centro criptológico nacional

Guía de Seguridad de las TIC
CCN-STIC 1105

Procedimiento de empleo seguro
EMMA

Julio de 2021

MINISTERIO DE DEFENSA

Catálogo de Publicaciones de la Administración General del Estado
https://cpage.mpr.gob.es

Edita:

MINISTERIO DE DEFENSA
SUBSECRETARÍA DE DEFENSA
SECRETARÍA GENERAL TÉCNICA

centro criptológico nacional

CENTRO CRIPTOLOGICO NACIONAL
cn=CENTRO CRIPTOLOGICO NACIONAL, 2.5.4.97=VATES-S2800155J, ou=CENTRO CRIPTOLOGICO NACIONAL, o=CENTRO CRIPTOLOGICO NACIONAL, c=ES
2022.04.05 18:08:52 +02'00'

Pº de la Castellana 109, 28046 Madrid

NIPO: 083-21-133-8

Fecha de Edición: abril de 2022

Open Cloud Factory ha participado en la realización y modificación del presente documento.

ÍNDICE

1. INTRODUCCIÓN

1. EMMA es una plataforma de control de acceso a la red (NAC) para entornos LAN y WAN que permite a las organizaciones 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.
2. Es una solución de *software* que facilita a la organización la visibilidad y control de los activos (dispositivos + identidades) conectados a la red (cableada, *wifi* y VPN).
3. Cada caso de uso requiere configurar un conjunto de nodos para que se comuniquen entre sí. Este documento pretende mostrar cómo 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.
4. Para encontrar información más detallada sobre los diferentes casos de uso del producto, así como los requisitos para cada caso de uso, se puede consultar el documento Despliegue de Nodos.

2. OBJETO Y ALCANCE

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

6. Se puede encontrar información más detallada sobre los nodos y casos de uso en las siguientes guías:

   a) Despliegue de Nodos

   b) Cumplimiento – Operación

   c) Visibilidad – Instalación

   d) Visibilidad – Operación

   e) VPN – Instalación

   f) VPN – Operación

7. El presente documento se ha realizado para la versión 1.2.1 del producto. El HW no influye en el documento y el SO dependerá de la OVA instalada.

3. ORGANIZACIÓN DEL DOCUMENTO

8. En este apartado se pretende dar una breve introducción de los apartados que se encuentran en el documento.

   a) Apartado 4. FASE DE DESPLIEGUE E INSTALACIÓN

   En este apartado se explicará cómo se realiza la entrega de los productos, tanto los nodos de software como de hardware. También se darán recomendaciones a tener en cuenta durante la fase de despliegue e instalación del producto.

   - Se explicará la gestión y el registro de las licencias que se deberán generar e incluir en los distintos nodos.
   - También se recogen las recomendaciones a tener en cuenta acerca del entorno de instalación de los nodos, así como las consideraciones previas a tener en cuenta antes de realizar la instalación.

   b) Apartado 5. FASE DE CONFIGURACIÓN

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

   - Se indicarán los diferentes métodos de autenticación implementados, tanto para la autenticación de usuarios como entre los distintos nodos.
   - Se indica cómo se realiza la administración tanto local como remota de los distintos nodos y la configuración de los administradores, así como la creación y gestión de los diferentes roles que se pueden configurar.
   - Se explicará qué servicios y puertos vienen configurados por defecto y cómo se pueden configurar, así como la configuración de las distintas interfaces para los distintos nodos.
   - Cómo se lleva a cabo la gestión de certificados, cómo generarlos y cómo llevar a cabo el control de vigencia y revocación.
   - La configuración de los distintos servidores de autenticación que usa el producto.
   - Se detallará la configuración del protocolo SNMP (*Simple Network Management Protocol*) en distintos casos de uso.
   - Se explicarán los entornos de HA (*High Availability*).
   - Los procesos de registro de eventos y cómo se efectúa este registro, su almacenamiento local así como el almacenamiento remoto, dónde se explicarán los diferentes estándares para el envío de estos a servidores externos. También se explicará cómo se efectúan los backups del sistema.

c) Apartado 6. FASE DE OPERACIÓN

En este apartado se recogen las tareas recomendadas para la fase de operación o mantenimiento del producto.

4. FASE DE DESPLIEGUE E INSTALACIÓN

9. En este apartado se explicará cómo se realiza la entrega de los productos, tanto los nodos *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

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

    a) **En el caso de los productos de software**, el fabricante se pone en contacto con el cliente o *partner* certificado para facilitarles el enlace de un repositorio donde podrán descargar los nodos necesarios para el caso de uso deseado. Para que puedan tener acceso, se generarán explícitamente unas credenciales para que los clientes o *partners* puedan acceder por HTTPS al repositorio y descargar la OVA correspondiente, el envío de dichas credenciales se realizara vía email con destinatarios únicos y mails separados. El usuario y contraseña nunca se enviarán en el mismo mail. Es decir, la descarga de cualquier nodo se hará siempre bajo la autorización por parte del equipo de *Profesional Services* o de pre-venta de *Open Cloud Factory***.**

    b) **En el caso de los nodos de hardware**, como el Intel NUC (*AllinOne*), la configuración se realiza a nivel interno por el equipo de *Professional Services.* Una vez configurados se envían directamente al cliente para su puesta en marcha en la infraestructura. El producto se entrega en la caja original con el NUC ya embalado. El proceso de envío se gestiona a través de la administración Open Cloud Factory, que se encarga de realizar las gestiones con la compañía de transporte**.**

4.2 ENTORNO DE INSTALACIÓN SEGURO

11. La instalación física de los nodos la gestiona el cliente bajo su responsabilidad.

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

13. La instalación de algunos de los nodos de la solución se puede realizar sobre un centro de procesamiento de datos (CPD) *on-premise* o en un entorno *cloud*. **La versión cualificada es la que se despliega *on-premise*.**

14. 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.

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

4.3 REGISTRO Y LICENCIAS

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

    - En el caso de ser un proyecto nuevo, el fabricante recibe una petición por parte del cliente, en la que se especifican los detalles que se deben configurar.
    - Se deben tener el nombre y el correo de la empresa, así como el tiempo (en días) que se quiera definir. Una vez se conocen los datos se generará la licencia con la configuración pertinente.
    - 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 *Open Cloud Factory* *(license@opencloudfactory.com)* para que se realice el registro.
    - 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.
    - En los proyectos que están en producción donde ya se tienen licencias registradas e instaladas, el sistema de chequeo de EMMA 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.
    - Una vez se notifique la necesidad de renovar la licencia, se genera y se envía a cliente para instalarla.

17. Se puede ver el proceso para la instalación de la licencia en el documento *'Despliegue de Nodos'*.

4.4 CONSIDERACIONES PREVIAS

18. Antes de iniciar la instalación y configuración del producto, se debe tener en cuenta ciertos aspectos, puesto que influirán en la forma de configurarlo de forma segura.
19. Las tecnologías de acceso a la red requieren una segmentación de red adecuada, tener un diseño claro ayudará a implementar una solución NAC.
20. 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.

21. Una vez se conozcan las especificaciones del cliente, se revisa la información referente a la electrónica de red, fabricante, modelos, versión, etc. para ver la compatibilidad con nuestros nodos.

22. 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 Open Cloud Factory.

23. 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 EMMA.
        - 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. (se detalla 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 se deben dejar en los diferentes servidores que lo requieran.
    - Definir qué nodos se van a instalar y 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.

4.5 INSTALACIÓN

24. Se puede encontrar la descripción de cómo llevar a cabo la instalación de los diferentes nodos en el documento de 'Despliegue de Nodos'.

Sección	Título de la sección
4	*Instalación y configuración de ON Core*
5	*Instalación y configuración de ON Analytics*
6	*Instalación y Configuración ON Sensor*
7	*Instalación y Configuración ON Agent*
8	*Instalación y configuración de la CMI*
9	*Instalación y configuración de la CMIX*

5. FASE DE CONFIGURACIÓN

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

5.1 AUTENTICACIÓN

5.1.1 AUTENTICACIÓN DE USUARIOS

26. En lo que se refiere a la autenticación de usuarios mediante 802.1x (en los casos de UNAC y VPN) se debe tener en cuenta que desde EMMA, se puede autenticar tanto a usuarios locales (se tiene 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.

27. 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 configurados.

28. El orden por defecto que se sigue es primero Usuarios locales, luego LDAP/AD.

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

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

31. En el caso de los usuarios externos, como desde EMMA no se tiene acceso a las credenciales de dicho usuario, se mandará la petición al servidor externo para ver si tiene registrado al usuario y si las credenciales coinciden.

32. En el caso del acceso al portal de administración web, cada usuario tendrá un rol asignado (se ve la descripción de los diferentes roles y su configuración en el apartado 5.2.2 CONFIGURACIÓN DE ADMINISTRADORES).

33. 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. EMMA filtra según los usuarios o grupos que estén configurados en el servidor externo, asignando a ciertos usuarios o a grupos usuarios un rol determinado.

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

34. En EMMA ciertos nodos que requieren autenticación para que puedan comunicarse entre ellos: la API y el *Captive Portal.*

35. 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.

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

37. La *API key* se genera desde el portal de administración (cada API key se genera para una dirección IP origen definida) y se añade en el fichero de configuración de la segunda máquina, así se puede verificar que la fuente con la que se intercambia información es fiable.

38. En lo que se refiere a comunicación con dispositivos externos, como dispositivos de red, se establece comunicación 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*). La conexión a los dispositivos se realiza por SSH (usando SSH v2).

5.1.3 AUTENTICACIÓN DE SESIONES

39. Se puede acceder a los nodos vía SSH tanto por clave como por certificado, aunque este se puede desactivar para que solo se pueda acceder mediante el uso de claves.

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

41. Para acceder al portal, se puede habilitar el uso de OTP como doble factor de autenticación.

5.2 ADMINISTRACIÓN DEL PRODUCTO

5.2.1 ADMINISTRACIÓN LOCAL Y REMOTA

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

43. Para la administración remota, se accede a la red privada del cliente con la VPN para poder acceder a los diferentes nodos.

44. A nivel interno, a la hora acceder vía CLI, **no se debe utilizar el servicio de telnet** ya que se expondría toda la información intercambiada con los servidores, con el consiguiente riesgo que esto supone. Por defecto, telnet no está habilitado en ninguno de los nodos de la solución, las conexiones vía CLI las se establecen mediante el SSH v2.

45. Otra práctica 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.

46. Para acceder mediante la interfaz gráfica de usuario, se usa el protocolo HTTPS.

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

5.2.2 CONFIGURACIÓN DE ADMINISTRADORES

48. 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 EMMA.

49. 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.

50. Por ejemplo, se puede 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.

51. Se pueden ver los roles configurados en *ON CMDB -> Security -> Role*s. Por defecto el producto viene con cuatro roles configurados:

    a) *Administrador*: Cuenta con todos los permisos de administración y gestión (no se puede modificar sus listas de control de acceso, ACL, ya que vienen definidas).

    b) *User*: Cuenta con los permisos básicos para la navegación dentro del portal de administración web a nivel de lectura (actualmente este rol está obsoleto ya que ha sido reemplazado por el de *readonly*).

    c) *Otpmanager*: Cuenta con permisos de acceso a la configuración de OTP.

    d) *Readonly*: Cuenta con permisos para visualizar todas las ventanas del menú.

52. Se pueden modificar los permisos de cada uno de los roles que ya hay configurados por defecto excepto el rol de administrador.

53. Para ello, se modifican los permisos, ACL, 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.

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

55. Una vez definidos los diferentes roles, se pueden asignar a los nuevos usuarios que se creen.

56. Al crear un nuevo usuario, también se debe tener en cuenta la política de configuración segura de contraseñas. Se recomienda que las contraseñas cumplan con los siguientes requisitos mínimos:

a) Longitud de la contraseña: mínimo 8 caracteres.

b) Complejidad: uno o más caracteres en minúscula, uno o más caracteres en mayúscula, uno o más números, uno o más caracteres especiales,

c) El cambio de contraseña debe realizarse periódicamente.

57. Respecto a la configuración de los parámetros de sesión:

a) *Timeout* de autenticación: es el tiempo que permanece el usuario autenticado en la sesión, transcurrido este tiempo, el usuario debe volver a autenticarse.

b) El *timeout* de inactividad: hace referencia al tiempo que puede permanecer una sesión inactiva, se define a una hora, una vez haya trascurrido ese tiempo, se realiza una desconexión automática de esa sesión.

c) 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.

d) Número máximo de intentos fallidos de autenticación, y tiempo de espera tras superar un umbral. Al superar los cinco intentos fallidos de autenticación a la hora de conectarnos a la aplicación, se bloquea el acceso de ese usuario durante cinco minutos, una vez transcurrido ese tiempo, el usuario volverá a tener cinco intentos para realizar la autenticación.

e) Está definido un sistema de bloqueo incremental, por lo que cada vez que el usuario falle al autenticarse, se incrementa 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.

5.3 CONFIGURACIÓN DE INTERFACES, PUERTOS Y SERVICIOS

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

Sección	Título de la sección
4	Instalación y configuración de ON Core
5	Instalación y configuración de ON Analytics
6	Instalación y Configuración ON Sensor
7	Instalación y Configuración ON Agent
8	Instalación y configuración de la CMI
9	Instalación y configuración de la CMIX

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

60. 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.

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

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

63. En el momento en el que se ponga en producción cualquiera de los nodos, se debe aplicar una configuración para que quede el acceso lo más restringido posible, habilitando solamente los puertos y servicios que se vayan a utilizar.

64. La configuración de los diferentes nodos se realiza mediante la red interna, a través de la cual se administran todos los nodos. Esta red debe ser privada para que solamente pueda acceder el personal autorizado.

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

66. La configuración para los distintos nodos es la siguiente:

- En el caso de *ON Core*:

Source	*Destination*	*Port*	*Service*
Core	*Core*	*TCP/22*	*SSH*
Core	*Analytics*	*TCP/22*	*SSH*
Core	*Aggregator*	*TCP/22*	*SSH*
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*

Source	Destination	Port	Service
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
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

- En el caso de *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
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

- En el caso de 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

Source	Destination	Port	Service
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

5.4 CONFIGURACIÓN DE PROTOCOLOS SEGUROS

67. 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.

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

69. El servicio HTTP no está habilitado en el portal de administración, siempre que se intente acceder a través del navegador se redirige el tráfico a HTTPS, que usa cifrado de capa TLSv1.2.

70. Esta configuración se realiza en el fichero de *opennac_ssl.conf* que se encuentra en el directorio */etc/httpd/conf.d.*

71. Con la siguiente línea, se indica que se deshabilitan todas las versiones que no sean *TLSv1.2: SSLProtocol -all +TLSv1.2.*

72. 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 está cifrada con SSH2.

73. Esta configuración se encuentra en el fichero de configuración *sshd_config* que se encuentra en el directorio de */etc/ssh.*

74. En este mismo fichero se desactivan aquellos Ciphers que son débiles. También se configuran 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.

5.5 GESTIÓN DE CERTIFICADOS

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

    a) RSA con claves de, al menos, 3072 bits de longitud.

b) ECDSA con curvas P-256 o superior.

c) DSA con claves de, al menos, 3072 bits de longitud.

d) Funciones Hash SHA-256 o superior.

76. Cuando un proyecto requiere habilitar la autenticación con certificados digitales, o la verificación de la identidad del servidor, se pide al cliente:

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

b) Que mediante los archivos CSR generados en EMMA, firmen mediante su CA, certificados de servidor.

77. Para generar los archivos CSR, se utiliza el siguiente comando, donde se especifica el tipo de cifrado que se va a usar:

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

78. Una vez se reciban los certificados ya firmados por el cliente, se deben importar al servidor.

79. 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.

80. Para convertir a formato PEM, se usa el siguiente comando:

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

81. 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.

82. Para validar la información de un certificado mediante openssl, se puede ejecutar el siguiente comando:

openssl x509 -in aaa_cert.pem -noout -text

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

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

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

85. 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.

86. 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.

87. 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.

88. 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.

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

90. El CN 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.

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

92. 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.

5.6 SERVIDORES DE AUTENTICACIÓN

93. Como ya se ha comentado anteriormente, para la autenticación de usuarios, se hace uso de LDAP habitualmente o de Directorio Activo como repositorio de identidades.

94. La comunicación con LDAP se hará preferiblemente mediante LDAPs sobre TLSv1.2.

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

96. 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).*

97. El SP es el encargado de recibir la conexión del usuario, redirigirlo al IdP y concederle el acceso que se había solicitado.

98. 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.

99. 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*.

100. EMMA asume el rol de *Identity Provider*. Con acceso a las bases de datos de usuarios, los autentica a la vez que evalúa y valida las políticas de acceso. Si, por otra parte, hubiera que autorizar al usuario, podría hacerlo tanto EMMA como el SP.

101. 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.

5.7 SINCRONIZACIÓN HORARIA

102. Para la configuración horaria de los diferentes nodos, se hace 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. La configuración es la siguiente:

103. Primero se debe parar el servidor NTP antes de modificar sus parámetros, se debe 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.

104. Se modifica el fichero /etc/ntp.conf y se incluyen los servidores apropiados:

    server <ip_servidor_ntp>

105. Se arranca el servicio:

    [root@core ~]# service ntpd start

5.8 ACTUALIZACIONES

106. Se puede consultar la descripción de cómo llevar a cabo la actualización de los diferentes nodos en el documento de Despliegue de Nodos.

Sección	Título de la sección
4	Instalación y configuración de ON Core

Sección	Título de la sección
5	Instalación y configuración de ON Analytics
6	Instalación y Configuración ON Sensor
7	Instalación y Configuración ON Agent
8	Instalación y configuración de la CMI
9	Instalación y configuración de la CMIX

5.9 AUTO-CHEQUEOS

107. El producto no realiza autochequeos (*self-test*) de integridad.
108. Los paquetes de EMMA vienen firmados con nuestra clave o la del proveedor (en este caso CentOS).
109. Al descargar los paquetes, se mirará si la firma se ha modificado o no para así poder validar la descarga.
110. Todos los paquetes que se instalen desde el repositorio vienen firmados. Si está instalado el paquete *opennac-gpg-key,* se garantiza que no se pueda instalar ningún paquete que no esté firmado.
111. Por lo que se garantiza que, al instalar un nuevo paquete, si ese paquete ha sido modificado o se ha intentado manipular, y no coincide con la firma, se sabe que ha sido manipulado y no se procede con su instalación.
112. Se ejecuta para ver si ha habido alguna modificación:

    [root@core ~]# rpm -qaV

5.10 SNMP

113. Desde EMMA 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).
114. Para configurar la comunicación mediante SNMP v1 y v2, es necesario configurar una "*community*" de *Read-Only (public)* y otra de *Read-Write (private*).
115. También está 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.

116. EMMA soporta 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, se hace un filtrado de IP desde el *firewall* para que solo nos puedan consultar ciertos dispositivos autorizados, así se consigue tener más acotada la visibilidad a dispositivos externos.

117. 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 EMMA cuando se deba enviar cierta información o informar de un problema. EMMA 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).

5.11 ALTA DISPONIBILIDAD

118. EMMA dispone dos (2) configuraciones de alta disponibilidad:

- Escenario de replicación de MySQL desde un nodo principal a diferentes nodos worker.

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

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

121. En primer lugar, se debe crear una instancia de EMMA Core que funcione como nodo principal. Una vez el nodo principal esté en funcionamiento, se crea una instancia (o las que se requieran) de nodo worker.

122. Se empieza configurando el nodo principal, se indica en el archivo /etc/my.cnf que se trata del principal.

123. Seguidamente, se reanuda el servicio de MySQL para aplicar los cambios.

124. Se accede 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.

125. Posteriormente se ejecuta 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.

126. Se procede con el volcado de información en la base de datos de opennac.

127. Se debe 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. Se aplican los cambios necesarios en el fichero de /etc/sysconfig/iptables, donde se filtra 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

128. Una vez configuradas las reglas de firewall, se reinicia el servicio de *iptables* para que se apliquen los cambios:

service iptables restart

129. Finalmente, se envia mediante SCP al *worker* la base de datos exportados del nodo principal.

scp opennac.sql root@<worker_ip>

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

131. Se añade la información del nodo principal (nombre de host i IP) en el fichero de /etc/hosts.

132. Se importa la DB de opennac enviada desde el nodo principal.

133. Se debe modificar también el fichero /etc/my.cnf para indicar que se trata de un nodo worker y se indica el server-id pertinente para ese nodo.

134. Seguidamente, se reanuda el servicio de MySQL para aplicar los cambios.

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

136. 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

- Configuración del clúster Analytics

137. El clúster de ON Analytics es el encargado de tratar, almacenar y mostrar toda la información que se recibe 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*).

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

139. Esta arquitectura consta de distintos nodos:

a) *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.

b) *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.

c) *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.

140. Se añade en el fichero de */etc/hosts* la información de los distintos nodos de la arquitectura añadiendo el nombre de host i la IP asociada IP.

141. 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.

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

143. Se deberá modificar el fichero de *iptables* para permitir la comunicación entre los distintos nodos. Se aplican los cambios necesarios en el fichero de /etc/sysconfig/iptables, donde se filtra por IP y puerto:

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

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

    service iptables restart

145. El envío de la información desde ON Core/ON Sensor hacia ON Analytikcs, 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.

146. Para más detalle sobre la configuración de seguridad en las comunicaciones del *Elastic Stack, (Logstash-Elasticsearch-Kibana*) se recomienda consultar el documento de despliegue de nodos, el Anexo D: Seguridad de Comunicaciones entre nodos.

5.12 AUDITORÍA

147. 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.

5.12.1 REGISTRO DE EVENTOS

148. Básicamente se registran dos tipos de logs que se pueden clasificar en:

    a) 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.

    b) Logs de programas: Registran cronológicamente los eventos más importantes mientras se usa una aplicación o programa. En este caso, los logs pueden ser generados por la propia aplicación o por Rsyslogd.

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

150. Dentro de estas carpetas también se encuentran 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:

    - *redis.log-20200521.gz*

151. Nombre del log junto a la fecha de creación de dicho registro comprimido.

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

    - *opennac-analytics.log*

153. 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*

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

    - *opennac-captive-analytics.log*

155. Este registro contiene información relacionada con el portal cautivo que luego se enviará a los análisis de EMMA. 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*

156. 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*

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

158. 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*

159. 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.

5.12.2 ALMACENAMIENTO LOCAL

160. 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.

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

162. Si se quiere recolectar más información y proporcionar visibilidad avanzada y seguimiento histórico del comportamiento de dispositivos en la red, se usa 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 *on-premise*.

163. Para controlar el comportamiento en caso de que se alcance el límite de almacenamiento se hace uso de la herramienta de *logrotate*.

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

165. 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.

166. En el directorio de /k/ se encuentra la configuración del log de los distintos servicios que se ejecutan en el sistema, donde se configura de manera

independiente cada uno de ellos. Si alguno no tuviera aquí su configuración tomaría la del fichero anterior.

167. Por otro lado, en el fichero */etc/logrotate.conf* se establece la configuración base para los diferentes paquetes de los servicios que se quieran rotar.

168. En este fichero se pueden 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.

169. El sistema almacena los logs antiguos, pero no indefinidamente. La configuración general del servicio se establece en el fichero de *logrotate.conf*, para indicar cada cuánto tiempo se crea un fichero de log y cuánto tiempo se deja 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.

5.12.3 ALMACENAMIENTO REMOTO

170. Desde EMMA, además de almacenar los registros (*audit*), se pueden enviar registros a servidores remotos. Para ello, se usa:

a) *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.

b) *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 procesa (*Logstash*), almacene (*Elasticsearch*) y se muestre al usuario (*Kibana*).

c) *Logstash*. Esta herramienta nos permitirá recolectar y procesar los datos recibidos del *ON Core* a través de Beats. En *Logstash*, los datos se filtran y moldean para posteriormente mandarlos a *Elasticsearch*.

d) *Elasticsearch*. Cuando 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.

e) *Kibana* generará los *dashboards* con diferentes tipos de gráficos y tablas a partir de los datos que extrae de *Elasticsearch*.

f) Por ejemplo, en el índice *opennac-** la información se elimina cuando lleva 15 días almacenada.

g) *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.

171. Para poder leer y recibir los distintos registros a través de la red, se debe configurar diferentes módulos.

Módulos de input:

Se configura en este módulo el *im_udp*, que se encargará de tratar los mensajes entrantes a través de UDP. Es recomendable utilizar los protocolos de transporte TCP o SSL para evitar la pérdida de mensajes.

También se configura dentro del módulo de input el *im_tcp*, que gestiona los mensajes entrantes a través de TCP. Se configura el *im_ssl* para que se encargue de gestionar los mensajes entrantes a través de TCP con seguridad SSL / TLS.

De cada uno de estos módulos se configura el puerto por el cual se recibirá la información y si fuera necesario, se definen las funciones que deberá ejecutar cada módulo.

Módulos de output:

Para el reenvío y almacenamiento de registros, se configura en los módulos de output el *om_file.* Esta configuración lee los mensajes de registro del socket y escribe los mensajes en un archivo, no se realiza ningún procesamiento adicional. En este módulo se definen los ficheros donde se almacenan los registros y se definen ciertas acciones que deberán ejecutarse.

5.13 BACKUP

172. 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.

173. Los *backups* se realizan mediante un script donde se configuran parámetros como:

- La carpeta / directorio en el que se desea hacer el *backup*.
- El tiempo de retención del backkup.
- La carpeta de destino en la que se almacena el backkup.
- El nombre del fichero.
- Número de logs que se desean retener.
- Si se quiere hacer o no el volcado de la base de datos.

174. 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*.

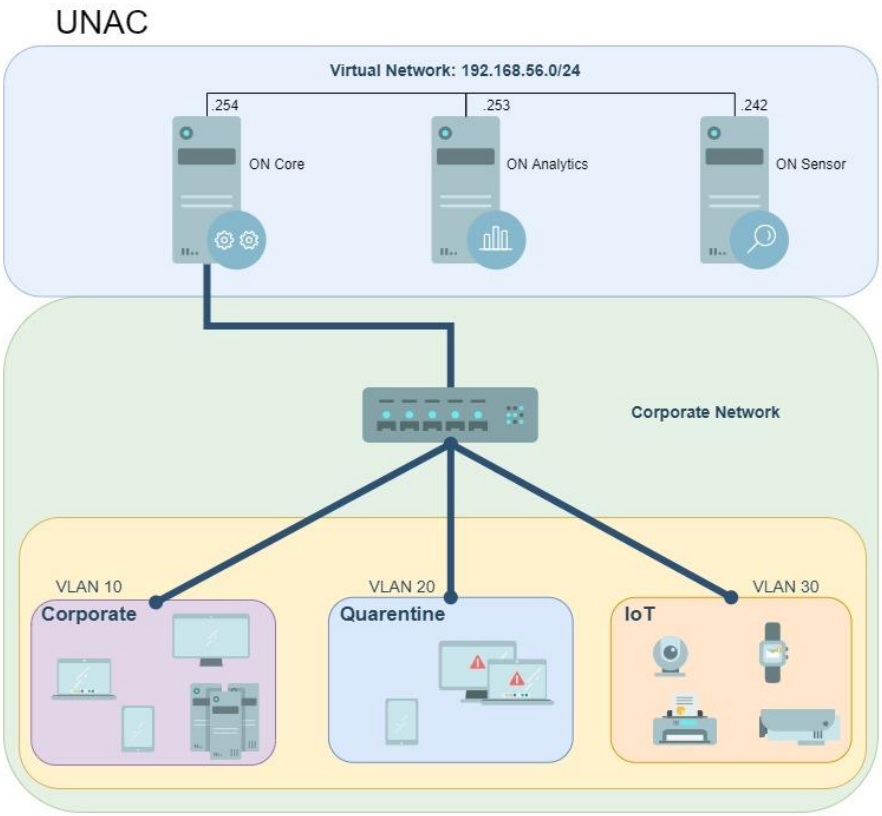
175. El formato de los ficheros sigue este formato: *opennac-<nombre_máquina>-yyyymmdd_hhmmss.tgz*

5.14 SERVICIOS DE SEGURIDAD

Caso de uso – UNAC

176. 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.

177. Una arquitectura básica de una implementación UNAC es la siguiente:



178. 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.

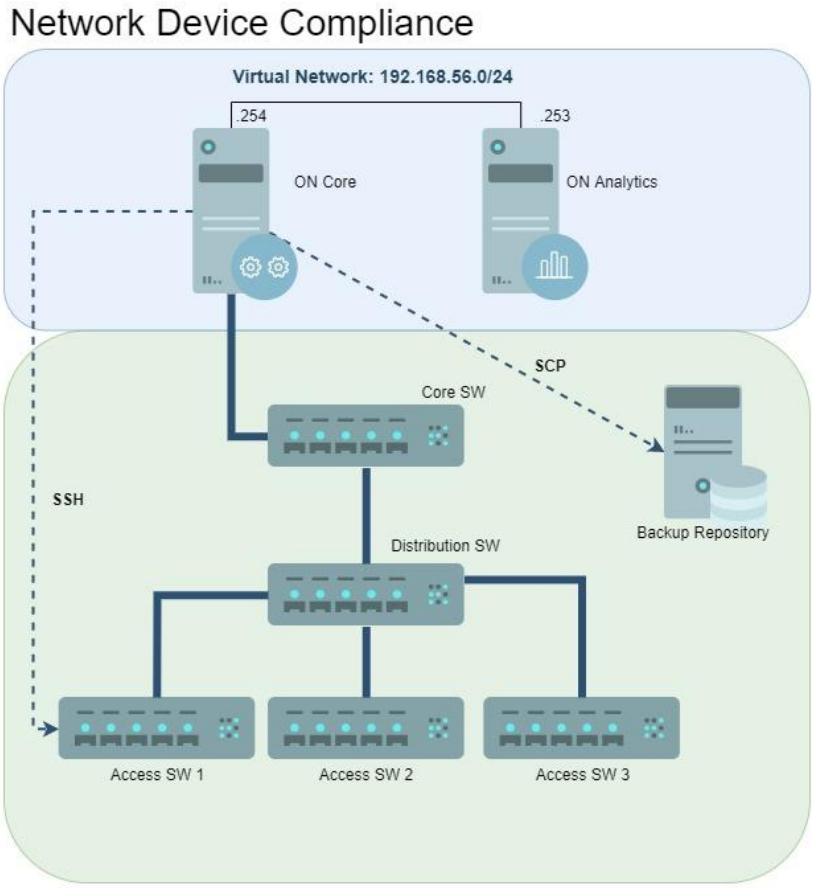
179. Dependiendo de la política a la que aplica la conexión del dispositivo, este se asignará a un segmento de la red o VLAN (autorización y segmentación), ON Sensor monitorizará la red y *ON Analytics* se encargará de recibir, normalizar y almacenar todos los datos generados por ON Core y ON Sensor,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

Caso de uso - *Network Device Compliance*

180. El caso de uso de *Network Device Compliance (NDC)* nos permitirá validar configuraciones de los dispositivos de red a través de la ejecución de test que

recuperaran información a través del backup (local o remoto) o comandos directos a estos dispositivos. Para esta implementación, simplemente será necesario el uso de *ON Core* y *ON Analytics*. Nos permite analizar requisitos de diferentes dispositivos de red para auditoria.

181. Un ejemplo de arquitectura básica de NDC es el siguiente:



182. En esta arquitectura, ON Core tiene accesos a la red local o corporativa, así como a los dispositivos de red que la componen. ON Analytics recibe a través de la red virtual interna los datos de ON Core para, posteriormente, mostrar los resultados de los tests al usuario.

183. ON Core puede acceder a los ficheros de backup a través de una conexión directa al dispositivo mediante SSH o a través de SCP mediante un repositorio remoto donde los dispositivos de red almacenan los backup. En ambos casos será necesario configurar las credenciales en la CMDB.

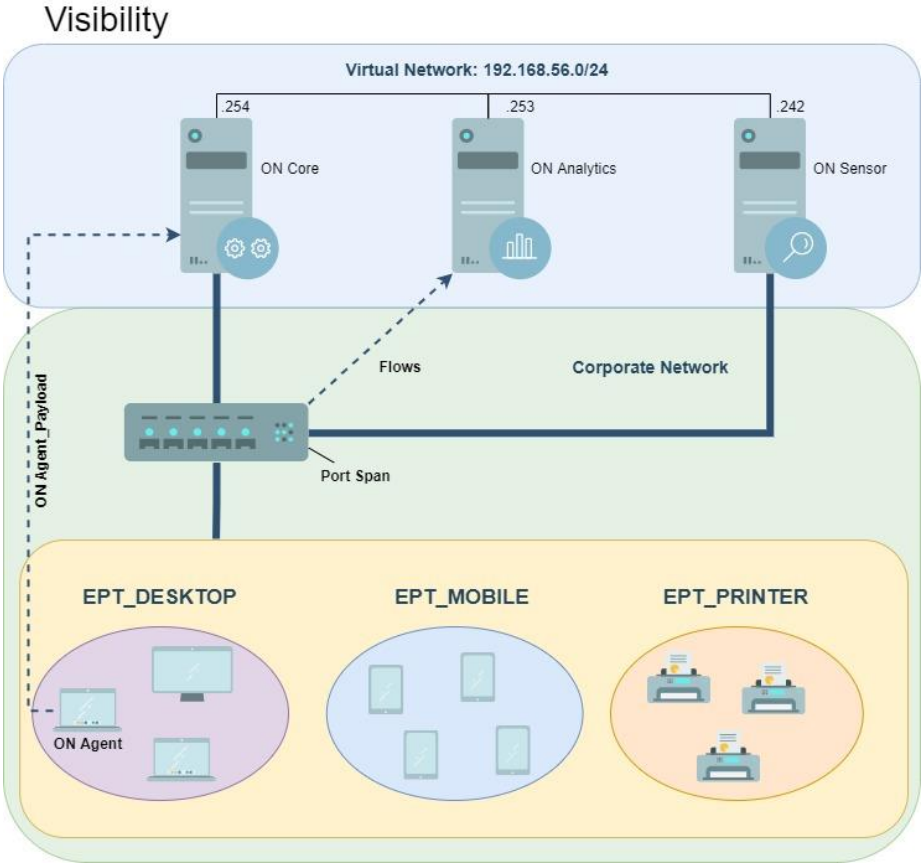
Caso de uso – *Visibility*

184. 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.

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

186. Este caso de uso requiere del despliegue de los nodos *ON Core*, *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.

187. Un ejemplo de arquitectura básica para el caso de uso de visibilidad es el siguiente:



188. El proceso de *visibility* consta de dos (2) fases principales:

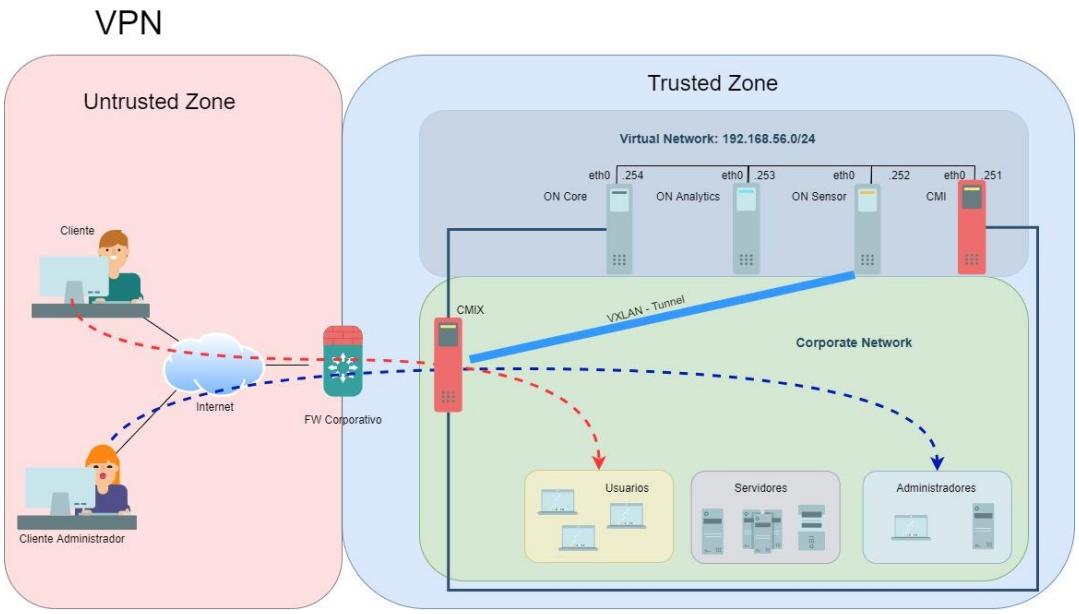
- Discovery: Proceso de descubrimiento de dispositivos conectados a la red, consiste en detectar una IP o MAC en la red no vista antes.
- Profiling: Cuando se detecta un nuevo dispositivo, este debe ser perfilado, es decir, determinar el tipo de dispositivo de que se trata (DESKTOP, MOBILE, PRINTER,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

189. 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 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.

Caso de uso - VPN

190. El caso de uso de VPN es una solución desarrollada para aportar accesibilidad y control en la capa de acceso a la red mediante el acceso remoto a la red.

191. El siguiente diagrama muestra una arquitectura de red de ejemplo, que servirá de guía durante la lectura y aplicación de este documento. La solución está compuesta de dos dispositivos principales, representados en dos servidores de color rojo, estos son *el Front End (FE*), basado en una *CMIX*, y el *Back End (BE),* basado en una CMI. Además, en el ejemplo existe un Firewall Corporativo el cual conecta la DMZ donde está el FE, y las redes internas donde típicamente estará el BE, además de Internet donde estarán los usuarios remotos.



192. En este ejemplo, Son dos (2) clientes diferentes que desean acceder a la red corporativa mediante una VPN.

193. Cuando un cliente quiere acceder a la red interna desde internet, este se conecta al finalizador de túneles. Este finalizador de túneles gestionará las conexiones VPN. El usuario realiza una petición de autenticación la cual se redirige al servidor ON Core para que la valide, este autentica y autoriza mediante las políticas. Una vez finalizada la autenticación y autorización, se le devuelve al Finalizador de túneles las claves que contienen la zona dinámica (definida en el Finalizador de túneles) a la que se debe asignar el usuario.

194. Adicionalmente el finalizador de túneles, mediante un túnel VXLAN, mandará todo el tráfico que este usuario genere al nodo ON Sensor.

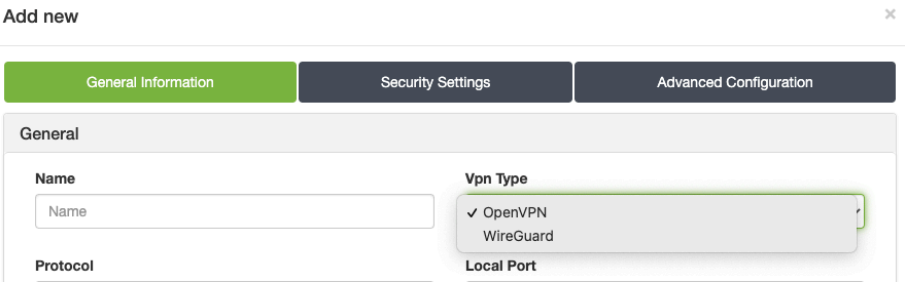
5.15 CONFIGURACIÓN SEGURA VPN

195. El caso de uso de VPN se puede configurar mediante dos (2) tecnologías diferentes: OpenVPN y WireGuard. Sin embargo, **solo la configuración que hace uso de la tecnología WireGuard ha sido evaluada dentro del proceso de cualificación e inclusión en el Catálogo de Productos y Servicios TIC.**

196. Para acceder a la configuración del caso de uso de VPN accede a través del menú:

    > Manage > VPN Road Warrior > Add new

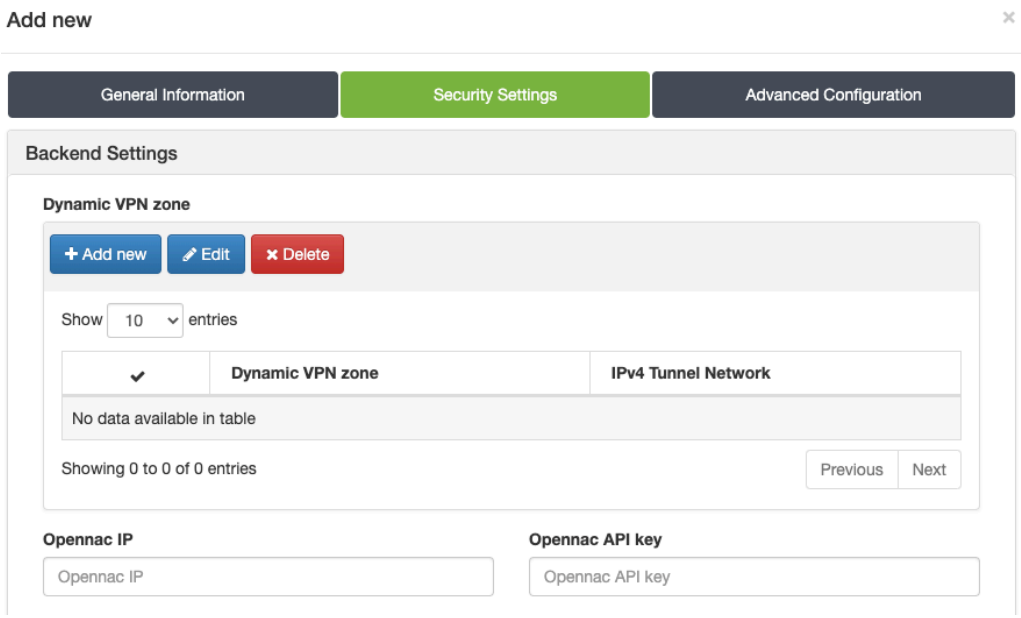
    Y elegir el tipo de VPN:



5.15.1 *WIREGUARD*

197. Wireguard utiliza un conjunto de protocolos criptográficos predefinidos. Estas configuraciones no son modificables por el usuario, sino que es el propio fabricante el que incluirá nuevos protocolos por defecto en el caso de que estos se vean comprometidos o surjan nuevos estándares más seguros.

198. Los protocolos usados por *wireguard* son: *Noise protocol framework, Curve25519, ChaCha20, Poly1305, BLAKE2, SipHash24* y *HKDF.*



6. FASE DE OPERACIÓN

199. El procedimiento de revisión y aprobación de conexiones dentro de una arquitectura debería ser el siguiente:

200. 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.

201. Las comprobaciones deberán realizarse al menos cada cuatro o seis meses y en cada cambio aprobado que se quiera realizar.

202. A nivel de *hardware* y *software* se realizan las diferentes comprobaciones para asegurar que no se ha introducido hardware o software no autorizado.

203. 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. Se usa *Openscap en* https://www.open-scap.org/tools/scap-workbench/ para probar la implementación del *Hardening* en Centos.

204. Los nodos se configurarán previamente para reducir las vulnerabilidades en el mismo, para ellos se eliminan 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.

205. Inicialmente, se instala un sistema base mínimo evitando instalaciones de grupos de aplicaciones y aplicaciones que no sean estrictamente necesarias.

206. Se revisarán los puertos y servicios abiertos, así como las diferentes reglas 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.

207. Se deberán revisar los paquetes instalados y se eliminarán aquellos que no sean necesarios. Esta comprobación se hace mediante:

    rpm -qa | less

    rpm -qa | grep <service>

    lsof -i | grep -v ESTABLISHED

208. Para eliminar paquetes o servicios instalados:

    systemctl stop <service>

    systemctl disable <service>

    yum remove <package>

209. Como se explica en el apartado 5.12 AUDITORÍA, se registrarán y almacenarán temporalmente de forma local ciertos eventos o backups de sistema, para poder

auditar los distintos nodos instalados. Se configura *Syslog* para poder enviar una copia de los registros a una máquina dedicada remota para hacer la gestión de estos.

7. CHECKLIST

210. Se incluye una *checklist* que contiene todas las recomendaciones sobre la configuración de cada apartado a modo de ejemplo:

ACCIONES	SÍ	NO	OBSERVACIONES
DESPLIEGUE E INSTALACIÓN			
Verificación de la entrega segura del producto			
Instalación en un entorno seguro			
Registro de los equipos			
Registro de las licencias			
Actualización de firmware			
CONFIGURACIÓN			
MODO DE OPERACIÓN SEGURO			
Autenticación			
Administración del producto			
Configuración de interfaces, puertos y servicios			
CONFIGURACIÓN DE PROTOCOLOS SEGUROS			
Gestión de certificados			
SERVIDORES DE AUTENTICACIÓN			
Sincronización horaria			
Actualizaciones			
Auto-chequeos			
SNMP			
FASE DE OPERACIÓN			
Auditoría			

ACCIONES	SÍ	NO	OBSERVACIONES
Registro de eventos			
Almacenamiento local			
Almacenamiento remoto			
Backup			

8. REFERENCIAS

- Documento de despliegue de nodos
- Documento Cumplimiento – Operación de EMMA
- Documento Visibilidad - Instalación
- Documento Visibilidad - Operación
- Documento VPN - Instalación
- Documento VPN - Operación

Para acceder a la documentación aquí referenciada, se debe remitir un correo electrónico a la dirección emma@ccn-cert.cni.es.

9. ABREVIATURAS

ACL	Lista de Control de Acceso
AD	*Active Directory*
CN	*Common Name*
CPD	*Centro de Procesamiento de Datos*
CRL	*Certificate Revocation List*
CRM	*Customer Relationship Management*
DB	Base de Datos
DB	*Database*
ENS	Esquema Nacional de Seguridad.
HA	*High Availavility*
LDAP	*Lightweight Directory Access Protocol*
NTP	*Network Time Protocol*
OSCP	*Offensive Security Certified Professiona*
SAML	*Security Assertion Markup Language*
SCP	*Secure Copy Protocol*
SSH	*Secure Shel*
SSL	*Secure Sockets Layer*
TCP	Transmission Control Protocol
TTL	*Time To Live*

PUBLICACIONES DE DEFENSA

GOBIERNO DE ESPAÑA
MINISTERIO DE DEFENSA
SUBSECRETARÍA DE DEFENSA
SECRETARÍA GENERAL TÉCNICA
SUBDIRECCIÓN GENERAL DE PUBLICACIONES Y PATRIMONIO CULTURAL