S21sec labs

La seguridad digital del futuro, hoy

Referencia:

Titulo: Declaración de Seguridad de Bitacora

Fecha: 03/11/2008

Cliente: -

Contacto:

Titulo del documento	
Tipo de documento:	Common Criteria
Nombre del fichero:	Declaracion de Seguridad de Bitacora.pdf
Clave cliente:	-
Código documento:	-
Versión documento:	3.0
Versión producto.	4.0.2
Estado:	
Fecha:	03/11/2008
Autor:	S21sec

Revisión, aprobación		
Revisado por:		Fecha:
Aprobado por:		Fecha:

Histórico de cambios			
Versión	Fecha	Descripción de la acción	Páginas
1.0	22/05/2008	Creación	32
2.0	19/09/2008	Revisión	39
3.0	03/11/2008	Revisión	39
	dd/mm/aaaa		

Table of Contents

Table of contents

List of tables

1 Introduction

1.1 ST reference

1 **Title:** Declaración de Seguridad Bitacora

2 **Version:** 3.0

3 **Author:** S21sec

4 **Date:** 3 de Noviembre de 2008

5

1.2 TOE reference

Bitacora 4.0.2

1.3 TOE overview

1.3.1 TOE usage

6 Bitacora es una solución de centralización, gestión y correlación de logs que permite a nuestra organización recolectar y consolidar los logs provenientes de los diferentes sistemas operativos, los dispositivos de red, las bases de datos y las aplicaciones.

7 Bitacora posibilita el cumplimiento legal y de auditoria para la retención y análisis de logs, permitiéndonos así alcanzar e incluso superar los requerimientos regulativos al poder archivar históricamente toda la información recogida en un formato consolidado y consultable.

8 Bitacora nos facilita una interfaz singular de consulta llamado Consola de consultas que permite recuperar datos concretos de ficheros de una semana, día o incluso de una hora concreta, filtrándolos, por ejemplo, según un identificador de usuario, una máquina, una fecha, etc. El acceso a Bitacora se realiza mediante un navegador web, se recomienda Internet Explorer (6.X o 7.X) o Firefox 2.X, a la URL: https://IP_Bitacora.

9 En una red corporativa se genera un volumen inmenso de datos en los ficheros de logs. Estos logs pueden corresponder a la actividad en un sistema operativo, en una aplicación, en una red, etc., y se almacenan en el host o servidor donde se produjeron, de forma dispersa y desvinculada.

10 El proceso del planificador organiza las tareas. El planificador se encarga de asignar tareas a los recolectores de eventos y proporcionarles el código necesario para ejecutarlas.

11 Los eventos recibidos en diferido llegan desordenados, pero están sujetos al potente mecanismo de ordenación de Bitacora que los organiza por día..

12 Los eventos ordenados se almacenan dentro del sistema de ficheros en una organización de ficheros y directorios en una estructura de datos en disco. Dentro de un día se guardan todos los eventos que se han producido en dicho día, totalmente ordenados. En un día, solamente existen eventos desde las 00:00:00 hasta las 23:59:59.

13 A partir de la ordenación y almacenamiento, los eventos son accesibles para usuarios autorizados mediante una aplicación. Utilizando la interfaz amigable de la Consola de consultas, los usuarios pueden formular consultas SQL (SELECTs) utilizando además criterios de consulta complejos para recuperar los datos almacenados.

14 Sin embargo, el almacén de eventos tiene una estructura de directorios y ficheros y no una estructura de una base de datos relacional. Además, no existe integridad referencial entre los datos. Por tanto, es el Motor de Consultas (Aleph Engine) quien se encarga de transformar las entradas en los ficheros de logs a tablas posibilitando su consulta mediante SQL, así simulando la estructura de una base de datos.

15 La única forma de acceso a Bitacora es a través del interfaz web que expone Bitacora. La máquina servidora en la que está instalado Bitacora deberá estar correctamente aislada y securizada para que nadie pueda acceder a través de otros interfaces a la misma. Una vez instalado Bitacora, el único acceso físico permitido a la máquina servidora de Bitacora será, a los usuarios autorizados, para reiniciar Bitacora a través del botón de arranque y/o parada de la máquina física.

16 Los administradores y usuarios autorizados a acceder a Bitacora se consideran confiables y competentes en el uso de la aplicación.

1.3.2 TOE type

17 Bitacora es una solución software de centralización, gestión y correlación de logs que permite a nuestra organización recolectar y consolidar los logs provenientes de los diferentes sistemas operativos, los dispositivos de red, las bases de datos y las aplicaciones.

18 Bitacora posibilita el cumplimiento legal y de auditoría para la retención y análisis de logs, permitiéndonos así alcanzar e incluso superar los requerimientos regulativos al poder archivar históricamente toda la información recogida en un formato consolidado y consultable.

19 Bitacora tiene una arquitectura modular y funcionalidad altamente configurable, lo que la hace apropiada para resolver las necesidades de un amplio abanico de usuarios con necesidades de gestión de la seguridad de redes y sistemas complejos. Está basada en desarrollos propios pero requiere de componentes de terceros, tales como un sistema operativo, una base de datos, un servidor de aplicaciones o un servidor web, todos ellos de libre distribución.

20 Bitacora requiere de un ordenador de propósito general, conectado a la red de donde se recibe los logs, y del siguiente software de terceros que no se considera parte del TOE:

- UBUNTU Server 6.06 LTS, http://www.ubuntu.com
- java version "1.5.0_13", http://www.sun.com
- Paquete SSL (openssl y libssl-dev)
- Compilador gcc
- Paquete make
- Paquete 'expect'
- Navegador web: Internet explorer 6.X, 7.X o Firefox 2.X

Se considera parte del TOE el siguiente software de terceros:

- Tomcat 5.5.26, http://tomcat.apache.org
- Apache 2.2.8, http://httpd.apache.org
- PostgreSQL 8.2.3, http://www.postgresql.org
- Liferay 4.3.3, http://www.liferay.com/

21 No hay requisitos al hardware distintos de los establecidos por el sistema operativo, UBUNTU Server 6.06 LTS, para su funcionamiento.

1.4 TOE description

22 Los componentes del sistema de gestión de logs Bitacora se describen a continuación:

1.4.1 Recolector de logs (SFTP)

23 Los logs son recopilados vía SFTP. El recolector consiste en un cliente de SFTP que se conecta a un servidor para recoger los logs provenientes de la fuente. Las recogida de esta información se realiza de manera cifrada vía protocolo SSH-2 y autenticada mediante logado (requiere de usuario y contraseña para recoger dicha información) y certificados (se produce autenticación mutua de servidor - cliente).

1.4.2 Motor de Consultas (ALEPH ENGINE)

24 Los eventos de logs que se producen en las distintas fuentes (servidores de aplicaciones, servidores web, sistemas operativos, etc.) son recogidos por el Recolector de Bitacora para luego ser almacenados en un sistema de

ficheros. Como se ha comentado en el párrafo anterior, los usuarios del sistema utilizan la Consola de consultas para acceder y analizar estos datos de forma gráfica, a través de consultas SQL, un lenguaje de recuperación de datos estándar de bases de datos relacionales.

25 Pero Bitacora no es base de datos relacional, y por tanto, para poder recuperar los datos mediante SQL hace falta una transformación de las entradas de los logs (almacenados en la estructura directorio-fichero) en una estructura consultable como es una tabla de una base de datos relacional.

26 Este proceso se lleva a cabo por medio del Motor de consultas (Aleph), la parte servidor del programa, mediante una transformación XML. Esta transformación consiste en un mapeo entre directorios y tablas relacionales mediante ficheros XML ya configurados con la definición de tablas, la definición de ficheros y la definición del formato de las fechas.

27 Para poder consultar las tablas definidas es necesario disponer de un perfil que tenga asociados los permisos adecuados para poder consultar dichas tablas. Todo aquel usuario que no disponga de los permisos adecuados no podrá acceder a la información de los logs contenidos en esas tablas.

28 Adicionalmente, toda acción que ejecute el Motor de consultas está precedida de una autenticación con el Portal.

1.4.3 El Portal

29 El Portal es el elemento integrador de prácticamente todos los módulos que integran Bitacora. El portal es elemento integrador ya que alberga las interfaces web a través de las cuales se accede al planificador y al motor de consultas. Otra funcionalidad básica del Portal es la de autenticar a todos y cada uno de los sistemas y subsistemas que integran el producto, así como la autenticación de los usuarios que acceden al mismo. Para ello el portal proporciona una autenticación local que es la que debe ser usada. Aunque el portal proporciona la posibilidad de realizar la autenticación a través de otros sistemas (LDAP, NTLM, CAS y OpenID) la configuración evaluada tiene dichos mecanismos de autenticación deshabilitados por lo que no se consideran sistemas de autenticación válidos.

1.4.4 Planificador

30 Los distintos Agentes de Bitacora (sailors), que se encargan de recibir o recolectar logs, no saben qué es lo que tienen que hacer. Los sailors reciben instrucciones en forma de tareas y código que han de ejecutar para completarlas.

31 El planificador es el responsable de asignar estas tareas a los sailors, y proporcionarles el código necesario para llevarlos a cabo y poder recoger la información de los logs. Mediante la Consola del planificador, UATU, el administrador puede planificar la ejecución de estas tareas configurando los respectivos parámetros.

32 Este planificador se encarga de lanzar las tareas de descarga de logs, y de desencadenar el proceso que comprueba la integridad de los mismos a lo largo del tiempo.

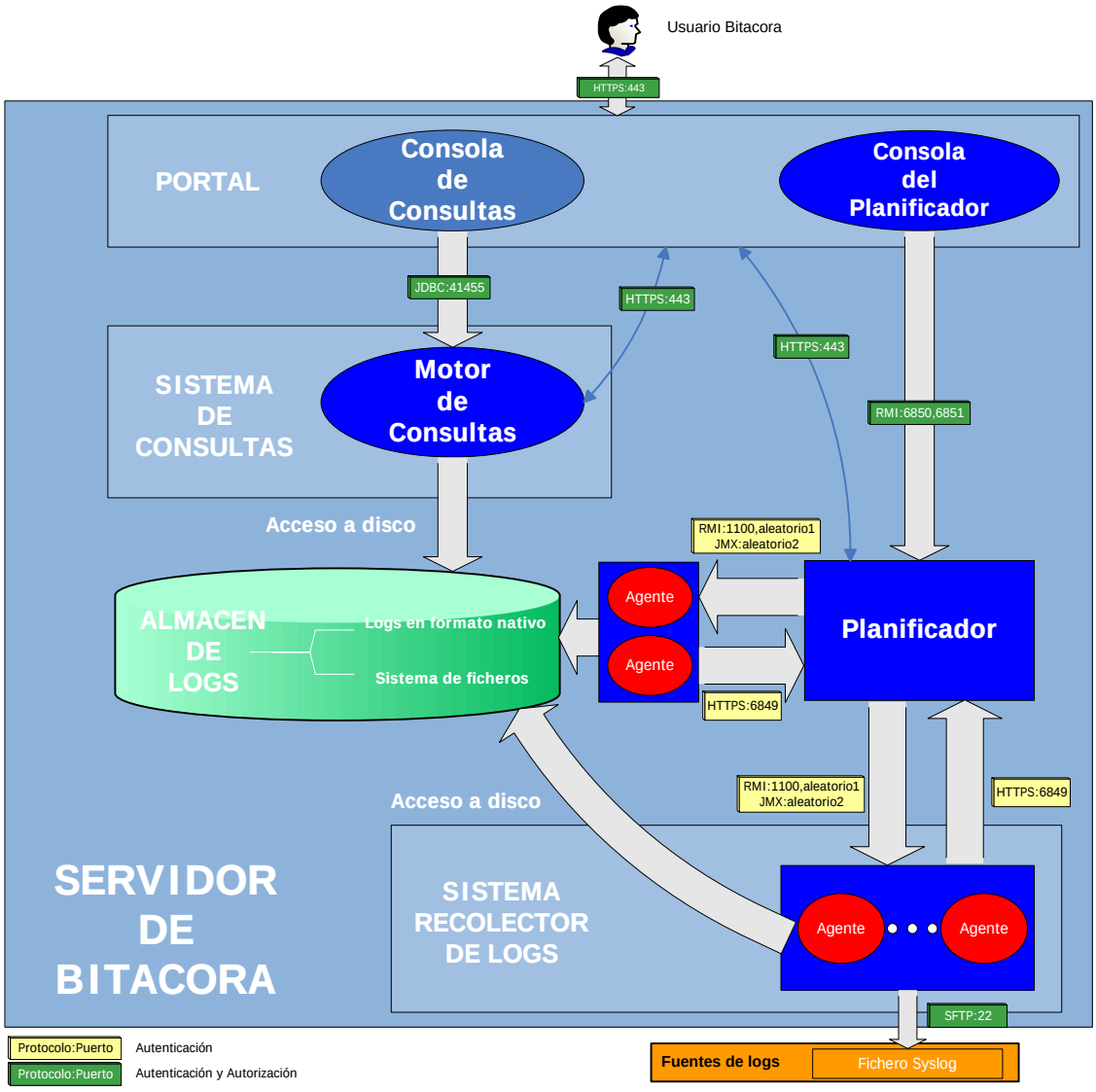
33 Del mismo modo que el resto de subsistemas, se requiere una autenticación previa del subsistema con el Portal.

1.4.5 Consola de consultas

34 La Consola de consultas es la aplicación web que posibilita al usuario del sistema acceder a información de los logs que ha sido recolectada por el sistema de recolección de datos de Bitacora. Esta consola de gestión está ligada a un potente motor de consulta que permite al usuario realizar cualquier función de consulta sobre los logs almacenados en el sistema como si se tratara de una base de datos. Mediante esta consola se posibilita la creación, modificación y eliminación de nuevas fuentes así como la política de acceso a las mismas basada perfiles.

35 A continuación se muestran dos diagramas explicativos de la arquitectura de Bitacora.

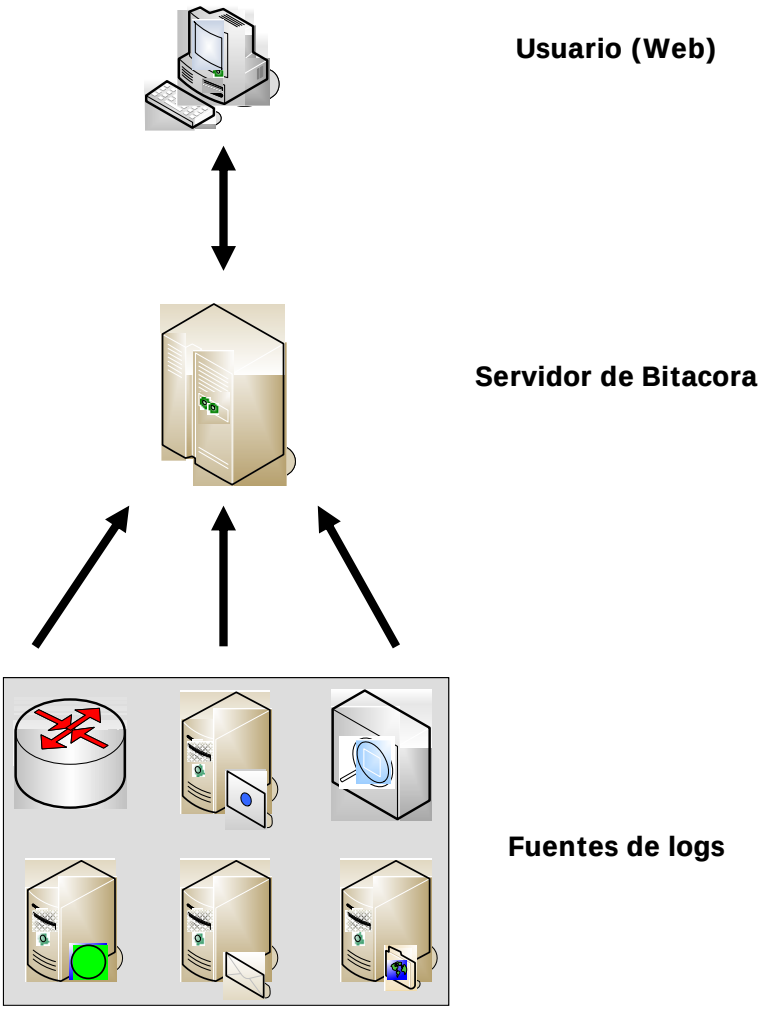
36 Arquitectura lógica:



Arquitectura lógica de Bitacora

37

38 Arquitectura física:



Arquitectura física de Bitacora

1.5 Características de seguridad de Bitacora

39 Bitacora es una potente solución de centralización, gestión y correlación de logs que permite a nuestra organización recolectar y consolidar los logs provenientes de los diferentes sistemas operativos, los dispositivos de red, las bases de datos y las aplicaciones. A estas capacidades funcionales, añade las siguientes características de seguridad:

- Garantiza la autenticidad de los datos recolectados, por cuanto exige la autenticación de las fuentes de log.

- Garantiza la confidencialidad de los logs, mediante el acceso a los mismos a través del interfaz de Bitacora, y a partir de la definición de las políticas de control de acceso de usuarios a aplicaciones de análisis de dichos logs que implementa Bitacora.

- Garantiza la integridad de los logs ya que estos son encriptadas en el momento de la recepción, y posteriormente procesados junto con los logs de los días anteriores para encontrar cualquier cambio que pudiese haber sufrido dichos logs desde su recepción.

2 PP conformance claims

2.1 CC Conformance Claim

40 Esta declaración de seguridad es conforme, en su estructura y contenido, a los requisitos de la norma Common Criteria, versión 3.1, revisión 2 y nivel de evaluación EAL2

41 Todos los requisitos de seguridad, tanto funcionales como de garantía, incluidos en esta declaración de seguridad se han extractado de las correspondientes partes 2 y 3 de dicha norma, sin que se haya añadido o extendido ningún requisito.

2.2 PP Claim, Package Claim

42 Esta declaración de seguridad no satisface ningún Perfil de Protección, sino que refleja las propiedades y soluciones de seguridad del producto Bitacora.

3 Security Problem Definition

3.1 TOE assets

3.1.1 Activos de Bitacora

- **A.LOGS**;

  La autenticidad, integridad y confidencialidad de los datos de log recabados y procesados por Bitacora.

3.2 Assumptions

3.2.1 Configuración segura

- **A.ENV**; La plataforma de uso de Bitacora, está configurada de manera que no presenta caminos o modos de acceso directo a los activos de Bitacora. Las vulnerabilidades propias de los elementos de terceros no son competencia de Bitacora por lo que deberán ser solucionadas por los fabricantes de los elementos de terceros.

  La única forma de acceso a Bitacora es a través del interfaz web que expone Bitacora. La máquina servidora en la que está instalado Bitacora deberá estar correctamente aislada y securizada para que nadie pueda acceder a través de otros interfaces a la misma. Una vez instalado Bitacora, el único acceso físico permitido a la máquina servidora de Bitacora será, a los usuarios autorizados, para reiniciar Bitacora a través del botón de arranque y/o parada de la máquina física.
  Los administradores y usuarios autorizados a acceder a Bitacora se consideran confiables y competentes en el uso de la aplicación.

3.3 Threats

3.3.1 Ataques a los activos de Bitacora

Los atacantes que se identifican a continuación están caracterizados con un potencial de ataque “básico”, conforme a las garantías del nivel de evaluación EAL2.

- **t.alogs**;

  Un atacante genera logs falsos, consiguiendo remitirlos a Bitacora como de una fuente fiable. Igualmente, un atacante consigue evitar la recepción de logs auténticos, mediante ataque directo a Bitacora.

- **t.access**;

  Un atacante consigue acceder a datos de log a los que no tiene concedido permiso de lectura.

- **t.int**;

  Un atacante consigue, a través de los interfaces de Bitacora, modificar o suprimir los datos de log almacenados en un repositorio.

4 Security Objectives.

4.1 Security Objectives for the TOE

- **o.alogs**;

  Bitacora deberá garantizar la autenticidad de los datos de log recibidos, mediante la importación de logs únicamente de fuentes autenticadas.

- **o.access**;

  Bitacora deberá permitir la definición de usuarios, y de las políticas de acceso de los mismos a las distintas aplicaciones que acceden a los repositorios de logs. El objetivo de las políticas de control de acceso es garantizar la confidencialidad de los datos. Estas políticas de control de acceso se respetarán por todos los módulos que accedan a los repositorios de logs.

- **o.int**;

  Bitacora deberá detectar e informar de cualquier modificación no autorizada de los repositorios de logs, a los efectos de poder demostrar su integridad.

4.2 Security Objectives for the Operational Environment.

- **o.env**;

  La plataforma de uso de Bitacora, incluyendo todos los elementos de terceros requeridos para su funcionamiento, se configurará de manera que no presentan caminos o modos de acceso directo a los activos de Bitacora. Los manuales de instalación y uso del producto incluirán los detalles de dicha instalación segura. Las vulnerabilidades propias de los elementos de terceros no son competencia de Bitacora por lo que deberán ser solucionadas por los fabricantes de los elementos de terceros.

  La única forma de acceso a Bitacora es a través del interfaz web que expone Bitacora. La máquina servidora en la que está instalado Bitacora deberá estar correctamente aislada y securizada para que nadie pueda acceder a través de otros interfaces a la misma. Una vez instalado Bitacora, el único acceso físico permitido a la máquina servidora de Bitacora será, a los usuarios autorizados, para reiniciar Bitacora a través del botón de arranque y/o parada de la máquina física.
  Los administradores y usuarios autorizados a acceder a Bitacora se consideran confiables y competentes en el uso de la aplicación.

4.3 Security Objectives rationale.

4.3.1 Objectivos para Bitacora

43 La siguiente tabla muestra la correspondencia trivial entre los objetivos de seguridad exigibles al producto Bitacora, y los correspondientes ataques que mitigan.

Table 1 Correspondencia entre objetivos de seguridad y amenazas

	t.alogs	t.access	t.int
o.alogs	x		
o.access		x	
o.int			x

4.3.2 Objetivos para el entorno de Bitacora

44 El objetivo de seguridad para el entorno o.env es una traslación directa de la hipótesis de uso del producto en un entorno seguro, a.env.

5 Security Requirements for the TOE

5.1 Functional Security Requirements.

5.1.1 Requisitos para la gestión de fuentes y usuarios

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 **The TSF shall be capable of performing the following management functions: [assignment: *Creación, borrado y modificación de usuarios, fuentes y aplicaciones, de sus identificadores y datos de autenticación.*].**

FMT_SMR.1 Security roles

Dependencies: FIA_UID.2 User identification before any action

FMT_SMR.1.1 **The TSF shall maintain the roles [assignment: *administrador, tecnico*].**

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

FMT_MSA.1.FUENTES Management of security attributes

Dependencies: FDP_ACC.1.FUENTES Subset access control
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1 **The TSF shall enforce the [assignment: *"política de autorización de fuentes"*] to restrict the ability to [selection: *insert, query, modify, delete*] the security attributes [assignment: *identidad y origen de las fuentes*] to [assignment: *Administrador*].**

FMT_MSA.1.USUARIOS Management of security attributes

Dependencies: FDP_ACC.1.USUARIOS Subset access control
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1 **The TSF shall enforce the [assignment: *"política de autorización de usuarios"*] to restrict the ability to [selection: *insert, query, modify, delete*] the security attributes [assignment: *identificador de usuario*] to [assignment: *Administrador*].**

FMT_MSA.3 Static attribute initialization

Dependencies: FMT_MSA.1.FUENTES Management of security attributes
FMT_MSA.1.USUARIOS Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3.1 **The TSF shall enforce the [assignment: *"política de autorización de fuentes", "política de autorización de usuarios"*] to provide [selection,**

choose one of: *restrictive*] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 **The TSF shall allow the [assignment: *Administrador*] to specify alternative initial values to override the default values when an object or information is created.**

5.1.2 Requisitos para el control de acceso

FIA_UID.2 User identification before any action

FIA_UID.2.1 **The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.**

FIA_UAU.2 User authentication before any action

Dependencies: FIA_UID.1 Timing of identification

No se considera como dependiente de FIA_UID.1 ya que el usuario no puede realizar ninguna acción sin estar previamente identificado, siempre y cuando se sigan las directrices que se indican en los documentos de Bitacora, y por ello, no se ha incluido FIA_UID.1, sino que se ha incluido el requisito FIA_UID.2, el cual se considera suficiente como dependencia del requisito actual.

FIA_UAU.2.1 **The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.**

FDP_ACF.1.FUENTES Security attribute based access control

Dependencies: FDP_ACC.1.FUENTES Subset access control
FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1.FUENTES **The TSF shall enforce the [assignment: *"política de autorización de fuentes"*] to objects based on the following: [assignment:**

- ***Lista de objetos: repositorios de log***
- ***Lista de sujetos: fuentes de datos de log***
- ***Lista de atributos: identidad y origen de la fuente***

].

FDP_ACF.1.2.FUENTES **The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *Las fuentes podrán exportar sus datos a Bitacora si están autorizadas para ello, mediante el alta de la identidad y origen de cada fuente.*].**

FDP_ACF.1.3.FUENTES **The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *Se aceptarán datos de fuentes autenticadas.*].**

FDP_ACF.1.4.FUENTES **The TSF shall explicitly deny access of subjects to objects based on the [assignment: *No se aceptarán datos de ninguna fuente no autenticada.*].**

FDP_ACF.1.USUARIOS Security attribute based access control

Dependencies: FDP_ACC.1.USUARIOS Subset access control
FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1.USUARIOS **The TSF shall enforce the [assignment: *"política de autorización de usuarios"*] to objects based on the following: [assignment:**

- ***Lista de objetos: aplicaciones registradas***
- ***Lista de sujetos: usuarios***
- ***Lista de atributos: identificador de usuario***

].

FDP_ACF.1.2.USUARIOS **The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *Los usuarios podrán acceder a las aplicaciones a las que hayan sido autorizados.*].**

FDP_ACF.1.3.USUARIOS **The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [assignment: *none.*].**

FDP_ACF.1.4.USUARIOS **The TSF shall explicitly deny access of subjects to objects based on the [assignment: *none.*].**

FDP_ACC.1.FUENTES Subset access control

Dependencies: FDP_ACF.1.FUENTES Security attribute based access control

FDP_ACC.1.1.FUENTES **The TSF shall enforce the [assignment: *"política de autorización de fuentes"*] on [assignment:**

- ***Lista de objetos: repositorios de log***
- ***Lista de sujetos: fuentes de datos de log***
- ***Lista de operaciones: importación de datos de log***

].

FDP_ACC.1.USUARIOS Subset access control

Dependencies: FDP_ACF.1.USUARIOS Security attribute based access control

FDP_ACC.1.1.USUARIOS **The TSF shall enforce the [assignment: *"política de autorización de usuarios"*] on [assignment:**

- ***Lista de objetos: aplicaciones registradas***
- ***Lista de sujetos: usuarios***
- ***Lista de operaciones: ejecución de las aplicaciones***

].

5.1.3 Requisitos para la importación autenticada de logs

FDP_ITC.1 Import of user data without security attributes

Dependencies: FDP_ACC.1.FUENTES Subset access control
FMT_MSA.3 Static attribute initialisation

User application notes

45 This component is used to specify the import of user data that does not have reliable (or any) security attributes associated with it. This function requires that the security attributes for the imported user data be initialised within the TSF. It could also be the case that the PP/ST author specifies the rules for import. It may be appropriate, in some environments, to require that these attributes be supplied via a trusted path or a trusted channel mechanism.

FDP_ITC.1.1 **The TSF shall enforce the [assignment: *"política de autorización de fuentes"*] when importing user data, controlled under the SFP, from outside of the TOE.**

FDP_ITC.1.2 **The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.**

FDP_ITC.1.3 **The TSF shall enforce the following : [assignment: *sólo se aceptarán datos de fuentes autorizadas*].when importing user data controlled under the SFP from outside the TOE**

5.1.4 Requisitos para la detección de cambios en la integridad de los repositorios de log

FDP_SDI.2 Stored data integrity monitoring and action

Dependencies: No dependencies.

FDP_SDI.2.1 **The TSF shall monitor user data stored in containers controlled by the TSF for [assignment: : "modificación y eliminacion"] on all objects, based on the following attributes: [assignment: "contenido de los logs"].**

FDP_SDI.2.2 **Upon detection of a data integrity error, the TSF shall [assignment: "generar aviso de error en una fuente de integridad"].**

5.2 Assurance Security Requirements

46 The development and the evaluation of the TOE shall be done in accordance to the following security assurance requirements:

- EAL2

ADV_ARC.1 Security architecture description

Dependencies: ADV_FSP.2 Security-enforcing functional specification
ADV_TDS.1 Basic design

Developer action elements:

ADV_ARC.1.1D **The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.**

ADV_ARC.1.2D **The developer shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities.**

ADV_ARC.1.3D **The developer shall provide a security architecture description of the TSF.**

Content and presentation of evidence elements:

ADV_ARC.1.1C **The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.**

ADV_ARC.1.2C **The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.**

ADV_ARC.1.3C **The security architecture description shall describe how the TSF initialization process is secure.**

ADV_ARC.1.4C **The security architecture description shall demonstrate that the TSF protects itself from tampering.**

ADV_ARC.1.5C **The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.**

ADV_FSP.2 Security-enforcing functional specification

Dependencies: ADV_TDS.1 Basic design

Developer action elements:

ADV_FSP.2.1D **The developer shall provide a functional specification.**

ADV_FSP.2.2D **The developer shall provide a tracing from the functional specification to the SFRs.**

Content and presentation of evidence elements:

ADV_FSP.2.1C **The functional specification shall completely represent the TSF.**

ADV_FSP.2.2C **The functional specification shall describe the purpose and method of use for all TSFI.**

ADV_FSP.2.3C **The functional specification shall identify and describe all parameters associated with each TSFI.**

ADV_FSP.2.4C **For each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.**

ADV_FSP.2.5C **For SFR-enforcing TSFIs, the functional specification shall describe direct error messages resulting from processing associated with the SFR-enforcing actions.**

ADV_FSP.2.6C **The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.**

ADV_TDS.1 Basic design

Dependencies: ADV_FSP.2 Security-enforcing functional specification

Developer action elements:

ADV_TDS.1.1D **The developer shall provide the design of the TOE.**

ADV_TDS.1.2D **The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.**

Content and presentation of evidence elements:

ADV_TDS.1.1C **The design shall describe the structure of the TOE in terms of subsystems.**

ADV_TDS.1.2C **The design shall identify all subsystems of the TSF.**

ADV_TDS.1.3C **The design shall describe the behaviour of each SFR-supporting or SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing.**

ADV_TDS.1.4C **The design shall summarise the SFR-enforcing behaviour of the SFR-enforcing subsystems.**

ADV_TDS.1.5C **The design shall provide a description of the interactions among SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other subsystems of the TSF.**

ADV_TDS.1.6C **The mapping shall demonstrate that all behaviour described in the TOE design is mapped to the TSFIs that invoke it.**

AGD_OPE.1 Operational user guidance

Dependencies: ADV_FSP.2 Security-enforcing functional specification

Developer action elements:

AGD_OPE.1.1D **The developer shall provide operational user guidance.**

Content and presentation of evidence elements:

AGD_OPE.1.1C **The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.**

AGD_OPE.1.2C **The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.**

AGD_OPE.1.3C **The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.**

AGD_OPE.1.4C **The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.**

AGD_OPE.1.5C **The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.**

AGD_OPE.1.6C **The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.**

AGD_OPE.1.7C **The operational user guidance shall be clear and reasonable.**

AGD_PRE.1 Preparative procedures

Dependencies: No dependencies.

Developer action elements:

AGD_PRE.1.1D **The developer shall provide the TOE including its preparative procedures.**

Content and presentation of evidence elements:

AGD_PRE.1.1C **The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.**

AGD_PRE.1.2C **The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.**

ALC_CMC.2 Use of a CM system

Dependencies: ALC_CMS.2 Parts of the TOE CM coverage

Developer action elements:

ALC_CMC.2.1D **The developer shall provide the TOE and a reference for the TOE.**

ALC_CMC.2.2D **The developer shall provide the CM documentation.**

ALC_CMC.2.3D **The developer shall use a CM system.**

Content and presentation of evidence elements:

ALC_CMC.2.1C **The TOE shall be labeled with its unique reference.**

ALC_CMC.2.2C **The CM documentation shall describe the method used to uniquely identify the configuration items.**

ALC_CMC.2.3C **The CM system shall uniquely identify all configuration items.**

ALC_CMS.2 Parts of the TOE CM coverage

Dependencies: No dependencies.

Developer action elements:

ALC_CMS.2.1D **The developer shall provide a configuration list for the TOE.**

Content and presentation of evidence elements:

ALC_CMS.2.1C **The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; and the parts that comprise the TOE.**

ALC_CMS.2.2C **The configuration list shall uniquely identify the configuration items.**

ALC_CMS.2.3C **For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.**

ALC_DEL.1 Delivery procedures

Dependencies: No dependencies.

Developer action elements:

ALC_DEL.1.1D **The developer shall document procedures for delivery of the TOE or parts of it to the consumer.**

ALC_DEL.1.2D **The developer shall use the delivery procedures.**

Content and presentation of evidence elements:

ALC_DEL.1.1C **The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.**

ASE_INT.1 ST introduction

Dependencies: No dependencies.

Developer action elements:

ASE_INT.1.1D **The developer shall provide an ST introduction.**

Content and presentation of evidence elements:

ASE_INT.1.1C **The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.**

ASE_INT.1.2C **The ST reference shall uniquely identify the ST.**

ASE_INT.1.3C **The TOE reference shall identify the TOE.**

ASE_INT.1.4C **The TOE overview shall summarize the usage and major security features of the TOE.**

ASE_INT.1.5C **The TOE overview shall identify the TOE type.**

ASE_INT.1.6C **The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.**

ASE_INT.1.7C **The TOE description shall describe the physical scope of the TOE.**

ASE_INT.1.8C **The TOE description shall describe the logical scope of the TOE.**

ASE_CCL.1 Conformance claims

Dependencies: ASE_INT.1 ST introduction
ASE_ECD.1 Extended components definition
ASE_REQ.2 Derived security requirements

Developer action elements:

ASE_CCL.1.1D **The developer shall provide a conformance claim.**

ASE_CCL.1.2D **The developer shall provide a conformance claim rationale.**

Content and presentation of evidence elements:

ASE_CCL.1.1C **The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.**

ASE_CCL.1.2C **The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.**

ASE_CCL.1.3C **The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.**

ASE_CCL.1.4C **The CC conformance claim shall be consistent with the extended components definition.**

ASE_CCL.1.5C **The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.**

ASE_CCL.1.6C **The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.**

ASE_CCL.1.7C **The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.**

ASE_CCL.1.8C **The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.**

ASE_CCL.1.9C **The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.**

ASE_CCL.1.10C **The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.**

ASE_SPD.1 Security problem definition

Dependencies: No dependencies.

Developer action elements:

ASE_APD.1.1D **The developer shall provide a security problem definition.**

Content and presentation of evidence elements:

ASE_SPD.1.1C **The security problem definition shall describe the threats.**

ASE_SPD.1.2C **All threats shall be described in terms of a threat agent, an asset, and an adverse action.**

ASE_SPD.1.3C **The security problem definition shall describe the OSPs.**

ASE_SPD.1.4C **The security problem definition shall describe the assumptions about the operational environment of the TOE.**

ASE_OBJ.2 Security objectives

Dependencies: ASE_SPD.1 Security problem definition

Developer action elements:

ASE_OBJ.2.1D **The developer shall provide a statement of security objectives.**

ASE_OBJ.2.2D **The developer shall provide a security objectives rationale.**

Content and presentation of evidence elements:

ASE_OBJ.2.1C **The statement of security objectives shall describe the security objectives for the TOE and the security objectives for the operational environment.**

ASE_OBJ.2.2C **The security objectives rationale shall trace each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective.**

ASE_OBJ.2.3C **The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.**

ASE_OBJ.2.4C **The security objectives rationale shall demonstrate that the security objectives counter all threats.**

ASE_OBJ.2.5C **The security objectives rationale shall demonstrate that the security objectives enforce all OSPs.**

ASE_OBJ.2.6C **The security objectives rationale shall demonstrate that the security objectives for the operational environment uphold all assumptions.**

ASE_ECD.1 Extended components definition

Dependencies: No dependencies.

Developer action elements:

ASE_ECD.1.1D **The developer shall provide a statement of security requirements.**

ASE_ECD.1.2D **The developer shall provide an extended components definition.**

Content and presentation of evidence elements:

ASE_ECD.1.1C **The statement of security requirements shall identify all extended security requirements.**

ASE_ECD.1.2C **The extended components definition shall define an extended component for each extended security requirement.**

ASE_ECD.1.3C **The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.**

ASE_ECD.1.4C **The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.**

ASE_ECD.1.5C **The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.**

ASE_REQ.2 Derived security requirements

Dependencies: ASE_OBJ.2 Security objectives
ASE_ECD.1 Extended components definition

Developer action elements:

ASE_REQ.2.1D **The developer shall provide a statement of security requirements.**

ASE_REQ.2.2D **The developer shall provide a security requirements rationale.**

Content and presentation of evidence elements:

ASE_REQ.2.1C **The statement of security requirements shall describe the SFRs and the SARs.**

ASE_REQ.2.2C **All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.**

ASE_REQ.2.3C **The statement of security requirements shall identify all operations on the security requirements.**

ASE_REQ.2.4C **All operations shall be performed correctly.**

ASE_REQ.2.5C **Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.**

ASE_REQ.2.6C **The security requirements rationale shall trace each SFR back to the security objectives for the TOE.**

ASE_REQ.2.7C **The security requirements rationale shall demonstrate that the SFRs meet all security objectives for the TOE.**

ASE_REQ.2.8C **The security requirements rationale shall explain why the SARs were chosen.**

Para identificar el nivel de seguridad requerido por el sistema, se ha realizado un estudio el cual ha concluido que tal y como está la demanda del mercado competencia de Bitacora, se hace conveniente la evaluación de nivel EAL2.

ASE_REQ.2.9C **The statement of security requirements shall be internally consistent.**

ASE_TSS.1 TOE summary specification

Dependencies: ASE_INT.1 ST introduction
ASE_REQ.2 Derived security requirements
ADV_FSP.2 Security-enforcing functional specification

Developer action elements:

ASE_TSS.1.1D **The developer shall provide a TOE summary specification.**

Content and presentation of evidence elements:

ASE_TSS.1.1C **The TOE summary specification shall describe how the TOE meets each SFR.**

ATE_COV.1 Evidence of coverage

Dependencies: ADV_FSP.2 Security-enforcing functional specification
ATE_FUN.1 Functional testing

Developer action elements:

ATE_COV.1.1D **The developer shall provide evidence of the test coverage.**

Content and presentation of evidence elements:

ATE_COV.1.1C **The evidence of the test coverage shall show the correspondence between the tests in the test documentation and the TSFIs in the functional specification.**

ATE_FUN.1 Functional testing

Dependencies: ATE_COV.1 Evidence of coverage

Developer action elements:

ATE_FUN.1.1D **The developer shall test the TSF and document the results.**

ATE_FUN.1.2D **The developer shall provide test documentation.**

Content and presentation of evidence elements:

ATE_FUN.1.1C **The test documentation shall consist of test plans, expected test results and actual test results.**

ATE_FUN.1.2C **The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.**

ATE_FUN.1.3C **The expected test results shall show the anticipated outputs from a successful execution of the tests.**

ATE_FUN.1.4C **The actual test results shall be consistent with the expected test results.**

ATE_IND.2 Independent testing - sample

Dependencies: ADV_FSP.2 Security-enforcing functional specification
AGD_OPE.1 Operational user guidance
AGD_PRE.1 Preparative procedures
ATE_COV.1 Evidence of coverage
ATE_FUN.1 Functional testing

Developer action elements:

ATE_IND.2.1D **The developer shall provide the TOE for testing.**

Content and presentation of evidence elements:

ATE_IND.2.1C **The TOE shall be suitable for testing.**

ATE_IND.2.2C **The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.**

AVA_VAN.2 Vulnerability analysis

Dependencies: ADV_ARC.1 Security architecture description
ADV_FSP.1 Basic functional specification
ADV_TDS.1 Basic design
AGD_OPE.1 Operational user guidance
AGD_PRE.1 Preparative procedures

Developer action elements:

AVA_VAN.2.1D **The developer shall provide the TOE for testing.**

Content and presentation of evidence elements:

AVA_VAN.2.1C **The TOE shall be suitable for testing.**

5.3 Rationale for the Security Requirements

47 El objetivo o.alogs, correspondiente a la garantía de la autenticidad de los datos de logs recibidos se satisface mediante la creación de fuentes, la disposición de roles basadas en las políticas de autorización de usuarios y fuentes, la asignación de los permisos adecuados a los usuarios para las fuentes y la disposición de claves de encriptación y de certificados de autenticación de fuentes y usuarios. Estos requisitos son suficientes ya que todo usuario está autenticado, teniendo acceso sólo a lo que debe tener acceso, por lo que solo el administrador puede definir nuevas fuentes; y toda fuente debe estar autenticada, por lo que solo se aceptan logs de fuentes en las que se confía. De este modo, se asegura que los logs que se han recibido son auténticos.

48 El objetivo o.access relativo al acceso a la información que es únicamente necesaria para el desempeño de sus competencias se satisface mediante la creación de fuentes, la disposición de roles basadas en las políticas de autorización de usuarios y fuentes, la asignación de los permisos adecuados a los usuarios para las fuentes y la identificación y autenticación de los usuarios. Se consideran que los requisitos son suficientes ya que ningún usuario puede acceder a la información de logs para la cual no tiene permisos.

49 El objetivo o.int referente a la integridad de los log se cumple con la consideración de que Bitacora proporciona un mecanismo de control de integridad que permite detectar cambios en los logs. Cualquier modificación de los logs de forma no controlada con posterioridad a la recepción de los mismos, es detectada e informada en el sistema.

50 La siguiente tabla muestra de forma resumida la correspondencia entre los objetivos de seguridad exigibles al producto Bitacora, y los correspondientes requisitos funcionales que los satisfacen.

Table 2 Correspondencia entre objetivos de seguridad y requisitos funcionales

	o.alogs	o.access	o.int
fmt_smf.1	X	X	
fmt_smr.1	X	X	
fmt_msa.1.FUENTES	X		
fmt_msa.1.USUARIOS	X	X	
fmt_msa.3	X	X	
fia_uid.2		X	
fia_uau.2		X	
fdp_sdi.2			X
fdp_acf.1.FUENTES	X		
fdp_acf.1.USUARIOS		X	
fdp_acc.1.FUENTES	X		
fdp_acc.1.USUARIOS		X	
fdp_itc.1	X		

6 TOE Summary Specification

En este apartado se describe, a alto nivel, cómo implementa cada uno de los requisitos especificados en la sección **5.1 Functional Security Requirements**, así como su relación con cada uno de los TSFIs asociados.

6.1 Requisitos para la gestión de fuentes y usuarios

FMT_SMF.1 Specification of Management Functions

El portal es el componente de Bitacora que permite la creación, borrado y modificación de usuarios, aplicaciones y de sus identificadores y datos de autenticación.

La Consola del planificador permite crear, borrar y modificar fuentes para la recolección de los logs y la Consola de consultas permite la creación y definición de fuentes para la consulta de las mismas.

Para las funciones descritas en el requisito, Bitacora ofrece una serie de interfaces que permiten realizar las acciones descritas:

- AÑADIR USUARIO
- BUSCAR USUARIO
- MODIFICAR USUARIO
- BORRADO DE USUARIO
- CAMBIO DE CONTRASEÑA DE UN USUARIO
- DESCARGA DE FICHERO POR SFTP
- AÑADIR FUENTE
- USUARIOS Y PERFILES - IMPORTAR XML

FMT_SMR.1 Security roles

Bitacora gestiona los permisos de los usuarios en base a dos perfiles lógicos: administrador y tecnico. Para ello se apoya en la gestión de roles que tiene definido el portal (roles de portal y roles de comunidad) y la gestión de perfiles para el acceso a fuentes a través de la Consola de consultas. Los permisos de un usuario en Bitacora queda definido por la tupla: Rol de portal + Rol de Comunidad + Perfil de la Consola de consultas. Los roles además pueden ser creados, borrados y modificados. Para ello Bitacora presenta los interfaces:

- BUSCAR ROLES
- AÑADIR ROL
- ELIMINAR ROL
- USUARIOS Y PERFILES - ASOCIAR PERFILES CON USUARIO
- USUARIOS Y PERFILES - ASOCIAR USUARIOS CON PERFIL
- USUARIOS Y PERFILES - CREAR PERFIL
- USUARIOS Y PERFILES - ELIMINAR PERFIL
- PERMISOS

FMT_MSA.1.FUENTES Management of security attributes

Bitacora permite gestionar a los usuarios con perfil “Administrador”:

Los atributos de seguridad referentes a la recolección de logs a través de la Consola del Planificador

La gestión de las fuentes de logs desde el menú de administración de la Consola de consultas.

Para realizar estas acciones proporciona los interfaces:

- AÑADIR FUENTE
- ASIGNAR ÁRBOL
- USUARIOS Y PERFILES - EXPORTAR XML
- USUARIOS Y PERFILES - IMPORTAR XML
- USUARIOS Y PERFILES - LIMPIAR CACHE DE ARBOLES
- USUARIOS Y PERFILES - LIMPIAR CACHE DE ALEPH

FMT_MSA.1.USUARIOS Management of security attributes

La gestión de los atributos de seguridad son permitidos sólo a aquellos usuarios con perfil "Administrador". Para ello, Bitacora se apoya en la gestión de usuarios del portal y en la gestión de permisos a través de la Consola de consultas a través de los interfaces:

- AÑADIR USUARIO
- BUSCAR USUARIO
- MODIFICAR USUARIO
- BORRADO DE USUARIO
- CAMBIO DE CONTRASEÑA DE UN USUARIO
- BUSCAR ORGANIZACIONES
- AÑADIR ORGANIZACIÓN
- ELIMINAR ORGANIZACIÓN
- BUSCAR GRUPOS DE USUARIO
- AÑADIR GRUPO DE USUARIOS
- ELIMINAR GRUPO DE USUARIOS
- PERMISOS
- USUARIOS Y PERFILES - ASOCIAR PERFILES CON USUARIO
- USUARIOS Y PERFILES - ASOCIAR USUARIOS CON PERFIL
- USUARIOS Y PERFILES - CREAR PERFIL
- USUARIOS Y PERFILES - ELIMINAR PERFIL
- USUARIOS Y PERFILES - COPIAR USUARIOS

FMT_MSA.3 Static attribute initialization

Bitacora presenta una configuración inicial (usuarios, roles, atributos referentes a la autenticación, política de contraseñas y fuentes) que permite al usuario con perfil "Administrador" gestionar la configuración de la aplicación de una forma más sencilla. Esta configuración inicial se puede modificar a través de los interfaces:

- AUTENTIFICACIÓN - GENERAL
- ASOCIACIONES POR DEFECTO A LOS USUARIOS
- ACCESO BÚSQUEDAS
- BÚSQUEDAS - BUSCAR
- BÚSQUEDAS LIBRES - BUSCAR
- BUSCAR POLÍTICAS DE CONTRASEÑAS
- AÑADIR POLÍTICA DE CONTRASEÑAS
- ELIMINAR POLÍTICAS DE CONTRASEÑAS

6.2 Requisitos para el control de acceso

FIA_UID.2 User identification before any action

Bitacora dispone de un sistema de autenticación que impide la realización de cualquier acción a un usuario no registrado en Bitacora. El portal es el componente de Bitacora que

contiene un sistema de autenticación local y que proporciona los siguientes interfaces para el acceso a Bitacora y configuración del mismo a Bitacora:

- HOME
- LOGIN
- AUTENTIFICACIÓN - GENERAL

FIA_UAU.2 User authentication before any action

FDP_ACF.1.USUARIOS Security attribute based access control

FDP_ACC.1.USUARIOS Subset access control

Bitacora comprueba los permisos que tiene un usuario antes de permitir la realización de cualquier acción. En Bitacora los permisos se definen a nivel de portal y a nivel de la Consola de consultas a través de los interfaces:

- BUSCAR ROLES
- AÑADIR ROL
- ELIMINAR ROL
- USUARIOS Y PERFILES - ASOCIAR PERFILES CON USUARIO
- USUARIOS Y PERFILES - ASOCIAR USUARIOS CON PERFIL
- USUARIOS Y PERFILES - CREAR PERFIL
- USUARIOS Y PERFILES - ELIMINAR PERFIL
- PERMISOS

FDP_ACF.1.FUENTES Security attribute based access control

FDP_ACC.1.FUENTES Subset access control

Bitacora autentica las fuentes de las que recolecta logs a través de la Consola del planificador que a su vez se apoya en el protocolo SFTP para realizar la descarga de forma segura y autenticada. Así mismo, la autenticación de Bitacora en la fuente de la que se recolectan los logs se realiza mediante logado con usuario y contraseña y el certificado correspondiente. Bitacora exige que la primera conexión con la fuente sea segura y en las sucesivas conexiones se realiza la autenticación de la fuente y la comunicación se realiza de forma segura mediante protocolo SSL. Los interfaces que integran este requisito son:

- DESCARGA DE FICHERO POR SFTP
- INTERFAZ SFTP

6.3 Requisitos para la importación autenticada de logs

FDP_ITC.1 Import of user data without security attributes

Para poder recolectar logs en Bitacora es necesario que la fuente esté autenticada con Bitacora, mediante la comprobación del certificado de la fuente. Así mismo, la autenticación de Bitacora en la fuente de la que se recolectan los logs se realiza mediante logado con usuario y contraseña y el certificado correspondiente. A partir de ese momento, la comunicación se realiza de forma segura mediante protocolo SSL:

Para realizar estas acciones proporciona los interfaces:

- DESCARGA DE FICHERO POR SFTP
- INTERFAZ SFTP

6.4 Requisitos para la detección de cambios en la integridad de los repositorios de log

FDP_SDI.2 Stored data integrity monitoring and action

Bitacora dispone de una serie de procesos que aseguran la integridad de los datos de los repositorios de logs y que se ejecutan de forma automática tras cada descarga de logs y para cada fuente de forma independiente. Además Bitacora realiza un firmado de los logs y la comprobación diaria de que no han sido modificados o eliminados. Estos procesos generan información de integridad que se presentan como una fuente más en el sistema y que debe ser consultada en la Consola de consultas tal y como indique el procedimiento de comprobación de integridad especificado en la documentación.

Para realizar estas acciones proporciona los interfaces:

- DESCARGA DE FICHERO POR SFTP
- BÚSQUEDAS - BUSCAR
- BÚSQUEDAS LIBRES - BUSCAR

Pamplona . San Sebastián . Barcelona
Madrid . Sevilla . México DF . Buenos Aires