ThreadID: <thread ID of the thread that created the event> UserData -> WinVerifyTrust -> CertificateInfo -> displayName : <name in server certificate>

	certificate name. Note that this name must match the first part of the server's URL in the HTTPS case. There may be multiple CAPI2 81 events per TLS authentication. At least one of the events will provide the display name of the certificate in the Certificate Info portion of the event.		
36888	Schannel in the System Event Log	A fatal alert was generated and sent to the remote endpoint. This may result in termination of the connection. The TLS protocol defined fatal error code is <TLS fatal error code>. The Windows SChannel error state is <SChannel error state>."	Logged: <Date and time of event> Reason: <reason for failure> In the Details view of the event: <TLS fatal error code> <SChannel error state>

The events in the System Event Log are correlated with the events in the CAPI2 operational log.

The correlation between the System Event Log events and the CAPI2 operational log events is done by first matching the SystemTime of the system event as closely as possible with the CAPI2 event. On the server side the ProcessID and ThreadID for the events must also match.

9.6 TOE Access (FTA)

9.6.1 Session Locking

TSF-initiated Session Locking (FTA_SSL.1) and User-initiated Locking (FTA_SSL.2)

9.6.1.1 Interfaces

The functional specification evidence associated with the interfaces for FTA_SSL.1 and FTA_SSL.2 and related audits are indicated in the table below (the legend for the below table is in section 1):

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
14316	User Inactivity Interval	Interactive logon: Machine inactivity limit: http://technet.microsoft.com/en-us/library/jj966265(v=ws.10).aspx	FTA: TOE Access	TSF-initiated Session Locking (FTA_SSL.1)
15279	User Inactivity Interval	Audit other account logon events: http://technet.microsoft.com/en-us/library/dd772704(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15710	User Inactivity Interval	Audit other account logon events: http://technet.microsoft.com/en-us/library/dd772704(v=ws.10).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15280	User initiated Locking	How do I lock or unlock my PC?: http://windows.microsoft.com/en-us/windows-8/lock-unlock-pc	FTA: TOE Access	User-initiated Locking (FTA_SSL.2)
15281	User initiated Locking	How do I lock or unlock my PC?: http://windows.microsoft.com/en-us/windows-8/lock-unlock-pc	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15711	User initiated Locking	How do I lock or unlock my PC?: http://windows.microsoft.com/en-	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		us/windows-8/lock-unlock-pc		
15206	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FTA: TOE Access	TSF-initiated Session Locking (FTA_SSL.1)
15207	LsaLogonUser	LsaLogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378292(v=vs.85).aspx	FTA: TOE Access	User-initiated Locking (FTA_SSL.2)
15270	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FTA: TOE Access	TSF-initiated Session Locking (FTA_SSL.1)
15273	LogonUser	LogonUser: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FTA: TOE Access	User-initiated Locking (FTA_SSL.2)
15271	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FTA: TOE Access	TSF-initiated Session Locking (FTA_SSL.1)
15274	LogonUserEx	LogonUserEx: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378189(v=vs.85).aspx	FTA: TOE Access	User-initiated Locking (FTA_SSL.2)
15272	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.	FTA: TOE Access	TSF-initiated Session Locking (FTA_SSL.1)

		85).aspx		
15275	LogonUserExExW	LogonUserExExW: http://msdn.microsoft.com/en-us/library/windows/desktop/bb540756(v=vs.85).aspx	FTA: TOE Access	User-initiated Locking (FTA_SSL.2)

9.6.1.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log. These audits are do not distinguish the difference of TSF- vs. user-initiated session locking.

To enable audit policy subcategories for Logon operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:"Logon" /success:enable /failure:enable
- auditpol /set /subcategory:"Logoff" /success:enable /failure:enable

Event Id	Policy Subcategory	Message	Fields
4800	Logoff	The workstation was locked.	Logged: <Date and time of event> Security ID: <SID of logon user> Account Name: <name of logon account> Account Domain: <domain of logon account>
4801	Logon	The workstation was unlocked.	Logged: <Date and time of event> Security ID: <SID of logon user> Account Name: <name of logon account> Account Domain: <domain of logon account>
4625	Logon	An account failed to logon.	Logged: <Date and time of event> Security ID: <SID of logon user> Account Name: <name of logon account> Account Domain: <domain of logon account>

9.6.2 Security Audit (FAU)

Audit Review (FAU_SAR.1), Restricted Audit Review (FAU_SAR.2), Selective Audit (FAU_SEL.1), Protected Audit Trail Storage (FAU_STG.1), Action in Case of Possible Audit Data Loss (FAU_STG.3), Prevention of Audit Data Loss (FAU_STG.4), Management of TSF Data for Audit Selection (FMT_MTD.1(AuditSel)), Management of TSF Data for Audit Data (FMT_MTD.1(Audit)), Management of TSF Data for Audit Log Failure (FMT_MTD.1(AuditFail)), and Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg))

9.6.2.1 Interfaces

The functional specification evidence associated with the interfaces for FAU_GEN.1(OSPP), FAU_GEN.2, FAU_SAR.1, FAU_SAR.2, FAU_SEL.1, FAU_STG.1, FAU_STG.3, FAU_STG.4 and related management interfaces are indicated in the table below (the legend for the below table is in section 1):

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
15774	Startup and shutdown of the TOE and audit function	Audit Security State Change: http://technet.microsoft.com/en-us/library/dd772631(v=ws.10).aspx Event Log Performance Monitoring Events: http://technet.microsoft.com/en-us/library/dd772682(v=ws.10).aspx Shutdown command: http://technet.microsoft.com/en-us/library/cc732503.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15775	Startup and shutdown of the TOE and audit function	Audit Security State Change: http://technet.microsoft.com/en-us/library/dd772631(v=ws.10).aspx Event Log Performance Monitoring Events:	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		http://technet.microsoft.com/en-us/library/dd772682(v=ws.10).aspx Shutdown command: http://technet.microsoft.com/en-us/library/cc732503.aspx		
16374	Startup and shutdown of the TOE and audit function	Audit Security State Change: http://technet.microsoft.com/en-us/library/dd772631(v=ws.10).aspx Event Log Performance Monitoring Events: http://technet.microsoft.com/en-us/library/dd772682(v=ws.10).aspx Shutdown command: http://technet.microsoft.com/en-us/library/cc732503.aspx	FAU: Security Audit	Audit Review (FAU_SAR.1)
16375	Startup and shutdown of the TOE and audit function	Audit Security State Change: http://technet.microsoft.com/en-us/library/dd772631(v=ws.10).aspx Event Log Performance Monitoring Events: http://technet.microsoft.com/en-us/library/dd772682(v=ws.10).aspx Shutdown command: http://technet.microsoft.com/en-us/library/cc732503.aspx	FAU: Security Audit	Restricted Audit Review (FAU_SAR.2)
14545	Viewing Audit Logs	Get-EventLog: http://technet.microsoft.com/en-	FMT: Security	Management of TSF Data for Audit Data

		us/library/hh849834.aspx	Management	(FMT_MTD.1(Audit))
14551	Clearing Audit Logs	Clear-EventLog: http://technet.microsoft.com/en-us/library/hh849789.aspx	FAU: Security Audit	Protected Audit Trail Storage (FAU_STG.1)
14552	Clearing Audit Logs	Clear-EventLog: http://technet.microsoft.com/en-us/library/hh849789.aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
14557	Clearing Audit Logs	Clear-EventLog: http://technet.microsoft.com/en-us/library/hh849789.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
14558	Clearing Audit Logs	Clear-EventLog: http://technet.microsoft.com/en-us/library/hh849789.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15707	Setting CrashOnAuditFail for Audit Log	Auditpol: http://technet.microsoft.com/en-us/library/cc731451.aspx	FMT: Security Management	Management of TSF Data for Audit Log Failure (FMT_MTD.1(AuditFail))
15708	Changing the Audit Log Size	Set Maximum Log Size: http://msdn.microsoft.com/en-us/library/windows/desktop/aa378184(v=vs.85).aspx	FMT: Security Management	Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg)
15727	Configure administrator alarm of possible audit data loss	See section 7.1.5 of the *Microsoft Windows 8 Microsoft Windows RT Common Criteria Supplemental Admin Guidance*	FMT: Security Management	Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg)
15726	Control Event Log behavior when the log reaches its	Set Log Retention Policy: http://technet.microsoft.com/en-	FMT: Security	Management of TSF Data for Audit Log Failure

	maximum size	us/library/cc721981.aspx	Management	(FMT_MTD.1(Audit Fail))
16369	Wevtutil.exe	Wevtutil: http://technet.microsoft.com/en-us/library/cc732848.aspx	FMT: Security Management	Management of TSF Data for Audit Log Failure (FMT_MTD.1(Audit Fail))
15820	Selecting the set of audited events	Auditpol: http://technet.microsoft.com/en-us/library/cc731451.aspx	FAU: Security Audit	Selective Audit (FAU_SEL.1)
15819	Selecting the set of audited events	Auditpol: http://technet.microsoft.com/en-us/library/cc731451.aspx	FMT: Security Management	Management of TSF Data for Audit Selection (FMT_MTD.1(Audit Sel))
15838	Selecting the set of audited events	Auditpol: http://technet.microsoft.com/en-us/library/cc731451.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15839	Selecting the set of audited events	Auditpol: http://technet.microsoft.com/en-us/library/cc731451.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15833[82]	Notify administrator of possible audit data loss	Event ID 1103 — Security Channel Configuration: http://technet.microsoft.com/en-us/library/cc774990(v=ws.10).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15832[82]	Notify administrator of possible audit data loss	Event ID 1103 — Security Channel Configuration: http://technet.microsoft.com/en-us/library/cc774990(v=ws.10).aspx	FAU: Security Audit	Action in Case of Possible Audit Data Loss (FAU_STG.3)

[82] This is an internal interface associated with the auditing subsystem that generates an alarm to the administrator upon reaching the configured audit storage threshold and if configured to do so shuts down the system when the audit storage becomes full.

15834[82]	Notify administrator of possible audit data loss	Event ID 1103 — Security Channel Configuration: http://technet.microsoft.com/en-us/library/cc774990(v=ws.10).aspx	FAU: Security Audit	Prevention of Audit Data Loss (FAU_STG.4)
5981	ElfrClearELFW	ElfrClearELFW: http://msdn.microsoft.com/en-us/library/cc231416(v=prot.20).aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
15835	ElfrClearELFW	ElfrClearELFW: http://msdn.microsoft.com/en-us/library/cc231416(v=prot.20).aspx	FAU: Security Audit	Protected Audit Trail Storage (FAU_STG.1)
5982	ElfrClearELFW	ElfrClearELFW: http://msdn.microsoft.com/en-us/library/cc231416(v=prot.20).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15836	ElfrClearELFW	ElfrClearELFW: http://msdn.microsoft.com/en-us/library/cc231416(v=prot.20).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
5988	ElfrBackupELFW	ElfrBackupELFW: http://msdn.microsoft.com/en-us/library/cc231414.aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
8359	ElfrReadELW	ElfrReadELW: http://msdn.microsoft.com/en-us/library/cc231426.aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
6200	EvtRpcClearLog	EvtRpcClearLog: http://msdn.microsoft.com/en-us/library/cc231379.aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))

15846	EvtRpcClearLog	EvtRpcClearLog: http://msdn.microsoft.com/en-us/library/cc231379.aspx	FAU: Security Audit	Protected Audit Trail Storage (FAU_STG.1)
8324	EvtRpcClearLog	EvtRpcClearLog: http://msdn.microsoft.com/en-us/library/cc231379.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
15847	EvtRpcClearLog	EvtRpcClearLog: http://msdn.microsoft.com/en-us/library/cc231379.aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
15851	EvtRpcExportLog	EvtRpcExportLog: http://msdn.microsoft.com/en-us/library/cc231381.aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
8325	EvtRpcOpenLogHandle	EvtRpcOpenLogHandle: http://msdn.microsoft.com/en-us/library/cc231395.aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
6209	EvtRpcQueryNext	EvtRpcQueryNext: http://msdn.microsoft.com/en-us/library/cc231397(v=prot.20).aspx	FMT: Security Management	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
16358	Explorer - Advanced Security Settings Audit Tab (SACL)	Set, view, change, or remove special permissions: http://technet.microsoft.com/en-us/library/cc786378(v=ws.10).aspx	FAU: Security Audit	Selective Audit (FAU_SEL.1))

16368	Registry - Advanced Security Settings, Audit Tab (SACL)	Registry Editor: http://technet.microsoft.com/en-us/library/cc755256.aspx	FAU: Security Audit	Selective Audit (FAU_SEL.1))
15726	Wevtutil.exe	Wevtutil: http://technet.microsoft.com/en-us/library/cc732848.aspx	FAU: Security Audit	Management of TSF Data for Audit Data (FMT_MTD.1(Audit))
16393	Configure administrative and other operational logs to overwrite old events	Set Log Retention Policy: http://technet.microsoft.com/en-us/library/cc721981.aspx	FAU: Security Audit	Management of TSF Data for Audit Storage Threshold (FMT_MTD.1(AuditStg)
16394	Configure administrative and other operational logs to overwrite old events	Set Log Retention Policy: http://technet.microsoft.com/en-us/library/cc721981.aspx	FAU: Security Audit	Action in Case of Possible Audit Data Loss (FAU_STG.3)

9.6.2.2 Audit Policy

Audits are outlined in the table below with details for each audit Id. The indicated audits may be viewed in the Event Viewer application (eventvwr.exe) by a user with administrator credentials on the local computer in the Security event log.

The table below, and in all other “Audit Policy” sections in this document, describes the information in the audit records necessary to associate the audit record and the user that cause the event leading to the creation of the audit record using the Field values “Account Name” and “Account Domain”. Audit records that are not attributable to a user identify do not include these fields (e.g. Id 1102 and 1103 in the table below are generated by the LSA Audit subsystem in response to audit log storage conditions).

To enable audit policy subcategories for Startup/Shutdown operations, run the following commands at an elevated command prompt:

- auditpol /set /subcategory:”Security State Change” /success:enable /failure:enable
- auditpol /set /subcategory:”Audit Policy Change” /success:enable /failure:enable
- auditpol /set /category:” Privilege Use” /success:enable /failure:enable

The policy subcategory value “N/A” indicates the audit is enabled by default and cannot be disabled

Event Id	Policy Subcategory	Message	Fields
1100	Security State Change	The event logging service has shut down	Logged: <Date and time of event> Keywords: <Outcome as Success>
1102	N/A	The audit log was cleared.	Logged: <Date and time of event> Account Name: <sam account name of user who cleared the log> Account Domain: <domain of user who cleared the log> Keywords: <Outcome as Success>
1103	N/A	The security audit log is now <the configured value > percent full.	Logged: <Date and time of event> Keywords: <Outcome as Success>
1104	N/A	The security audit log is full.	Logged: <Date and time of event> Keywords: <Outcome as Success>
4608	Security State Change	Windows is starting up.	Logged: <Date and time of event> Security ID: S-1-5-19 Account Name: Local Service Account Domain: Nt Authority Keywords: <Outcome as Success>
4719	Audit Policy Change	System audit policy was changed.	Logged: <Date and time of event> Account Name: <sam account name of user who cleared the log> Account Domain: <domain of user who cleared the log> Category: <Audit category that was modified> Subcategory: <Audit subcategory that was modified> Changes: <The modification to the set of events.> Keywords: <Outcome as Success>
4673	Sensitive Privilege Use / Non Sensitive Privilege Use	A privileged service was called.	Logged: <Date and time of event> Security ID: <SID of user account that viewed the log> Account Name: <user account name that viewed

			the log> Account Domain: <domain of user accout that viewed the log> Keywords: <Outcome as Success>

See section 2.1.3.2 Audit Policy (FDP DAC) for the audit entries for changing SACLs on objects.

9.7 Cryptographic Support (FCS)

9.7.1.1 Interfaces

The functional specification evidence associated with the interfaces for Cryptographic Support and related audits are indicated in the table below (the legend for the table below:

SCE Id	Interface Name	Search Term: Interface Documentation	Security Functional Class	Security Functional Requirement
16461	BCryptDecrypt	BCryptDecrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375391(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))
7712	BCryptDecrypt	BCryptDecrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375391(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
4459	BCryptEncrypt	BCryptEncrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375421(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))

7715	BCryptEncrypt	BCryptEncrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375421(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
7766	BCryptDestroyKey	BCryptDestroyKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375404(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
14464	BCryptDestroyKey	BCryptDestroyKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375404(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4456	BCryptDestroyKey	BCryptDestroyKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375404(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Zeroization (FCS_CKM_EXT.4)
8950	BCryptDestroySecret	BCryptDestroySecret: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375407(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16465	BCryptDestroySecret	BCryptDestroySecret: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375407(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4473	BCryptDestroySecret	BCryptDestroySecret: http://msdn.microsoft.com/en-	FCS: Cryptographi	Cryptographic Key

		us/library/windows/desktop/aa375407(v=vs.85).aspx	c Support	Zeroization (FCS_CKM_EXT.4)
7767	BCryptFinalizeKeyPair	BCryptFinalizeKeyPair: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375439(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16486	BCryptFinalizeKeyPair	BCryptFinalizeKeyPair: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375439(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4462	BCryptFinalizeKeyPair	BCryptFinalizeKeyPair: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375439(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Generation for Asymmetric Keys (FCS_CKM.1(ASYM))
16547	BCryptFinalizeKeyPair	BCryptFinalizeKeyPair: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375439(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Generation for Asymmetric Keys (FCS_CKM.1(AUTH))
7751	BCryptFinalizeKeyPair	BCryptFinalizeKeyPair: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375439(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
4468	BCryptGenerateSymmetricKey	BCryptGenerateSymmetricKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375453(v=vs.	FCS: Cryptographic Support	Cryptographic Key Generation for Symmetric Keys (FCS_CKM.1(SYM))

		85).aspx		
7756	BCryptGenerateSymmetricKey	BCryptGenerateSymmetricKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375453(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16487	BCryptSecretAgreement	BCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375496(v=vs.85).aspx	FCS: Cryptographic Support	Audit Data Generation (FAU_GEN.1(OSPP))
4471	BCryptSecretAgreement	BCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375496(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for DH Key Agreement (FCS_COP.1(DH KA))
16498	BCryptSecretAgreement	BCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375496(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for ECDH Key Agreement (FCS_COP.1(EC KA))
7725	BCryptSecretAgreement	BCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375496(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
8953	BCryptGenRandom	BCryptGenRandom: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375458(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

4469	BCryptGenRandom	BCryptGenRandom: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375458(v=vs.85).aspx	FCS: Cryptographic Support	Random Number Generation (FCS_RBG_EXT.1)
7716	BCryptGenRandom	BCryptGenRandom: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375458(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16497	BCryptHashData	BCryptHashData: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375468(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Keyed Hash Message Authentication (FCS_COP.1(HMAC))
4470	BCryptHashData	BCryptHashData: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375468(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Hashing (FCS_COP.1(HASH))
7717	BCryptHashData	BCryptHashData: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375468(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
4461	BCryptFinishHash	BCryptFinishHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375443(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Hashing (FCS_COP.1(HASH))
7754	BCryptFinishHash	BCryptFinishHash: http://msdn.microsoft.com/en-	FCS: Cryptographi	Cryptographic Services

		us/library/windows/desktop/aa375443(v=vs.85).aspx	c Support	(FCS_SRV_EXT.1)
16489	BCryptSignHash	BCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375510(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16490	BCryptSignHash	BCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375510(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4474	BCryptSignHash	BCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375510(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Digital Signature (FCS_COP.1(SIGN))
7723	BCryptSignHash	BCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375510(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16483	BCryptVerifySignature	BCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375515(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16491	BCryptVerifySignature	BCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375515(v=vs.	FAU: Security Audit	User Identity Association (FAU_GEN.2)

		85).aspx		
4475	BCryptVerifySignature	BCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375515(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Digital Signature (FCS_COP.1(SIGN))
7720	BCryptVerifySignature	BCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa375515(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
8951	NCryptDeleteKey	NCryptDeleteKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376251(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16493	NCryptDeleteKey	NCryptDeleteKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376251(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4702	NCryptDeleteKey	NCryptDeleteKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376251(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Zeroization (FCS_CKM_EXT.4)
7762	NCryptFinalizeKey	NCryptFinalizeKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376265(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

16494	NCryptFinalizeKey	NCryptFinalizeKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376265(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4704	NCryptFinalizeKey	NCryptFinalizeKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376265(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Generation for Asymmetric Keys (FCS_CKM.1(ASYM))
16548	NCryptFinalizeKey	NCryptFinalizeKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376265(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Generation for Asymmetric Keys (FCS_CKM.1(AUTH))
7760	NCryptFinalizeKey	NCryptFinalizeKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376265(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
7763	NCryptSecretAgreement	NCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376289(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16499	NCryptSecretAgreement	NCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376289(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for DH Key Agreement (FCS_COP.1(DH KA))
4709	NCryptSecretAgreement	NCryptSecretAgreement: http://msdn.microsoft.com/en-	FCS: Cryptographi	Cryptographic Operation for ECDH Key Agreement

		us/library/windows/desktop/aa376289(v=vs.85).aspx	c Support	(FCS_COP.1(EC KA))
7750	NCryptSecretAgreement	NCryptSecretAgreement: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376289(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
7764	NCryptSignHash	NCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376295(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16495	NCryptSignHash	NCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376295(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
7680	NCryptSignHash	NCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376295(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Digital Signature (FCS_COP.1(SIGN))
7681	NCryptSignHash	NCryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376295(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
7765	NCryptVerifySignature	NCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376298(v=vs.	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		85).aspx		
16496	NCryptVerifySignature	NCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376298(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
4708	NCryptVerifySignature	NCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376298(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Digital Signature (FCS_COP.1(SIGN))
7695	NCryptVerifySignature	NCryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa376298(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16504	CryptDecrypt	CryptDecrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379913(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))
16505	CryptDecrypt	CryptDecrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379913(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16501	CryptEncrypt	CryptEncrypt: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379924(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Data Encryption/Decryption (FCS_COP.1(AES))
16502	CryptEncrypt	CryptEncrypt: http://msdn.microsoft.com/en-	FCS:	Cryptographic Services

		us/library/windows/desktop/aa379924(v=vs.85).aspx	Cryptographic Support	(FCS_SRV_EXT.1)
16537	CryptDestroyKey	CryptDestroyKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379918(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16538	CryptDestroyKey	CryptDestroyKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379918(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16507	CryptDestroyKey	CryptDestroyKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379918(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Zeroization (FCS_CKM_EXT.4)
16539	CryptGenKey	CryptGenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379941(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16540	CryptGenKey	CryptGenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379941(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16515	CryptGenKey	CryptGenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379941(v=vs.	FCS: Cryptographic Support	Cryptographic Key Generation for Asymmetric Keys (FCS_CKM.1(ASYM))

		85).aspx		
16549	CryptGenKey	CryptGenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379941(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Generation for Asymmetric Keys (FCS_CKM.1(AUTH))
16514	CryptGenKey	CryptGenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379941(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16513	CryptGenKey	CryptGenKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379941(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Key Generation for Symmetric Keys (FCS_CKM.1(SYM))
16541	CryptImportKey	CryptImportKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380207(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16534	CryptImportKey	CryptImportKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380207(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for DH Key Agreement (FCS_COP.1(DH KA))
16535	CryptImportKey	CryptImportKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380207(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for ECDH Key Agreement (FCS_COP.1(EC KA))

16536	CryptImportKey	CryptImportKey: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380207(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16542	CryptGenRandom	CryptGenRandom: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379942(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16517	CryptGenRandom	CryptGenRandom: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379942(v=vs.85).aspx	FCS: Cryptographic Support	Random Number Generation (FCS_RBG_EXT.1)
16518	CryptGenRandom	CryptGenRandom: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379942(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16520	CryptHashData	CryptHashData: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380202(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Hashing (FCS_COP.1(HASH))
16521	CryptHashData	CryptHashData: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380202(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16523	CryptGetHashParam	CryptGetHashParam: http://msdn.microsoft.com/en-	FCS: Cryptographi	Cryptographic Operation for

		us/library/windows/desktop/aa379947(v=vs.85).aspx	c Support	Hashing (FCS_COP.1(HASH))
16524	CryptGetHashParam	CryptGetHashParam: http://msdn.microsoft.com/en-us/library/windows/desktop/aa379947(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16534	CryptSignHash	CryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380280(v=vs.85).aspx	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))
16544	CryptSignHash	CryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380280(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16527	CryptSignHash	CryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380280(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Digital Signature (FCS_COP.1(SIGN))
16528	CryptSignHash	CryptSignHash: http://msdn.microsoft.com/en-us/library/windows/desktop/aa380280(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)
16545	CryptVerifySignature	CryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa381097(v=vs.	FAU: Security Audit	Audit Data Generation (FAU_GEN.1(OSPP))

		85).aspx		
16546	CryptVerifySignature	CryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa381097(v=vs.85).aspx	FAU: Security Audit	User Identity Association (FAU_GEN.2)
16531	CryptVerifySignature	CryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa381097(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Operation for Digital Signature (FCS_COP.1(SIGN))
16532	CryptVerifySignature	CryptVerifySignature: http://msdn.microsoft.com/en-us/library/windows/desktop/aa381097(v=vs.85).aspx	FCS: Cryptographic Support	Cryptographic Services (FCS_SRV_EXT.1)

The NCrypt interfaces are higher level wrappers of the BCrypt interfaces. Therefore the testing of the NCrypt interfaces is accomplished by testing the BCrypt interfaces. Below is a table which maps the NCrypt interface to the BCrypt interface that it wraps.

NCrypt Interface	**BCrypt Interface**
NCryptDeleteKey	BCryptDestroyKey
NCryptFinalizeKey	BCryptFinalizeKeyPair
NCryptSecretAgreement	BCryptSecretAgreement
NCryptSignHash	BCryptSignHash
NCryptVerifySignature	BCryptVerifySignature