National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme Validation Report

ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070

Report Number: **CCEVS-VR-VID10342-2011**
Dated: **October 11, 2011**
Version: **1.0**

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6940
Fort George G. Meade, MD 20755-6940

ACKNOWLEDGEMENTS

Validation Team

Mr. Paul A. Bicknell

The MITRE Corporation

Bedford, MA

Ms. Vicky Ashby

The MITRE Corporation

McLean, Virginia

Common Criteria Testing Laboratory

Mr. Herb Markle

CygnaCom Solutions

McLean, Virginia

Much of the material in this report was extracted from evaluation material prepared by the CCTL. The CCTL team deserves credit for their hard work in developing that material. Many of the product descriptions in this report were extracted from the ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070 Security Target.

Table of Contents

List of Figures and Tables

1. Executive Summary

This Validation Report (VR) documents the evaluation and validation of the product ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070.

This VR is not an endorsement of the IT product by any agency of the U.S. Government and no warranty of the IT product is either expressed or implied.

The Target of Evaluation (TOE) is a Network Access Control System that consists of the following components: the CounterACT Appliance, the CounterACT Enterprise Manager, SecureConnector and the CounterACT Console used for managing the product.

CounterACT combines clientless Network Access Control (NAC) and threat protection to ensure all devices connecting to the network are in compliance with network security and access policies and are free of self-propagating malware. CounterACT integrates into a network environment and enables enterprises to tailor enforcement actions to match the level of policy violations, while avoiding disruptions during device interrogation.

The evaluation was performed by the CygnaCom Common Criteria Testing Laboratory (CCTL), and was completed in September 2011. The information in this report is derived from the Evaluation Technical Report (ETR) and associated test reports, all written by the CygnaCom CCTL. The evaluation team determined that the product is Common Criteria version 3.1 R3 [CC] Part 2 extended and Part 3 conformant, and meets the assurance requirements of EAL 4 augmented with ALC_FLR.2 from the Common Methodology for Information Technology Security Evaluation, Version 3.1 R3, [CEM]. This Security Target claims no Protection Profile conformance.

The evaluation and validation were consistent with National Information Assurance Partnership (NIAP) Common Criteria Evaluation and Validation Scheme (CCEVS) policies and practices as described on their web site www.niap-ccevs.org. The Security Target (ST) is contained within the document ForeScout CounterACT v6.3.3 Security Target.

2. Identification

Target of Evaluation: ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070

Evaluated Software and Hardware:

ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070 product consisting of the following components:

- **CounterACT Appliance**:
    - All appliance hardware (Models: CT-Remote, CT-100, CT-1000, CT-2000, and CT-4000),
    - All ForeScout software installed on the appliance including proprietary protocols and the following Hotfix and Plugins:
        - Hotfix (version 6.11070)
        - Host Property Scanner (version 9.11050)
        - HPS-Vulnerability DB (1.11060; may be updated by the user)
        - NBT Scanner (version 3.0)
        - User Directory (version 4.9110)
        - Switch (version 7.10021)
        - Macintosh/Linux (version 6.11040)
        - DNS Client (version 1.8040)
        - Reports (version 3.11020)
        - Syslog (version 2.9060)
    - All 3rd party software installed on the appliance
- **CounterACT Enterprise Manager**:
    - All appliance hardware (Models: CEM-5/A, CEM-10/A, CEM-25/A, CEM-50/A and CEM-100A)
    - All ForeScout software installed on the appliance including proprietary protocols and the following Hotfix and Plugins:
        - Hotfix (version 6.11070)
        - Host Property Scanner (version 9.11050)
        - HPS-Vulnerability DB (1.11060; may be updated by the user)
        - NBT Scanner (version 3.0)

        - User Directory (version 4.9110)
        - Switch (version 7.10021)
        - Macintosh/Linux (version 6.11040)
        - DNS Client (version 1.8040)
        - Reports (version 3.11020)
        - Syslog (version 2.9060)
    - All 3rd party software installed on the appliance including:
- **CounterACT Console**: software only component
- **SecureConnector (version 3.325)**: software only component

Developer:	ForeScout Technologies, Inc.
CCTL:	CygnaCom Solutions 7925 Jones Branch Dr, Suite 5400 McLean, VA 22102-3321
Evaluators:	Herb Markle
Validation Scheme:	National Information Assurance Partnership CCEVS
Validators:	Paul A. Bicknell, Vicky Ashby
CC Identification:	Common Criteria for Information Technology Security Evaluation, Version 3.1 R3, July 2009
CEM Identification:	Common Methodology for Information Technology Security Evaluation, Version 3.1 R3, July 2009

3. Security Policy

The TOE enforces the following security policies as described in the ST:

3.1. Security Audit Functions

The TOE's auditing capabilities include the generation of information about system processing, use of the administrative functions and attempted access to the protected network. The TOE provides authorized personnel access to the audit data and the ability to interpret and sort the data. The TOE protects the audit data from modification and unauthorized deletion.

Security Audit relies on the Operational Environment to provide reliable timestamps for the audit records. This functionality may optionally rely on an external syslog server in the Operational Environment to archive audit records. It also relies on the Environment to provide a secure channel between the TOE and the external time-server and the optional syslog server.

3.2. Network Access Control Functions

The TOE provides its own Network Access Control separate from that of the Operational Environment between subjects and objects covered by the TOE's access control policies. The TOE supports three types of Network Access Control policies: NAC, Virtual Firewall, and Threat Protection. All three types of policies may be used simultaneously for network protection. The TOE provides administrative functions for authorized administrators to define these policies.

Network Access Control depends on the Operational Environment to provide secure communications between the TOE and the network endpoints. User data protection may rely on an external e-mail server in the Operational Environment if e-mail notifications are configured in a policy. It also depends on the Environment to provide a secure channel between the TOE and the e-mail server if it is present.

3.3. User Identification and Authentication Functions

Each TOE user must be successfully identified and authenticated by the TSF or an external authentication service invoked by the TSF before access is allowed to the TOE. The TSF maintains security attributes for each individual TOE user for the duration of the user's login session. The TOE also supports a password policy, authentication failure handling and masks the user's authentication data upon input.

User Identification and Authentication may rely on the Operational Environment to provide an optional external authentication service if that method of authentication of TOE users is configured for the system. It also depends on the Environment to provide a secure channel between the TOE and the authentication server if it is present.

3.4. Security Management Functions

The TOE provides role-based security management functions through the use of the administrative GUI. The ability to manage various security attributes, system parameters and all TSF data is controlled and limited to those users who have been assigned the appropriate administrative role and permissions.

Security Management relies on a management console in the Operational Environment to host the CounterACT console application. Security management also depends on the Operational Environment to provide secure communications between the TOE and the DNS Server, Network Switch(es), optional User Directory Server, optional E-mail Server and between the TOE and network endpoints.

3.5. Protection of Security Functions

The TOE protects data being transferred between the distributed TOE components from disclosure and modification by the implementation of secure internal interfaces.

3.6. Vulnerability Scanning Functions

The TOE further protects the targeted network through the ability to conduct vulnerability scans. The TOE has the ability to collect configuration and posture data from endpoints attempting network access, analyze the collected data and perform administrator configured remediation actions if a potential vulnerability is detected.

Vulnerability Scanning depends on the Operational Environment for secure communications between the TOE and the network endpoints. Vulnerability scanning may rely on an external e-mail server in the Operational Environment if e-mail notifications are configured to be sent when a vulnerability is detected. It also depends on the Environment to provide a secure channel between the TOE and the e-mail server if it is present.

3.7. Assumptions

The ST identifies the following assumptions about the use of the product:

1. The TOE assumes there will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.
2. The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.
3. Those responsible for the TOE will ensure the communications between the TOE components and external IT Entities are via secure channels.
4. The TOE assumes that its users will protect their authentication data.

3.8. Clarification of Scope

All evaluations (and all products) have limitations, as well as potential misconceptions that need clarifying. This text covers some of the more important limitations and clarifications of this evaluation. Note that:

1. As with any evaluation, this evaluation only shows that the evaluated configuration meets the security claims made, with a certain level of assurance (EAL 4 in this case).
2. This evaluation only covers the specific version of the product identified in this document, and not any earlier or later versions released or in process.
3. As with all EAL 4 evaluations, this evaluation did not specifically search for, nor seriously attempt to counter, vulnerabilities that were not "obvious" or vulnerabilities to objectives not claimed in the ST. The CEM defines an "obvious" vulnerability as one that is easily exploited with a minimum of understanding of the TOE, technical sophistication and resources.
4. Cryptographic protection is provided by the TOE; however, the cryptography used in this product was not analyzed or tested to conform to cryptographic standards during this evaluation.
5. The following product components and functionality will not be included in the TOE or the evaluation:
    a. The CounterACT Assets Portal Product Component and its Functionality
    b. Command Line Tools (CLI Functionality) (not used during run-time operation of the TOE)
    c. Plugins not bundled with CounterACT Appliance
    d. Updates to CounterACT Appliance Plugins, except for the HPS-Vulnerability DB Plugin
    e. High Availability Option (requires separate license)
    f. Payment Card Industry (PCI) Kit (requires PCI Plugin)
    g. Cryptographic Functionality of the SSL interfaces between TOE components
    h. TOE reception of syslog messages from the external Syslog Server (requires installation of NTsyslog on Domain Controller)
    i. Remote Management Module 2 (RMM2) integration
6. The Operational Environment needs to provide the following capabilities:
    a. Host Platform for CounterACT Console application
    b. Network Authentication Services
    c. Network Switches
    d. Optional External Servers/Controllers
        - Domain Controller

- DHCP Server
- NTP Server
- E-mail Server
- Syslog Server
- User Directory Servers:
    - Microsoft Active Directory
    - Sun Java System Directory Server
    - Novell eDirectory
    - IBM Lotus Notes
    - Radius
    - TACACS

4. Architectural Information

ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070 (CounterACT) combines Network Access Control (NAC) and threat protection to ensure all connecting devices are in compliance with network security policies and are free of self-propagating malware (worms). CounterACT integrates into a network environment and enables enterprises to tailor enforcement actions to achieve a level of policy enforcement through network appliances managed via a single control point that interrogates and controls access to the network devices.

The ForeScout CounterACT TOE is comprised of the following components:

- CounterACT Appliance (Appliance)
- CounterACT Enterprise Manager (Enterprise Manager)
- CounterACT Console (Console)
- SecureConnector

The CounterACT Appliance performs compliance testing and enforcements, and provides protection against self-propagating threats. It automatically identifies and manages suspicious network activity, handles vulnerabilities and Network Access Control (NAC) compliance issues, and lets administrators create network security zones via a virtual firewall. The CounterACT appliance also stores and manages information about network threats and activity, as well as the action taken at hosts in the network. Multiple CounterACT Appliances can be deployed to ensure maximum protection of an organization.

NAC Policies, Virtual Firewall Policies, and Threat Protection Policies are all methods of Network Access Control. All three types of policies may be in force at the same time at one customer installation. Of the three types of policies, NAC Policies are the most flexible and significant to the user. Vulnerability Scanning can be integrated within the NAC Policies defined at a site.

Plugins are additional software modules that can be integrated into the CounterACT Appliance to expand the scope of endpoint inspections and enforcement capabilities. Information gleaned from Plugins is incorporated into CounterACT NAC tools used for creating policies; in the Information Panel and events table as well as in existing reports or in newly designed reports designed to support the Plugin. Tools are available to install/uninstall, configure, test as well as start and stop Plugins at any time.

When multiple CounterACT Appliances are present (up to 100 Appliances), these devices can be managed as one through a central CounterACT Enterprise Manager. The Enterprise Manager is an aggregation device that communicates with multiple CounterACT Appliances distributed across an enterprise. It manages the CounterACT Appliance activity and policies and collects information about malicious activity that was detected by each Appliance, including infection attempts, identification, and suppression actions taken. Administrators use the Enterprise Manager to define and distribute network policies throughout the LAN to all CounterACT Appliances. The Enterprise Manager collects security event data for reporting, and shares relevant security information gathered from individual Appliances with the rest of the CounterACT Appliances on the

network. The connection between multiple CounterACT Appliances and the Enterprise Manager is authenticated and encrypted using SSL on port 13000 using TCP. The Enterprise Manager also contains the Hotfix and set of Plugins that is bundled with the product as described in the previous section.

The CounterACT Console is the CounterACT management application GUI used for configuring, viewing and managing important information about Network Access Control policies, malicious activities, vulnerable network hosts, and more. The Console lets administrators define the conditions under which hosts are identified and handled by CounterACT. Access to the Enterprise Manager or an Appliance via the Console is authenticated by verifying an Enterprise Manager or Appliance IP address, user ID and password or by authenticating the user via an external User Directory server.

SecureConnector is a lightweight, small-footprint executable that can optionally be run at the endpoint so that CounterACT can monitor and control otherwise unmanageable hosts on the network.

SecureConnector creates a tunnel from the host to the Appliance. The tunnel created is used to remotely inspect the host, as if it was a domain member. The port closes when network users reboot or disconnect from the network, and reopens at reconnection. During operation, the host does not listen to incoming connections as it establishes the encrypted SSL connection with the Appliance. SecureConnector can be configured to dissolve at reboot or disconnection from the network, leaving no footprints. Alternatively, it can be configured to install normally so that it remains upon reboot or disconnection; in this case it can be removed via the uninstall option in the Console GUI.

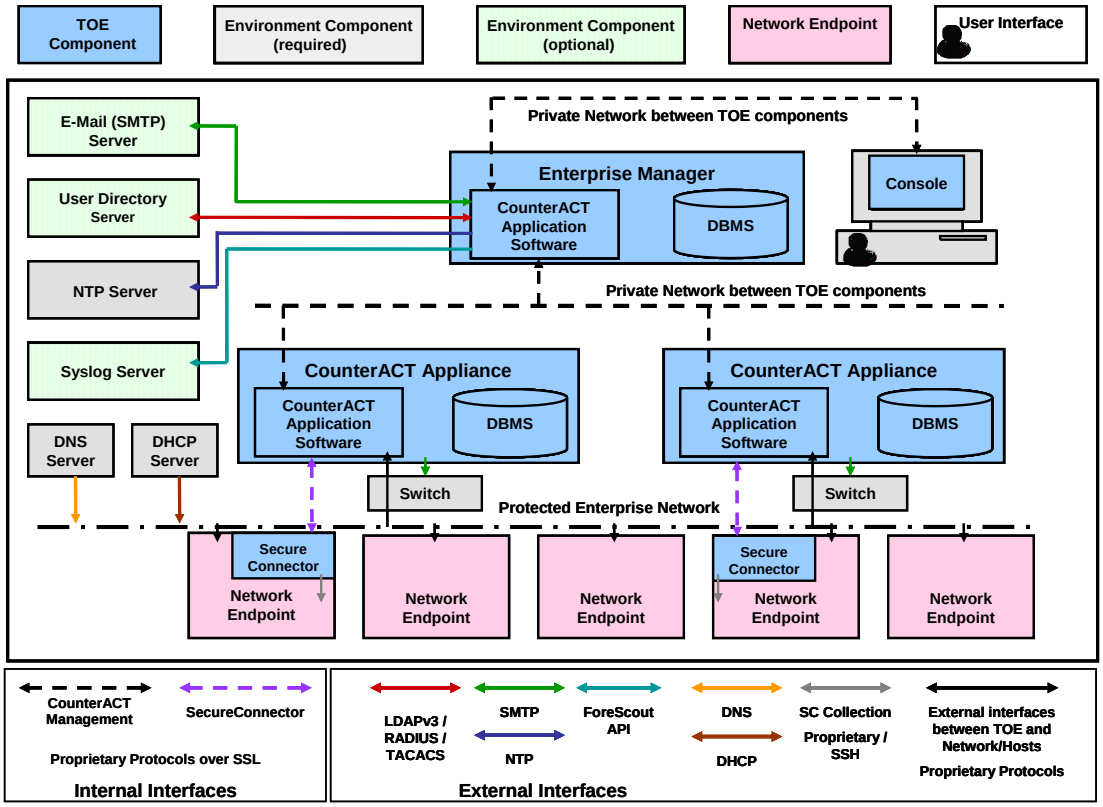

Figure 1: TOE Boundary

5. Documentation

The TOE is physically delivered to the End-User. The guidance is part of the TOE and is delivered in printed form and as PDFs on the installation media.

5.1. Guidance Documentation

The following documents are developed and maintained by ForeScout and delivered to the end user of the TOE:

[1] ***CounterACT Installation Guide***, Version 6.3.3, May 31, 2009

[2] ***CounterACT Release Notes***, Version 6.3.3, July 2009

[3] ***CounterACT Console User Manual***, Version 6.3.3, June 2009

[4] ***CounterACT 6.3.3 Hotfix 6.11070 Release notes****;* July 2011

[5] ***ForeScout CounterACT v6.3.3 Common Criteria Supplement to the Administrative Guidance ,*** Version 1.0, Sept. 7, 2011

5.2. Security Target (ST)

Security Target (ST)

[1] *ForeScout CounterACT v6.3.3 Security Target*, Version 2.0, Sept. 7, 2011

6. IT Product Testing

At EAL 4, the overall purpose of the testing activity is “independently testing a subset of the TSF, whether the TOE behaves as specified in the design documentation, and to gain confidence in the developer's test results by performing a sample of the developer's tests”.

At EAL 4, the developer’s test evidence must “show the correspondence between the tests provided as evaluation evidence and the functional specification. This section describes the testing efforts of the Vendor and the evaluation team.

The objective of the Evaluator’s independent testing sub-activity is “to demonstrate that the security functions perform as specified. Evaluator testing includes selecting and repeating a sample of the developer tests”.

6.1. Developer Testing

The developer testing effort involved executing all the TOE’s described functions.

6.1.1. Overall Test Approach

All of the Developer test cases are manual, i.e. all test steps including setup and cleanup steps were performed by a user entering commands a terminal running the Administrative GUI and visually verifying the results. All developer test cases test TOE security functions by stimulating an external interface.

Although the developer tests are performed using the Administrative GUI, the Evaluator determined that the test cases as described in the test documentation adequately exercise the internal interfaces.

The Developer executed all of their test procedures and provided a generated report of the actual results. The Developer's actual results were consistent with their expected results for the test procedures provided. All actual results were visually verified with no additional evidence being provided.

6.1.2. Test Results

The Developer's tests covered all of the security relevant behavior of the TOE:

- 100% of the TOE SFRs claimed in the Security Target.
- 100% of the External TSF Interfaces.
- 100% of each subsystem’s described security features and behaviour

The Developer ran the test suite twice in July, 2011. Later, a third run was performed as a Hotfix was found to be needed during the vulnerability analysis.

6.2. Evaluator Independent Testing

The testing was performed at Evaluator’s Home Office in Canastota, NY.

The Evaluator performed the following activities during independent testing:

- Execution the Developer's Functional Tests
- Team-Defined Functional Testing
- Vulnerability/Penetration Testing

6.2.1. Execution the Developer's Functional Tests

The sampling of the Developer's Functional test cases was executed after the TOE was installed in the evaluated configuration consistent with the Security Target.

The Evaluator chose Developer Functional tests to provide:

- Complete coverage of all SFRs
- Complete coverage of all TSFIs
- Complete coverage of all Subsystems and Internal Interfaces
- Represented 95% of the complete Developer test cases.

The test configurations used by the Evaluator were the same as that used by the developer.

The test results and screenshots for the test cases were recorded during the Evaluator testing. Overall success of the testing was measured by 100% of the retests being consistent with expected results. Anomalies were documented along with suggested / required solutions.

All of the Developer's Functional Tests rerun by the Evaluator received a 'Pass' verdict.

6.2.2. Evaluator-Defined Functional Testing

The Evaluator-Defined Functional tests were devised to augment the Developer Functional tests in order to exercise functionality in greater depth than the Developer tests provided. In particular, these tests were developed to exercise the primary security functionality of the TOE, NAC enforcement. The Developer's tests focused on testing the functionality of the machine under pre-determined configuration (i.e. known assets that should show up when scanned). The Evaluator explored the TOE in a more realistic operational environment with assets/network segments being added and unknown assets trying to obtain access. Additional laptops/desktops that form a new network and individual assets were used. The adding of the machines and network were part of the testing (rather than preconfigured test setups).

The Evaluator categorized team-defined testing into three sections:

- NAC testing: The Evaluator explored the NAC policies and actions by using different inputs than what the Developer's test pre-determined. These tests included: IM Testing, Peer to Peer testing, Personal Firewall Testing and Windows Vulnerability testing
- Add Host/Network: The Developer's test configuration was setup following the Developer's instructions using a server that has several virtual hosts for the

managed hosts. The Evaluator wanted to mimic a more operational scenario where hosts are going to be added/removed onto a network to determine/ensure that appropriate actions occur.

- Add illegitimate host: This test specifically dealt with NAC policies that pertain to found hosts that are not legitimate (and have no intentions to be)

All of the Team-Defined Tests executed as expected and received a 'Pass' verdict.

6.2.3. Vulnerability/Penetration Testing

The Penetration tests for TOE were developed according to the following strategy:

- The Evaluator will perform a systematic vulnerability analysis of the TOE.
- The Evaluator will note possible security vulnerabilities by examining the Vulnerability Analysis, Functional Specification, TOE Design Document and TOE Security Target.
- The Evaluator will analyze the different components that comprise the TOE for existing vulnerabilities.
- The Evaluator will search public vulnerability databases for vulnerabilities that corresponded to these components.
- The Evaluator will identify hypothesized vulnerabilities requiring low attack potential that apply to the TOE.
- The Penetration tests will cover hypothesized vulnerabilities and potential misuse of guidance.
- The tests for potential misuse of guidance will cover installing the TOE from the guidance documentation and sampling the documented administrator procedures.

The Evaluator examined the external interfaces for means to bypass security. Scenarios for penetration testing were developed during vulnerability analysis of the product and after the Evaluator gained familiarity with the operation of the TOE.

Password Policy: Ensure minimum standards as documented are sufficient and that the policy cannot be confused by entering bad combinations. The passwords entry should also be tested against large input (more than 256 characters). Tests:

Test of Password Policy Entry

In this test, the entries for the password policy will tested for minimum and maximum values.

Attempt at confusing Password Policy

In this test, the entries for the password policy will tested for conflicting settings.

Large input (buffer overflow)

In this test, the entries for the password will tested for overflow type conditions.

SecureConnector modes: Determine if there are any undocumented differences in behavior that could lead to a weakness. Test:

SecureConnector Test

Install SecureConnector as a service, in permanent, and in dissolvable mode. Test the ability to limit the removal of the SecureConnector via a password.

Input Parameters: Verify limitations and determine if there is any way to input invalid parameters. Such as using network segments instead of full IP addresses or ranges above 255. Test:

Console restriction testing

In this test, the entries for the IP address restriction policy that controls console access will be tested for conflicting settings.

Access Control Lists: Determine default behavior of TOE if ACL is not correctly generated or gets corrupted. Test:

CLI Access Testing

In this test, the entries for the ACL that controls CLI access will be tested for incorrect input settings.

Scan for Vulnerabilities: Run a vulnerability scan against the TOE.

The Penetration test cases were executed after the TOE was installed in the evaluated configuration consistent with the Security Target

Additional testing/verification: Additional verification was done to ensure that the Hotfix did indeed update the third party software to the correct version.

All of the Vulnerability/Penetration Tests received a ‘Pass’ verdict.

7. Results of Evaluation

The evaluation was conducted based upon version 3.1 Revision 3 of the CC and the CEM.

The evaluation team concluded that the ForeScout CounterACT v6.3.3-309 with Hotfix v6.11070 met all “EAL4 augmented with ALC_FLR.2” evaluation criteria.

8. Validators Comments/Recommendations

The validators were satisfied with the evaluation team's evaluation and testing efforts. The validators did not identify any gaps or missing information. The CCTL was well prepared, and the material was complete and correct.

9. Security Target

ForeScout CounterACT v6.3.3 Security Target, Version 2.0, Sept. 7, 2011, is compliant with the Specification of Security Targets requirements found within Annex B of Part 1of the CC.

10. Glossary

10.1.Acronyms

The following are product specific and CC specific acronyms. Not all of these acronyms are used in this document.

ARP	Address Resolution Protocol
CLI	Command Line Interface
DBMS	Database Management System
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
GUI	Graphical User Interface
HTTP	HyperText Transmission Protocol
HTTPS	HyperText Transmission Protocol, Secure
IP	Internet Protocol
IPS	Intrusion Protection System
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
MAC	Media Access Control
MIB	Management Information Base
NAC	Network Access Control
NAT	Network Address Translation
NetBIOS	Network Basic Input/Output System.
NIC	Network Interface Controller
NTP	Network Time Protocol
OID	Object ID
P2P	Peer-to-Peer
PCI	Payment Card Industry
PDF	Portable Document Format

RADIUS	Remote Authentication Dial In User Service
SMTP	Simple Mail Transport Protocol
SNMP	Simple Network Management Protocol
SSH	Secure Shell Network Protocol
SSL	Secure Sockets Layer,
TACACS	Terminal Access Controller Access-Control System
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security,
UDP	User Datagram Protocol
USB	Universal Serial Bus
VLAN	Virtual Local Area Network
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
WAN	Wide Area Network

10.2. Terminology

This section defines the product-specific and CC-specific terms. Not all of these terms are used in this document.

Action	Measures taken at network endpoints; ranging from notices, warnings and alerts to remediation, access restrictions and complete blocking. Actions can be incorporated into NAC policies or applied manually on selected network endpoints.
ActiveResponse	A patented technology created by ForeScout Technologies that effectively mitigates human attackers, worms and other self-propagating malware. ActiveResponse technology pinpoints and halts threats at the earliest stages of the infection process.
ActiveResponse range	The range of addresses protected by ActiveResponse technology.

Admission event	Network events that indicate the admission of an endpoint into the network. For example when it physically connects to a switch port; when its IP address changes or when it sends out a DHCP request.
Appliance	A CounterACT component, consisting of dedicated hardware and software that executes inspection and policy enforcement. The Appliance monitors traffic going through the enterprise network and, as needed, generates response traffic into the network in order to provide IPS, NAC and firewall functionality.
ARP request	Address Resolution Protocol Request: A request sent by a host on an IP network in order to find the hardware (MAC) address of another host whose network address (IP address) is known. ARP requests are monitored and used by CounterACT to detect hosts in the network.
Bite Event	An event in which a malicious host tries to gain access to the protected network using CounterACT bait (part of the ActiveResponse technology). When a network device (endpoint) tries to gain access to the protected network using a system mark.
Cell	A group of endpoints (hosts) that are monitored and protected by a single Appliance.
Channel	A set of input and output interfaces used by a CounterACT Appliance. A channel consists of: • a monitor interface that examines traffic going through the network • a response interface that generates traffic back into the network • a mapping of VLAN tagging between them
Condition	In NAC policies, a pre-defined set of host properties, logical conditions and Boolean relations connecting them.
Console	The CounterACT GUI application used for creating NAC, firewall and IPS policies, generating reports, viewing and managing detection information, and managing CounterACT Appliances.
Endpoint	A Network Host discovered by CounterACT, for example desktop, laptop, server, etc.

Enterprise Manager	A CounterACT component that manages multiple Appliances distributed across the network.
Firewall policy	A CounterACT policy that lets the user create network security zones, giving more control over network traffic. The CounterACT firewall is virtual — providing (out-of-band) firewall protection, without being located inline.
Fstool	A command line toolset used at the Appliance and Enterprise Manager for extended configuration and troubleshooting.
Hijack	Actions that let CounterACT intercept and replace endpoint Web (HTTP) sessions with customized Web pages to realize a NAC function. For example, replace a Web session with a notification page indicating that the host does not comply with network policies. Endpoints can be prevented from using the network until they comply, or until they acknowledge an informatory message, etc.
Host	An endpoint; a network machine handled by CounterACT.
Host block	An IPS blocking option that prevents a host from communicating with the enterprise network for a specified time period.
Host inspection	Examination of network hosts by CounterACT. The purpose of inspection is to retrieve host properties and to verify compliance with NAC policies. Hosts that are defined within the CounterACT Internal Range are inspected.
HTTP local host login	A NAC action that lets CounterACT interrogate unmanageable guest hosts. It allows guests to provide CounterACT with credentials which in turn can be used to remotely inspect the host for compliance with the policy.
Internal network range	The range of network hosts in an organization that CounterACT is configured to inspect.
IPS policy	Same as Threat Protection Policy. A policy that allows the user to define how CounterACT should handle hosts that attempt to attack or infect the network.
Irresolvable host	A Host that could not be properly inspected, and as a result not all properties required by the NAC policy were resolved.

Legitimate e-mail servers	Mail servers/hosts from which mail traffic is expected and should be allowed. Some hosts in the network may generate excessive or suspicious mail traffic that will be detected as a mail infection. For mail servers, this traffic actually qualifies as legitimate activity.
Legitimate traffic rules	Rules for allowing specific network activity. Activity defined in these rules will be ignored by CounterACT when it detects malicious network traffic.
Malicious Host	A machine at which self-propagating malware is detected, or operated by a malicious operator (attacker).
Malware	Software designed specifically to damage or disrupt a system, such as a virus or a Trojan horse. Malware includes both viruses and spyware.
Manageable hosts	Hosts that are accessible for deep inspection by CounterACT.
Management Interface	An Appliance network interface through which the CounterACT Appliance is managed. The management interface is typically also used to perform queries, deep inspection and HTTP hijacking based on CounterACT policies. The interface needs be connected to a switch port and/or VLAN that has access to all network endpoints that it needs to interact with.
Manual action	NAC actions applied manually to endpoints from the Console
Manually added host	Hosts that users manually introduce into CounterACT for IPS related activities — for example adding an endpoint IP that should be ignored by CounterACT.
Mark	Virtual resource information generated by the TOE that is sent to suspected malware programs that are probing the network for information.
Mark naming rules	Instructions that CounterACT uses to create customized marks as part of the ActiveResponse technology. These rules should reflect the naming conventions used for host and user names in your network — for example host names that always begin with a fixed text string.
Monitor interface	The Appliance interface used to monitor network traffic. Typically, network traffic would be mirrored to a port on a switch, to which the monitoring interface would in turn be connected.

NAC policy	A set of rules instructing CounterACT how to detect and handle network endpoints for the purpose of maintaining Network Access Control, compliance and security.
Plugins	Functionality enhancement modules that can be incorporated into CounterACT. Plugins enable deeper inspection as well as broader control over network endpoints. Bundled plugins are pre-packaged with CounterACT. Other plugins may be available from ForeScout or from a third party.
Response interface	An Appliance interface through which CounterACT sends generated traffic into the network. Response traffic is used to: • Protect against self propagating malware, worms and hackers. • Carry out firewall blocking. • Perform NAC Policy actions — for example hijacking Web browsers.
SecureConnector	A lightweight, small-footprint executable that runs at the endpoint so that CounterACT can inspect it. SecureConnector opens an encrypted tunnel to CounterACT allowing it to remotely inspect it, similar to how domain member host would be inspected. SecureConnector can be used when CounterACT cannot otherwise manage the endpoint (unmanageable). SecureConnector can be deployed via a NAC action or using other methods.
Segment	An option that lets the user organize and display the enterprise network into logical groups, which can then be used in NAC policy, reports etc.
Unmanageable host	A host that CounterACT cannot inspect. In general, Windows hosts are unmanageable if they cannot be accessed by CounterACT via ports 139 or 445 or do not allow remote inspection (e.g. registry, file system). This is typical, for example, when endpoints are guests or in cases where domain credentials are not available.
Virtual firewall policy	A CounterACT policy used to create traffic rules for both protecting and making available network services, resources and segments.
Worm	A self-replicating computer program that uses a network to send copies of itself to other nodes (hosts on the network) and it may do so without any user intervention.

11. Bibliography

URLs

[1] Common Criteria Evaluation and Validation Scheme (CCEVS): (http://www.niap-ccevs.org/cc-scheme).

[2] CygnaCom Solutions CCTL (http://www.cygnacom.com).

CCEVS Documents

[1] Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and general model, July 2009 Version 3.1 Revision 3 Final, CCMB-2009-07-001.

[2] Common Criteria for Information Technology Security Evaluation - Part 2: Security functional components, July 2009 Version 3.1 Revision 3 Final, CCMB-2009-07-002.

[3] Common Criteria for Information Technology Security Evaluation - Part 3: Security assurance components, July 2009, Version 3.1 Revision 3 Final, CCMB-2009-07-003.

[4] Common Methodology for Information Technology Security Evaluation - Evaluation methodology, July 2009, Version 3.1 Revision 3 Final, CCMB-2009-07-004.