Guía de Seguridad de las TIC
CCN-STIC 1022

Procedimiento de Empleo Seguro
OpenNAC Enterprise



Noviembre 2026

Edita:

CCN
c=ES, l=MADRID, o=CCN,
ou=SELLO ELECTRONICO,
2.5.4.97=VATES-S2800155J,
serialNumber=S2800155J,
cn=CCN
2026.11.04 14:07:01 +01:00

Fecha de Edición: Noviembre de 2026

LIMITACIÓN DE RESPONSABILIDAD

El presente documento se proporciona de acuerdo con los términos en él recogidos, rechazando expresamente cualquier tipo de garantía implícita que se pueda encontrar relacionada. En ningún caso, el Centro Criptológico Nacional puede ser considerado responsable del daño directo, indirecto, fortuito o extraordinario derivado de la utilización de la información y software que se indican incluso cuando se advierta de tal posibilidad.

AVISO LEGAL

Quedan rigurosamente prohibidas, sin la autorización escrita del Centro Criptológico Nacional, bajo las sanciones establecidas en las leyes, la reproducción parcial o total de este documento por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares del mismo mediante alquiler o préstamo públicos.

Esta publicación se distribuye bajo licencia CC BY-NC-ND 4.0 que permite compartir el material en cualquier medio o formato con la condición de reconocer adecuadamente la procedencia «Edita: Ministerio de Defensa. Secretaría General Técnica.» No se puede modificar. No se puede utilizar con fines comerciales.

ÍNDICE

1 INTRODUCCIÓN 3
2 OBJETIVO Y ALCANCE 3
3 ORGANIZACIÓN DEL DOCUMENTO 4
4 FASE PREVIA A LA INSTALACIÓN 5
4.1 ENTREGA SEGURA DEL PRODUCTO 5
4.2 ENTORNO DE INSTALACIÓN SEGURO 5
4.3 REGISTRO Y LICENCIAS 5
4.4 CONSIDERACIONES PREVIAS 6
5 FASE DE INSTALACIÓN 8
6 FASE DE CONFIGURACIÓN 9
6.1 AUTENTICACIÓN 9
6.1.1 AUTENTICACIÓN DE USUARIOS 9
6.1.2 AUTENTICACIÓN ENTRE NODOS DEL PRODUCTO Y/O CON SERVIDORES O DISPOSITIVOS EXTERNOS 9
6.1.3 AUTENTICACIÓN DE SESIONES 10
6.2 ADMINISTRACIÓN DEL PRODUCTO 10
6.2.1 ADMINISTRACIÓN LOCAL Y REMOTA 10
6.2.2 CONFIGURACIÓN DE ADMINISTRADORES 11
6.3 CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS 15
6.4 CONFIGURACIÓN DE PROTOCOLOS SEGUROS 18
6.5 GESTIÓN DE CERTIFICADOS 19
6.5.1 CERTIFICADO MÓDULO RADIUS 19
6.5.2 OTROS CAMPOS DEL CERTIFICADO 21
6.5.3 VALIDACIÓN DE LA VIGENCIA 21
6.5.4 VALIDACIÓN DE REVOCACIÓN 21
6.5.5 CERTIFICADOS DEL PORTAL DE ADMINISTRACIÓN (HTTPS) 22
6.6 SERVIDORES DE AUTENTICACIÓN 22
6.7 SINCRONIZACIÓN HORARIA 23
6.8 ACTUALIZACIONES 23
6.9 AUTOCHEQUEOS 23
6.10 SNMP 24
6.11 ALTA DISPONIBILIDAD 24
6.11.1 REPLICACIÓN DE MYSQL DIFERENTES NODOS WORKER 24
6.11.2 CONFIGURACIÓN DEL CLÚSTER ANALYTICS 26
6.12 AUDITORÍA 27
6.12.1 REGISTRO DE EVENTOS 27
6.12.2 ALMACENAMIENTO LOCAL 28
6.12.3 ALMACENAMIENTO REMOTO 29
6.13 BACKUP 34
6.14 SERVICIOS DE SEGURIDAD 34
6.14.1 CASO DE USO – UNAC 35
6.14.2 CASO DE USO – VISIBILITY 35
7 REFERENCIAS 36
8 ABREVIATURAS 38

1 INTRODUCCIÓN

OpenNAC Enterprise by Cipherbit es una plataforma de control de acceso a la red (NAC) para entornos LAN / WAN que permite a las empresas autenticar, autorizar y auditar todos los accesos basándose en un conjunto de reglas o políticas. Para ello, los dispositivos de red (conmutadores y puntos de acceso) pueden admitir el protocolo 802.1x, aunque no es obligatorio.

Se ofrece una solución de software que aumenta la seguridad de la red al brindar a la organización un 100% de visibilidad y control de los activos (dispositivos + identidades) conectados a la red (cable, Wi-Fi y VPN).

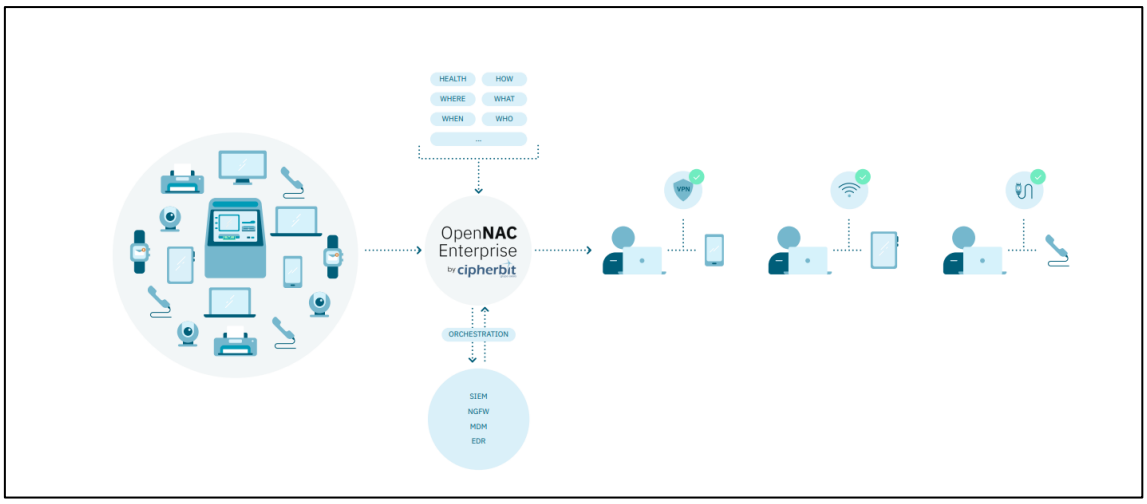


Ilustración 1. Arquitectura OpenNAC

En este documento se pretende mostrar cómo se llevan a cabo ciertos procesos dentro de la empresa para garantizar un entorno seguro a la hora de realizar la gestión de licencias y certificados, gestión local y remota de los distintos nodos, configuración de interfaces, puertos y servicios, así como los protocolos de comunicación y gestión que se usan en cada caso.

Cada caso de uso requiere configurar un conjunto de componentes, también llamados nodos, para que se comuniquen entre sí, veremos cómo se comunican, que información envían y que protocolos se usan para este intercambio.

Para encontrar información más detallada sobre los diferentes casos de uso del producto, así como los requisitos o nodos desplegados para cada caso de uso, se puede consultar la documentación oficial del producto [REF1].

2 OBJETIVO Y ALCANCE

El objetivo de este documento es detallar la configuración y el tipo de conexiones seguras que se aplican en los distintos protocolos, servicios, etc. de los nodos de la solución para los distintos casos de uso.

El producto OpenNAC ha sido incluido en el catálogo CPSTIC, para conocer el listado, la categoría o nivel y la familia consulte el sitio web del CPSTIC [REF31].

3 ORGANIZACIÓN DEL DOCUMENTO

a. Apartado 4. En este apartado se recogen aspectos y recomendaciones a considerar, antes de proceder a la instalación del producto.

b. Apartado 5. En este apartado se recogen recomendaciones a tener en cuenta durante la fase de instalación del producto.

c. Apartado 6. En este apartado se recogen las recomendaciones a tener en cuenta durante la fase de configuración del producto, para lograr una configuración segura.

4 FASE PREVIA A LA INSTALACIÓN

En este apartado se explicará cómo se realiza la entrega de los productos, tanto los nodos de software como de hardware, la gestión y el registro de las licencias y se detallarán las recomendaciones a tener en cuenta acerca del entorno de instalación de los nodos.

4.1 ENTREGA SEGURA DEL PRODUCTO

Para garantizar que la entrega de los diferentes nodos de la solución sea segura, se deberán seguir ciertos pasos:

a. De cara a obtener acceso a la descarga de los productos de software, el contacto con el cliente o partner se realizará por correo electrónico desde una cuenta oficial de la empresa al correo "*soporte_one@oesia.com*", a través de la cual se facilitará el enlace al repositorio desde el que podrán descargar los nodos necesarios para el caso de uso correspondiente.

b. Para permitir el acceso mediante HTTPS al repositorio [REF2] y descargar la OVA correspondiente a cada nodo, se generarán credenciales específicas para cada cliente o partner. El usuario y la contraseña se enviarán por correo electrónico en mensajes separados y dirigidos a un único destinatario, garantizando que nunca se incluyan ambos en el mismo envío.

c. Es decir, la descarga de cualquier nodo se hará siempre bajo la autorización por parte del equipo de Profesional Services o de preventa de OpenNAC Enterprise by Cipherbit.

4.2 ENTORNO DE INSTALACIÓN SEGURO

La instalación de los distintos nodos la gestiona el cliente bajo su responsabilidad. Se recomienda aplicar las medidas del Esquema Nacional de Seguridad (ENS) y seguir las guías del Centro Criptológico Nacional (CCN).

A nivel físico se debe tener un control de las máquinas que vayamos a instalar, de manera que estén en un entorno aislado con acceso solo a usuarios autorizados.

La instalación de algunos de los nodos de la solución se puede realizar sobre un centro de procesamiento de datos (CPD) onpremise o en un entorno cloud.

En caso de que la instalación de alguno de los nodos deba instalarse sobre un CPD, se deberá tener en cuenta que solo podrá acceder al personal autorizado, cualquier acción o modificación que se realice quedará registrada en los logs.

Los accesos se pueden hacer mediante el uso de contraseñas cifradas o certificados.

4.3 REGISTRO Y LICENCIAS

Para realizar el registro completo de una nueva licencia se deben seguir ciertos pasos:

a. En el caso de ser un proyecto nuevo, primero de todo recibiremos una petición por parte del cliente, donde especificará los detalles que deberemos configurar.

b. Se deben tener el nombre y el correo de la empresa, así como el TTL (en días) que se quiera definir. Una vez conocemos los datos se generará la licencia con la configuración pertinente.

c. La instalación de la licencia la puede realizar equipo de Professional Services (que registrará automáticamente la licencia) o se puede enviar directamente al cliente para que ellos gestionen la instalación. En caso de enviarla al cliente, el envío se hará vía email añadiendo en copia a la cuenta interna de OpenNAC Enterprise by Cipherbit (license-opennac@oesia.com) para que se realice el registro.

d. Para que el cliente tenga constancia de las fechas de inicio y fin de la licencia (la caducidad), estas, se especificarán en este mismo correo para que los responsables del CRM puedan tener constancia de estas fechas y adjunten la información al proyecto.

e. En los proyectos que están en producción donde ya tenemos licencias registradas e instaladas, el sistema de chequeo de OpenNAC Enterprise informará periódicamente a medida que se acerque la fecha de expiración de la licencia, para evitar problemas, esta debe ser renovada antes de que caduque.

f. Una vez se notifique la necesidad de renovar la licencia, la generaremos y la enviaremos a cliente para instalarla.

Para realizar la instalación de la licencia, seguir los pasos indicados en el apartado “5.1.9 License” de la documentación del producto [REF3].

4.4 CONSIDERACIONES PREVIAS

Antes de iniciar la instalación y configuración del producto, debemos tener en cuenta ciertos aspectos, puesto que influirán a la hora de configurarlo de forma segura.

Las tecnologías de acceso a la red requieren una segmentación de red adecuada, tener un diseño claro ayudará a implementar una solución NAC.

Al iniciar un nuevo proyecto, se recibe una solicitud por parte de cliente donde se especifica el caso de uso, necesidades y la configuración que se desea implementar.

Una vez se conozcan las especificaciones del cliente, se revisará la información referente a la electrónica de red, fabricante, modelos, versión, etc., donde se comprobará la compatibilidad con nuestros nodos.

En caso de que en la instalación del cliente exista una electrónica de red que no esté homologada para funcionar junto al producto, deberá gestionarse la posibilidad de, previamente a la instalación, realizar una homologación a cargo del equipo de OpenNAC Enterprise.

A nivel de requerimientos de red, se debe tener en cuenta los siguientes aspectos de cara al despliegue:

- Es importante definir los requisitos de red de Capa 2, Capa 3 y Capa 4 para garantizar una implementación exitosa.
    - Capa 2: la arquitectura de red debe estar implementada antes de la instalar OpenNAC Enterprise.
    - Capa 3: Es posible que se requieran cambios de visibilidad de IP entre diferentes piezas de software, enrutamiento y cortafuegos.

    - Capa 4: Puertos TCP / UDP que deben estar abiertos en la red para evitar problemas de integración. (detallaremos más adelante la configuración de los diferentes puertos).
- Tener en cuenta la gestión de los certificados, ya que estos se generarán para que el cliente los firme, y una vez firmados con su entidad y los dispositivos confíen en nosotros deberemos dejarlos en los diferentes servidores que lo requieran.
- Definir que nodos se van a instalar y definir la comunicación que habrá entre ellos, para así ver si es necesario crear diferentes API keys, y asignarlas a los diferentes servidores que existan en la infraestructura, de manera que puedan establecer conexión entre ellos.

Tanto de forma previa a la instalación, como en posteriores revisiones de la misma, se deberá seguir el siguiente procedimiento de revisión y aprobación de conexiones dentro de la arquitectura:

- Las conexiones se describen en el esquema de arquitectura diseñado y cumplen con los requisitos comerciales tal como se diseñaron y aprobaron en el proyecto inicial, este esquema deberá ser utilizado como referencia en las revisiones periódicas.
- Las comprobaciones deberán realizarse al menos cada cuatro o seis meses y en cada cambio aprobado que se quiera realizar.
- A nivel de hardware y software se realizan las diferentes comprobaciones para asegurar que no se ha introducido hardware o software no autorizado.
- Antes de publicar una nueva versión de producto o antes de actualizar el SO, se realizarán procesos de Hardening para reforzar al máximo posible la seguridad de un equipo. Usaremos Openscap [REF4].

Para probar la implementación del Hardening en Centos:

- Los nodos se configurarán previamente para reducir las vulnerabilidades en el mismo, para ello eliminaremos ciertos parámetros, servicios, usuarios, etc; innecesarios en el sistema, así como cerrando puertos que tampoco estén en uso, o añadiendo ciertas configuraciones para aumentar los requisitos de seguridad para verificar archivos hash con RPM, asegurar que los parches de software estén instalados, etc.
- Inicialmente, se instala un sistema base mínimo evitando instalaciones de grupos de aplicaciones y aplicaciones que no sean estrictamente necesarias.
- Se revisarán los puertos y servicios abiertos, así como las diferentes reglas que tengamos definidas en el fichero de `/etc/sysconfig/iptables` en caso se haber realizado cambios, se comprobará que siguen habilitados los puertos y servicios que sean estrictamente necesarios para el correcto funcionamiento de los distintos nodos.

Se deberán revisar los paquetes instalados y se eliminarán aquellos que no sean necesarios. Esta comprobación la podemos hacer mediante:

```
rpm -qa | less
rpm -qa | grep <service>
lsof -i | grep -v ESTABLISHED
```

Para eliminar paquetes o servicios instalados:

```
systemctl stop <service>
systemctl disable <service>
yum remove <package>
```

5 FASE DE INSTALACIÓN

Una vez concedido el acceso al repositorio y habiendo realizado los pasos previos, se podrá llevar a cabo la instalación del TOE. Existen dos métodos principales para su despliegue:

1. Utilizar un sistema operativo vacío e instalar y configurar manualmente todos los paquetes del TOE.
2. Utilizar un archivo OVA de OpenNAC, que ya incluye los paquetes principales preinstalados.

Para este producto cualificado se empleará la segunda opción, implementando el componente ON Core en una plataforma de virtualización Proxmox. Para ello, consulte el apartado 2.1.1 “Proxmox OVA Deployment” de la documentación [REF5].

Además, para verificar la integridad del archivo OVA, se utilizará un hash MD5 incluido en el repositorio de OpenNAC. En este caso, un administrador de plataforma de confianza deberá calcular el hash MD5 del archivo OVA descargado y compararlo con el proporcionado por el fabricante.

La configuración básica del sistema en el que se despliega el componente “On Core” puede consultarse en la sección 9.2.1 “On Core” [REF6].

De forma adicional al componente básico “On Core”, las siguientes referencias incluyen los enlaces en los que se describe el procedimiento de instalación de los distintos nodos complementarios:

- Instalación y configuración de ON Analytics [REF7]
- Instalación y configuración de ON Aggregator [REF8]
- Instalación y configuración ON Sensor [REF9]
- Instalación y configuración ON Agent [REF10]
- Instalación y configuración ON VPNGW [REF11]

Nota: Aunque la solución permite extender sus capacidades mediante el uso de todos los nodos previamente enumerados, la cualificación se ha llevado a cabo utilizando específicamente el componente ON Core, el cual por sí mismo proporciona capacidades suficientes para ejercer la funcionalidad de seguridad.

Tener en cuenta que es imprescindible cambiar todas las contraseñas por defecto en cada uno de los nodos instalados.

6 FASE DE CONFIGURACIÓN

En este apartado se recogerán las recomendaciones principales a tener en cuenta durante la fase de configuración y administración del producto.

6.1 AUTENTICACIÓN

6.1.1 AUTENTICACIÓN DE USUARIOS

En lo que se refiere a la autenticación de usuarios (en los casos de UNAC y VPN) mediante 802.1x debemos tener en cuenta que desde OpenNAC Enterprise, podemos autenticar tanto a usuarios locales (tenemos su información almacenada en la base de datos local), usuarios que se encuentren en servidores externos (AD y LDAP) o mediante el uso de certificados.

A la hora de hacer la autenticación de un usuario, se seguirá un cierto orden para hacer la consulta y ver si ese usuario está registrado o no, tanto en la base de datos local como en los servidores externos que tengamos configurados.

El orden por defecto que se sigue es:

- Usuarios locales
- LDAP/AD

Primero se mirará en la base de datos local, se comprobará si el usuario existe y si las credenciales coinciden, si encontramos al usuario, pero las credenciales no coinciden, el usuario no se autenticará.

Si no se ha encontrado el usuario en la base de datos local, iremos a la siguiente base de datos/repositorio que tengamos activo.

En el caso de los usuarios externos, como desde OpenNAC Enterprise no tenemos acceso a las credenciales de dicho usuario, mandaremos la petición al servidor externo para ver si tiene registrado al usuario y si las credenciales coinciden.

En el caso del acceso al portal de administración web, cada usuario tendrá un rol asignado (veremos la descripción de los diferentes roles y su configuración en el apartado 6.2.2 CONFIGURACIÓN DE ADMINISTRADORES de este documento).

Si se trata de un usuario local, se le asignará un rol al crearlo. Si se trata de usuarios que proviene de servidores externos, la primera vez que accedan al portal no tendrán ningún rol definido y se les asignara a posteriori. Lo que hacemos desde OpenNAC Enterprise, es filtrar según los usuarios o grupos que estén configurados en el servidor externo, asignando a ciertos usuarios o a grupos de usuarios un rol determinado.

6.1.2 AUTENTICACIÓN ENTRE NODOS DEL PRODUCTO Y/O CON SERVIDORES O DISPOSITIVOS EXTERNOS

En el caso de OpenNAC Enterprise, ciertos nodos requieren autenticación para que puedan comunicarse entre ellos: la API y el Captive Portal.

Por defecto los módulos del portal de administración, API y captive portal se gestionan en la misma máquina, por lo que no es necesario realizar ninguna autenticación ni comprobación entre los nodos para que puedan comunicarse entre ellos.

En el caso de que tengamos en una máquina el portal de administración y la API, y en una segunda máquina tengamos el portal cautivo, deberemos generar una API key que intercambiarán ambas máquinas para verificar que las peticiones que se realicen desde ambos nodos vienen de una fuente fiable.

La API key la generaremos desde el portal de administración (cada API key se genera para una IP origen definida) y la añadiremos en el fichero de configuración de la segunda máquina, así podremos verificar que la fuente con la que nos intercambiamos información es fiable.

En lo que se refiere a comunicación con dispositivos externos, como dispositivos de red, nos comunicamos con ellos ya sea para añadir una configuración (NetConf), hacer copias de seguridad de la configuración que tenga el dispositivo (NetBackup) o para ver si el dispositivo cumple con ciertos requisitos de configuración (Network Device Compliance), esta conexión a los dispositivos se realizará por SSH (usando SSH v2).

6.1.3 AUTENTICACIÓN DE SESIONES

A la hora de acceder a los nodos vía SSH podemos garantizar el acceso tanto por clave como por certificado, aunque este se puede desactivar para que solo se pueda acceder mediante el uso de claves.

Se pueden generar claves RSA públicas para que diferentes usuarios puedan usarlas para acceder a los distintos nodos.

Para acceder al portal, podemos habilitar el uso de OTP como doble factor de autenticación.

6.2 ADMINISTRACIÓN DEL PRODUCTO

6.2.1 ADMINISTRACIÓN LOCAL Y REMOTA

La administración, tanto local como remota, de los elementos de la solución, se puede realizar por CLI o por GUI.

Para la administración remota, accederemos a la red privada del cliente con la VPN que nos facilitará para que podamos acceder a los diferentes nodos.

A nivel interno, a la hora se conectarnos vía CLI, descartamos el uso del servicio de telnet ya que expondríamos toda la información que intercambiamos con los servidores, con el consiguiente riesgo que esto supone, por lo que, por defecto, telnet no está habilitado en ninguno de los nodos de la solución, las conexiones vía CLI las hacemos mediante el SSH v2. Para el correcto uso de SSH por los administradores de los sistemas, se deberá modificar la contraseña por defecto en el primer acceso, tal y como se expone en la sección 5.1.6.1.1 “SSH Password” de la documentación [REF12].

Otra práctica que realizamos para mantener la privacidad es evitar mostrar por CLI ninguna contraseña en ficheros de log o al ejecutar determinados comandos para evitar que quede expuesta.

Para la copia y transferencia de archivos descartamos el uso de FTP (File Transfer Protocol), por defecto no está habilitado, en su lugar usaremos SCP (Sercure Copy Protocol).

Para acceder mediante la interfaz gráfica de usuario, usamos el protocolo HTTPS.

Los usuarios autorizados tendrán acceso al portal de administración a través de la siguiente URL: "*https://<ip_management>/admin", donde "ip_management*" es la dirección IP privada obtenida al configurar las interfaces de red en el TOE.

A través de esta interfaz web, se deberán realizar las siguientes configuraciones durante el primer acceso:

- Solicitud de licencia al fabricante; consultar el apartado *4.3 REGISTRO Y LICENCIAS* de este documento.
- El usuario administrador deberá cambiar la contraseña predeterminada por una nueva durante el primer acceso; para ello, deberá seguir los siguientes pasos:
  1. Haga clic en el icono "*usuario*" situado en la esquina superior derecha de la interfaz web.
  2. Aparecerá un menú desplegable en el que deberá hacer clic en "*Configuración*".
  3. La contraseña se puede cambiar desde el apartado "*Cambiar contraseña*".

Para información adicional sobre este aspecto, consulte el apartado 5.1.6.1.3 "*Administration Portal Password*" de la documentación [REF13].

6.2.2 CONFIGURACIÓN DE ADMINISTRADORES

El primer usuario administrador no requiere creación manual, ya que viene preconfigurado en la imagen OVA oficial proporcionada por el fabricante. Tras desplegar la OVA, el administrador puede acceder directamente al Portal de Administración utilizando las credenciales iniciales suministradas junto con el paquete de instalación.

Mediante el uso de roles, los usuarios con perfil de administrador pueden generar diferentes perfiles con distintos niveles de privilegios y luego asociarlos a un usuario.

Los roles se utilizan para administrar el acceso al portal de administración web de OpenNAC Enterprise.

Esto permite tener un control de acceso más granular sobre las áreas funcionales de la interfaz de administración web y del uso de la API.

Por ejemplo, podemos crear un perfil de rol de administrador que brinde acceso a las áreas de configuración de red y de la interfaz web y un perfil separado para sus administradores de seguridad que brinde acceso a definiciones, registros e informes de políticas de seguridad.

Podemos ver los roles que tenemos configurados en `ON CMDB -> Security -> Roles`.

Por defecto el producto viene con seis roles configurados:

- **Administrador:** Cuenta con todos los permisos de administración y gestión (no podremos modificar sus ACL's (lista de controles de acceso), ya viene definida).
- **Otpmanager:** Cuenta con permisos de acceso a la configuración de OTP.
- **Readonly:** Cuenta con permisos para visualizar todas las ventanas del menú.
- **Audit:** Cuenta con permisos para auditar logs.
- **Operator:** Cuenta con permisos para tareas de operación.
- **userDeviceViewer:** Cuenta con permisos para operar la vista de Dispositivos de Usuarios.

Podemos modificar a nuestra conveniencia los permisos de cada uno de los roles que ya tenemos configurados por defecto, excepto el rol de administrador.

Para ello, modificaremos los permisos, ACLs, que hacen referencia a un determinado rol, tanto para habilitar la visualización de determinados menús, como para habilitar los permisos para editar y realizar configuraciones.

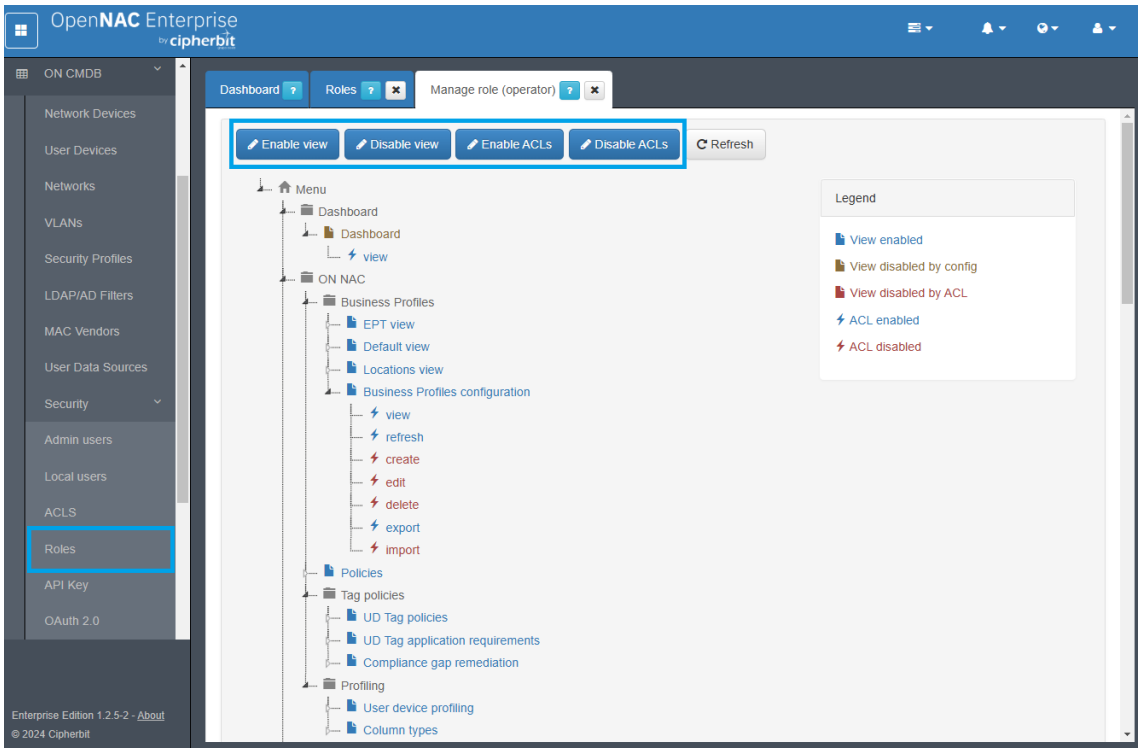


Ilustración 2.Edición de ACLs de roles

También tenemos la posibilidad de crear nuevos roles. Al crear un nuevo rol, por defecto, este tendrá habilitados los mismos permisos que el rol de readonly.

Una vez definidos los diferentes roles, estos pueden asignarse tanto a los nuevos usuarios que se creen como a los usuarios ya existentes.

Al crear un nuevo usuario, también deberemos tener en cuenta la política de configuración segura de contraseñas. Se recomienda que las contraseñas cumplan con los requisitos del ENS, de acuerdo con lo establecido en la guía CCN-STIC 807 [REF14]:

- Longitud de la contraseña: mínimo 12 caracteres.
- Complejidad:
  - Uno o más caracteres en minúscula.
  - Uno o más caracteres en mayúscula.
  - Uno o más números.
  - Uno o más caracteres especiales.
- La contraseña no puede ser el nombre de usuario.
- La contraseña no puede ser una matrícula de coche.
- No puede ser una de las ultimas 10 contraseñas.
- El cambio de contraseña debe realizarse periódicamente.

Configuración de parámetros de sesión:

Para configurar los parámetros relativos a la sesión de usuario en el portal de administración web, deberemos acceder a la sección `Configuration -> Configuration vars -> Advanced`, allí encontraremos los siguientes parámetros:

- **Timeout de autenticación:** Es el tiempo que permanece el usuario autenticado en la sesión, transcurrido este tiempo, el usuario debe volver a autenticarse.
- **Timeout de inactividad:** El timeout de inactividad hace referencia al tiempo que puede permanecer una sesión inactiva, por defecto está definido a una hora, sin embargo se recomienda modificarlo a un máximo de 30 minutos. Una vez haya trascurrido ese tiempo, se hará una desconexión automática de esa sesión. Este parámetro puede ser modificado por aquellos usuarios que dispongan de los roles con permisos para realizar dicho cambio en la pestaña "*Configuration > Configuration Vars > Advanced > Edit Inactivity timeout*" de la consola de gestión.
- **Número máximo de sesiones concurrentes de un administrador:** Por defecto viene habilitado un máximo de 5 sesiones concurrentes de un administrador, parámetro que se puede modificar desde el portal de administración.
- **Número máximo de intentos fallidos de autenticación:** Al superar los cinco intentos fallidos de autenticación a la hora de conectarnos a la aplicación, bloquearemos el acceso de ese usuario durante cinco minutos, una vez transcurrido ese tiempo, el usuario volverá a tener cinco intentos para realizar la autenticación.
- **Caducidad de la contraseña del usuario administrador:** Este parámetro obliga a cambiar la contraseña del usuario administrador cada cierto tiempo. El parámetro se define en días. Por defecto, la contraseña debe renovarse cada 90 días. Este parámetro puede ser modificado.
- **Tiempo de espera tras superar un umbral:** Está definido por defecto un sistema de bloqueo incremental (no configurable), por lo que cada vez que el usuario falle al autenticarse, incrementaremos el tiempo de espera, por lo que la primera vez que falle, deberá esperar 5min, la segunda 2*5min, la tercera 3*5min y así de manera incremental.

Adicionalmente, se deberá realizar la configuración pertinente del servidor de retransmisión de correo SMTP para poder realizar la configuración inicial de OTP. Consultar el apartado 9.2.7.1.3 "*Notification Service Configuration*" [REF15].

Una vez configurado el servicio SMTP, se podrá proceder a la configuración del código OTP para cada usuario autorizado creado en la consola de administración. Consultar el apartado 3.1.10.4 "*OTP*" de la documentación [REF16].

Adicionalmente, se recomienda realizar la configuración de una política de denegación para cualquier dispositivo de usuario, dispositivo de red u origen. Esta política se colocará en una lista de políticas para que actúe como regla por defecto, de manera que, si un evento no coincide con ninguna de las políticas anteriores, se le denegará el acceso. Para ello, se deben seguir los siguientes pasos:

- Hacer clic en la sección "*ON NAC*" situada en la parte izquierda de la interfaz web.
- Aparecerá un menú desplegable en el que se deberá hacer clic en "*Politicas*".
- Hacer clic en el botón "*Añadir nueva*".
- En la pestaña "*General*", asignar un nombre a la política y establecer el campo "habilitado" en "*sí*".
- Dejar la pestaña "*Precondiciones*" con la configuración predeterminada, sin definir ninguna configuración adicional.
- En la pestaña "*Postcondiciones*", hacer clic en "*Establecer VLAN*", seleccionar la VLAN 4095 (ACCESO DENEGADO) y hacer clic en el botón "*Confirmar*".

Tener en cuenta que, para el correcto funcionamiento de esta política de denegación, es necesario eliminar la política "*catch all*" que viene configurada por defecto en la lista de políticas.

Adicionalmente, para garantizar que exista siempre una política de denegación por defecto, incluso en ausencia de una política preconfigurada, deberá habilitarse la opción "Denegar el acceso como política por defecto" en la sección "Configuración → Variables de configuración → Avanzado → Evaluación de políticas" del Portal del administrador.

Por último, para garantizar la autorización adecuada a la API, se deberá eliminar las dos claves de API predeterminadas proporcionadas por el TOE, y, a continuación, crear dos nuevas claves de API con privilegios de "*operador*" y "*administrador*" para las solicitudes procedentes de la dirección IP "*127.0.0.1*" a través del Portal del administrador en "*CMDB*" → "*Seguridad*" → "*Clave de API*". Una vez creadas, estas claves deben incluirse en los siguientes archivos del sistema operativo:

- La clave API de "*operador*" deberá incluirse en los siguientes archivos y en el siguiente formato:
    - *"/usr/share/opennac/admonportal-html5/application/configs/globals.ini"*:
      apiKey = "xxxx-yyyyy-zzzzz"
    - *"/usr/share/opennac/captive-portal/application/configs/globals.ini"*:
      api.apiKey = "xxxx-yyyyy-zzzzz"
    - *"/etc/raddb/mods-available/opennac"*:
      apiKey = xxxx-yyyyy-zzzzz
    - *"/etc/collectd.d/curl_json.conf"*:
      Header "X-Opennac-API-key: xxxx-yyyyy-zzzzz" (on all lines containing the "Header" reference)
- La clave API de "administrador" deberá incluirse en los siguientes archivos y en el siguiente formato:
    - *"/usr/share/opennac/queues/config.ini":*
      token = "xxxx2-yyyyy2-zzzzz2"
    - *"/usr/share/opennac/api/application/configs/api.ini"*:
      apiKey.key = "xxxx2-yyyyy2-zzzzz2"
- Los principales servicios de los componentes TOE se reiniciarán ejecutando los siguientes comandos tras aplicar los cambios:

```
systemctl restart radiusd
systemctl restart httpd
systemctl restart php-fpm
systemctl restart opennac
systemctl restart collectd
```

6.3 CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS

En lo que se refiere a la configuración de interfaces, puertos y servicios, la configuración de estos ya viene por defecto.

Podemos encontrar la descripción de cómo llevar a cabo la configuración de las interfaces de los diferentes nodos en las siguientes referencias:

- Instalación y configuración de ON Core, sección 9.2.1.1.1 de la documentación [REF6]
- Instalación y configuración de ON Analytics, sección 9.2.2.1.1 de la documentación [REF7]
- Instalación y configuración de ON Aggregator, sección 9.2.3.1.1 de la documentación [REF8]
- Instalación y configuración ON Sensor, sección 9.2.4.1.1 de la documentación [REF9]
- Instalación y configuración ON Agent, sección 6.1.9 de la documentación [REF10]
- Instalación y configuración ON VPNGW, sección 9.2.5.1.2 de la documentación [REF11]

Hacemos un control de los servicios abiertos al público ya que están filtrados por el firewall (mediante la configuración que se aplica en las iptables según IP y puerto) y que estarán escuchando información en la interfaz de servicio.

En lo que se refiere a los servicios locales, estos solo escuchan información y peticiones de localhost, los servicios y puertos sin uso están deshabilitados.

Para hacer el filtrado, accedemos desde CLI a los distintos nodos (vía SSH2) y modificaremos las reglas de firewall en el fichero de iptables (el fichero se encuentra en `/etc/sysconfig/iptables`).

Inicialmente viene por defecto una configuración restrictiva para que solamente se pueda acceder a los servicios principales.

En el momento en el que pongamos en producción cualquiera de los nodos deberemos aplicar una configuración para que quede el acceso lo más restringido posible, habilitando solamente los puertos y servicios que vayamos a utilizar dependiendo del caso de uso que configuremos.

Para llevar a cabo la configuración de los diferentes nodos, lo haremos mediante la red interna, a través de la cual administraremos todos los nodos. Esta red debe ser privada para que solamente pueda acceder el personal autorizado.

La siguiente tabla muestra los puertos y servicios que se pueden habilitar en los distintos nodos, así como aquellos que deberemos dejar abiertos para que pueda establecerse una comunicación entre los distintos nodos de la arquitectura configurada.

La configuración para los distintos nodos es la siguiente:

ON CORE			
SOURCE	**DESTINATION**	**PORT**	**SERVICE**
Core	Core	TCP/22	SSH
Core	Analytics	TCP/22	SSH
Core	Aggregator	TCP/22	SSH

ON CORE			
SOURCE	DESTINATION	PORT	SERVICE
Core	Sensor	TCP/22	SSH
Core	Core	TCP/80	HTTP
Core	Core	TCP/443	HTTPS
Core	Core Principal (VIP)	TCP/3306	MySQL
Core	Core Principal (VIP)	TCP/6379	Redis
Core	Core Principal (VIP)	UDP/25826	Collectd
Core	Analytics (VIP)	TCP/5601	Kibana
Core	Analytics (VIP)	TCP/9200	ElasticSearch
Core	Aggregator	TCP/5000	FileBeat
Core	Network Devices	UDP/161	SNMP
Core	Network Devices	UDP/3799	CoA
Core	Network Devices	TCP/22	SSH
Core	MTA Relay	TCP/25	SMTP
Core	NTP SERVER	UDP/123	NTP
Core	Proxy Server	TCP/8080 [*]	HTTP / HTTPS
Core	DNS	UDP/53	DNS
Core	AD SERVERS	UDP/TCP/88	KERBEROS
Core	AD SERVERS	UDP/TCP/135	DCOM/RPC
Core	AD SERVERS	UDP/TCP/137	NETBIOS
Core	AD SERVERS	UDP/TCP/138	NETBIOS
Core	AD SERVERS	UDP/TCP/139	NETBIOS
Core	AD SERVERS	UDP/TCP/389	LDAP
Core	AD SERVERS	TCP/445	SMB

ON CORE			
SOURCE	DESTINATION	PORT	SERVICE
Core	AD SERVERS	TCP/464	KPASSWD
Core	AD SERVERS	TCP/636	LDAPs
Core	Palo Alto Fw	TCP/443	HTTPS
Core HTTP (VIP)	Core	TCP/80	HTTP
Core HTTP (VIP)	Core	TCP/443	HTTPS
Core API (VIP)	Core	TCP/80	HTTP
Core API (VIP)	Core	TCP/443	HTTPS
Core API (VIP)	Core	TCP/4730	Gearman (Queues)
Core RADIUS (VIP)	Core	UDP/162	SNMPTRAP
Core RADIUS (VIP)	Core	UDP/1812	RADIUS
Core RADIUS (VIP)	Core	UDP/1813	RADIUS
Core DHCP (VIP)	Core	UDP/67	IP HELPER
Core DNS (VIP)	Core	UDP/53	DNS
Core DNS (VIP)	Core	TCP/53	DNS

Tabla 1. On core

ON ANALYTICS Y ON AGGREGATOR			
SOURCE	DESTINATION	PORT	SERVICE
Analytics	Core	TCP/22	SSH
Analytics	Sensor	TCP/22	SSH
Analytics	Aggregator	TCP/22	SSH
Analytics	Analytics	TCP/22	SSH
Analytics	Analytics	TCP/9200	ElasticSearch
Analytics	Analytics	TCP/9300	ElasticSearch
Analytics	NTP SERVER	UDP/123	NTP
Analytics	DNS	UDP/53	DNS
Analytics	Proxy Server	TCP/8080	HTTP / HTTPS

ON ANALYTICS Y ON AGGREGATOR			
SOURCE	DESTINATION	PORT	SERVICE
Aggregator	Core	TCP/22	SSH
Aggregator	Sensor	TCP/22	SSH
Aggregator	Analytics	TCP/22	SSH
Aggregator	Aggregator	TCP/22	SSH
Aggregator	Core API (VIP)	TCP/80	HTTP
Aggregator	Core API (VIP)	TCP/443	HTTPS
Aggregator	NTP SERVER	UDP/123	NTP
Aggregator	DNS	UDP/53	DNS
Aggregator	Proxy Server	TCP/8080 [*]	HTTP / HTTPS
Analytics (VIP)	Analytics	TCP/5601	Kibana
Analytics (VIP)	Analytics	TCP/9200	ElasticSearch
Aggregator (VIP)	Aggregator	TCP/5002	SYSLOG

Tabla 2. On analytics on aggregator

ON SENSOR			
SOURCE	DESTINATION	PORT	SERVICE
Sensor	Core	TCP/22	SSH
Sensor	Sensor	TCP/22	SSH
Sensor	Analytics	TCP/22	SSH
Sensor	Aggregator	TCP/22	SSH
Sensor	Analytics	TCP/5000-5015	FileBeat
Sensor	Core Principal (VIP)	TCP/6379	Redis
Sensor	Core Principal (VIP)	TCP/4730	Gearman (Queues)
Sensor	Core API (VIP)	TCP/4730	Gearman (Queues)
Sensor	NTP SERVER	UDP/123	NTP
Sensor	DNS	UDP/53	DNS
Sensor	Proxy Server	TCP/8080	HTTP / HTTPS

Tabla 3. On sensor

6.4 CONFIGURACIÓN DE PROTOCOLOS SEGUROS

Los distintos protocolos de red que utiliza el producto para garantizar la seguridad y la integridad de los datos en tránsito a través de la red vienen configurados por defecto en la OVA.

Para las comunicaciones a nivel de capa de transporte, usaremos TLSv1.2 que viene configurado por defecto. Todas las versiones de SSL /TLS antes de TLSv1.2 ahora están en desuso y se consideran inseguras.

No tenemos el servicio HTTP habilitado en el portal de administración, siempre que intentemos acceder a través del navegador redirigiremos a HTTPS, que usa cifrado de capa TLSv1.2.

Esta configuración la podremos realizar en el fichero de `opennac_ssl.conf` que encontraremos en el directorio `/etc/httpd/conf.d`.

Con la siguiente línea, estaremos diciendo que se deshabilitan todas las versiones que no sean TLSv1.2: `SSLProtocol -all +TLSv1.2`.

Para las conexiones vía SSH, el producto solo soporta SSH2. SSH2 es una versión mucho más segura, eficiente y portable de SSH. Incluye SFTP que esta encriptada con SSH2.

Esta configuración la encontraremos en el fichero de configuración `sshd_config` que encontraremos en el directorio de `/etc/ssh`.

Vemos en este mismo fichero que vienen desactivados aquellos Ciphers que son débiles.

También encontraremos los métodos de intercambio de claves que se utilizan para generar claves por conexión, como, por ejemplo, para la configuración de los ficheros de VPN.

Finalmente, para configurar el producto de modo que utilice los conjuntos de cifrados recomendados según la guía CCN-STIC 807 [REF14], se deberán ejecutar los siguientes comandos:

```
/bin/cp -f /usr/share/opennac/common/hardening/crypto-policies/OPENNAC-HIGH-SECURE.pmod /etc/crypto-policies/policies/modules/OPENNAC-HIGH-SECURE.pmod
update-crypto-policies --set DEFAULT:OPENNAC-SECURE:OPENNAC-HIGH-SECURE
reboot
```

6.5 GESTIÓN DE CERTIFICADOS

Importante, debe tenerse en cuenta que siempre que se trabaje con certificados, estos deberán usar algoritmos y funciones criptográficas admitidos por la guía CCN-STIC-807 [REF14]. Esto significa que sólo podrán usarse los siguientes algoritmos y funciones:

- RSA con claves de, al menos, 3072 bits de longitud.
- ECDSA con curvas P-256 o superior.
- DSA con claves de, al menos, 3072 bits de longitud.
- Funciones Hash SHA-256 o superior.

6.5.1 CERTIFICADO MÓDULO RADIUS

Cuando un proyecto requiere habilitar la autenticación con certificados digitales, o la verificación de la identidad del servidor, debemos pedir al cliente:

- Que nos proporcione la CA, Certificate Authority, completa (root e intermediate en caso de existir).

- Que mediante los archivos CSR generados en OpenNAC Enterprise, nos firmen mediante su CA, certificados de servidor. Para generar los archivos CSR, lo haremos mediante el siguiente comando, donde especificaremos el tipo de cifrado que se va a usar:

```
openssl req -new -newkey rsa:3072 -nodes -keyout opennac_server.key -
out
opennac_server.csr
```

Una vez recibamos los certificados ya firmados por el cliente, deberemos importarlos al servidor.

Tanto los certificados de servidor como la CA, es recomendable tenerlos en formato PEM, y en el caso de una CA en cadena, agruparla entera en un mismo fichero .pem con el orden correcto, empezando con la raíz, seguido de los certificados intermedios que pudiera haber.

Para convertir a formato PEM, podemos usar el siguiente comando:

```
openssl x509 -inform der -in certificate.cer -out certificate.pem
```

Modificar la configuración del módulo Radius para que haga uso de dichos certificados. Se almacenarán los certificados en el directorio `/etc/raddb/certs` y se modificará los ficheros de configuración `/etc/raddb/eap.conf` y el fichero `etc/raddb/modules/inner-eap` con los nuevos parámetros.

Para validar la información de un certificado mediante openssl, podemos ejecutar el siguiente comando:

```
openssl x509 -in aaa_cert.pem -noout -text
```

Para verificar que los certificados desplegados en el módulo de Radius son correctos y están firmados por la CA correspondiente, podemos ejecutar el siguiente comando:

```
openssl verify -verbose -CAfile /etc/radb/certs/ca.pem -untrusted
/etc/radb/certs/server.pem
```

Este comando determinará si el certificado (server.pem) está firmado por la CA (ca.pem), que puede ser simple o en cadena.

Respecto a la funcionalidad de autenticación en la red (NAC) por parte de los usuarios y de los dispositivos de red se realizará mediante el protocolo EAPTLS (RADIUS) utilizando certificados. Por este motivo, la configuración de los certificados de ON Core se llevará a cabo con SHA256 como algoritmo de hash y mecanismos de cifrado RSA (longitud de clave de 3072 bits o superior), de acuerdo con la guía CCN-STIC-807 [REF14]. Consultar el apartado 5.1.10.1.2 "*Internal Certificate Authority*" [REF17].

1. Además, con este mismo fin, y para garantizar que solo se utilice este protocolo, se ejecutarán los siguientes comandos a través de la CLI del sistema operativo:

```
cp -a /etc/raddb/sites-available/default_only_tls_opennac /etc/raddb/sites-
available/default
cd /etc/raddb/mods-enabled/
ln -s ../mods-available/eap_only_tls_opennac /etc/raddb/mods-enabled/
unlink /etc/raddb/mods-enabled/inner-eap_opennac
unlink /etc/raddb/mods-enabled/eap_opennac
unlink /etc/raddb/sites-enabled/inner-tunnel
```

2. El protocolo también debe configurarse para utilizar TLS v1.3. Para ello, hay que comprobar que la propiedad "*tls_max_version*" definida en el archivo "*/etc/raddb/modsavailable/eap_only_tls_opennac*" esté establecida en "*1.3*". Además, el parámetro "*ecdh_curve*" del archivo mencionado deberá especificar "*prime256v1*".

   Tras esto, se debe reiniciar el servicio utilizando el siguiente comando:

```
systemctl restart radiusd
```

Por último, el dispositivo frontera (ejemplo: switch) utilizado en el despliegue deberá configurarse adecuadamente para definir el producto OpenNAC como "*Servidor de autenticación*". Para ello consultar el apartado 9.3 "*Network Devices Compatibility and Configuration*" de la documentación [REF18]. Los usuarios o dispositivos de red, en el esquema RADIUS, actúan como "*supplicant*", intentando acceder a la red protegida a través del "*autenticador*". Estos dispositivos deberán configurarse para presentar sus certificados 802.1X durante el proceso de autenticación. La configuración del "*supplicant*" deberá realizarse siguiendo el apartado 4.2.4.2.2 "*Supplicant configuration for 802.1x Authentication*" de la documentación [REF19].

6.5.2 OTROS CAMPOS DEL CERTIFICADO

La extensión de Key Usage define el propósito (cifrado, firma, firma de certificado) de la clave contenida en el certificado. La restricción de uso podría emplearse cuando se deba restringir una clave que podría usarse para más de una operación.

Esta extensión debe aparecer en certificados que contienen claves públicas que se utilizan para validar firmas digitales en otros certificados de clave pública o CRL. Cuando aparezca esta extensión, debería marcarse como crítica.

El CN (Common Name) representa el nombre del servidor protegido por el certificado SSL. El certificado es válido solo si el nombre de host de la solicitud coincide con el nombre común del certificado. La mayoría de los navegadores web muestran un mensaje de advertencia cuando se conectan a una dirección que no coincide con el nombre común en el certificado.

6.5.3 VALIDACIÓN DE LA VIGENCIA

El healthcheck que se ejecuta en los distintos nodos, se encargará de avisar meses antes de la fecha de la expiración del certificado.

Este parámetro puede configurarse en el script de `check_cert.sh`, por defecto el valor viene configurado a 90 días.

6.5.4 VALIDACIÓN DE REVOCACIÓN

El CRL (Certificate Revocation List) al igual que el OCSP (Online Certificate Status Protocol) nos permiten determinar el estado de vigencia de certificados digitales.

El CRL es un registro utilizado en la operación para mantener un listado de aquellos certificados que han sido revocados y, por tanto, ya no son válidos y en los que no se debería confiar.

Los servidores OCSP se encargan de responder a las peticiones para saber si un certificado sigue vigente o no.

Los certificados los genera el cliente y ellos tienen la gestión, por lo que se encargarán de realizar la validación de revocación de los certificados.

6.5.5 CERTIFICADOS DEL PORTAL DE ADMINISTRACIÓN (HTTPS)

Por último, para que los navegadores web reconozcan el Portal de Administración como un sitio HTTPS de confianza, es necesario cargar certificados de confianza en la configuración del servidor web con SHA256 como algoritmo de hash y mecanismos de cifrado RSA (longitud de clave de 3072 bits o superior), de acuerdo con la guía CCN-STIC 807 [REF14]. Consulte el apartado 5.1.10.2 "*HTTPD Certificates Configuration*" [REF20] con la precaución de modificar el comando relacionado con la generación del certificado para que especifique "*rsa:3072*", de la siguiente manera:

```
openssl req -x509 -nodes -days 11499 -subj '/C=ES/ST=Madrid/L=Madrid/CN=opennac.test' -newkey rsa:3072 -keyout /etc/httpd/ssl/server.key -out /etc/httpd/ssl/server.crt
```

6.6 SERVIDORES DE AUTENTICACIÓN

Como ya se ha comentado anteriormente, para la autenticación de usuarios habitualmente hacemos uso de LDAP o de AD's como repositorio de identidades.

Para la comunicación con LDAP lo haremos preferiblemente, si el cliente lo soporta, mediante LDAPs sobre TLSv1.2.

Para la comunicación entre componentes, cualquier comunicación externa se realizará con HTTPS (mediante TLSv1.2), haremos uso del estándar de SAML.

SAML es un protocolo que define una infraestructura XML para el intercambio de credenciales (autenticación y autorización) entre distintos dominios de seguridad. Comúnmente, estos dominios se dividen en Service Provider (SP) e Identity Provider (IdP).

El SP (rol que asume OpenNAC Enterprise) es el encargado de recibir la conexión del usuario, redirigirlo al IdP y concederle el acceso que se había solicitado.

Por otro lado, el IdP es el que autentica al usuario mediante el medio que tenga establecido. Como consecuencia se genera un token que el usuario deberá entregar al SP para poder acceder al recuso solicitado.

Para que el Service Provider pueda reconocer ese token, se debe hacer una configuración previa entre ambos sistemas llamada federación. Éstos generan las llaves pública y privada y se intercambian junto con unos metadatos, que incluyen la información sobre a qué SP hay que devolver la llamada, etc. La información contenida en los metadatos es importante debido a que un Identity Provider puede servir a diferentes Service Providers.

Usualmente, al autenticar al usuario, la respuesta del IdP va acompañada de una serie de datos (claims) que previamente ha pedido el SP para poder autorizar al usuario en la aplicación.

6.7 SINCRONIZACIÓN HORARIA

Para la configuración horaria de los diferentes nodos, lo haremos mediante el Network Time Protocol (NTP), un protocolo que permite sincronizar los relojes de los sistemas informáticos a través del enrutamiento de paquetes en redes con latencia variable.

Primero debemos parar el servidor NTP antes de modificar sus parámetros, debemos introducir un servidor NTP válido, por ejemplo "hora.roa.es":

```
[root@core ~]# service ntpd stop
[root@core ~]# ntpdate <ip_servidor_ntp>
```

En caso de disponer de un servidor NTP propio se puede configurar. Se modifica el fichero `/etc/ntp.conf` y se incluyen los servidores apropiados:

```
server <ip_servidor_ntp>
```

Se arranca el servicio:

```
[root@core ~]# service ntpd start
```

6.8 ACTUALIZACIONES

Podemos encontrar la descripción de cómo llevar a cabo la actualización de los diferentes nodos en las siguientes referencias:

- Actualización de ON Core [REF21]
- Actualización de ON Analytics [REF22]
- Actualización de ON Aggregator [REF23]
- Actualización de ON Sensor [REF24]
- Actualización de ON Agent [REF25]
- Actualización de ON VPNGW [REF26]

6.9 AUTOCHEQUEOS

El producto no realiza autochequeos (self-test) de integridad.

Los paquetes de OpenNAC Enterprise vienen firmados con nuestra clave o la del proveedor (en este caso Rocky Linux 9).

Al descargar los paquetes, se mirará si la firma se ha modificado o no para así poder validar la descarga.

Todos los paquetes que se instalen desde el repositorio vienen firmados. Si tenemos instalado el paquete `opennac-gpg-key`, podremos garantizar que no se pueda instalar ningún paquete que no esté firmado.

Por lo que garantizamos que, al instalar un nuevo paquete, si ese paquete ha sido modificado o se ha intentado manipular, y no coincide con nuestra firma, por lo que sabremos que es corrupto, asegurándonos de que no instalamos nada que no tenga nuestra firma.

Podemos ejecutar para ver si ha habido alguna modificación:

```
[root@core ~]# rpm -qaV
```

Aunque el producto no realiza comprobaciones de integridad durante su ejecución, este ofrece una monitorización continua de los principales servicios que lo componen y que aportan la funcionalidad de seguridad mediante la configuración de la capacidad "Healthcheck". Se recomienda seguir la sección "9.2.1.1.8 Healthcheck configuration" de la documentación [REF27] para configurar adecuadamente esta característica.

6.10 SNMP

Desde OpenNAC Enterprise by Cipherbit se ofrece compatibilidad con SNMP v1, SNMP v2 y SNMP v3 para la comunicación con distintos dispositivos de red. La configuración se aplica en los dispositivos de red que están registrados en la base de datos de la gestión de configuración (CMDB).

Para configurar la comunicación mediante SNMP v1 y v2, es necesario configurar una "community" de Read-Only (public) y otra de Read-Write (private).

También tenemos la posibilidad de hacer la configuración mediante SNMP v3, protocolo que se recomienda siempre que el dispositivo lo soporte, ya que refuerza las prestaciones de seguridad, incluyendo autentificación, privacidad y control de acceso. Cabe destacar que SNMPv3 no se trata de un estándar que reemplaza a SNMPv1 y/o SNMPv2, sino que define una serie de capacidades adicionales de seguridad y administración a ser utilizadas en conjunción con SNMPv2 (preferiblemente) o SNMPv1.

Soportamos tanto SNMP v1, SNMP v2 y SNMP v3 para que agentes externos puedan monitorizar el comportamiento de los diferentes nodos. Si se desea realizar esta monitorización, haremos un filtrado de ip desde el firewall para que solo nos puedan consultar ciertos dispositivos autorizados, así conseguimos tener más acotada la visibilidad a dispositivos externos.

Por otro lado, un agente SNMP podría también mandar un mensaje a un gestor SNMP sin el envío previo de una solicitud por parte de éste. Este tipo de mensaje es conocido como Traps. Los SNMP Traps son generalmente enviados para reportar eventos. Un agente los envía al servidor de OpenNAC Enterprise cuando se deba enviar cierta información o informar de un problema. OpenNAC Enterprise utiliza SNMP Traps para descubrir diferentes redes y como fuente de información para los casos de uso de visibilidad. Uno de los eventos es "SNMP Traps MAC learn", que nos indica cuándo se ha conectado y dónde está conectado un dispositivo identificando su dirección física (MAC).

6.11 ALTA DISPONIBILIDAD

Contamos con dos configuraciones de alta disponibilidad.

6.11.1 REPLICACIÓN DE MYSQL DIFERENTES NODOS WORKER

Para poder configurar un esquema con un clúster central junto con diferentes nodos de replicación en funcionamiento, deberemos seguir los siguientes pasos:

a. Las configuraciones necesarias para la configuración del cluster, se realizarán siempre a través de la interfaz de administración de los servidores, siempre deberemos trabajar en una red privada de manera que el acceso este limitado solamente al personal autorizado.

b. En primer lugar, deberemos crear una instancia de ON Core que funcione como nodo principal. Una vez el nodo principal esté en funcionamiento, crearemos una instancia (o las que se requieran) de nodo worker.

c. Empezaremos configurando el nodo principal, indicaremos en el archivo `/etc/my.cnf` que se trata del principal.

d. Seguidamente, reanudaremos el servicio de MySQL para aplicar los cambios.

e. Accederemos a MySQL para crear los usuarios únicos de replicación y conceder permisos y privilegios para que los distintos nodos puedan hacer la replicación de la DB.

f. Posteriormente ejecutaremos el comando flush privileges para recargar las tablas de permisos en la base de datos MySQL permitiendo que los cambios surtan efecto sin recargar o reiniciar el servicio MySQL.

Haremos el volcado de información en la base de datos de OpenNAC Enterprise:

a. Deberemos modificar el fichero de iptables en el nodo principal añadiendo las reglas pertinentes para cada nodo worker, de tal manera que se pueda efectuar la replicación. Aplicaremos los cambios necesarios en el fichero de /etc/sysconfig/iptables, donde filtramos por IP y puerto, para abrir el puerto de MySQL (3306) para que los nodos de replicación puedan comunicarse con el nodo principal.

```
-A INPUT -s <worker_ip> -p tcp -m state --state NEW -m tcp --dport 3306
-j ACCEPT
```

b. Una vez configuradas las reglas de firewall, reiniciaremos el servicio de iptables para que se apliquen los cambios:

```
service iptables restart
```

c. Finalmente, enviaremos mediante SCP al worker la base de datos que hemos exportado del nodo principal.

```
scp opennac.sql root@<worker_ip>
```

En los nodos de replicación se debe aplicar la siguiente configuración:

a. Añadiremos la información del nodo principal (nombre de host i IP) en el fichero de /etc/hosts.

b. Importaremos la DB de OpenNAC Enterprise que hemos enviado desde el nodo principal.

c. Se debe modificar también el fichero /etc/my.cnf para indicar que se trata de un nodo worker e indicaremos el serverid pertinente para ese nodo.

d. Seguidamente, haremos reanudaremos el servicio de MySQL para aplicar los cambios.

e. Se accederá a MySQL para conceder permisos y privilegios para que los distintos nodos puedan hacer la replicación de la DB.

f. Finalmente se hará un start slave; para iniciar la ejecución del nodo worker y se reiniciarán todos los servicios para que los nodos empiecen a funcionar con toda normalidad.

```
service opennac start
service gearmand start
```

```
service httpd start
service radiusd start
```

6.11.2 CONFIGURACIÓN DEL CLÚSTER ANALYTICS

El clúster de ON Analytics es el encargado de tratar, almacenar y mostrar tota la información que recibimos por parte de los otros nodos. Para ello, se utiliza el Elastic Stack con sus nodos: Filebeat (envío de datos), Logstash (ingesta y tratamiento de los datos), Elasticsearch (almacenamiento) y Kibana (dashboards).

El clúster de Analytics se compone de: ON Aggregator y ON Analytics. Estos dos roles pueden estar en un mismo servidor o separados.

Esta arquitectura consta de distintos nodos que se muestran a continuación.

6.11.2.1 ON AGGREGATOR – LOGSTASH

Servidor que ejecuta Logstash como servicio principal, que recibe un gran volumen de datos directamente de uno o más servidores ON Core u ON Sensor. Luego, los datos entrantes se manipulan y transforman mediante un conjunto de filtros que ayudan a extraer la información de los eventos de registro y les dan contexto. La información resultante de este procesamiento se reenvía a otros servidores para su almacenamiento.

6.11.2.2 ON ANALYTICS – ELASTICSEARCH

Este nodo ejecuta Elasticsearch, el motor analítico y de búsqueda de texto completo. Los servidores de On Analytics reciben datos de On Aggregators, los indexan y los almacenan de forma distribuida, lo que permite consultar la información y mostrarla en forma de gráficos o informes.

6.11.2.3 ON ANALYTICS – KIBANA

Este nodo ejecuta Kibana, este realiza la presentación visual de la información en forma de paneles o informes. Los servidores ON Core pueden conectarse directamente con Kibana y mostrar los paneles a través de la interfaz de administración web.

Añadiremos en el fichero de `/etc/hosts` la información de los distintos nodos de la arquitectura añadiendo el nombre de host y la IP asociada.

Los nodos de On Analytics ya tienen una configuración predefinida de reglas de firewall con las iptables necesarias para permitir la conexión a ciertos servicios. Sin embargo, se necesitan reglas adicionales según el nodo que se configure en los servidores.

Para la comunicación entre los nodos del clúster deben comunicarse entre sí a través del puerto TCP/9300.

Se deberá modificar el fichero de iptables para permitir la comunicación entre los distintos nodos. Aplicaremos los cambios necesarios en el fichero de `/etc/sysconfig/iptables`, donde filtramos por IP y puerto:

```
-s <node_ip> -A INPUT -p tcp -m state --state NEW -m tcp --dport 9300-
j ACCEPT
```

Una vez configuradas las reglas de firewall, reiniciaremos el servicio de iptables para que se apliquen los cambios:

```
service iptables restart
```

6.11.2.4 SEGURIDAD DE COMUNICACIÓN ENTRE NODOS

El envío de la información desde ON Core/ON Sensor hacia ON Analytics, se realiza mediante Filebeat (transmisión) y Logstash (recepción). Para securizar esta comunicación se hace uso de certificados para validar la identidad de cada uno de los servidores. Esta verificación se realiza mediante TLSv1.2 y nos permite cifrar la comunicación entre ambos.

Para más detalle sobre la configuración de seguridad en las comunicaciones del Elastic Stack, (Logstash-Elasticsearch-Kibana) podemos consultar la sección 2.7 “Connections Security” de la documentación [REF28].

6.12 AUDITORÍA

En este apartado se explicará cómo se registran y almacena los eventos dentro del sistema para verificar el funcionamiento de los distintos nodos de la solución.

6.12.1 REGISTRO DE EVENTOS

Básicamente se registran dos tipos de logs que podemos clasificar en:

- **Logs de sistema:** Registran información relacionada con el funcionamiento del sistema operativo. Algunos ejemplos de logs de sistema son los que registran información sobre los servicios, los que registran los accesos al equipo, los mensajes del sistema, etc.
- **Logs de programas:** Registran cronológicamente los eventos más importantes mientras estamos usando una aplicación o programa. En este caso, los logs pueden ser generados por la propia aplicación o por Rsyslogd.

La mayor parte de los logs del sistema se almacenan en directorio `/var/log` y en diferentes subdirectorios dentro de este. En su interior podemos encontrar una serie de ficheros que almacenarán los logs de diferentes aplicaciones o dispositivos del sistema, la extensión de estos archivos será `.log`.

Dentro de estas carpetas también encontraremos estos mismos ficheros de log con la extensión `.gz`. Se trata de ficheros de log antiguos comprimidos y almacenados por el sistema durante un tiempo establecido para su consulta, siguen el siguiente formato (Nombre del log junto a la fecha de creación de dicho registro comprimido):

```
redis.log-20200521.gz
```

Además de los diferentes logs del sistema y de las diferentes aplicaciones, los logs que se registran destinados a la auditoría, los encontramos en `/var/log/opennac`.

```
opennac-analytics.log
```

Este registro contiene la información relacionada con los dispositivos. Esto se almacena como json. Este registro se envía al ON Analytics para que pueda tratar la información registrada.

```
opennac-audit.log
```

Registra todos los eventos generados por los usuarios en OpenNAC Enterprise. Contiene información relacionada con los cambios / acciones realizadas por un usuario a través del portal de administración de OpenNAC Enterprise.

```
opennac-captive-analytics.log
```

Este registro contiene información relacionada con el portal cautivo que luego se enviará a los análisis de OpenNAC Enterprise. Esta información podría ser el acceso del usuario y la ruta que siguió (como WebAuth_Guest_Email) dentro del portal cautivo.

```
opennac-nd-analytics.log
```

Contiene información relacionada con los dispositivos de red. Esta información se almacena como un json y luego se envía al ON Analytics para que pueda tratar la información registrada.

```
opennac-poleval-audit.log
```

Este registro contiene la información relacionada con la auditoría de la ejecución de la evaluación de políticas.

Hay algo interesante en este registro llamado Fake Evals. Después de agregar / eliminar una etiqueta a un dispositivo, se realiza una evaluación de política falsa para evaluar si ese dispositivo, con estos cambios, debería cambiar su VLAN o no, o si coincide con otra política o no. Si la respuesta es positiva, si coincide con otra categoría y tiene que cambiar de VLAN, se fuerza una verdadera evaluación de la política. Se realiza una solicitud de puerto de alternancia y la regla se vuelve a evaluar para que el dispositivo coincida con la política adecuada.

```
opennac-poleval.log
```

Este registro contiene la información relacionada con los resultados de las evaluaciones de políticas. Contiene datos como el MAC del dispositivo, el estado de la evaluación y el tiempo que tardó en completarse.

6.12.2 ALMACENAMIENTO LOCAL

El producto realiza un almacenamiento local de los registros de auditoría. Estos logs se almacenan de forma local y se envían al ON Analytics que se encargará de tratar los datos recibidos, la comunicación se hace mediante el protocolo TLS 1.2.

La información con la que trabaja ON Analytics se gestiona mediante el elastic stack con sus nodos de Elasticsearch, Logstash, Kibana y Beats.

Si se quiere recolectar más información y proporcionar visibilidad avanzada y seguimiento histórico del comportamiento de dispositivos en la red, usaremos opcionalmente ON Sensor, este nodo decodifica los protocolos y rastrea todas las actividades de la red donde se implementa. El sensor se implementa cerrado al tráfico de clientes y solo se puede instalar en formato onpremise.

Para controlar el comportamiento en caso de que se alcance el límite de almacenamiento haremos uso de la herramienta de `logrotate`.

El logrotate es un proceso que se encarga de rotar, comprimir y enviar los logs del sistema en Linux.

Cuando un paquete individual (httpd, mysqld, radiusd, yum, etc.) se instala, éste añade en el directorio `/etc/logrotate.d/` su fichero de configuración logrotate encargado de la gestión de los logs.

En el directorio de `/etc/logrotate.d/` encontraremos la configuración del log de los distintos servicios que se ejecutan en el sistema, donde podremos configurar de manera independiente cada uno de ellos. Si alguno no tuviera aquí su configuración tomaría la del fichero anterior.

Por otro lado, tendremos el fichero `/etc/logrotate.conf`, donde se establece la configuración base para los diferentes paquetes de los servicios que queramos rotar.

En este fichero podemos configurar las políticas de rotación comunes para todos los logs. En cada fichero de configuración dentro de `/etc/logrotate.d` se pueden sobrescribir las políticas comunes, de forma local, redefiniendo el valor para la variable de configuración deseada.

El sistema almacena los logs antiguos, pero no indefinidamente. La configuración general del servicio la podemos establecer en el fichero de `logrotate.conf`, para indicar cada cuánto tiempo creamos un fichero de log y cuánto tiempo lo dejamos sin eliminar en el sistema. Algunos de los parámetros a utilizar son:

- daily | weekly | monthly | yearly = Cada cuánto se crea un fichero de log nuevo.
- rotate 4 = Cuántos ficheros de log se mantienen máximo.
- compress = Comprimir los ficheros de log rotados.

Para la configuración de la rotación de los registros de auditoría, consultar el apartado 7.6 "*Adjust Logrotate*" de la documentación [REF29]. En concreto, para este producto cualificado, se configurarán 30 rotaciones con un límite de tamaño de 2MB. Para definir el límite de tamaño de los registros, se debe incluir la directiva "maxsize" en el archivo de configuración. Concretamente, se debe añadir una línea que indique "maxsize 2M" a las siguientes directivas de archivos de registro:

- *"/etc/logrotate.d/opennac",* que afecta a los siguientes archivos de registro:
  - */var/log/opennac/backup.log*
  - */var/log/opennac/cron.log*
  - */var/log/opennac/opennac-*.log*
  - */var/log/opennac/php.log*
- *"/etc/logrotate.d/opennac-healthcheck"*, que afecta a los siguientes archivos de registro:
  - *"/var/log/opennac/healthcheck.log"*
  - *"/var/log/opennac/health_syslog.log"*
- *"/etc/logrotate.d/opennac-radius"*, que afecta a los siguientes archivos de registro:
  - *"/var/log/opennac/ntlm_auth_exec_time*.log"*
  - *"/var/log/opennac/opennac_otp_auth.log"*
  - *"/var/log/opennac/radius-crlupdate.log"*
  - *"/var/log/opennac/radius-radacct-purge.log"*

6.12.3 ALMACENAMIENTO REMOTO

Desde OpenNAC Enterprise, además de almacenar los registros (audit), se pueden enviar registros a servidores remotos. Para ello usaremos las siguientes herramientas externas a OpenNAC:

- **Syslogd:** Permite ser configurado para escuchar y aceptar conexiones remotas, lo que implica poder recibir datos y almacenarlos de clientes externos (syslog de otros servidores). Esto es perfecto para crear un servidor syslog central y enviarle todos los logs de otros servidores, con la finalidad de tenerlos y gestionarlos todos en el mismo sitio.
- **Filebeat:** Nos permitirá enviar todos los logs e información que se hayan generado en el nodo ON Core hacia el servicio Logstash del nodo ON Analytics para que la información se procese (Logstash), almacene (Elasticsearch) y se muestre al usuario (Kibana).
- **Logstash:** Esta herramienta nos permitirá recolectar y procesar los datos que recibimos del ON Core a través de Beats. En Logstash, los datos se filtran y moldean para posteriormente mandarlos a Elasticsearch.
- **Elasticsearch:** Cundo los datos se han moldeado con Logstash, llegan a Elasticsearch donde son indexados y almacenados para que el usuario pueda ejecutar consultas y Kibana pueda generar los dashboards a partir de estos datos.
- **Kibana:** Kibana generará los dashboards con diferentes tipos de gráficos y tablas a partir de los datos que extrae de Elasticsearch. Por ejemplo, en el índice opennac-* la información se elimina cuando lleva 15 días almacenada.
- **NXLog:** es una herramienta de administración de registros que nos permite analizar los problemas operativos en los registros del servidor, los registros del sistema operativo, los registros de las aplicaciones, etc. El archivo de configuración de NXLog se compone de distintos bloques y directivas. Para poder leer y recibir los distintos registros a través de la red, deberemos configurar diferentes módulos.

Los pasos que deben seguirse para instalar y configurar el servicio RSyslog en el componente ON Core son los siguientes (para obtener más información consultar el apartado 5.1.2.3 "*Sending OpenNAC logs to SIEM*" de la documentación [REF30]):

- Instalar RSyslog con TLS:

```
dnf install -y rsyslog-gnutls gnutls-utils
```

- Crear los certificados con OpenSLL:

    g. Crear la CA:

```
openssl genrsa -out ca.key 4096
openssl req -x509 -new -nodes -key ca.key -sha256 -days 3650 -out ca.pem -subj "/C=ES/ST=Barcelona/L=Terrassa/O=OpenNAC/CN=OpenNAC-CA"
```

    h. Crear el certificado de cliente RSyslog:

```
openssl genrsa -out rsyslog-key.pem 3072
openssl req -new -key rsyslog-key.pem -out rsyslog.csr -subj "/C=ES/ST=Barcelona/L=Terrassa/O=OpenNAC/CN=rsyslog-client"
openssl x509 -req -in rsyslog.csr -CA ca.pem -CAkey ca.key -CAcreateserial -out rsyslog-cert.pem -days 365 -sha256
```

    i. Crear el certificado del servidor Logstash:

```
openssl genrsa -out logstash-key.pem 3072
openssl req -new -key logstash-key.pem -out logstash.csr -subj "/C=ES/ST=Barcelona/L=Terrassa/O=SIEM/CN=siem.mydomain.local"
openssl x509 -req -in logstash.csr -CA ca.pem -CAkey ca.key -CAcreateserial -out logstash-cert.pem -days 365 -sha256
```

- Crear los directorios de certificados y aplicar los permisos necesarios:
- Configuración de RSyslog mediante la creación del archivo "*/etc/rsyslog.d/0-siem.conf*" con el siguiente contenido:

Nota: En la siguiente configuración, el valor "target" debe identificar al servidor Syslog remoto, y el campo "Common Name" (CN) del certificado del servidor presentado durante el protocolo de enlace TLS debe coincidir con este valor.

```
# TLS global configuration
global(
  DefaultNetstreamDriver="gtls"
  DefaultNetstreamDriverCAFile="/etc/rsyslog.certs/ca.pem"
  DefaultNetstreamDriverCertFile="/etc/rsyslog.certs/rsyslog-cert.pem"
  DefaultNetstreamDriverKeyFile="/etc/rsyslog.certs/rsyslog-key.pem"
)

module(load="imfile")

input(type="imfile"
File="/var/log/httpd/opennac-*_log"
Tag="opennac-apache"
Severity="info"
Facility="local5"
reopenOnTruncate="on"
)
input(type="imfile"
File="/var/log/radius/radius.log"
Tag="opennac-radius"
Severity="info"
Facility="local5"
reopenOnTruncate="on"
)

# Send ALL logs to the SIEM via TLS with a persistent queue
if ($programname startswith "opennac") or ($programname == "simplesamlphp")
then {
  action(
    type="omfwd"
    target="logstash.mydomain.local"
    port="9222"
    protocol="tcp"
    StreamDriver="gtls"
    StreamDriverMode="1"
    StreamDriverAuthMode="x509/name"
    gnutlsPriorityString="@SYSTEM:-RSA:-GROUP-FFDHE2048"
    template="RSYSLOG_FileFormat"
```

```
    # Persistent queue for resilience
        queue.type="LinkedList"
        queue.filename="siem_tls_queue"
        queue.maxdiskspace="1g"
        queue.size="100000"
        queue.dequeuebatchsize="1000"
        queue.highwatermark="90000"
        queue.lowwatermark="20000"
        queue.saveonshutdown="on"
        action.resumeRetryCount="-1"
      )
    }

    if ($programname == "opennac-apache") or ($programname == "opennac-radius")
    then {
    stop
}
```

- Asegurarse que el nombre de host del certificado SIEM se pueda resolver, ya sea mediante una consulta DNS o añadiendo el nombre de host al archivo "/etc/hosts", por ejemplo:

```
echo "10.10.39.106 siem.mydomain.local" >> /etc/hosts
```

- Reiniciar el servicio Rsyslog:

```
systemctl restart rsyslog
systemctl status rsyslog
```

- Verificar la conexión TLS:

```
gnutls-cli --priority=NORMAL --x509certfile=/etc/rsyslog.certs/rsyslog-cert.pem --
x509keyfile=/etc/rsyslog.certs/rsyslog-key.pem --
x509cafile=/etc/rsyslog.certs/ca.pem siem.mydomain.local -p 9222
```

6.13 BACKUP

Los backups del sistema se generan de forma local en los distintos nodos, estos backups se pueden exportar, por ejemplo, mediante SCP para que se puedan gestionar de forma externa.

Los backups se realizan mediante un script donde configuraremos parámetros como:

- La carpeta / directorio en el que se desea hacer el backup.
- El tiempo (en días) de retención del backup.
- La carpeta de destino en la que almacenaremos el backup.
- El nombre del fichero.
- Número de logs que queremos retener.
- Si se quiere hacer o no el volcado de la base de datos

Por defecto la retención de los ficheros de backup es de los 7 días anteriores, estos ficheros se almacenarán por defecto en el directorio de `/backup`.

El formato de los ficheros sigue este formato:

```
opennac-<nombre_máquina>-yyyymmdd_hhmmss.tgz
```

6.14 SERVICIOS DE SEGURIDAD

Dependiendo del caso de uso que se quiera adoptar con la solución, se desplegarán unos u otros componentes de la arquitectura, tal y como se expondrá en las siguientes subsecciones.

Un ejemplo de arquitectura completa, incluyendo la implementación UNAC, el caso de uso de visibilidad y el uso de VPN, sería el siguiente:

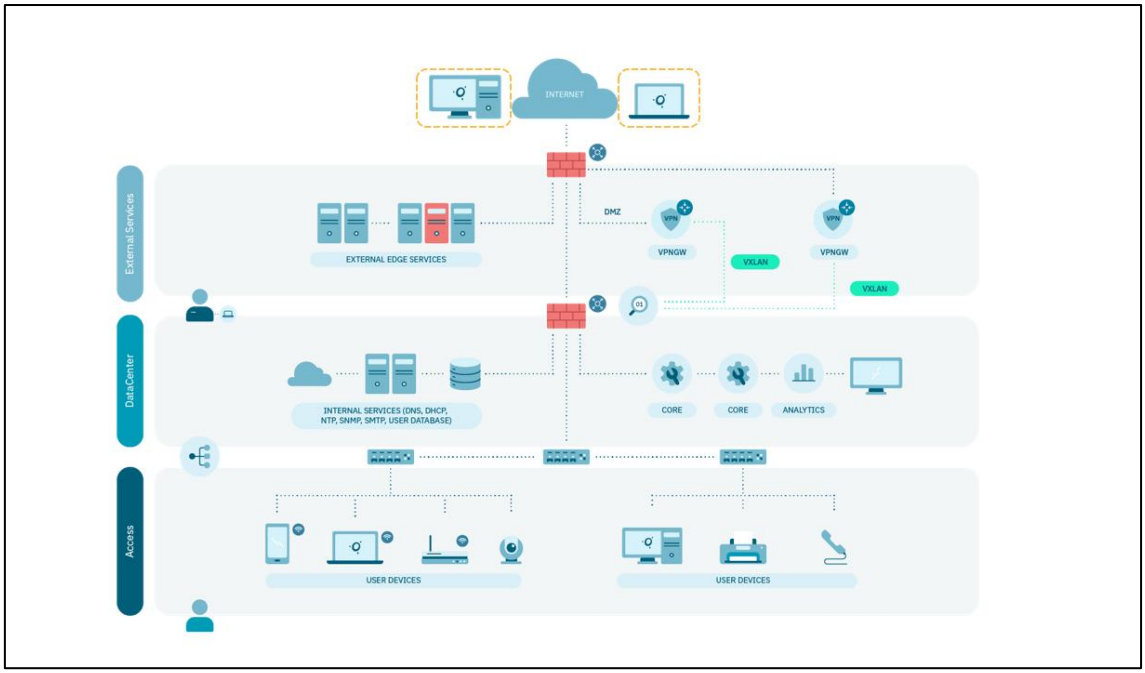


Ilustración 3. Arquitectura con UNAC y caso de uso de visibilidad

6.14.1 CASO DE USO – UNAC

El caso de uso de UNAC o control de acceso a la red, nos permite controlar que dispositivos pueden acceder a la red, cuales no y determinar a qué vlan debe pertenecer cada uno de ellos.

Cuando un dispositivo se conecta a la red corporativa, es necesario que este se autentique utilizando, por ejemplo, 802.1x. Dicha autenticación será validada por el servidor RADIUS localizado en ON Core.

Dependiendo de la política a la que aplica la conexión del dispositivo, este se asignará a un segmento de la red o VLAN (autorización y segmentación), ON Sensor monitorizará la red y ON Analytics se encargará de recibir, normalizar y almacenar todos los datos generados por ON Core y ON Sensor, utilizando el elastic stack para que los administradores de red puedan visualizar esta información a través de Kibana en el portal de administración.

6.14.2 CASO DE USO – VISIBILITY

Visibility nos permite obtener una visión global y profunda de los activos conectados a la red, así como el tráfico que se genera en ella.

Con el caso de uso de visibilidad, se realizará el descubrimiento, categorización y perfilado de los activos de la red.

Este caso de uso requiere del despliegue de los nodos ON Core y ON Analytics. De forma adicional, es posible utilizar ON Sensor para obtener mayor cantidad de tráfico y ON Agent en los equipos clientes para obtener más información sobre los equipos conectados a la red.

El proceso de visibility consta de dos fases principales:

- **Discovery:** Proceso de descubrimiento de dispositivos conectados a la red, consiste en detectar una IP o MAC en la red no vista antes.
- **Profiling:** Cuando se detecta un nuevo dispositivo, este debe ser perfilado, es decir, determinar el tipo de dispositivo de que se trata (DESKTOP, MOBILE, PRINTER, etc). Cuando se haya perfilado mediante las reglas de perfilado, se añadirá un tag EPT_<TIPO_DISPOSITIVO>.

Al conectar un dispositivo, este es detectado mediante alguno de los métodos de visibility (Network Device, Agente o a través de ON Sensor). Una vez hemos descubierto el dispositivo, se ejecutarán, en caso de estar configurados, los plugins que nos permitirán obtener información relativa al dispositivo y finalmente se le aplicarán las reglas de perfilado, donde como resultado se aplicará un tag al dispositivo que nos indicará su tipo.

7 REFERENCIAS

[REF1] Casos de Uso del producto

[REF2] Enlace Repositorio oficial

[REF3] Instalación de licencia

[REF4] Herramienta Openscap

[REF5] Despliegue OVA Proxmox

[REF6] Instalación y configuración de ON Core

[REF7] Instalación y configuración de ON Analytics

[REF8] Instalación y configuración de ON Aggregator

[REF9] Instalación y Configuración ON Sensor

[REF10] Instalación y Configuración ON Agent

[REF11] Instalación y configuración ON VPNGW

[REF12] Contraseña SSH

[REF13] Contraseña Portal de Administración

[REF14] Guía CCN-STIC-807

[REF15] Configuración de SMTP Mail Relay

[REF16] Configuración de OTP

[REF17] Configuración CA Interna

[REF18] Configuración y Compatibilidad de Dispositivos de Red

[REF19] Configuración del "Supplicant" para Autenticación 802.1x

[REF20] Configuración Certificados HTTPD

[REF21] Actualización de ON Core

[REF22] Actualización de ON Analytics

[REF23] Actualización de ON Aggregator

[REF24] Actualización de ON Sensor

[REF25] Actualización de ON Agent

[REF26] Actualización de ON VPNGW

[REF27] Configuración Healthcheck

[REF28] Seguridad en las conexiones

[REF29] Configuración Logrotate

[REF30] Configuración envío logs OpenNAC a SIEM

[REF31] CPSTIC

8 ABREVIATURAS

ACL	Access Control List
AD	Active Directory
AES	Advanced Encryption Standard
API	Application Programming Interface
CPD	Centro de Procesamiento de Datos
CA	Certificate Authority
CRL	Certificate Revocation List
CSR	Certificate Signing Request
CoA	Change of Authorization
CLI	Command Line Interface
CMDB	Configuration Management Database
CRM	Customer Relationship Management
DSA	Digital Signature Algorithm
DNS	Domain Name System
DHCP	Dynamic Host Configuration Protocol
ECDSA	Elliptic Curve Digital Signature Algorithm
FTP	File Transfer Protocol
GPG	GNU Privacy Guard
GUI	Graphical User Interface
HA	High Availability
HKDF	HMAC-based Extract-and-Expand Key Derivation Function
HTTPS	HyperText Transfer Protocol Secure
IdP	Identity Provider
LDAP	Lightweight Directory Access Protocol
LAN	Local Area Network
WireGuard	Modern VPN protocol
NAC	Network Access Control
NTP	Network Time Protocol
OTP	One-Time Password
OCSP	Online Certificate Status Protocol
OpenVPN	Open Source Virtual Private Network
OVA	Open Virtual Appliance
ON	OpenNAC

PEM	Privacy-Enhanced Mail (certificate format)
RPM	Red Hat Package Manager
RADIUS	Remote Authentication Dial-In User Service
RSA	Rivest-Shamir-Adleman (cryptographic algorithm)
SCP	Secure Copy Protocol
SHA	Secure Hash Algorithm
SSH	Secure Shell
SSL	Secure Sockets Layer
SAML	Security Assertion Markup Language
SCAP	Security Content Automation Protocol
SP	Service Provider
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SO	Sistema Operativo
Syslog	System Logging Protocol
TTL	Time To Live
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UNAC	Unified Network Access Control
UDP	User Datagram Protocol
VXLAN	Virtual Extensible LAN
VPN	Virtual Private Network
VPNGW	VPN Gateway
WAN	Wide Area Network