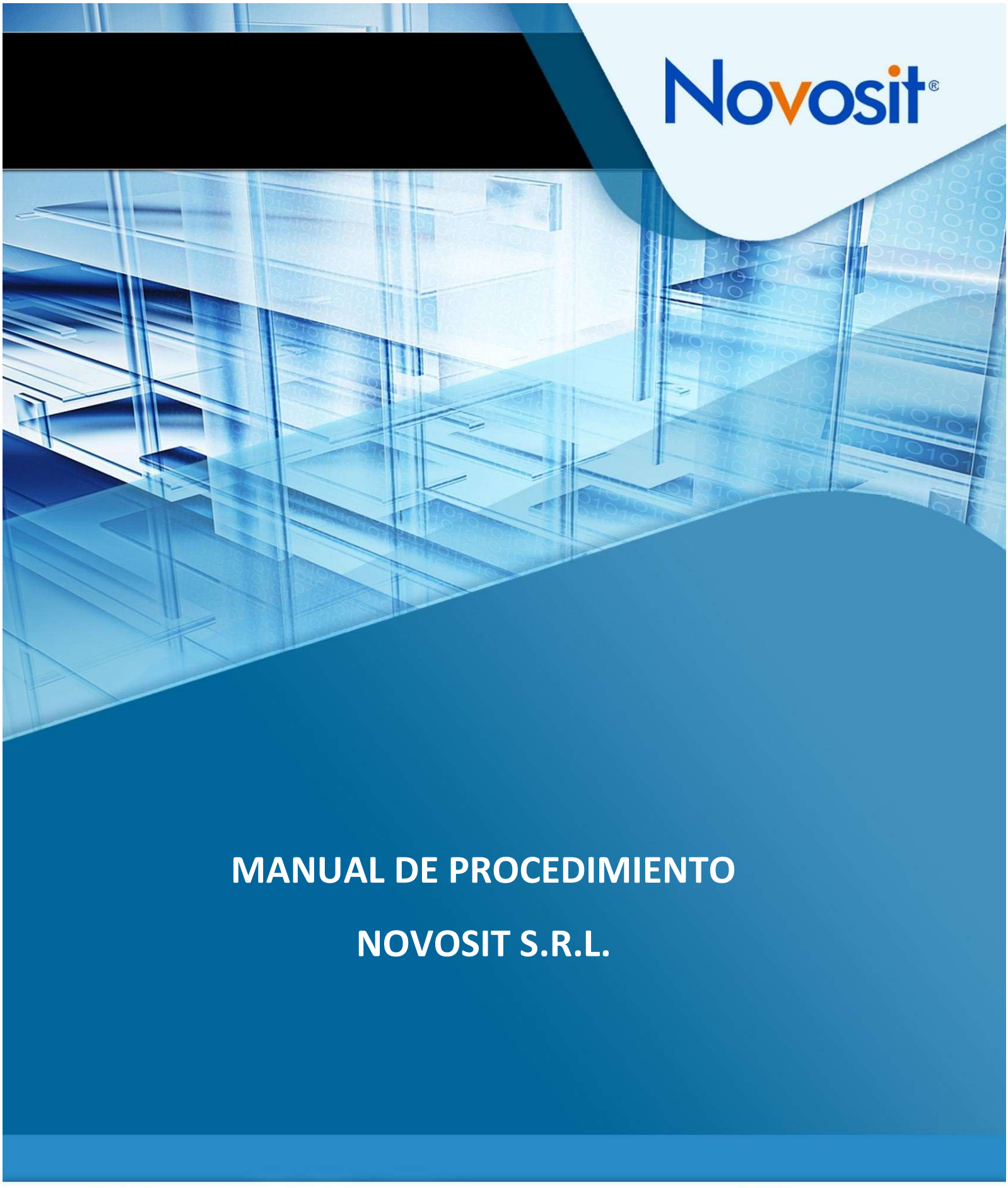


MANUAL DE PROCEDIMIENTO

NOVOSIT S.R.L.

OnBase by Hyland
ProDoctivity Digital All
Microsoft Partner Gold Independent Software Vendor (ISV)
PARASCRIPT
NovoSign

Información general

Control documental

Clasificación de seguridad:	**Uso interno**
Versión:	**1**
Fecha edición:	19/10/2020
Fichero:	**NOVOSIT_Manual_Procedimientos_v1**

Estado formal

Preparado por:	Revisado por:	Aprobado por:
Nombre: Alejandro Grande Fecha: 19/10/2020	Nombre: Arturo Díaz Fecha: 06/11/2020	Nombre: Francis Reyes Fecha: 06/11/2020
	Nombre: Arturo Díaz Fecha: 06/09/2021	Nombre: Francis Reyes Fecha: 06/09/2021
	Nombre: Ramón Sarante Fecha: 18/10/2022	Nombre: Francis Reyes Fecha: 18/10/2022
	Nombre Ramón Sarante Fecha: 06/11/2023	Nombre: Francis Reyes Fecha: 06/11/2023
	Nombre: Ramón Sarante Fecha: 02/12/2024	Nombre: Francis Reyes Fecha: 02/12/2024

Control de versiones

Versión	Partes que cambian	Descripción cambio	Autor cambio	Fecha cambio
1.0	Original	Creación del documento		19/10/2020

Índice

Introducción

NOVOSIT SRL , en lo sucesivo NOVOSIT, ha elaborado el presente documento que contiene el Manual de Procedimiento, en el que se recogen y describen los diferentes procedimientos de seguridad que se implementan en la Prestación de Servicios de Confianza, todo ello a través de Uanataca, S.A., como proveedor de los servicios de infraestructura de clave pública de NOVOSIT.

Normas y referencias

Norma ISO/IEC 27001, puntos A.11.2.5, A.12.1.3, A.12.4 y A.12.6.1.

1 Gestión de dispositivos

1.1 Objetivo

Describir el procedimiento para la gestión de altas, bajas o modificación de dispositivos para la prestación de servicios.

1.2 Descripción

N/A.

1.3 Ámbito de aplicación

Este procedimiento afecta a cualquier dispositivo utilizado para prestación de los servicios.
En este procedimiento se describe el proceso a seguir por el personal que provee de las Infraestructuras.

1.4 Documentación relacionada

N/A

1.5 Anexos

N/A

1.6 Responsabilidad

Se identifican las siguientes responsabilidades en la aplicación de este procedimiento:

- Responsable de Infraestructuras:
    - Gestionar las solicitudes de asignación, devolución o baja de dispositivos. Adicionalmente definirá los procesos de copia de seguridad, formateo seguro, resguardo y control de inventario de los dispositivos.

- Usuarios y peticionarios en general:
    - Realizar las peticiones de servicio siguiendo las instrucciones descritas en este procedimiento.

Es responsabilidad del departamento de Infraestructuras mantener vigente éste documento, así como su aplicación.

1.7 Procedimiento

La petición de servicio se realizará a través de la herramienta de ticketing, https://bit4id.mantishub.io/my_view_page.php

- Alta
- Baja
- Modificación

Todos los dispositivos están actualizados en un inventario que será actualizado cada vez que se realice un alta, baja o modificación de dispositivo.

1.7.1 Alta de dispositivo

- El usuario se conectará a la herramienta de ticketing y reportará la solicitud. La petición de servicio debe contener la siguiente información:

    - Dispositivo requerido.
    - Fecha para la cual se requiere el alta.

- Accesos y aplicaciones que debe tener instaladas.

- Una vez complementada toda la información, se envía la petición de servicio. El usuario que ha realizado la petición será informado del estado por correo electrónico.

1.7.2 Baja de dispositivo

- El interesado contactará con el departamento de Infraestructuras mediante la herramienta de ticketing e informará del material que debe ser procesado para su baja o reutilización.
- En el caso de que el material debe ser desechado se realizará de manera segura, habiendo tomado las medidas de seguridad para que no pueda sustraerse información.
- El dispositivo será llevado a un centro de recogida de material para su reciclado de manera anónima asegurando que el dispositivo no posee identificación alguna que lo pueda relacionar con los servicios prestados por Uanataca.

1.8 Flujo de procesos

N/A

2 Mantenimiento de sistemas parches y actualizaciones de seguridad

2.1 Objetivo

Describir el procedimiento para la valoración y aplicación de actualizaciones sobre el software base y parches de seguridad emitidos por los fabricantes.

2.2 Descripción

Los fabricantes de software publican actualizaciones sobre sus sistemas mediante nuevas versiones o parches que corrigen problemas o vulnerabilidades de seguridad. Se describen dos procedimientos para realizar su evaluación y en tipo de intervención a realizar:
Correctivo: Mantenimiento que se realiza a cabo para corregir los errores detectados durante la explotación del sistema.
Evolutivo: Mantenimiento que se realiza para mejorar o añadir alguna característica nueva al software.

2.3 Ámbito de aplicación

Este procedimiento afecta a sistemas operativos, software de seguridad, bases de datos y aplicaciones.

2.4 Documentación relacionada

N/A

2.5 Anexos

2.9.1 Requerimientos de seguridad

2.9.2 Formulario de solicitud de mantenimiento de programas

2.6 Responsabilidad

Es responsabilidad de departamento técnico evaluar la idoneidad de si una actualización debe ser aplicada sobre el sistema y en caso de llevarse a cabo cómo debe ser la intervención.

Es responsabilidad del departamento técnico mantener vigente y la aplicación de este documento.

2.7 Procedimiento

El presente procedimiento se realiza tomando como referencia marcos metodológicos de desarrollos o mantenimiento de programas. Los marcos metodológicos para el desarrollo de programas establecen un procedimiento sistemático que definen como debe realizarse la solicitud, el circuito de aprobaciones, la priorización de modificaciones, el seguimiento, los procedimientos de traspasos a producción, elaboración de documentación técnica, funcional y de usuario, fase de pruebas, formación, etc.

A continuación se indican las actividades y tareas para los desarrollos correctivos y evolutivos:

- Registro de la solicitud de mantenimiento
    - Realización por parte del Responsable de referencia
    - Asignación de la solicitud

- Análisis de la solicitud

- Verificación y estudio de la solicitud
- Preparación de la propuesta de solución

- Desarrollo de la solicitud
  - Planificación de tareas
  - Ejecución de las pruebas

- Seguimiento y aceptación
  - Seguimiento del desarrollo
  - Realización de pruebas
  - Aprobación y cierre de la solicitud

2.7.1 Realización de la solicitud de mantenimiento

2.7.1.1 Realización de la solicitud

- El Responsable debe rellenar el formulario de Solicitud de Mantenimiento, completando los distintos apartados según corresponda.

2.7.1.2 Asignación de la solicitud

- La solicitud será asignada a uno de los Jefes de Proyectos de desarrollos a medida o aplicaciones estándar, según sea la solicitud.

- Una vez asignada la solicitud, el Jefe de Proyecto contactará a la persona que realiza la petición para revisar la solicitud y planificar el inicio del trabajo.

2.7.2 Análisis de la solicitud

En esta actividad, como su propio nombre indica, se llevará a cabo un análisis de la petición y un diagnóstico de esta para dar una respuesta sobre la solicitud.

2.7.2.1 Verificación y estudio de la solicitud

- El Jefe de Proyecto realizará inicialmente la verificación que incluirá:
    - Revisión del tipo de solicitud (mantenimiento correctivo, evolutivo o nuevo desarrollo).
    - Factibilidad del tipo de solicitud
    - Análisis de los requerimientos de negocio y técnicos de la solicitud

- Una vez realizada la verificación se realizará el estudio que será diferente según el tipo de solicitud:
    - Si se trata de una solicitud de mantenimiento correctivo, y según el acuerdo de nivel de servicio establecido para los sistemas de información afectados, se evalúa la criticidad del problema. Esta evaluación permitirá determinar si la solución es a corto plazo, es decir, urgente o inmediata, o si es a medio o a largo plazo:
        - Si el problema es crítico, su análisis y solución comienza inmediatamente con el fin de reanudar rápidamente el nivel de servicio. Sin embargo, este modo de actuación no elimina la necesidad de una revisión posterior del problema para valorar los posibles efectos secundarios, establecer una solución definitiva y actualizar todos los productos implicados.

        - Si no es crítico, la petición se clasifica para proceder en la tarea siguiente a determinar cuál es la solución más adecuada.

    - En el caso de un mantenimiento evolutivo se delimita su alcance determinando si se trata de una modificación a los sistemas de información inicialmente afectados o de una incorporación para cubrir nuevas funcionalidades no contempladas hasta el momento en dichos sistemas de información.

2.7.2.2 Preparación de la propuesta de solución

Para la preparación de la propuesta se tomarán en cuenta los siguientes aspectos siempre y cuando aplique:

- Identificar solicitudes en curso que puedan afectar a los mismos sistemas.
- Alcance valorado.
- Prioridad inicialmente asignados.
- Requisitos planteados para cada solicitud y análisis, aun alto nivel, de los sistemas de información/páginas webs implicadas.

En cualquiera de las situaciones anteriores, se hace una estimación preliminar del esfuerzo requerido mediante los indicadores establecidos en el acuerdo de nivel de servicio para cada sistema de información, según la tecnología aplicada, naturaleza y tamaño del sistema de información y los tipos de lenguajes utilizados, bases de datos, etc.

2.7.3 Desarrollo de la solicitud

2.7.3.1 Planificación de tareas

- El Responsable debe rellenar el formulario de Solicitud de Mantenimiento, completando los distintos apartados según corresponda.
  - Planificación de tareas
  - Ejecución de las pruebas

2.7.3.2 Ejecución de las pruebas

- El Responsable debe rellenar el formulario de Solicitud de Mantenimiento, completando los distintos apartados según corresponda.

2.8 Flujo de procesos

El registro y gestión de las solicitudes, a través del Sistema de Solicitudes de Servicio, se realiza según el flujo representado en la Figura 1:

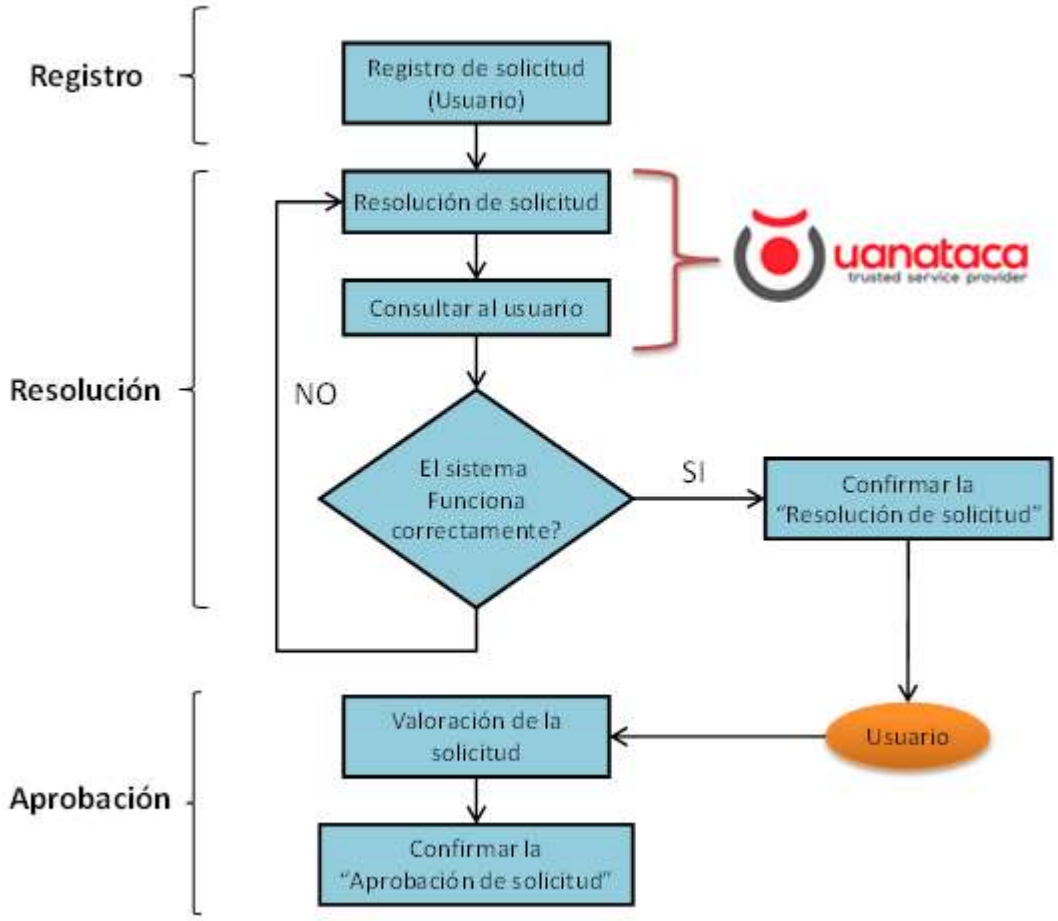


Figura 1: Flujo de registro y gestión de solicitudes.

2.9 Anexo

2.9.1 Requerimientos de seguridad

FASE DE REQUERIMIENTOS

- Control de la autenticación
- Control de roles y privilegios
- Requerimientos orientados a la evaluación del riesgo
- Aprobación de los privilegios por tipo de perfil
- Matriz control de roles y privilegios

FASE DE ANÁLISIS Y DISEÑO

- Acceso a componentes y administración del sistema
- Trazas para la auditoría
- Gestión de las sesiones
- Datos históricos
- Gestión adecuada de los errores
- Segregación, separación de Roles

FASE DE DESARROLLO E IMPLANTACACIÓN

- Securización del entorno de desarrollo
- Elaboración documentación técnica
- Desarrollo seguro (requerir estándar de seguridad)
- Seguridad en las comunicaciones
- Seguridad en paso al entorno de producción

DESARROLLO SEGURO

- Validación de entradas
- Codificación de las salidas
- Gestión de logs de los cambios
- Uso de criptografía
- Gestión de errores y logs

- Gestión de archivos
- Gestión de la memoria
- Estandarización y reutilización de funciones de seguridad

FASES DE PRUEBAS

- Control de calidad y de seguridad
- Inspección del código por fases
- Comprobación de la gestión de configuraciones
- Test de vulnerabilidades e intrusión

FASE DE MANTENIMIENTO

- Aseguramiento basado en gestión de riesgos
- Test de vulnerabilidades e intrusión después de cambios

2.9.2 Formulario de solicitud de mantenimiento de programas

Las solicitudes de mantenimiento de programas deberán realizarse cumplimentando la información requerida en seguida:

1. Titulo

*Descripción corta (1 línea de la tarea)

2. Descripción exacta de la tarea

*Descripción exacta del desarrollo con imágenes orientativas de donde tiene que quedar cualquier nueva opción.

3. Pasos para reproducirlo

*Explicación detallada que los pasos que debería seguir alguien que no conoce la aplicación para poder ver el problema o ver el sitio donde se solicita el nuevo desarrollo.

4. Datos Técnicos

*Módulos, clases, tablas a editar o tener en cuenta.

3 Desarrollo y mantenimiento de programas

3.1 Objetivo

Describir el procedimiento para la realización de nuevos desarrollos de sistemas y/o realización de cambios a los programas de los sistemas productivos o en funcionamiento.

3.2 Descripción

En este apartado se describen los conceptos de los tipos de mantenimientos a realizar:
Correctivo: Mantenimiento que se realiza a cabo para corregir los errores detectados durante la explotación del sistema.
Evolutivo: Mantenimiento que se realiza para mejorar o añadir alguna característica nueva al programa cuyo coste no sea superior al 60% de dedicación respecto al resto de mantenimientos.

3.3 Ámbito de aplicación

Este procedimiento afecta a cualquier nuevo desarrollo de un sistema informático o página web. Igualmente afecta a los mantenimientos (correctivos y evolutivos) que se realicen a los programas que conforman los sistemas productivos o en funcionamiento.
En este procedimiento se describe el proceso a seguir por las Business Area (en adelante BA) para la realización de peticiones de nuevos desarrollos o mantenimientos, así como el proceso de ejecución de las tareas a realizar por los Jefes de Proyectos del departamento de Information and Communication Technologies (en adelante ICT).

3.4 Documentación relacionada

N/A

3.5 Anexos

3.9.1 Requerimientos de seguridad

3.9.2 Formulario de solicitud de mantenimiento de programas

SE INCLUIRA PLANTILLA PARA LA REALIZAR LAS PETICIONES DE DESARROLLOS Y MANTENIMIENTOS.

3.6 Responsabilidad

Se identifican las siguientes responsabilidades en la aplicación de este procedimiento:

- Responsables de Business Area:
    - Aprobar las peticiones de nuevos desarrollos o de mantenimientos (correctivos/evolutivos), describir los requerimientos de negocio y funcionales, priorizar la solicitud, participar en las pruebas de los desarrollos y aprobar el trabajo formalmente una vez finalizado.

- Jefe de proyecto de desarrollos a medida / Jefe de Proyecto aplicaciones estándar:

    - Gestionar las solicitudes recibidas, validar los requerimientos y prioridad con las BA, planificar las tareas y el plan de pruebas, realizar la documentación soporte según la solicitud.

- Security Project Manager:

    - Evaluar los riesgos de seguridad y cumplimiento regulatorio TI de las peticiones de nuevos desarrollos o cambios relevantes a realizar.

Es responsabilidad del departamento técnico mantener vigente y la aplicación de este documento.

3.7 Procedimiento

El presente procedimiento se realiza tomando como referencia marcos metodológicos de desarrollos o mantenimiento de programas. Los marcos metodológicos para el desarrollo de programas establecen un procedimiento sistemático que definen como debe realizarse la solicitud, el circuito de aprobaciones, la priorización de modificaciones, el seguimiento, los procedimientos de traspasos a producción, elaboración de documentación técnica, funcional y de usuario, fase de pruebas, formación, etc.
A continuación, se indican las actividades y tareas para los desarrollos correctivos y evolutivos:

- Registro de la solicitud de mantenimiento
    - Realización por parte del Responsable de la BA
    - Asignación de la solicitud

- Análisis de la solicitud
    - Verificación y estudio de la solicitud
    - Preparación de la propuesta de solución

- Desarrollo de la solicitud
    - Planificación de tareas
    - Ejecución de las pruebas

- Seguimiento y aceptación
    - Seguimiento del desarrollo
    - Realización de pruebas
    - Aprobación y cierre de la solicitud

3.7.1 Realización de la solicitud de mantenimiento

3.7.1.1 Realización de la solicitud

- El Responsable de la BA debe rellenar el formulario de Solicitud de Mantenimiento, completando los distintos apartados según corresponda.

3.7.1.2 Asignación de la solicitud

- La solicitud será asignada a uno de los Jefes de Proyectos de desarrollos a medida o aplicaciones estándar, según sea la solicitud.

- Una vez asignada la solicitud, el Jefe de Proyecto contactará a la persona que realiza la petición para revisar la solicitud y planificar el inicio del trabajo.

3.7.2 Análisis de la solicitud

En esta actividad, como su propio nombre indica, se llevará a cabo un análisis de la petición y un diagnóstico de la misma para dar una respuesta sobre la solicitud.

3.7.2.1 Verificación y estudio de la solicitud

- El Jefe de Proyecto realizará inicialmente la verificación que incluirá:
    - Revisión del tipo de solicitud (mantenimiento correctivo, evolutivo o nuevo desarrollo).
    - Factibilidad del tipo de solicitud
    - Análisis de los requerimientos de negocio y técnicos de la solicitud

- Una vez realizada la verificación se realizará el estudio que será diferente según el tipo de solicitud:

- Si se trata de una solicitud de mantenimiento correctivo, y según el acuerdo de nivel de servicio establecido para los sistemas de información afectados, se evalúa la criticidad del problema. Esta evaluación permitirá determinar si la solución es a corto plazo, es decir, urgente o inmediata, o si es a medio o a largo plazo:

    - Si el problema es crítico, su análisis y solución comienza inmediatamente con el fin de reanudar rápidamente el nivel de servicio. Sin embargo, este modo de actuación no elimina la necesidad de una revisión posterior del problema para valorar los posibles efectos secundarios, establecer una solución definitiva y actualizar todos los productos implicados.

    - Si no es crítico, la petición se clasifica para proceder en la tarea siguiente a determinar cuál es la solución más adecuada.

- En el caso de un mantenimiento evolutivo se delimita su alcance determinando si se trata de una modificación a los sistemas de información inicialmente afectados o de una incorporación para cubrir nuevas funcionalidades no contempladas hasta el momento en dichos sistemas de información.

3.7.2.2 Preparación de la propuesta de solución

Para la preparación de la propuesta se tomarán en cuenta los siguientes aspectos siempre y cuando aplique:

- Identificar solicitudes en curso que puedan afectar a los mismos sistemas.
- Alcance valorado.
- Prioridad inicialmente asignados.
- Requisitos planteados para cada solicitud y análisis, aun alto nivel, de los sistemas de información/páginas webs implicadas.

En cualquiera de las situaciones anteriores, se hace una estimación preliminar del esfuerzo requerido mediante los indicadores establecidos en el acuerdo de nivel de servicio para cada sistema de información, según la tecnología aplicada, naturaleza y tamaño del sistema de información y los tipos de lenguajes utilizados, bases de datos, etc.

3.7.3 Desarrollo de la solicitud

3.7.3.1 Planificación de tareas

- El Responsable de la BA debe rellenar el formulario de Solicitud de Mantenimiento, completando los distintos apartados según corresponda.
    - Planificación de tareas
    - Ejecución de las pruebas

3.7.3.2 Ejecución de las pruebas

- El Responsable de la BA debe rellenar el formulario de Solicitud de Mantenimiento, completando los distintos apartados según corresponda.

3.8 Flujo de procesos

El registro y gestión de las solicitudes, a través del Sistema de Solicitudes de Servicio, se realiza según el flujo representado en la Figura 1:

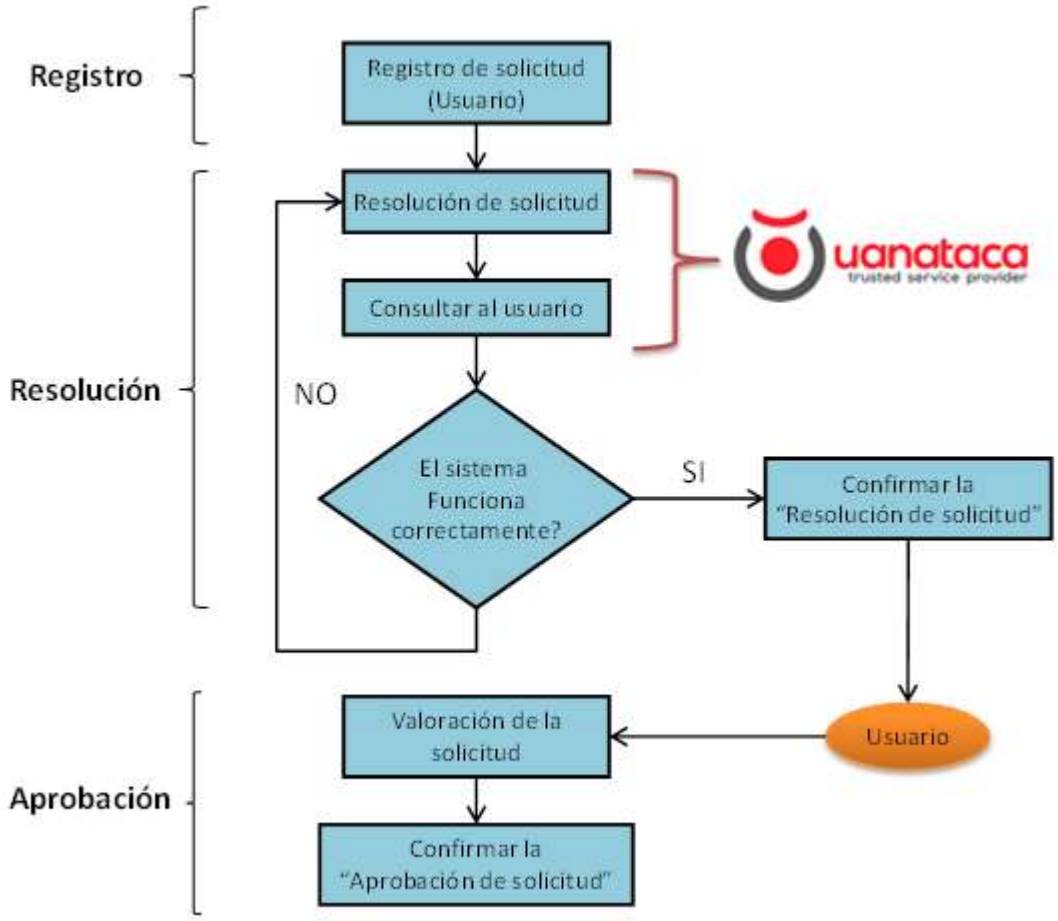


Figura 1: Flujo de registro y gestión de solicitudes.

3.9 Anexo

3.9.1 Requerimientos de seguridad

FASE DE REQUERIMIENTOS

- Control de la autenticación
- Control de roles y privilegios
- Requerimientos orientados a la evaluación del riesgo
- Aprobación de los privilegios por tipo de perfil
- Matriz control de roles y privilegios

FASE DE ANÁLISIS Y DISEÑO

- Acceso a componentes y administración del sistema
- Trazas para la auditoría
- Gestión de las sesiones
- Datos históricos
- Gestión adecuada de los errores
- Segregación, separación de Roles

FASE DE DESARROLLO E IMPLANTACACIÓN

- Securización del entorno de desarrollo
- Elaboración documentación técnica
- Desarrollo seguro (requerir estándar de seguridad)
- Seguridad en las comunicaciones
- Seguridad en paso al entorno de producción

DESARROLLO SEGURO

- Validación de entradas
- Codificación de las salidas
- Gestión de logs de los cambios
- Uso de criptografía
- Gestión de errores y logs
- Gestión de archivos
- Gestión de la memoria

- Estandarización y reutilización de funciones de seguridad

FASES DE PRUEBAS

- Control de calidad y de seguridad
- Inspección del código por fases
- Comprobación de la gestión de configuraciones
- Test de vulnerabilidades e intrusión

FASE DE MANTENIMIENTO

- Aseguramiento basado en gestión de riesgos
- Test de vulnerabilidades e intrusión después de cambios

3.9.2 Formulario de solicitud de mantenimiento de programas

Las solicitudes de mantenimiento de programas deberán realizarse cumplimentando la información requerida en el siguiente formulario:

1. Titulo

*Descripción corta (1 línea de la tarea)

2. Descripción sintética de la tarea

*Descripción exacta del desarrollo con imágenes orientativas (si aplicables) de donde tiene que quedar cualquier nueva opción.

3. Descripción detallada de la tarea

*Explicación detallada de las funcionalidades que se deberán garantizar como resultado del desarrollo en cuestión. Puede incluir diagramas de tipo UML para descripción de casos de uso o diagramas de sequencia.

4. Datos Técnicos

*Módulos, clases, tablas a editar o tener en cuenta.

4 Análisis de vulnerabilidades

4.1 Objetivo

Describir el procedimiento para el escaneo periódico del estado de la seguridad del sistema.

4.2 Descripción

N/A.

4.3 Ámbito de aplicación

Este procedimiento aplica a todos los servicios publicados en internet y que están a disposición de los clientes y de terceras partes.
Se describe cómo se realizará el proceso de escaneado de vulnerabilidades del sistema en aras de la mejora continua y con el objetivo garantizar la seguridad de los servicios prestados.

4.4 Documentación relacionada

N/A

4.5 Anexos

Se adjunta como anexo el documento de procedimiento y metodología a seguir para la realización del análisis de vulnerabilidades.

VULNERABILITY ASSESSMENT AND PENETRATION TESTING METHODOLOGY

1. INTRODUCTION

The goal of the Vulnerability Assement (VA) and Penetration Test (PT) is to identify vulnerabilities that can be exploited by attackers to violate an organization's privacy and security policies. This document outlines the methodology followed for VA and PT.

Each result is associated with a level of risk. The overall risk assessment for a particular vulnerability is determined considering the ease of exploitation, the impact on business and the likelihood of exploitation. The overall risk assessment is evaluated on a scale from zero to nine, a zero rating is associated with risk-free vulnerabilities, while a rating of nine is associated with critical-risk vulnerabilities.
The assessment shall be based on:

- **Ease of exploitation:** UNIVERSITÀ DEGLI STUDI DEL SANNIO determines this factor through the search for information or public code available in order to exploit the vulnerability to the particular operating system, service, network or application.
- **Business impact in case of exploits:** UNIVERSITÀ DEGLI STUDI DEL SANNIO determines this factor based on the knowledge of the sector, the type of access used and the sensitivity of the data to be compromised.
- **Probability of exploitation:** UNIVERSITÀ DEGLI STUDI DEL SANNIO determines this factor on the basis of additional knowledge of safety controls and associated risk and on the basis of the location of the system or application.

The risk scale of the UNIVERSITÀ DEGLI STUDI DEL SANNIO (defined on the basis of OWASP Risk Rating *Methodology) is illustrated below:*

Levels of probability and impact	
6 to 9	High
From 3 to 6	Medium
From 0 to 3	Low

Table 1 - OWASP Risk ratings

2. SCOPE

The security Consultants of UNIVERSITÀ DEGLI STUDI DEL SANNIO examine the web applications previously indicated through the use of test credentials provided by the CLIENT. Credentials are required to access the web application. After entering the authorized area of the site, the security Consultants provide analysis on the web application with zero knowledge of the application itself.

3. METHODOLOGY

The goal of VA and PT application is to protect confidential data by identifying weaknesses in application logic.

The activity includes an evaluation of the input validation checks on all data passed from the client to the web application. In addition to the input validation test, UNIVERSITÀ DEGLI STUDI DEL SANNIO performs a test of any inherent vulnerabilities of the technologies used (including the software used as Web Server).

In the activity carried out, UNIVERSITÀ DEGLI STUDI DEL SANNIO analyzes the authentication process, the session management functionalities and the presence of any vulnerabilities that allow access to parts of an application normally not allowed.

In order to simulate the threat from an attacker of the platform, tests were conducted using the login credentials of a user created ad-hoc for testing, having access to most of the application's features.

In particular, the methodology used follows the Penetration Testing Execution Standard (PTES) which recommends the following steps:

Information Gathering	It consists of collecting as much information as possible about the system being tested, with the aim of understanding its operation and use. This can be done through examples of lawful use of the platform, through inspection of the source codes of rendered pages, through analysis of client-server requests.
Threat Modeling	It consists of identifying the potential structural threats and vulnerabilities that could affect the system in question or the execution flow of the application (e.g. authentication process). This will provide an idea of which components are the most exposed to attacks and/ or that arouse more interest in a potential actor malevolent attacker.
Vulnerability Analysis	It consists of identifying the real vulnerabilities that afflict the system through automatic and manual tests. In this regard a number of computer tools including Nessus, Acunetix, Openvas, SPARTA, Nikto and Zed Attack Proxy will be used. Vulnerabilities detected through automatic tools will be tested manually to be sure of what is detected.
Exploitation	It consists of the exploitation of the vulnerabilities emerged during the previous phase, through scripts available fro m public sources or scripts forged for the specific goal. The successful exploitation of a vulnerability could allow unauthorized control of the system, access to sensitive information or sections of the application not normally accessible. In this phase some tools including Burpsuite and Metasploit will be used.
Post Exploitation	Once the system has been compromised, you can recover sensitive information stored in it, or use it as an entry point to the internal network with the aim of reaching the system on which the information you want to access occurs. In this phase techniques will be attempted that allow lateral movement to other sys tems connected to the same network of the compromised machine. For the correct execution of this phase, tools such as Mimikatz and other ad-hoc written programs will be used.
Reporting	It consists of the preparation of the report containing all the vulnerabilities found on the systems, the description of the scenarios that allowed an unauthorized access

	to the system and any sensitive information recovered and/or proof-of-concept of the compromise of the system.

Depending on the activity of Web Penetration Testing to realize, the Post-exploitation phase may not be carried out.

During the activity, the PTES standard (described above), formulated in a generic way to be applied to any type of system (infrastructural and application), has been reworked with respect to the concerned field a Web Application. It has also been applied paying particular attention to the major vulnerabilities surveyed by OWASP within the OWASP TOP 10 2017 ranking.

OWASP TOP 10 - 2017
A1: Injection
A2: Broken Authentication
A3: Sensitive Data Exposure
A4: XML External Entities (XXE)
A5: Broken Access Control
A6: Security Misconfiguration
A7: Cross-Site Scripting (XSS)
A8: Insecure Deserialization
A9: Using Components with Known Vulnerabilities
A10: Insufficient Logging & Monitoring

The identified vulnerabilities are classified according to the OWASP Top 10 2017 categories and assigned to each of them a level of risk referring to the CVSS v2 framework defined by the NIST (https://nvd.nist.gov/vuln-metrics/cvss).

4. Tools used

In order to perform the task properly, we use a toolchain of Web Application Testing that includes the use of the most established tools in the industry for each point of the chain to follow the methodology of Penetration Testing PTES:

1. Tool for Information Gathering

The following tools will be used to perform the tests of Information Gathering:

1. **TheHarvester**

TheHarvester is a tool to gather searchable information about your goals that could be anything from private information to websites to companies. TheHarvester can find email accounts, subdomain names, virtual hosts, open ports and banners and employee names from different public sources.

2. **Whatweb**

Whatweb is a tool that aims to extrapolate more technical information about one or more hosts. Technical information means all data relevant to a proper analysis of IT Security such as the server version Apache, PHP, CMS.

3. **Wafw00f**

Wafw00f is a tool that allows the detection and analysis of which Web Application Firewall is installed to protect the application from malicious users. It is an active reconnaissance tool because it connects to the web server, first it starts with a normal HTTP response and intensifies if necessary. You can replace or include personal headers, have SOCKS and HTTP proxy support and detect an entire group of WAF products from hosted solutions like Cloudflare and Incapsula to server side solutions like Modsecurity.

4. **Dnsmap**

Dnsmap is mainly thought to be used by the pentesters during the information gathering phase and enumeration of corporate resources. During enumeration phase, the security consultant typically discovers the company's netblocks IP target, domain names, phone numbers, etc...

2. **Threat modelling**

As for the threat modeling section, no tools are reported, as it is a conceptual phase of the Penetration Testing process. In this phase we study what is the best attack plan to best test the web application.

3. Vulnerability analysis tools

1. Nessus

Nessus is an automatic system vulnerability scan tool. The system can be of any type, from infrastructure services to web applications. Nessus is the global standard for preventing network attacks, identifying vulnerabilities,
and detecting configuration problems that hackers use to enter the network.
The Nessus vulnerability scanner is used by over a million users worldwide, making it the world leader in vulnerability assessments, security configuration, and security compliance. Its plug-in configuration makes it modular in the choice of components to use during security analysis, choosing the optimal configuration for each scan.
In fact, there is also a plug-in to scan web application testing.

2. Acunetix

Acunetix Web Application Scanner is one of the most established web application security scan tools. The advantage of this tool over competitors is the fact of using a very convenient graphical interface even for not vulnerability management experts. The automatic scanning tool is able to emulate attacks against web applications from SQL Injection, to cross-site scripting.

3. Nikto

Nikto is an open source software used in web application penetration testing. It includes a database of files and configurations notoriously vulnerable to various types of attacks.
The strong point of this software is the technology in the detection of errors: not all servers in fact comply with the RFC standard in the sending of the answer (200 – OK o 404 – Not Found), but they could respond with a
200 when the requested resource does not exist or is in the state “Forbidden”. Nikto checks the answers through checksum and reading the text in the answer, thus minimizing false positives.
Another strong point is the evasion technology: it includes methods to circumvent the controls applied by WAF or IPS.

4. Webscarab

Webscarab is a framework for web application analysis. Webscarab has several operating modes, implemented by numerous

plugins. In its most common use, Webscarab operates as a proxy, allowing the attacker to review and modify browser-created requests before they are sent to the server, and to review and modify the responses returned by the server before they are received by the browser. Webscarab is able to intercept both HTTP and HTTPS communication.

5. **DirBuster**

DirBuster is a tool for brute-forcing directories and filenames on web applications. It, through the use of 9 different lists to search hidden directories and files, manages to enumerate the hidden paths of the web application. This allows a security analyst to be able to obtain sensitive information or information that should have been properly hidden by the developer to an attacker.

6. **Skipfish**

Skipfish allows to analyze a website starting from a sitemap generated interactively through brute force methods to search for as many nodes as possible; follows a recursive analysis that can refer to a local database to further optimize scanning times.
The tests that can be completed are classified according to the following five attack levels:

- High risk, can lead to system compromise: SQL injection, XML/Xpath injection, shell command injection;
- Medium risk, may involve compromise of the data: directory transversal, XSS and CSS inclusion vectors, missing MIME types;
- Low risk, limited impact: Self-signed SSL certificates, External untrusted embedded content, HTTP credentials in Urls;
- Internal problems: Failed 404 behavior checks, IPS filtering detected;

7. **Zed Attack Proxy**

OWASP Zed Attack Proxy (ZAP) is an integrated tool dedicated to penetration testing that allows you to detect vulnerabilities in applications and websites. It is a simple and flexible solution that adapts to all levels of use and expertise.

4. **Exploitation**
1. **Burp Suite**

Burp Suite is a software designed to analyze the network traffic generated by web-based applications. Used to examine and manipulate the flow of data from the web browser to the reference server, such as an Internet site, is structured in a series of tools that allow the user to perform different Penetration Tests and attacks of various kinds.
It consists primarily of these components:

- Proxy: allows to inspect and manipulate the network traffic between the web browser and the application to be analyzed.
- Spider: indexes the content of the site, generating a list of Urls to check for security flaws.
- Scanner: this tool, which is only available with the "Professional" version, performs an automated scan for publicly known vulnerabilities.
- Intruder: useful to generate custom attacks, so no exploits from the Burp Suite database.
- Repeater: records every request made between client and server and allows to reproduce it later simulating the identity of the issuer.
- Sequencer: allows to verify the "randomness" of the data, for example the session IDs, the tokens used for resetting passwords, etc.

2. Metasploit Framework

Metasploit Framework is an open source tool for developing and running exploits against a remote machine.
The basic steps for exploiting a system using Metasploit Framework include:

- Choose and configure an exploit (code that penetrates a system by exploiting bugs; about 900 different exploits are included for Windows, Unix/Linux and Mac OS X).
- Check if the attacked system is susceptible to the selected exploit (optional).
- Choose and configure a payload (code to run on the attacked system after we have successfully penetrated; for example a remote shell or a VNC server).
- Choose the encoding technique so that the intrusion prevention system (IPS) ignores the encoded payload.
- Run the exploit.

This modular approach (which allows the combination of any exploit code with any playload) is the main advantage of Metasploit Framework.
In addition, it facilitates the operations of attackers and those who write exploits and payloads.

3. Sqlmap

Sqlmap is an open source software for penetration testing, which allows you to automate the detection and exploitation of defects in SQL injection and thus take control of DBMS servers. The software works on the command line. It supports multiple DBMS such as Mysql, Oracle, Postgresql, Microsoft SQL Server, Microsoft Access, IBM DB2, Sqlite, Firebird, Sybase, SAP Maxdb, HSQLDB and Informix. It also supports 6 different types of SQL njection. It can recognize the hash format of passwords and launch dictionary attacks on them.

4.6 Responsabilidad

Se identifican las siguientes responsabilidades en la aplicación de este procedimiento:

- Responsable de la auditoría:
  - Se encargará de establecer el calendario de análisis de vulnerabilidades y de su cumplimiento. Dará fe de la correcta ejecución y emitirá el informe del resultado.
  - Velará para que las medidas correctoras sean llevadas a cabo por el equipo técnico.

- Responsable de la infraestructura:
  - Se encargará de que se lleve a cabo los análisis de vulnerabilidades definidos por el responsable de auditoria en las fechas indicadas.
  - Será responsable de aplicar las medidas correctoras que minimicen el riesgo según el informe remitido por el Responsable de Auditoria.

Es responsabilidad del departamento de Infraestructuras mantener vigente este documento, así como su aplicación.

4.7 Procedimiento

En primera instancia se definen los objetivos por lo que se realizará el análisis de vulnerabilidades conjuntamente entre el Auditor, el departamento Legal y de Infraestructuras.

Una vez realizada la toma de requerimientos y datos para llevar a cabo el análisis se llevará a cabo según se describe a continuación.

Los análisis de vulnerabilidades se realizan siguiendo un procedimiento determinado que atiende a un estándar referente a la ejecución de tests de intrusión conocido como PTES (*Penetration Testing Execution Standard*).

El análisis de vulnerabilidad se encuentra en el punto tercero del estándar, determinado como "*Vulnerability Analysis".* Dicho apartado se traduce en la necesidad de identificar las vulnerabilidades reales que pueden afectar al sistema a través del uso de una serie de tests manuales y automáticos.

Dichos tests manuales y automáticos comprenden una serie de herramientas que serán descritas a continuación:

1. Nessus: Esta herramienta automática de escaneo de vulnerabilidades se utiliza en el sistema eligiendo la configuración óptima para poder realizar el mismo.

2. Acunetix: La herramienta de escaneo web de Acunetix permite realizar la emulación de ataques a aplicaciones web a través de distintos protocolos como *SQL Injection* o a través del uso de *scripts*. Además de ello, posee una interfaz gráfica que permite su uso de manera sencilla y efectiva para el supuesto que nos ocupa.

3. Nikto: Es una herramienta de código abierto que utiliza una base de datos con configuraciones determinadas para realizar la detección de errores en el acceso a recursos. Además de ello, la tecnología incluida en dicha aplicación incluye métodos para circunvalar protecciones o controles aplicados en web.

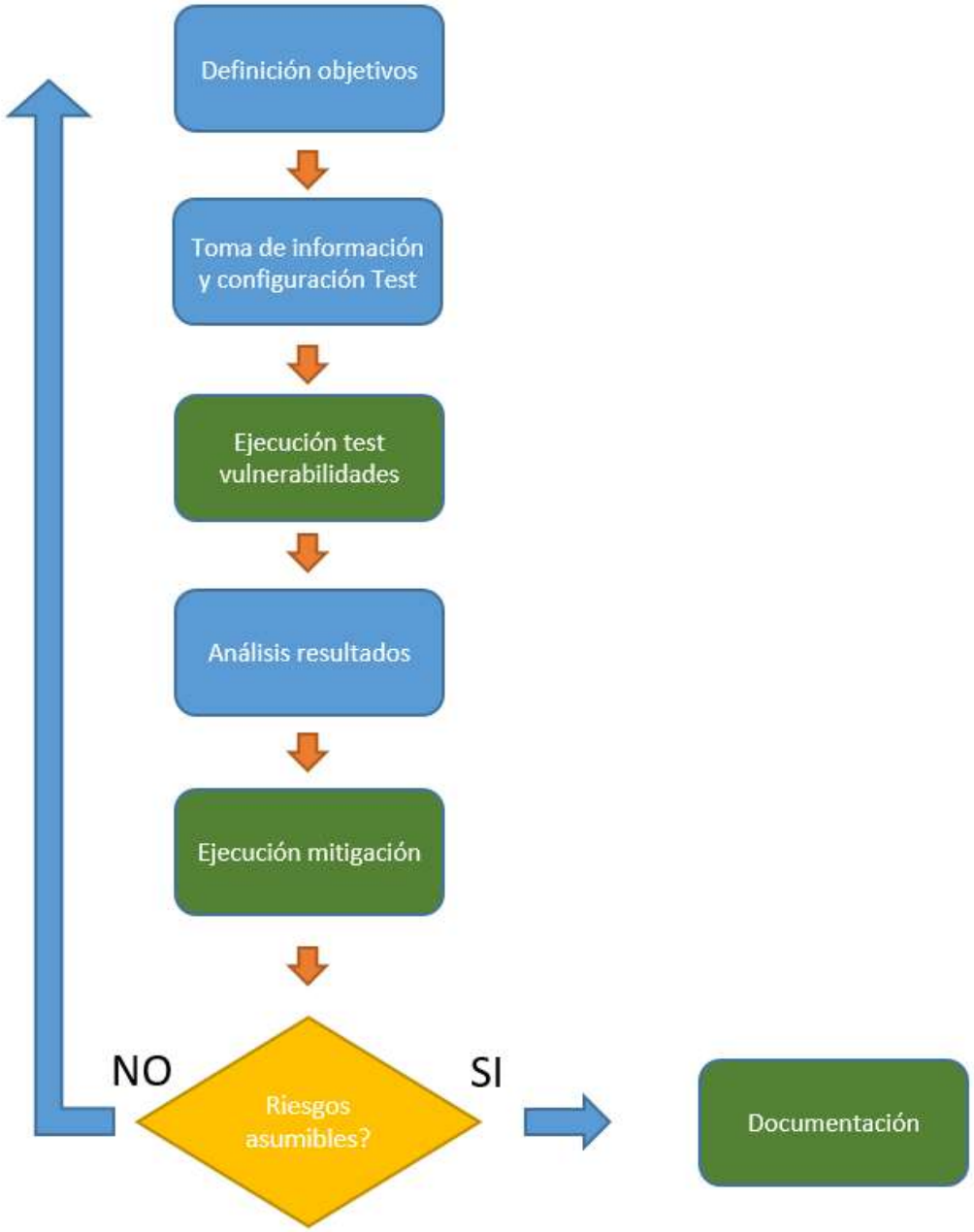
4. Zed Attack Proxy: Esta herramienta integrada y dedicada al test de intrusión permite analizar y descubrir vulnerabilidades en aplicaciones y sitios web, siendo muy flexible respecto de los usos de la herramienta, adaptándose a toda circunstancia y nivel de infraestructura y de experiencia de uso.

4.7.1 Análisis de resultados

Con el resultado del análisis y con las vulnerabilidades reales identificadas, se excluirán los falsos positivos, se llevarán a cabo las acciones necesarias para su mitigación. Se actualizará o configurará adecuadamente el elemento identificado.

Posteriormente se valoran aquellos que por su complejidad o falta de solución en el momento no puedan ser corregidos. En caso de aceptar el riesgo éste será documento e identificado.

4.8 Flujo de procesos



4.9 Documentación relacionada

N/A

5 Test de intrusión

5.1 Objetivo

El objetivo principal del test de intrusión es proporcionar un esquema de informes estándares basados en una metodología científica para la caracterización precisa de la seguridad mediante el examen y la correlación de evidencias de manera coherente y fiable.

Los resultados de la prueba de penetración serán notificados con el detalle de las pruebas de penetración para que puedan evaluar los impactos potenciales a la organización sugiriendo las medidas para reducir los riesgos.

Antes del inicio de la prestación de los servicios se realiza un test de intrusión y sus correspondientes correcciones para garantizar el nivel óptimo de seguridad.

En lo sucesivo, siempre que se vaya a proceder a un cambio significativo en los elementos que prestan el servicio se realizará un test de intrusión con la finalidad de garantizar que el nivel de seguridad se mantiene o es mejorado.

5.2 Descripción

La prueba de penetración es valiosa por varias razones:

1. Determinar la posibilidad de éxito de un ataque.
2. Identificación de vulnerabilidades de alto riesgo que resultan de una combinación de vulnerabilidades de menor riesgo explotadas en una secuencia particular.
3. Identificación de vulnerabilidades que pueden ser difíciles o imposibles de detectar con red automatizada o un software de análisis de vulnerabilidades.
4. Comprobar la capacidad de las medidas de defensa de los servicios para detectar con éxito y responder a los ataques.

En nuestro caso seguiremos lo dispuesto en el estándar PTES respecto de la realización del test de intrusión, acorde a lo dispuesto en el apartado cuarto titulado "*Exploitation".*
Atendiendo a esta fase, debe entenderse que consiste en la explotación de las vulnerabilidades que hayan surgido, de tal manera que se usan scripts públicos o privados para la realización de este. Este procedimiento queda dotado de una gran importancia, ya que si el sistema queda expuesto en el test de intrusión se puede obtener acceso al sistema, acceso a datos sensibles o a secciones privadas y que no pueden ser accedidas por el usuario.

Para tal fin, se utilizarán las siguientes herramientas:

1. Burp Suite: Burp Suite es un software diseñado para analizar el tráfico de la red generado por las aplicaciones web, de tal manera que dicho software se usa para examinar y manipular el flujo de datos que se emiten desde el navegador hasta el servidor en el que se aloja el recurso a visitar (p.ej una página web). Dicha herramienta se estructura de tal manera que tiene una serie de herramientas que permiten la realización de tests de intrusión y ataques de varios tipos.

El contenido del software incluye:

- Un Proxy: Que permite inspeccionar y manipular el tráfico de red entre el navegador y el aplicativo web a analizar.
- "*Spider*": Consiste en la indexación del lugar objetivo, generando una lista de URL para realizar las verificaciones oportunas, buscando fallas de seguridad.
- Escáner: Realiza un escáner automatizado respecto de vulnerabilidades públicamente conocidas, observando si el sistema objetivo es vulnerable a alguna de estas.
- Intruso: Genera el ataque de manera personalizada, sin atender a lo indicado en la base de datos de la aplicación.
- Repetidor: Realiza la grabación y guardado de cada petición realizada entre el cliente y el servidor y permite su reproducción simulando la identidad del ejecutante.
- Secuenciador: Permite verificar la aleatoriedad de los datos, como por ejemplo las identificaciones de sesión, los tokens utilizados para realizar el reinicio de contraseñas entre otros supuestos.

2. Metasploit Framework

Es una aplicación de código abierto que permite el desarrollo y el uso de vulnerabilidades contra una maquina remota. La metodología de uso básica de este aplicativo se basa en:

- Realizar la elección o configuración de una vulnerabilidad.
- Observar si el sistema atacado es vulnerable a dicha vulnerabilidad.
- Elegir y configurar código a ejecutar en el sistema atacado si se realiza la inserción en el sistema de forma satisfactoria.
- Elegir la encriptación para prevenir que el sistema de prevención de intrusiones detecte el código a ejecutar.
- Ejecutar el código en el sistema atacado.

3. Sqlmap: Es una aplicación de código abierto utilizada para tests de intrusión que permite automatizar la detección y explotación de defectos en la inyección SQL y tomar control de servidores manejadores de bases de datos (DBMS).

5.3 Anexos

Debe observarse el anexo incluido en el apartado 4.5 ya que el mismo incluye previsiones y metodología de uso.

6 Monitorización de eventos y análisis activa de los logs

6.1 Objetivo

El objetivo principal de un sistema de monitorización de eventos es detectar de manera proactiva eventos relacionados a la disponibilidad y utilización de los servicios, prevenir paradas de funciones provenientes de situaciones de carga superiores a las que el sistema puede sostener y detectar eventos potencialmente peligrosos analizando los logs del sistema.

Con esta finalidad, se han dotado herramientas y procedimientos automatizados.

6.2 Descripción

Para la monitorización se utiliza el servicio externo *Datadog*[1].

Se ha optado por un servicio externo en vez de una solución desarrollada internamente con el fin de focalizar las actividades del laboratorio en el desarrollo de componentes del Prestador de Servicios de Confianza. Además, con el fin de probar la accesibilidad de un servicio desde diferentes áreas geográficas, el servicio objeto de monitorización y el servicio que monitorea deben de estar necesariamente en ubicaciones distintas.

Adicionalmente, un servicio de monitorización proporcionado por una empresa líder y con años de experiencia, dispondrá de funcionalidades y fiabilidad difícilmente superables por una solución realizada “in-house”, a menos que se emplee gran esfuerzo y dedicación no compatibles con el objetivo principal de un Prestador de Servicios de Certificación.

Los motivos por los cuales ha sido elegido *Datadog* son:

[1] https://www.datadoghq.com

- El agente de Datadog ha sido desarrollado en el lenguaje de programación Python, con el cual los desarrolladores tienen gran experiencia y conocimientos profundos.
- El servicio tiene requisitos de seguridad/fiabilidad alineados con nuestras exigencias. Es un servicio de monitorización servidor, también utilizado ampliamente por grandes empresas y servicios web de gran difusión como Citrix, Spotify, Electronic Arts, The New York Times, The Washington Post, imgur, Slashdot y otros.
- La empresa que ofrece el servicio está consolidada. Figura en la lista "Cloud 100" de Forbes (que contempla las cien principales empresas que suministran servicios en la nube)2 y entre las mejores diez empresas de mayor crecimiento en Norteamérica en el 2016 según Deloitte[3].

A través de un panel de control intuitivo, el servicio permite la monitorización continua de los principales recursos disponibles de los servidores de producción, recuperación de desastres y test (ej. espacio de los discos, memoria RAM libre, procesadores, etc.). Permite verificar el correcto funcionamiento de todos los servicios en ejecución sobre los servidores, a través de métricas específicas. A través de dicho panel de control, se puede recuperar información relativa al correcto funcionamiento, como por ejemplo número de certificados emitidos en unidad de tiempo, número de usuarios activos en la autoridad de registro y solicitudes por segundo procesadas desde el interfaz API.

Toda esta información se restituye al usuario del servicio en la forma deseada. De hecho, es posible construir visualmente el panel de control introduciendo diagramas y gráficos en el orden y tipo deseado, con el objetivo de mejorar la legibilidad y priorizar aquellas de mayor importancia.

Se ha utilizado también el componente de alertas del servicio *Datadog*. Esta permite al personal operativo recibir las comunicaciones oportunas respecto a los parámetros de funcionamiento que no hayan sido definidos en un rango de elegibilidad. La comunicación se produce a través de email y por SMS en los casos de mayor gravedad.

---

[2] http://www.forbes.com/sites/alexkonrad/2016/09/07/the-cloud-100-the-next-leaders-in-cloud-computing/#6666d5b56669

[3] https://www2.deloitte.com/content/dam/Deloitte/us/Documents/technology-media-telecommunications/us-tmt-2016-fast-500-winners-by-rank.pdf

La comunicación vía SMS está disponible a través de la integración del servicio *Datadog* con un servicio de SMS Gateway *Skebby*[4]. Además, con el fin de garantizar que el personal operativo no ignore inadvertidamente las comunicaciones, se instala la aplicación *FireAlert2* [5] en el smartphone. Esta aplicación de móvil reacciona a SMS con palabras clave específicas en su interior, haciendo que el smartphone emita un sonido continuo, (aún si el smartphone está silenciado o el volumen al mínimo), hasta cuando el operador no intervenga manualmente para desactivarla. Esta aplicación la utilizan habitualmente bomberos, servicios médicos de emergencia, guardia costera, cruz roja, etc, con el objetivo de no pasar por alto comunicaciones de extrema importancia.

A continuación se muestra la lista de los servidores monitorizados con las métricas activadas y los límites para el envío de email (*severity=WARNING*) o email+SMS (*severity=ERROR*).

Nome server	Nome metrica	Limite WARNING	Limite ERROR
SERVICES1, SERVICES2	disk free	<20%	<10%
	ram free	<10%	<3%
	running services	/	<4
CA1, CA2	disk free	<20%	<10%
	ram free	<10%	<3%
	running services	/	<2
HSM1, HSM2	disk free	<20%	<10%
	ram free	<10%	<3%
	running services	/	<1
DB1, DB2	disk free	<30%	<20%
	ram free	<20%	<6%
	running services	/	<1
LOG1, LOG2	disk free	<30%	<20%
	ram free	<10%	<3%
	running services	/	<1

[4] https://www.skebby.it

[5] https://play.google.com/store/apps/details?id=de.hoernchen.android.firealert2

DISASTER RECOVERY	disk free	<30%	<20%
	ram free	<20%	<6%
	running services	/	<5

Además, para detectar eventos potencialmente peligrosos cuales:

1. Accesos al sistema de Registration Authority (RA)
2. Accesos al sistema de Validation Authority (VA)
3. Accesos de tipología SSH
4. Accesos a las subCA
5. Perdida de calibración de los relojes de los sistemas y/o sus desviaciones
6. Actualización de los relojes de los sistemas

Se ha previsto utilizar la herramienta sobre descrita DataDog en combinación con SysLog, para centralizar todos los logs y analizarlos. Para el análisis, los eventos potencialmente peligrosos descritos arriba han sido modelados como "metrics", para que sean automáticamente detectados por DataDog generando las alertas al personal responsable.

6.3 Evaluación de requisitos de capacidad

Se han programado evaluaciones con respecto de los requisitos de capacidad de la infraestructura con el fin de garantizar la potencia adecuada de procesado y disponibilidad del almacenamiento en la Prestación de Servicios de Confianza.

Los análisis del nivel de carga se realizarán de manera trimestral. (Enero, abril, julio y octubre).

En el supuesto que en el análisis se evidencia un nivel de carga que supere el 90% del nivel de carga actual soportado por la infraestructura, se adoptarán las medidas que sean necesarias para adaptar la infraestructura con el fin de soportar dichas capacidades.

6.4 Anexos

N/A.

7 Procedimiento para la inicialización de la CA Root y de las CA Sobordinadas

7.1 Descripción del procedimiento

1.- La inicialización del HSM que contiene el componente funcional de la Autoridad de Certificación Raíz, se realiza de acuerdo al procedimiento y protocolos que se detallan en la presente sección.

1. Se implementa una política de segregación de funciones Se implantará una política de tres (3) tarjetas ACS de un total de seis (6) para poder administrar el HSM.
2. Se implantará una política de una (1) tarjeta OCS de un total de seis (6) para operar el HSM.
3. Entrega y aceptación mediante firma de las tarjetas ACS y OCS a los correspondientes Claveros y ensobrado seguro de los PINs de las tarjetas.
4. Creación del par de claves y generación del certificado según se describe en la sección pertinente de "PROCEDIMIENTOS TÉCNICOS".

Claveros ACS CA raíz (ACS-R)
Claveros ACS CAs subordinadas (ACS-S)
Claveros OCS CA raíz (OCS-R)
Claveros OCS CAs subordinadas (OCS-S)

2.- Autoridad de Certificación de Información Subordinada 01 (online):

ACS-R	ACS-S	OCS-R	OCS-S
	x	x	x

a) Inicialización del HSM, según se describe en la sección pertinente de "PROCEDIMIENTOS TÉCNICOS".
   i. Se implantará una política de tres (3) tarjetas ACS de un total de seis (6) para poder administrar el HSM.
   ii. Se implantará una política de una (1) tarjeta OCS de un total de seis (6) para operar el HSM.

b) Entrega y aceptación mediante firma de las tarjetas ACS y OCS a los correspondientes Claveros y ensobrado seguro de los PINs de las tarjetas.
c) Creación del par de claves y generación del certificado según se describe en la sección pertinente de “PROCEDIMIENTOS TÉCNICOS”.
d) Realización de prueba de correcto funcionamiento del certificado de CA Subordinada emitiendo una CRL.

3.- Autoridad de Certificación de Información Subordinada 02 (online):

ACS-R	ACS-S	OCS-R	OCS-S
	x	x	x

a) Inicialización del HSM, según se describe en la sección pertinente de “PROCEDIMIENTOS TÉCNICOS”.
    i. Se implantará una política de tres (3) tarjetas ACS de un total de seis (6) para poder administrar el HSM.
    ii. Se implantará una política de una (1) tarjeta OCS de un total de seis (6) para operar el HSM.
b) Entrega y aceptación mediante firma de las tarjetas ACS y OCS a los correspondientes Claveros y ensobrado seguro de los PINs de las tarjetas.
c) Creación del par de claves y generación del certificado según se describe en la sección pertinente de “PROCEDIMIENTOS TÉCNICOS”.
d) Realización de prueba de correcto funcionamiento del certificado de CA Subordinada emitiendo una CRL.

7.2 PROCEDIMIENTOS TÉCNICOS

7.2.1 Autoridad de Certificación de Información Raíz

7.2.1.1 Inicialización del HSM

Creación de Security World y ACS:

1. Acceder al servidor con la cuenta “caoperator” (ver Prerrequisitos).
2. Abrir una ventana de terminal.

3. Ejecutar el siguiente comando para crear un nuevo Security World con la configuración escogida de ACS:

   > hsm world root

4. Seguir las instrucciones en pantalla (se indicará al usuario que coloque el switch del HSM en la posición I.)
5. Aparecerá el siguiente mensaje:

   Create Security World:
   Module 1: 0 cards of 6 written
   Module 1 slot 0: empty
   Module 1 slot 0: unknown card

6. Para cada tarjeta del ACS, anotar su número de serie, su clavero y seguir los siguientes pasos:

   Insert/change card in module (or change module mode)
   Module 1 slot 0: Enter new passphrase:

7. Insertar el PIN en la tarjeta inteligente:

   Module 1 slot 0: Confirm new passphrase:
   Retype the PIN for the smartcard
   Module #1 Slot #0: Processing ...

8. Al final de la creación de ACS se mostrará el siguiente mensaje:

   Card writing complete.
   security world generated on module #0; hknso = [security officer key hash]

   NOTA: no es necesario apuntar este hash, ya que se puede obtener de nuevo mediante el comando: > /opt/nfast/bin/nfkminfo

Creación del OCS:

1. Acceder al servidor con la cuenta “caoperator” (ver Prerrequisitos).
2. Abrir una ventana de terminal.
3. Ejecutar el siguiente comando para crear un nuevo OCS con la configuración escogida:

```
> hsm ocs root
```

4. Seguir las instrucciones en pantalla (se indicará al usuario que coloque el switch del HSM en la posición O.)
5. En este momento, insertar una tarjeta del ACS, para aprobar la operación:

```
FIPS insert ACS/OCS:
 Module 1: 0 cards read
 Module 1 slot 0: empty
Insert/change card in module 1 (or change module mode)
```

6. Extraer la tarjeta ACS cuando aparezca el siguiente mensaje:

```
FIPS 140-2 level 3 auth obtained
```

7. Para cada tarjeta del OCS, anotar su número de serie, su clavero y seguir los siguientes pasos:

```
Insert/change card in module (or change module mode)
Module 1 slot 0: Enter new passphrase:
```

8. Insertar el PIN en la tarjeta inteligente:

```
Module 1 slot 0: Confirm new passphrase:
Retype the PIN for the smartcard
Module #1 Slot #0: Processing ...
```

9. Al final de la creación de OCS se mostrará el siguiente mensaje:

```
cardset created; hkltu=[hash token user key]
```

NOTA: no es necesario apuntar este hash, ya que se puede obtener de nuevo mediante el comando: > /opt/nfast/bin/nfkminfo

7.2.1.2 Generación de claves y certificados

1. Insertar una tarjeta OCS en el lector de tarjetas Thales.
2. Hacer login en el servidor de la CA raíz con la cuenta de caoperator.
3. Abrir una ventana de terminal.
4. Ejecutar el siguiente comando para crear una nueva clave de aplicación para la aplicación de PKI y vincularla al PKCS#11 de EJBCA, usando el OCS rootca:

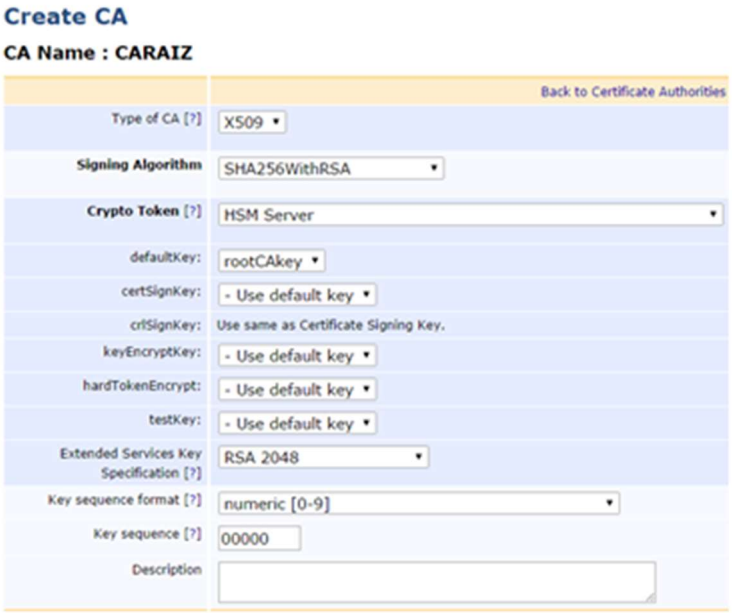
   > ca appkey rootca

5. Se solicitará que se introduzca el PIN de la tarjeta OCS insertada en el lector Thales.
6. Usando un navegador, conectarse a la aplicación de PKI en la siguiente dirección: <<ADDRESS_OF_CA>>/ejbca/adminweb/
7. Hacer login usando la tarjeta Bit4ID de Superadministrador de CA creada como prerrequisito, introduciéndola en el lector conectado al puerto usb estándar.
8. Ir a CA Functions> Crypto Tokens> Create new.
   a. Nombre: HSM Server
   b. Tipo: PKCS11
   c. Código de autenticación: introducir el PIN de la tarjeta OCS presente en el HSM
   d. Repetir código de autenticación: introducir el PIN de la tarjeta OCS presente en el HSM
   e. Auto-activación: No
9. El nuevo crypto-token creado no contiene ningún par de claves. **A continuación, se creará el par de claves correspondientes a la CA raíz**. Definir los siguientes parámetros tal y como se indica a continuación:
   a. Alias: rootCAkey
   b. Key size: RSA 4096
10. Pulsar el botón Generate new key pair

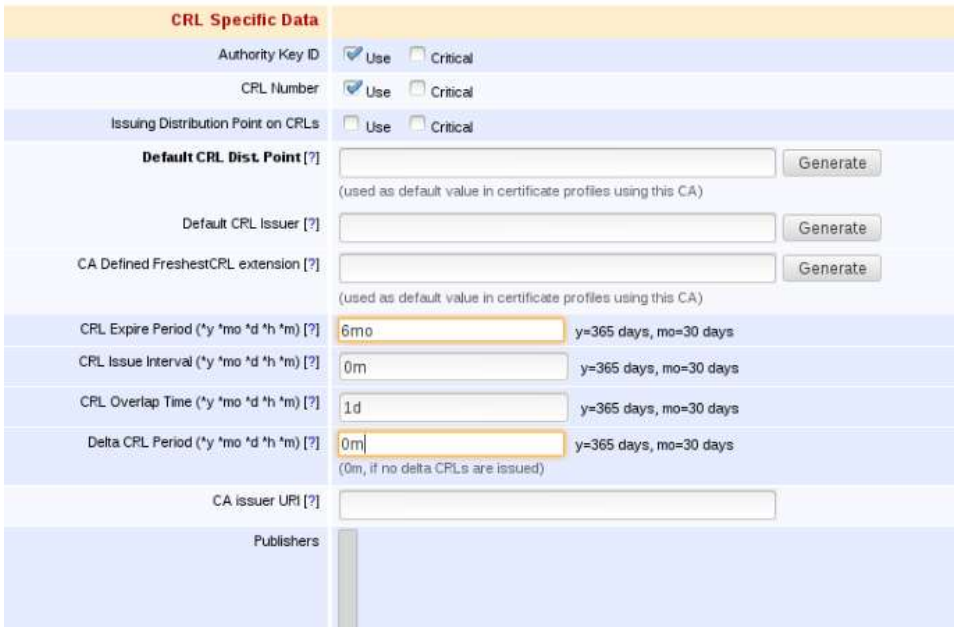
11. El nuevo par de claves aparecerá al final de la pantalla. Presionar en Test para comprobar que se ha generado correctamente. Debe aparecer el mensaje rootCAkey tested successfully.

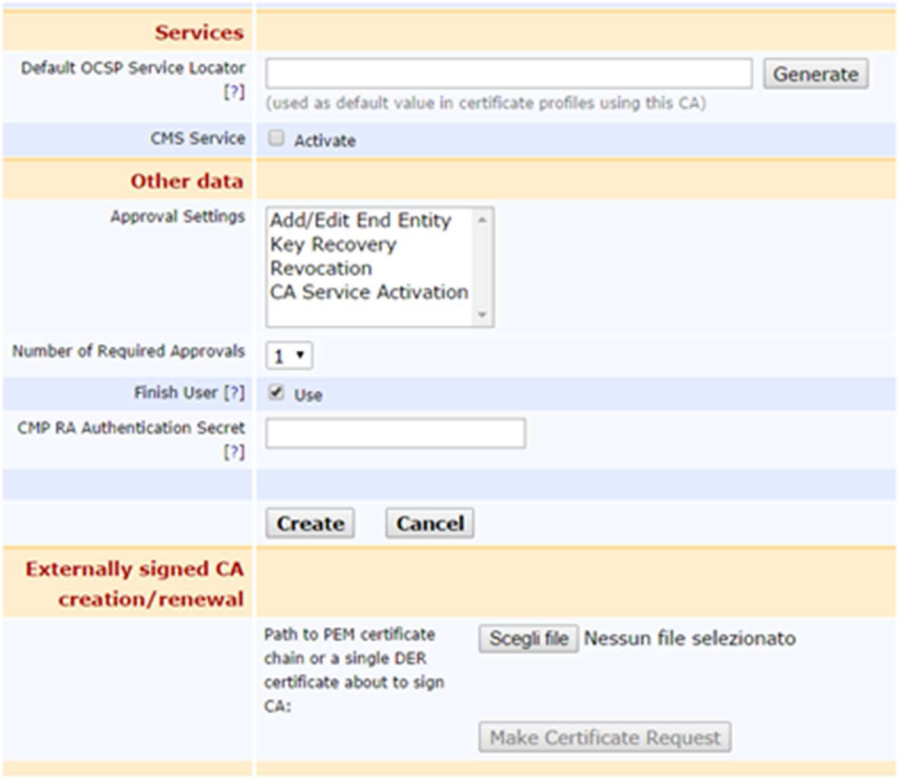
12. Ir a CA Functions> Certification Authorities> Add CA, insertando como nombre de la CA: CARAIZ

13. Rellenar los datos de solicitud de certificado tal y como muestran las siguientes imágenes:



NOTA: usar en el campo Subject DN el definido en el perfil del certificado.





14. Presionar el botón Create para finalizar el proceso de creación de la CA raíz.
15. Visualizar el certificado y verificar que contiene los datos correctos según el perfil establecido en la sección “Perfil de los certificados electrónicos de CA a emitir”.
16. Anotar el *hash* SHA-1 (huella digital) del certificado en la sección “RESULTADO”.
17. Descargar el certificado y salvarlo en un pendrive, que posteriormente servirá para importarlo en la CA subordinada:
    a. Ir a CA Functions> CA Structure & CRLs
    b. Presionar en Download PEM file
18. Seguir los siguientes pasos para crear la End Entity (Autoridad final para la CA subordinada):
    a. ir a RA Functions> Add End Entity
    b. Elegir el perfil de Autoridad final: eeCaSubordinada
    c. No cambiar los valores por defecto

d. Crear dos End Entity con usarios respectivamente “SUBCA1” y “SUBCA2”

e. Elegir para cada usuario la nueva contraseña usando los campos Password (or Enrollment Code) y Confirm Password

Con esto finaliza la creación de claves y generación de certificado de CA raíz.

7.2.2 Autoridad de Certificación de Información Subordinada

7.2.2.1 Inicialización del HSM

Creación de Security World y ACS:

1. Ejecutar el siguiente comando para crear un nuevo Security World con la configuración escogida de ACS:

   > hsm world sub

2. Seguir las instrucciones en pantalla (se indicará al usuario que coloque el switch del HSM en la posición I.)

3. Aparecerá el siguiente mensaje:

   Create Security World:
   Module 1: 0 cards of 6 written
   Module 1 slot 0: empty
   Module 1 slot 0: unknown card

4. Para cada tarjeta del ACS, anotar su número de serie, su clavero y seguir los siguientes pasos:

   Insert/change card in module (or change module mode)
   Module 1 slot 0: Enter new passphrase:

5. Insertar el PIN en la tarjeta inteligente:

   Module 1 slot 0: Confirm new passphrase:

Retype the PIN for the smartcard
Module #1 Slot #0: Processing ...

6. Al final de la creación de ACS se mostrará el siguiente mensaje:

Card writing complete.
security world generated on module #0; hknso = [security officer key hash]

NOTA: no es necesario apuntar este hash, ya que se puede obtener de nuevo mediante el comando: > /opt/nfast/bin/nfkminfo

Creación del OCS:

1. Ejecutar el siguiente comando para crear un nuevo OCS con la configuración escogida:

> hsm ocs sub

2. Seguir las instrucciones en pantalla (se indicará al usuario que coloque el switch del HSM en la posición O.)

3. En este momento, insertar una tarjeta del ACS, para aprobar la operación:

FIPS insert ACS/OCS:
Module 1: 0 cards read
Module 1 slot 0: empty
Insert/change card in module 1 (or change module mode)

4. Extraer la tarjeta ACS cuando aparezca el siguiente mensaje:

FIPS 140-2 level 3 auth obtained

5. Para cada tarjeta del OCS, anotar su número de serie, su clavero y seguir los siguientes pasos:

Insert/change card in module (or change module mode)
Module 1 slot 0: Enter new passphrase:

6. Insertar el PIN en la tarjeta inteligente:

```
Module 1 slot 0: Confirm new passphrase:
Retype the PIN for the smartcard
Module #1 Slot #0: Processing ...
```

7. Al final de la creación de OCS se mostrará el siguiente mensaje:

```
cardset created; hkltu=[hash token user key]
```

NOTA: no es necesario apuntar este hash, ya que se puede obtener de nuevo mediante el comando: `> /opt/nfast/bin/nfkminfo`

NOTA: Recordar que se debe dejar insertada una tarjeta OCS para poder operar.

7.2.2.2 Generación de claves y certificados

7.2.2.2.1 Creación del Crypto Token

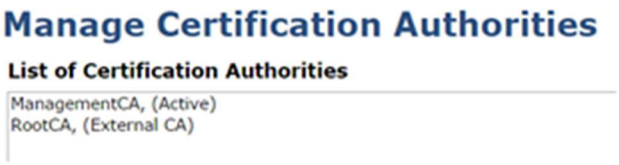
Nota: Este procedimiento es único para crear cualquier número de Autoridades de Certificación Subordinadas dentro de la presente jerarquía. Por lo cual, no deberá repetirse para la creación de CAs Subordinadas adicionales.

1. Insertar una tarjeta OCS en el lector de tarjetas ubicado en el panel frontal del rack.
2. Hacer login en el servidor de la CA subordinada.
3. Abrir una ventana de terminal.
4. Ejecutar el siguiente comando para crear una nueva clave de aplicación para la aplicación de PKI y vincularla al PKCS#11 de EJBCA, usando el OCS subca:

```
> ca appkey subca
```

5. Se solicitará que se introduzca el PIN de la tarjeta OCS insertada.
6. Usando un navegador, conectarse a la aplicación de PKI en la siguiente dirección: <<ADDRESS_OF_CA>>/ejbca/adminweb/
7. Hacer login usando la tarjeta Bit4ID de Superadministrador de CA creada como prerrequisito, introduciéndola en el lector conectado al puerto usb estándar.

8. Importar el certificado de CA raíz:
    a. Ir a CA Functions> Certification Authorities
    b. Presionar en Import CA certificate, rellenar la información solicitada y seleccionar el fichero que contiene el certificado de CA raíz en formato PEM creado anteriormente.
    c. Comprobar en CA Functions> Certification Authorities que el certificado de CA raíz se ha importado efectivamente como un certificado de CA raíz.

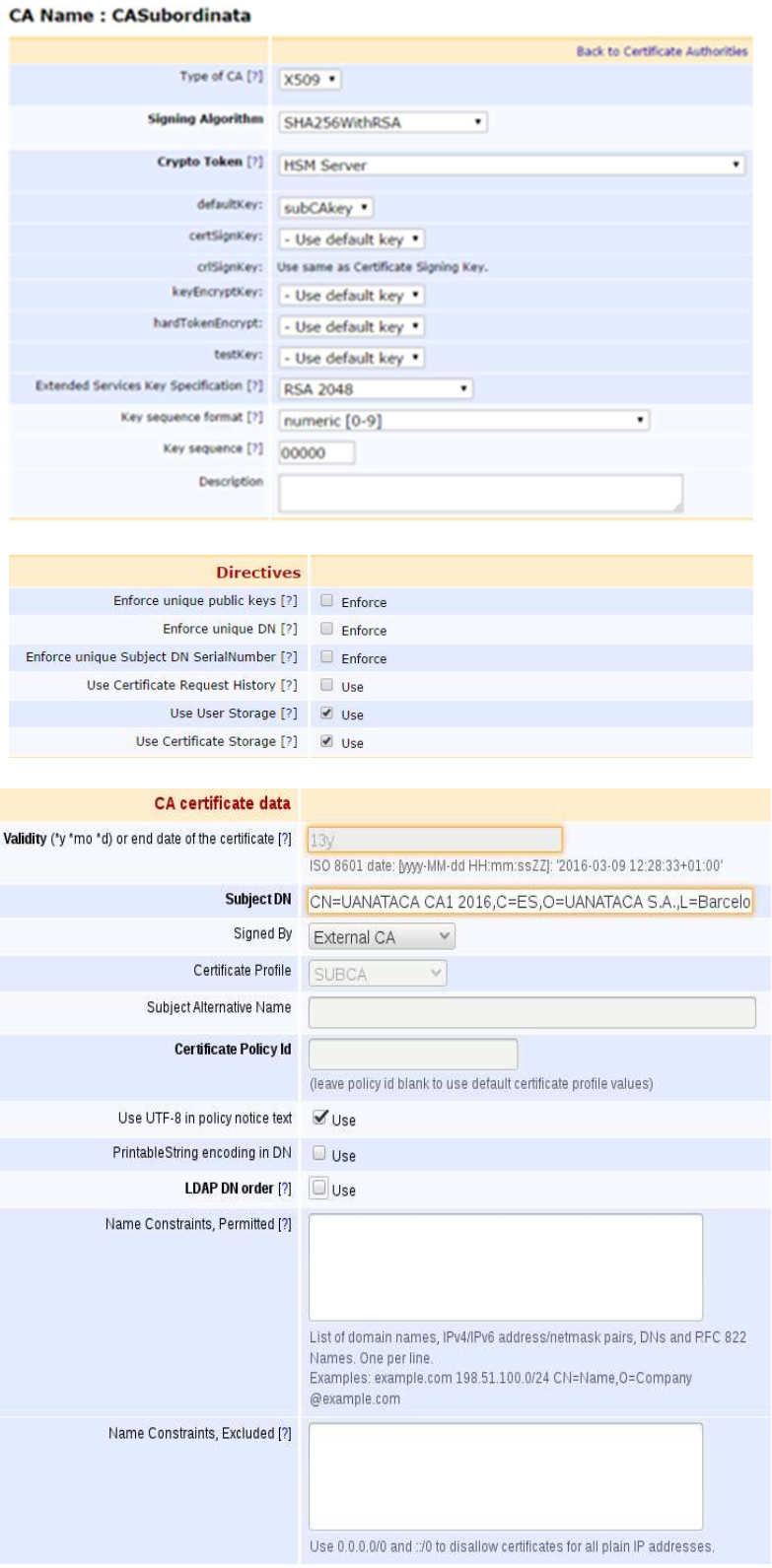


9. Ir a CA Functions> Crypto Tokens> Create new.
    a. Nombre: HSM Server
    b. Tipo: PKCS11
    c. Código de autenticación: introducir el PIN de la tarjeta OCS presente en el HSM
    d. Repetir código de autenticación: introducir el PIN de la tarjeta OCS presente en el HSM
    e. Auto-activación: No

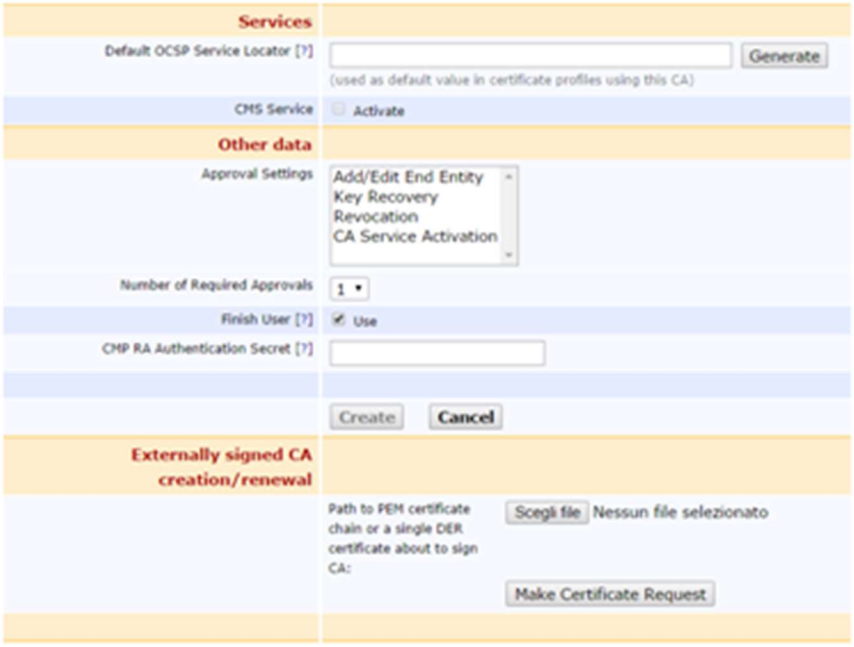
Nota: El nuevo crypto-token creado no contiene ningún par de claves.

7.2.2.2.2 Creación de la CA Subordinada 01

1. **A continuación, se creará el par de claves correspondientes a la CA subordinada 01**. Definir los siguientes parámetros tal y como se indica a continuación:
    a. Alias: subCAkey01
    b. Key size: RSA 4096
2. Pulsar el botón Generate new key pair
3. El nuevo par de claves aparecerá al final de la pantalla. Presionar en Test para comprobar que se ha generado correctamente. Debe aparecer el mensaje subCAkey01 tested successfully.
4. Ir a CA Functions> Certification Authorities> Add CA, insertando como nombre de la CA: CASubordinada01
5. Rellenar los datos de solicitud de certificado tal y como muestran las siguientes imágenes:



NOTA: usar en el campo Subject DN el definido en el perfil del certificado.



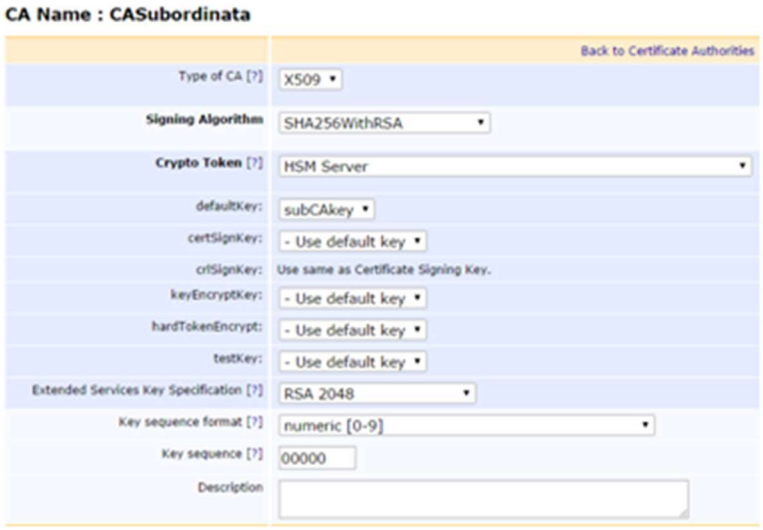
6. Presionar en Make Certificate Request y luego en el enlace para descargar la solicitud de certificado en formato PEM.
7. Salvar la petición (CSR en formato PEM) en el pendrive y regresar a la CA raíz.
8. Ir a Public Web> Create Certificate from CSR.
9. En la siguiente pantalla usar el nombre de usuario=SUBCA1 y seleccionar el fichero que contiene la petición CSR en formato PEM creada hace unos instantes.
10. Aparecerá una pantalla con el mensaje Certificate Created que confirmará la creación del certificado de CA subordinada.

11. Descargar el certificado presionando en el enlace y salvarlo en un pendrive, que posteriormente servirá para importarlo en la CA subordinada.
12. Visualizar el certificado y verificar que contiene los datos correctos según el perfil establecido en la sección “Perfil de los certificados electrónicos de CA a emitir”.
13. Anotar el *hash* SHA-1 (huella digital) del certificado en la sección “RESULTADO”.
14. Regresar a la CA subordinada.
15. Ir a CA Functions> Certification Authorities.
16. Seleccionar el nodo de CA subordinada cuyo estado sea: Waiting for Certificate Response.
17. Presionar en el botón Edit CA e ir al final de la pantalla y seleccionar el fichero del pendrive que contiene el certificado de la CA subordinada y presionar en Receive Certificate Response.

Con esto finaliza la creación de claves y generación de certificado de CA subordinada 01.

7.2.2.2.3 Creación de la CA Subordinada 02

18. **A continuación, se creará el par de claves correspondientes a la CA subordinada 02**. Definir los siguientes parámetros tal y como se indica a continuación:
    a. Alias: subCAkey02
    b. Key size: RSA 4096
19. Pulsar el botón Generate new key pair
20. El nuevo par de claves aparecerá al final de la pantalla. Presionar en Test para comprobar que se ha generado correctamente. Debe aparecer el mensaje subCAkey02 tested successfully.
21. Ir a CA Functions> Certification Authorities> Add CA, insertando como nombre de la CA: CASubordinada02
22. Rellenar los datos de solicitud de certificado tal y como muestran las siguientes imágenes:

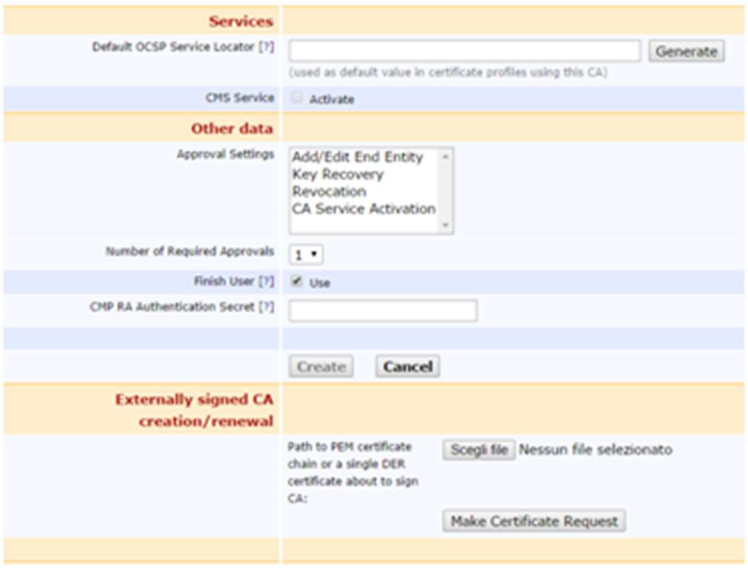


Directives	
Enforce unique public keys [?]	Enforce
Enforce unique DN [?]	Enforce
Enforce unique Subject DN SerialNumber [?]	Enforce
Use Certificate Request History [?]	Use
Use User Storage [?]	Use
Use Certificate Storage [?]	Use

CA certificate data	
Validity (*y *mo *d) or end date of the certificate [?]	13y ISO 8601 date: [yyyy-MM-dd HH:mm:ssZZ]: '2016-03-09 12:28:33+01:00'
Subject DN	CN=UANATACA CA2 2016,C=ES,O=UANATACA S.A.,L=Barcelo
Signed By	External CA
Certificate Profile	SUBCA
Subject Alternative Name	
Certificate Policy Id	(leave policy id blank to use default certificate profile values)
Use UTF-8 in policy notice text	Use
PrintableString encoding in DN	Use
LDAP DN order [?]	Use
Name Constraints, Permitted [?]	List of domain names, IPv4/IPv6 address/netmask pairs, DNs and RFC 822 Names. One per line. Examples: example.com 198.51.100.0/24 CN=Name,O=Company @example.com
Name Constraints, Excluded [?]	Use 0.0.0.0/0 and ::/0 to disallow certificates for all plain IP addresses.

NOTA: usar en el campo Subject DN el definido en el perfil del certificado.

23.



24. Presionar en Make Certificate Request y luego en el enlace para descargar la solicitud de certificado en formato PEM.
25. Salvar la petición (CSR en formato PEM) en el pendrive y regresar a la CA raíz.
26. Ir a Public Web> Create Certificate from CSR.
27. En la siguiente pantalla usar el nombre de usuario=SUBCA2 y seleccionar el fichero que contiene la petición CSR en formato PEM creada hace unos instantes.
28. Aparecerá una pantalla con el mensaje Certificate Created que confirmará la creación del certificado de CA subordinada.
29. Descargar el certificado presionando en el enlace y salvarlo en un pendrive, que posteriormente servirá para importarlo en la CA subordinada.

30. Visualizar el certificado y verificar que contiene los datos correctos según el perfil establecido en la sección “Perfil de los certificados electrónicos de CA a emitir”.
31. Anotar el *hash* SHA-1 (huella digital) del certificado en la sección “RESULTADO”.
32. Regresar a la CA subordinada.
33. Ir a CA Functions> Certification Authorities.
34. Seleccionar el nodo de CA subordinada cuyo estado sea: Waiting for Certificate Response.
35. Presionar en el botón Edit CA e ir al final de la pantalla y seleccionar el fichero del pendrive que contiene el certificado de la CA subordinada y presionar en Receive Certificate Response.

Con esto finaliza la creación de claves y generación de certificado de CA subordinada 02.

8 Recuperación de evidencias

8.1 Objetivo

Describir el procedimiento para la recuperación de evidencias, registros y logs relativos a la prestación de servicios de confianza.

8.2 Descripción

Se recogen los procedimientos necesarios para la solicitud, gestión y obtención de evidencias.

En concreto determina todo el circuito de gestión de peticiones, de la tipología de sujetos solicitantes, así como las distintas posibilidades de solicitud y el procedimiento técnico de recuperación de evidencias, atendiendo además a las posibles excepciones que puedan aplicarse como consecuencia del procedimiento.

8.3 Ámbito de aplicación

El presente procedimiento tiene como alcance todos los servicios de confianza ofrecidos,

8.4 Documentación relacionada

Véase Norma ISO 27001, ISO 27002 A.16.1.7

8.5 Anexos

N/A

8.6 Responsabilidad

Se identifican las siguientes responsabilidades en la aplicación de este procedimiento:

- Responsable de Seguridad:
  - Gestionar las solicitudes de solicitud de petición de evidencias.
  - Aprobar y/o denegar las solicitudes.

- Responsable de la infraestructura PKI:
  - Gestionar el equipo técnico encargado de realizar las actividades técnicas para la recopilación de las evidencias.

- Responsable de Operaciones:
  - Gestionar las actividades en el CMS para la recopilación de las evidencias.

8.7 Procedimiento

Las peticiones de recuperación de evidencias se realizarán a través de la herramienta de ticketing, https://bit4id.mantishub.io/my_view_page.php

8.7.1 Solicitudes

Las solicitudes para la recuperación de evidencias podrán ser:

- Por requisito legal: procedentes de procedimientos judiciales, legales, policiales o administrativos, en general por cualquier organismo de acuerdo con lo establecido con la normativa actual.

- Por petición externa: procedentes de entidades o personas ajenas

- Por petición interna: procedentes de empleados terceros vinculados

8.7.2 Registro de la solicitud

- El responsable de seguridad analizará la procedencia de las solicitudes y las registrará en la herramienta de ticketing.

- Realizado el análisis aprobará o denegará la solicitud, notificándolo a través de la herramienta de ticketing.

- Aprobada la solicitud, se asignará al equipo técnico y/o responsable para que se proceda a su preparación.

8.7.3 Recuperación de evidencias de la infraestructura

- El equipo técnico gestionará la solicitud a través de la herramienta de ticketing.
- El equipo técnico accederá a los sistemas de la infraestructura.
- Se accederá al sistema de logs, que se encuentra encriptado con la clave maestra del HSM, de tal manera que para realizar dicho acceso se requerirá al administrador de la CA y/o de Sistemas por tal de proveer el acceso necesario a dicho sistema de logs.
- Una vez se ha introducido la clave maestra del HSM, y se dispone del acceso a los logs, se descifra el data blob referente a los logs a los que se quiere acceder.
- Descifrado el data blob, se procede a la recolección de la información requerida, en un formato comprensible, para dar cumplimiento a la solicitud presentada debiendo remitirse con la mayor celeridad posible la información reclamada.

8.7.4 Recuperación de evidencias de los servicios de confianza

- El responsable de operaciones gestionará la solicitud a través de la herramienta de ticketing.
- El responsable accederá al CMS.

- Se procederá a la obtención de los registros mediante consulta al apartado de registro de auditoría propio de la aplicación.
- Se extraerá en un formato legible y comprensible la información solicitada.
- Si fuese necesario por motivo de la solicitud, el responsable de operaciones accederá a los registros físicos con el fin de justificar los registros.

8.7.5 Entrega de las evidencias

- Los registros se extraerán en un formato legible y comprensible.
- Los mismos serán cifrados o se procederá a su protección para que únicamente el solicitante tenga acceso.
- La notificación y envío se realizará por las vías ordinarias en función de la solicitud, asegurando que las credenciales y/o acceso se entregue una por una vía distinta, cuyo único conocedor sea el solicitante.

8.8 Flujo de procesos

N/A