



Edita:



© Centro Criptológico Nacional, 2019  
NIPO:083-19-180-6

Fecha de Edición: mayo de 2019

El Departamento de Ingeniería de Sistemas Telemáticos de la Universidad Politécnica de Madrid ha participado en la elaboración del presente documento y sus anexos.

#### **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.

## PRÓLOGO

El uso masivo de las tecnologías de la información y las telecomunicaciones (TIC), en todos los ámbitos de la sociedad, ha creado un nuevo espacio, el ciberespacio, donde se producirán conflictos y agresiones, y donde existen ciberamenazas que atentarán contra la seguridad nacional, el estado de derecho, la prosperidad económica, el estado de bienestar y el normal funcionamiento de la sociedad y de las administraciones públicas.

La Ley 11/2002, de 6 de mayo, reguladora del Centro Nacional de Inteligencia (CNI), encomienda al Centro Nacional de Inteligencia el ejercicio de las funciones relativas a la seguridad de las tecnologías de la información en su artículo 4.e), y de protección de la información clasificada en su artículo 4.f), a la vez que confiere a su Secretario de Estado Director la responsabilidad de dirigir el Centro Criptológico Nacional (CCN) en su artículo 9.2.f).

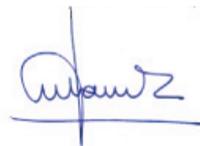
Partiendo del conocimiento y la experiencia del CNI sobre amenazas y vulnerabilidades en materia de riesgos emergentes, el Centro realiza, a través de su Centro Criptológico Nacional, regulado por el Real Decreto 421/2004, de 12 de marzo, diversas actividades directamente relacionadas con la seguridad de las TIC, orientadas a la formación de personal experto, a la aplicación de políticas y procedimientos de seguridad, y al empleo de tecnologías de seguridad adecuadas.

El Real Decreto 3/2010, de 8 de enero, por el que se regula el Esquema Nacional de Seguridad en el ámbito de la Administración Electrónica (ENS, en adelante), al que se refiere el apartado segundo del artículo 156 de la Ley 40/2015, de 1 de octubre, de Régimen Jurídico del Sector Público, establece la política de seguridad en la utilización de medios electrónicos que permita una protección adecuada de la información.

Precisamente el Real Decreto 3/2010 de 8 de enero, actualizado por el Real Decreto 951/2015, de 23 de octubre, fija los principios básicos y requisitos mínimos así como las medidas de protección a implantar en los sistemas de la Administración, y promueve la elaboración y difusión de guías de seguridad de las tecnologías de la información y las comunicaciones (STIC) por parte de CCN para facilitar un mejor cumplimiento de dichos requisitos mínimos.

En definitiva, la serie de documentos CCN-STIC se elabora para dar cumplimiento a los cometidos del Centro Criptológico Nacional y a lo reflejado en el Esquema Nacional de Seguridad, conscientes de la importancia que tiene el establecimiento de un marco de referencia en esta materia que sirva de apoyo para que el personal de la Administración lleve a cabo su difícil, y en ocasiones, ingrata tarea de proporcionar seguridad a los sistemas de las TIC bajo su responsabilidad.

Mayo de 2019



Félix Sanz Roldán

Secretario de Estado

Director del Centro Criptológico Nacional

## ÍNDICE

<b>1. INTRODUCCIÓN</b> .....	<b>7</b>
<b>2. OBJETO</b> .....	<b>7</b>
<b>3. ALCANCE</b> .....	<b>7</b>
<b>4. INFORMACIONES PREVIAS A LA CONFIGURACIÓN DEL EQUIPO</b> .....	<b>8</b>
4.1 CONFIGURACIÓN MEDIANTE LA LÍNEA DE COMANDOS .....	8
4.2 GESTIÓN DE USUARIOS .....	10
4.2.1 MÉTODOS DE AUTENTICACIÓN .....	11
4.3 GESTIÓN DE PERMISOS DE USUARIOS .....	14
4.3.1 MODO RBAC .....	15
4.4 ACTUALIZACIÓN DEL SOFTWARE DEL DISPOSITIVO .....	18
4.4.1 VERSIONES DE SOFTWARE INSTALADAS Y EN EJECUCIÓN .....	18
4.4.2 SELECCIÓN DE VERSIÓN Y CONFIGURACIÓN A EJECUTAR .....	19
4.4.3 ACTUALIZACIÓN DEL SISTEMA .....	20
4.5 SWITCH IDENTITY PROFILE .....	20
<b>5. SEGURIDAD EN EL ACCESO A LA ADMINISTRACIÓN</b> .....	<b>22</b>
5.1 CONFIGURACIÓN INICIAL DEL EQUIPO .....	22
5.2 GESTIÓN LOCAL DEL EQUIPO .....	23
5.2.1 PUERTO DE CONSOLA .....	23
5.2.2 PUERTO USB .....	24
5.3 GESTIÓN REMOTA DEL EQUIPO .....	25
5.3.1 REQUISITOS Y RECOMENDACIONES INICIALES .....	25
5.3.2 TELNET .....	26
5.3.3 SSH .....	27
5.3.4 HTTP/HTTPS .....	27
5.3.5 ACCESO MEDIANTE API REST .....	30
5.3.6 SNMP .....	31
5.4 POLÍTICAS DE CALIDAD DE CONTRASEÑAS .....	33
5.4.1 ALMACENAMIENTO Y CIFRADO DE CONTRASEÑAS .....	36
5.4.2 RECUPERACIÓN DE CONTRASEÑAS Y CONFIGURACIÓN DE FÁBRICA .....	36

<b>6. SERVICIOS DE RED DEL EQUIPO</b> .....	<b>39</b>
6.1 LLDP Y CDP .....	39
6.2 ICMP .....	40
6.3 TRANSFERENCIA DE FICHEROS.....	40
6.4 SINCRONIZACIÓN DE TIEMPO .....	41
<b>7. CONTROL DE TRÁFICO Y SEGURIDAD EN LOS PUERTOS DE RED</b> .....	<b>43</b>
7.1 APAGADO DE PUERTOS.....	43
7.2 MEDIDAS DE PROTECCIÓN DE PUERTOS .....	43
7.3 LIMITACIÓN DEL TRÁFICO DE BROADCAST Y OTROS TIPOS .....	45
7.4 USO DE VLAN COMO MEDIDA DE AISLAMIENTO .....	46
7.5 VLAN PRIVADAS .....	48
7.6 PROTECCIÓN FRENTE A ENVÍO DE MENSAJES DE CONTROL STP .....	50
7.7 LISTAS DE ACCESO IP .....	50
7.7.1 TIPOS DE ACLS .....	51
7.7.2 CREACION DE ACLS.....	52
7.7.3 APLICACION DE LISTAS DE ACCESO .....	53
7.8 LISTAS DE ACCESO MAC .....	54
<b>8. SISTEMAS DE CONTROL DE ACCESO</b> .....	<b>56</b>
8.1 CONTROL DE ACCESO MEDIANTE 802.1X .....	57
8.2 AUTENTICACION BASADA EN DIRECCIONES MAC .....	58
8.3 AUTENTICACIÓN BASADA EN WEB .....	59
8.4 AUTENTICACIÓN MEDIANTE PORTAL CAUTIVO EXTERNO .....	60
8.5 POLÍTICAS DE ACCESO BASADAS EN ROLES .....	61
8.5.1 DEFINICIÓN DE ROLES DE USUARIO .....	61
8.5.2 CONFIGURACIÓN DEL ATRIBUTO ROLE EN RADIUS .....	62
8.5.3 ACTIVACIÓN DE LA FUNCIONALIDAD .....	63
<b>9. PROTECCIÓN FRENTE ATAQUES</b> .....	<b>64</b>
9.1 DHCP SNOOPING.....	64
9.2 ARP SNOOPING .....	66
9.3 INUNDACIÓN MAC .....	68

9.4	VIRUS THROTTLING .....	69
9.5	MEDIDAS DE PROTECCIÓN DE ENLACES TRONCALES .....	71
9.5.1	MACSEC .....	71
9.5.2	PROTECCIÓN DEL PROTOCOLO DLDP .....	73
<b>10.</b>	<b>RESUMEN DE RECOMENDACIONES .....</b>	<b>74</b>
10.1	PROCEDIMIENTO DE CONFIGURACION INICIAL DE UN EQUIPO .....	74
10.2	OTRAS RECOMENDACIONES ADICIONALES .....	77
<b>11.</b>	<b>REFERENCIAS.....</b>	<b>78</b>

## 1. INTRODUCCIÓN

El presente documento pretende servir de guía para establecer una configuración segura de la familia de equipos HPE Aruba con sistema operativo ArubaOS-Switch. A lo largo de los diferentes capítulos, se ofrecen consejos y recomendaciones sobre la activación o desactivación de servicios y funcionalidades de los equipos de red con el fin de establecer una configuración lo más segura posible.

La estructura del documento y sus contenidos no exigen una lectura lineal del mismo. El lector puede utilizar el índice de contenidos para localizar y acceder al capítulo que trate el aspecto sobre el que desea mejorar la seguridad. Sin embargo, se recomienda realizar lectura rápida del documento completo para hacerse una idea de todas las funcionalidades descritas.

## 2. OBJETO

Analizar los mecanismos de seguridad disponibles para proteger los entornos de sistemas de información y comunicaciones que emplean switches HPE Aruba con sistema operativo ArubaOS-Switch. Como consecuencia, se establece un marco de referencia que contemple las recomendaciones STIC en la implantación y utilización de switches HPE Aruba con sistema operativo ArubaOS-Switch.

En líneas generales, en este documento no se valora la idoneidad de utilizar o no determinados protocolos, sino que se describe como deben ser securizados cada uno de ellos.

Queda fuera del alcance de este documento la configuración de los mecanismos para garantizar la calidad de servicio necesaria para la explotación del dispositivo puesto que se entiende que la calidad del servicio no afecta a la seguridad de este.

En el ámbito de este documento, se asume que existirá un usuario de nivel administrador que podrá configurar todas las funcionalidades requeridas, incluidas las definiciones de usuarios locales.

## 3. ALCANCE

Las autoridades responsables de la aplicación de la política de seguridad de las TIC (STIC) determinarán el análisis y aplicación de este documento a los switches HPE Aruba bajo su responsabilidad.

## 4. INFORMACIONES PREVIAS A LA CONFIGURACIÓN DEL EQUIPO

El objetivo de la guía es establecer un marco que permita incrementar la seguridad de un equipo. Por un lado describe los mecanismos que evitan accesos no deseados a la administración del equipo y a los datos almacenados en él. Por otro lado analiza los servicios y sus interacciones con la red en la que será integrado, con el objeto de diferenciar las acciones seguras y necesarias en el equipo, de las acciones que no lo son y deben ser sustituidas por otras.

Es necesario aplicar los consejos de seguridad que contiene esta guía, al menos los relativos a proteger el acceso a la gestión del equipo y la deshabilitación de los servicios no utilizados, antes de desplegar el equipo en la red, con el objeto de evitar posibles incidentes antes de que las protecciones estén activas.

Asimismo, antes de desplegar el equipo es imprescindible realizar una actualización del mismo con las últimas versiones estables del sistema operativo, con el objeto de protegerlo de los problemas de seguridad detectados en versiones anteriores.

Esta guía no pretende ser un curso de configuración de los conmutadores HPE Aruba sino un conjunto de recomendaciones y normas para efectuar su configuración de forma segura. Es importante, sin embargo, repasar algunos conceptos básicos relativos a la configuración mediante la línea de comandos, la gestión de usuarios o la actualización del software.

### 4.1 CONFIGURACIÓN MEDIANTE LA LINEA DE COMANDOS

Los equipos HPE Aruba con sistema ArubaOS-Switch son equipos completamente gestionables, permitiendo al administrador de red configurar el equipo mediante la línea de comandos, mediante el interfaz gráfico basado en web o a través de un API REST [1].

El interfaz gráfico ofrece solo un conjunto reducido de opciones básicas de configuración de puertos y enlaces, dado que tiene más por objeto la monitorización y resolución de incidencias. Por esta razón esta guía se centra en la configuración segura del equipo via el interfaz de línea de comandos.

El acceso a la línea de comandos (CLI) es posible mediante la conexión directa a la consola, o bien mediante el acceso remoto vía red IP utilizando SSH.

En la línea de comandos de un switch ArubaOS existen diferentes contextos de gestión (Figura 1). Cada contexto se identifica con un *prompt* específico de la línea de comandos. En condiciones en las que no se haya restringido por usuario existen los siguientes contextos:

- **Nivel Operador:** permite ejecutar únicamente comandos de inspección. El prompt está formado por el nombre del equipo seguido de ">".
- **Nivel Manager:** permite el control del equipo y todas las acciones administrativas, así como las acciones de actualización de software. El prompt

está formado por el nombre del equipo seguido de "#". Desde el contexto Operator se puede acceder a este modo ejecutando el comando "enable".

- **Configuración global.** Permite la modificación de la configuración del equipo. Se accede a este modo mediante la ejecución del comando "configure". Es imprescindible acceder desde el contexto manager. El prompt está formado por el nombre del equipo seguido de "(config)#". La pulsación del carácter "?" ofrece al administrador la lista de opciones posibles de configuración.
- **Configuración específica.** Permite la configuración particular de un aspecto específico. Algunos ejemplos son la configuración de VLAN, de interfaces de red, ACL, OSPF, etc. El prompt está formado por el nombre del equipo seguido de "(xxxxx)#", donde xxxx adopta un literal que describe el contexto en concreto en el que se encuentra el usuario. La pulsación del carácter "?" ofrece al administrador la lista de opciones posibles de configuración específicas de este contexto, junto con algunos comandos generales.

### Contexto ArubaOS-Switch



Figura 1: Contextos en la línea de comandos en ArubaOS-switch

Adicionalmente, existe el comando "menu" (Figura 2), que permite configurar los aspectos más básicos del equipo sin la necesidad de uso de comandos de línea. Es interesante para una configuración inicial.

```
ArubaOS-sw                                     14-Feb-2000 22:27:39
===== TELNET - MANAGER MODE =====
Main Menu

1. Status and Counters...
2. Switch Configuration...
3. Console Passwords...
4. Event Log
5. Command Line (CLI)
6. Reboot Switch
7. Download OS
8. Run Setup
0. Logout
```

**Provides the menu to display configuration, status, and counters.**

To select menu item, press item number, or highlight item and press <Enter>.

### Figura 2: Comando menú en ArubaOS-Switch

Casi todos los comandos de configuración tienen una forma “no”, que se utiliza normalmente para desactivar esa función. En ocasiones, desactivar una función requiere especificar además los parámetros originales que activaban dicha función. Por ejemplo:

```
Aruba(config)#no front-panel-security password-recovery
```

El comando “write memory” guarda la configuración del equipo en el fichero de configuración de arranque.

```
Aruba# write memory
```

El comando “show” se utiliza para obtener información de configuración y estado del conmutador. Por ejemplo, para obtener información sobre las VLANs creadas en el conmutador:

```
Aruba# show vlan
```

## 4.2 GESTIÓN DE USUARIOS

En los conmutadores ArubaOS la autenticación de usuarios puede ser local o remota. En el primer caso, los usuarios y sus credenciales de acceso se definen localmente en cada conmutador; en el segundo, la autenticación se realiza en servidores de autenticación externos de tipo RADIUS o TACACS. En general, es conveniente desde el punto de vista de la seguridad y la gestión utilizar la segunda opción, ya que facilita la gestión de usuarios y la protección de la información al estar ésta centralizada. De

cualquier modo, en redes de pequeño tamaño la autenticación local es una opción válida.

Por defecto, y con el objeto de facilitar su puesta en marcha inicial, la configuración de fábrica de los conmutadores no incluye ninguna autenticación activa: cualquier persona con acceso físico a la consola puede acceder al sistema. Por ello, una de las primeras tareas será la definición de los usuarios con permisos de acceso al sistema y la configuración del acceso de consola para que solicite las credenciales de acceso (ver sección 5).

## 4.2.1 MÉTODOS DE AUTENTICACIÓN

### 4.2.1.1 AUTENTICACIÓN LOCAL

En este caso los usuarios y sus credenciales son definidos y almacenados localmente en el conmutador. Dentro de la autenticación local, el modo más simple es el modo clásico (ver adicionalmente Modo RBAC en sección 4.2.2), en el que sólo existe un usuario para cada uno de los dos niveles de administración: usuario *operator* con permisos de nivel de operador y usuario *manager* con permisos de nivel de manager. Ambos usuarios carecen inicialmente de contraseña, por lo que una de las primeras tareas a realizar debe ser la protección de esas cuentas con una contraseña.

Para asignar una contraseña y, opcionalmente, modificar el nombre de usuarios de las dos cuentas, se puede utilizar el comando:

```
Aruba(config)# password [manager|operator] user-name <nombre_usuario> [plaintext|sha1|sha256]
```

Se recomienda el uso de *sha256* siempre que sea posible, en su defecto se debe usar *sha1* y nunca se debe usar texto plano.

La autenticación local se puede utilizar también como método secundario o de respaldo a la autenticación remota. Por ejemplo, en el caso de que la autenticación se realice mediante un servidor RADIUS externo y el equipo no alcance ninguno de los servidores RADIUS configurados, si la autenticación local se ha configurado como método de respaldo, se haría uso de los dos usuarios para controlar el acceso a la administración del equipo.

### 4.2.1.2 AUTENTICACIÓN MEDIANTE SERVIDOR RADIUS

En este caso los usuarios y sus credenciales se definen y almacenan en un servidor RADIUS externo, que permite centralizar la administración de los usuarios que acceden a la configuración de los equipos. De esta forma, las altas y bajas de usuarios, así como los cambios de credenciales, se pueden realizar sin necesidad de cambios en la configuración de los switches.

El servidor externo de autenticación a utilizar debe soportar el protocolo RADIUS (RFC 2865). Por ejemplo, la autenticación RADIUS está soportada en el gestor Aruba ClearPass

Policy Manager de HPE. En el ámbito del software libre, se puede utilizar FreeRADIUS (freeradius.org).

Como se ha mencionado, es frecuente y en muchos casos recomendable seleccionar el método de autenticación local como método de respaldo. De esta forma, si el equipo no tiene conectividad con los servidores de autenticación puede aún ser gestionado.

ArubaOS permite configurar métodos de autenticación diferentes para las distintas formas de acceso a la gestión del equipo: *Console*, *Telnet*, *WebUI* y *SSH*, incluso diferenciando en cada caso si el acceso es al modo operador o al modo manager. El siguiente comando permite definir para qué métodos de acceso al switch se utilizará la autenticación basada en RADIUS:

```
Aruba(config)# [no] aaa authentication <console|telnet|ssh|web> <enable|login> radius <método-secundario>
```

El parámetro *<console|telnet|ssh|web>* define el método de acceso al switch al que se aplicará la autenticación vía RADIUS, el parámetro *<enable|login>* define si se aplica al acceso al nivel de manager o de operador, y el parámetro *<método-secundario>* define el método secundario a utilizar en caso de que los servidores RADIUS no estén accesibles: *none*, si no se define método secundario, *local* si se quiere autenticación local, o *authorized*, en caso de que se quiera proporcionar acceso no autenticado (no recomendado).

Por ejemplo, los siguientes comandos configuran la autenticación RADIUS con respaldo local para los accesos vía consola y ssh en los dos niveles de privilegios:

```
Aruba(config)# aaa authentication console login radius local
Aruba(config)# aaa authentication console enable radius local
Aruba(config)# aaa authentication ssh login radius local
Aruba(config)# aaa authentication ssh enable radius local
```

Para consultar la configuración de los métodos de autenticación utilizados se puede utilizar el comando:

```
Aruba# show authentication
```

ArubaOS permite la configuración de múltiples servidores RADIUS, así como agruparlos para proporcionar un mecanismo de redundancia en caso de fallo de alguno de ellos.

Para definir un servidor RADIUS y, opcionalmente, la clave de sesión a utilizar se utiliza el comando:

```
Aruba(config)# radius-server host <dir-IP> key <clave>
```

Adicionalmente, conviene activar el procesamiento de mensajes de desconexión o cambio de autorizaciones, así como fijar el tiempo de validez de las peticiones de autorización dinámicas (0 significa sin límite de validez):

```
Aruba(config)# radius-server host <dir-IP> dyn-authorization
```

```
Aruba(config)# radius-server host <dir-IP> time-window d
```

Para agrupar los servidores RADIUS y mejorar con ello la fiabilidad del servicio de autenticación se utiliza el siguiente comando:

```
Aruba(config)# aaa server-group radius <nombre-grupo> host <dirIP-servidor-radius1>
```

ArubaOS permite crear múltiples grupos de tres servidores RADIUS. El comando anterior debe ejecutarse para cada uno de los servidores del grupo. Por ejemplo, los comandos siguientes crean un grupo denominado RAD-TEST incluyendo a los servidores 10.150.0.43, 44 y 45:

```
Aruba(config)# aaa server-group radius "RAD-TEST" host 10.150.0.43
Aruba(config)# aaa server-group radius "RAD-TEST" host 10.150.0.44
Aruba(config)# aaa server-group radius "RAD-TEST" host 10.150.0.45
```

Pueden existir simultáneamente varios grupos, con el objeto de asociar cada grupo a servicios distintos del switch. Existe un grupo por defecto que siempre está formado por los tres primeros servidores. Se puede eliminar un servidor de un grupo precediendo el comando con un “no”. Cuando se elimina el último servidor, también se elimina el grupo.

Para ver la configuración relativa a los servidores RADIUS configurados se utiliza el comando:

```
Aruba# show radius
```

Para definir el grupo de servidores a utilizar en cada caso se utiliza el parámetro *server-group*. Por ejemplo, para asociar la autenticación del login por ssh con el grupo anteriormente definido:

```
Aruba(config)# aaa authentication ssh login radius server-group RAD-TEST local
```

Para ver la configuración relativa a los grupos de servidores RADIUS configurados se utiliza el comando:

```
Aruba# show server-group radius
```

Para más información sobre RADIUS puede consultarse la guía “Access Security Guide” [2].

#### 4.2.1.3 AUTENTICACIÓN MEDIANTE SERVIDOR TACACS

En ArubaOS es posible también gestionar el acceso a la configuración de los equipos de forma centralizada mediante servidores TACACS. Al igual que RADIUS, la autenticación basada en TACACS está soportada en el gestor Aruba ClearPass Policy Manager.

La gestión de los servidores TACACS es muy similar a la de RADIUS: es posible configurar más de un servidor TACACS para proporcionar mayor disponibilidad; se pueden especificar grupos de servidores TACACS para asociarlos a distintos servicios; o

se puede especificar la autenticación local como método de respaldo en caso de perder la conectividad con los servidores TACACS.

Los comandos que utilizar para TACACS son muy similares a los de RADIUS. Por ejemplo, para definir autenticación vía TACACS para el acceso a la gestión:

```
Aruba(config)# [no] aaa authentication <console|telnet|ssh|web> <enable|login> tacacs  
[servergroup <group-name>] <método-secundario>
```

Siendo el significado de los parámetros el mismo que en el caso de RADIUS.

Para definir un servidor TACACS es necesario introducir el siguiente comando:

```
Aruba(config)# [no] tacacs-server host <dir-IP> key KEY-STR
```

Por ejemplo:

```
Aruba(config)# tacacs-server host 1.2.3.4 key XXXXXX
```

Dentro del servidor TACACS, para proporcionar el nivel correcto de privilegios es necesario especificar el atributo “Max-privilege” en las credenciales del usuario. Si Service-Type = 15 se proporciona acceso en modo manager y si Service-Type = 14 o menor, se proporciona acceso en modo operator.

Para mas informacion sobre TACACS puede consultarse la guía “Access Security Guide” [2].

### 4.3 GESTIÓN DE PERMISOS DE USUARIOS

En ArubaOS existen dos formas de controlar los permisos asignados a los usuarios que acceden al equipo para gestionarlo: el modo clásico y el modo RBAC.

- **Modo Clásico.** Es el modo por defecto y en él existen, como ya se ha mencionado previamente, dos usuarios con niveles de permisos preasignados: *operator* y *manager*.
  - **Nivel Operador** (usuario *operator*), con acceso parcial a la gestión del conmutador (solo lectura), que le permite únicamente inspeccionar el estado del equipo y los eventos que en él se producen.
  - **Nivel Manager** (usuario *manager*), con acceso total al conmutador y a todos los contextos de configuración con permisos de lectura y escritura, posibilitando con ello la realización de todo tipo de cambios en la configuración.
- **Modo RBAC** (*Role Based Access Control*). Este nuevo modo, introducido en la versión 16.01 de ArubaOS, permite definir usuarios adicionales a los dos por defecto, así como especificar de forma precisa los permisos asignados a cada usuario.

Tal como se describe en la siguiente sección, mediante la creación de perfiles (*roles*), este modo permite crear nuevos usuarios definiendo sus capacidades para la

configuración, inspección y ejecución de comandos relativos a las diversas funcionalidades que ofrece el equipo. La definición de los roles es local al equipo y la autenticación de los usuarios se puede realizar localmente o mediante servidores RADIUS o TACACS.

### 4.3.1 MODO RBAC

El modo RBAC permite hasta 64 perfiles o roles diferentes:

- 3 roles por defecto: *operator*, *manager* y *default-security-group*.
- 16 roles precreados, denominados *Level-0*, *Level-1*, hasta llegar a *Level-15*.
- 45 roles adicionales totalmente definibles por el administrador.

De los 16 roles precreados, algunos tienen predefinidos los comandos que pueden ejecutar:

- *Level-0* (*Network-Diagnostic*), permite ejecutar los comandos: *ping*, *traceroute*, *ssh* y *telnet*.
- *Level-1* (*Network-Operator*), permite ejecutar los comandos del *Level-0* y todos los comandos “*show*” excepto “*show history*”, y todos los comandos “*display*” excepto “*display history*”.
- *Level-9* (*Designated-Administrator*), permite ejecutar todos los comandos, excepto los relativos a la definición de usuarios.
- *Level-15* (*Administrator*), permite ejecutar cualquier comando del equipo.

Los niveles 9 y 15 no pueden ser modificados por el usuario. El resto de roles precreados (del 2 al 8 y del 10 al 14) no tienen ningún comando definido y pueden ser modificados por el administrador.

En cuanto a los tres roles por defecto, el rol *operator* tiene los mismos permisos que *Network-Operator*, el rol *manager* tiene los mismos permisos que *Administrator*, y el rol *default-security-group*, permite únicamente ver, copiar y borrar los logs de seguridad. Ninguno de los tres puede ser modificado.

Los roles definidos en un equipo pueden consultarse con el comando:

```
Aruba# show authorization group
```

#### 4.3.1.1 Reglas para la definición de roles

La definición de un rol se realiza mediante reglas que especifican qué comandos de configuración, comandos de inspección y comandos de línea de acciones directas desde el equipo pueden ser ejecutadas. Es posible configurar hasta 1000 reglas por cada uno de los roles.

Las reglas pueden especificar los comandos o funcionalidades permitidas o denegadas. Es posible, además, configurar que se guarde registro en los logs del equipo cuando se utiliza una regla.

Existen cuatro tipos de reglas para la definición de perfiles RBAC:

- **Reglas de Comandos**, que permiten definir qué comandos concretos están permitidos o denegados para ese perfil. Por ejemplo:
  - Añadir al grupo (role) *network-admin* la regla 1 que permite ejecutar los comandos “*router ospf*” e “*ip address*”.

```
Aruba(config)# aaa authorization group "network-admin" 1 match-command
"command:router ospf;ip address" permit log
```

- Añadir al grupo *network-admin* la regla 2 que bloquea el uso del comando “*router ospf enable*”.

```
Aruba(config)# aaa authorization group "network-admin" 2 match-command
"command:configure router ospf enable" deny log
```

- **Reglas de Funcionalidades (Features)**, que permiten los permisos asociados a un rol en función de conjuntos de funcionalidades. En la versión 16.04 de ArubaOS se definen los siguientes conjuntos (features): *aaa*, *arp*, *cdp*, *ping*, *snmp*, *radius*, *syslog*, *tacacs*, *access-list IP*, *vlan*, *spanning-tree*, *dhcp*, *gvrp*, *igmp*, *router*, *port-security*, *dldp*, *lldp*, *crypto*, *mac-access-list*, *telnet*, *smart-link group*, *sntp*, *mirror*, *Rmon*, *interface*, *ip*, *ipv6*, *QoS*, *mesh*, *Policy*, *redundancy*, *sflow*, *rate-limit*, *trunk*, *terminal*, *tftp*, *ssh*, *copy* y *macsec*.
- Es posible, además, seleccionar qué tipos de comandos se permiten para cada conjunto: ‘*r*’, para permisos de lectura, ‘*w*’ para permisos de escritura, y ‘*x*’ para permisos de ejecución (p.e., *copy*, *delete*, *reset*, etc.). Por ejemplo:
  - Añadir al grupo *network-admin* la regla 10 que permite el uso las features OSPF tanto en configuración (*w*), comandos de inspección (*r*) y comandos de acción (*x*).

```
Aruba(config)# aaa authorization group "network-admin" 10 match-command
"feature:rw:ospf" permit log
```

- **Reglas de VLAN**, que limitan las VLANes con las que un usuario perteneciente a un perfil puede operar. Solo se permite una regla de VLAN por rol; si no se especifica ninguna regla, se asume el acceso a todas las VLANes. Por lo tanto, si una regla bloquea una o varias VLANes, todas lo demás estarán permitidas y si una regla permite una o varias VLANes, todas las demás estarán bloqueadas. Por ejemplo:
  - Añadir al grupo *network-admin* la regla 30 que bloquea (*deny*) a los usuarios asignados a ese rol el uso de comandos relacionado con las VLANes 10-12, 20 y 30-40.

```
Aruba(config)# aaa authorization group "network-admin" 30 match-command "policy:vlan:10-12,20,30-40" deny log
```

- Reglas de Interfaces, que limitan los interfaces de red con los que un usuario perteneciente a un rol puede operar. Solo se permite una regla de Interfaces por rol; si no existe ninguna regla se asume acceso a todos los interfaces. Por ello, si una regla bloquea un interfaz, todos los demás están permitidos y si una regla permite un interfaz, todos los demás estarán bloqueados. Ejemplo:
  - Añadir al grupo *network-admin* la regla 40 que bloquea (deny) el acceso a comandos relacionados con los interfaces A10-A12, A20 y L20-L24 a todo usuario asignado a ese rol:

```
Aruba(config)# aaa authorization group "network-admin" 40 match-command "policy:interface:A10-A12,A20,L20-L24" deny log
```

Nota: las reglas de comandos no son validadas por el sistema, por lo que es importante que el administrador defina las reglas sin errores.

#### 4.3.1.2 Secuencia de configuración de RBAC

El primer paso consiste en la activación del modo RBAC a través de la elección del método de autorización, local o mediante RADIUS/TACACS.

- Activación de autorización en base a las definiciones de roles guardadas en el equipo:

```
Aruba(config)# aaa authorization commands local
```

- Activación del método de la autorización de la misma forma que se haya realizado la autenticación. Este método solo funcionaría en caso de que la autenticación local tenga lugar (como sería el caso de pérdida de conectividad con el RADIUS por ejemplo)

```
Aruba(config)# aaa authorization commands auto
```

El paso siguiente consiste en la creación de las reglas y grupos mediante:

```
Aruba(config)# aaa authorization group <nombre_grupo> <n> match-command "comando/funcionalidad" [permit|deny] [log]
```

Como se ha mencionado, es posible definir reglas que permitan o denieguen ese comando y configurar la generación de una entrada en el log del equipo cuando esa regla sea ejecutada (parámetro *log*). Se debe de definir un número de secuencia *n* comprendido entre 1 y 2147483647.

Finalmente, se deberán crear los usuarios y asociarlos a los roles mediante:

```
Aruba(config)# aaa authentication local-user <username> group <rolename> password [plaintext|sha1|sha256]
```

Se recomienda introducir como método de cifrado sha256 y nunca usar texto plano. En el comando se asume que *grupo* es lo mismo que *role*. Si a un usuario no se le asigna a ningún grupo explícitamente, queda asignado por defecto al rol Level-1 (Network-Operator).

Para más información puede consultarse el capítulo sobre RBAC de la guía “Access Security Guide for ArubaOS-Switch” correspondiente a su modelo de conmutador [2].

## 4.4 ACTUALIZACIÓN DEL SOFTWARE DEL DISPOSITIVO

Los conmutadores ArubaOS tienen capacidad para mantener dos versiones de software distintas, cada una de ellas con su propia versión de software de arranque (*Boot ROM*).

Asimismo, el equipo es capaz de almacenar varias configuraciones distintas y proporciona comandos para que el usuario especifique con cual de las dos imágenes software y con cual de las configuraciones almacenadas se iniciará el equipo tras el siguiente rearranque.

### 4.4.1 VERSIONES DE SOFTWARE INSTALADAS Y EN EJECUCIÓN

Para verificar las versiones de software instaladas en el equipo se utiliza el comando *show flash*, que muestra las versiones de las imágenes almacenadas en cada uno de los dos slots disponibles (*primary* y *secondary*), así como el slot por defecto utilizado para el arranque. Por ejemplo:

```
Aruba# show flash
Image                Size (bytes) Date      Version
-----
Primary Image       : 29072623 06/22/18 WC.16.06.0006
Secondary Image     : 15642343 04/27/16 WC.16.02.0003

Boot ROM Version
-----
Primary Boot ROM Version : WC.16.01.0004
Secondary Boot ROM Version : WC.16.01.0004

Default Boot Image   : Primary
Default Boot ROM     : Primary
```

Para verificar qué versiones del software están en ejecución se utiliza el comando *show versión*, que muestra:

- La versión de Boot ROM utilizada y el slot desde el que ha arrancado.
- La versión de ArubaOS en ejecución y el slot desde el que ha arrancado.

Por ejemplo:

```
Aruba# show version
Image stamp:
/ws/swbuildm/rel_washington_qaoff/code/build/lvm(swbuildm_rel_washington_qaoff_rel_washington)
Jun 22 2018 12:55:58
WC.16.06.0006
```

```

607
Boot Image:      Primary
Boot ROM Version:  WC.16.01.0004
Active Boot ROM:  Primary

```

#### 4.4.2 SELECCIÓN DE VERSIÓN Y CONFIGURACIÓN A EJECUTAR

El equipo tiene capacidad de almacenar varias versiones distintas de ficheros de configuración y permite al usuario especificar cual de las configuraciones se utilizará en cada una de las dos particiones (slots).

Para conocer las configuraciones disponibles se utiliza el comando:

```

Aruba# show config files

Configuration files:

id | act pri sec | name
-----+-----
 1 | *  *  *  | config1
 2 |          | config-otra
 3 |          | config2
 4 |          |
 5 |          |

```

En este ejemplo se muestra que existen tres configuraciones distintas y que la configuración *config1* es la activa y que está asociada a la partición primaria y secundaria.

Para cambiar la configuración de arranque (*startup*) asociada a cada partición se utiliza el comando *startup-default*. Por ejemplo, para asociar la configuración *config-otra* a la partición secundaria:

```

Aruba# startup-default secondary config config-otra
Aruba# show config files
Configuration files:
id | act pri sec | name
-----+-----
 1 | *  *  *  | config1
 2 |          *  | config-otra
 3 |          | config2
 4 |          |
 5 |          |

```

Si se rearranca el equipo, se cargará el software de la partición definida por defecto y la configuración asociada a esta. El comando *boot* permite ignorar esta configuración por defecto y seleccionar manualmente la imagen del sistema operativo (primary o secondary) y configuración que utilizará el equipo después del próximo arranque. Por ejemplo, para arrancar el equipo desde el slot secundario y utilizar la configuración *config-otra*:

```

Aruba# boot system flash secondary config config-otra

```

### 4.4.3 ACTUALIZACIÓN DEL SISTEMA

Es muy importante mantener el software del equipo actualizado a las últimas versiones publicadas, con el fin de incorporar todas las correcciones de seguridad incorporadas al software por el fabricante.

La actualización del sistema operativo puede realizarse desde la línea de comandos (CLI) o a través del interfaz gráfico (GUI). Antes de actualizar es muy recomendable hacer una copia de respaldo de la configuración (al menos dentro del propio conmutador o, para más seguridad, a un servidor externo) utilizando el comando copy:

```
Aruba# copy config config1 config config1.bak
```

Esta copia nos permitirá recuperar la configuración original en caso de tener que volver atrás en una actualización de versiones, ya que en ocasiones el formato de los comandos puede variar entre distintas versiones y la actualización provoca cambios no deseados en la configuración.

Mediante el CLI es posible usar TFTP y SFTP para actualizar el equipo via interfaces de red IP, así como XMODEM usando el puerto de consola. Se recomienda el uso de SFTP por ser un protocolo seguro. XMODEM debe usarse obligatoriamente si no se dispone de ninguna versión del software en el equipo y, por tanto, no puede usarse la red. No se recomienda en ningún caso el uso de TFTP.

Por ejemplo, para actualizar el software desde el CLI mediante sftp desde un servidor 10.1.1.1 a la partición secundaria puede utilizarse el comando siguiente:

```
copy sftp flash usuario@10.1.1.1 K_15_10_0001.swi secondary
The secondary image will be deleted.

Continue (y/n)?
Validating and Writing System Software to FLASH...
```

Adicionalmente a TFTP y SFTP, mediante el interfaz gráfico es posible actualizar el software subiendo la imagen desde un fichero local al equipo donde corre el navegador web. Esta es una opción muy cómoda, aunque en este caso para garantizar la seguridad se recomienda el uso de HTTPS en el acceso al interfaz web gráfico.

Puede encontrarse más información sobre la actualización del software en la sección “Managing switch software” de [1].

## 4.5 SWITCH IDENTITY PROFILE

Con el objeto de facilitar las tareas de creación y solicitud de certificados en caso de utilizarse en el equipo (por ejemplo, para el acceso a las gestión vía HTTPS), se recomienda crear un perfil de identidad del equipo, para evitar tener que completar los datos en cada certificado solicitado. Para ello se debe utilizar el comando:

```
Aruba(config)# crypto pki identity-profile [switch-id-profile] subject common-name [switch-name] country [country] state [state] locality [localityName] org [organization] org-unit [organization-unit]
```

Por ejemplo:

```
Aruba(config)# crypto pki identity-profile switch-id-profile subject common-name aruba-sw country ES state Madrid locality Madrid org CCN org-unit CERT
```

## 5. SEGURIDAD EN EL ACCESO A LA ADMINISTRACIÓN

Es imprescindible prevenir que usuarios no autorizados puedan acceder a los equipos y visualizar o cambiar la información de configuración almacenada en ellos. Es necesario, por lo tanto, implementar medidas de seguridad que impidan esos accesos no deseados, tanto de aquellos usuarios que intenten acceder a nuestros equipos desde dentro o fuera de nuestra red a través de los interfaces de datos, como de los que traten de hacerlo por medio del acceso físico al equipo y la conexión a los puertos de consola.

### 5.1 CONFIGURACIÓN INICIAL DEL EQUIPO

Es muy importante tener en cuenta que la configuración inicial del equipo cuando procede de fábrica o después de haberse reseteado es muy insegura. Dicha configuración es la siguiente:

- En el conmutador existen dos usuarios predefinidos, *manager* y *operator*, sin contraseña asignada.
- Todos los puertos están configurados en modo acceso y asociados a la VLAN 1 que tiene activo el servicio DHCP como cliente para obtener una dirección IP desde un servidor externo.
- Existen múltiples servicios activos: Telnet, SSH, SNMP con la comunidad *public* configurada, LLDP, así como la gestión del switch a través de la web.

Por ejemplo, dicha configuración de fábrica para un equipo 2930F sería la siguiente:

```
Aruba-2930F-24G-PoEP-4SFPP# show running-config
Running configuration:

; J1255A Configuration Editor; Created on release #WC.16.03.0004
; Ver #10:08.3f.ff.bb.ef.7c.59.fc.6b.fb.9f.fc.ff.ff.37.ef:7e
hostname "Aruba-2930F-24G-PoEP-4SFPP"
module 1 type j1255a
snmp-server community "public" unrestricted
vlan 1
  name "DEFAULT_VLAN"
  untagged 1-28
  ip address dhcp-bootp
  exit
```

Con esta configuración, se puede acceder libremente a la gestión del conmutador a través de la consola serie. Asimismo, en cuanto uno de los puertos de datos se conecte a una red que le proporcione una dirección IP por DHCP, la gestión del conmutador estará también accesible por telnet, SSH o WWW sin ningún control de acceso, o mediante SNMP utilizando la clave comunmente conocida "public".

Por ello, es imprescindible asegurar la configuración del equipo en un entorno seguro y aislado antes de desplegarlo en la red. Como mínimo, se deberán asignar claves de acceso a los usuarios de administración, desactivar todos los servicios que no se vayan

a utilizar o que no presenten la seguridad adecuada, asegurar los servicios que queden activos y deshabilitar todos los interfaces de red que no se vayan a utilizar, todo ello siguiendo las recomendaciones que aparecen en en los subapartados posteriores de esta guía.

## 5.2 GESTIÓN LOCAL DEL EQUIPO

### 5.2.1 PUERTO DE CONSOLA

Los equipos están dotados de un puerto de consola serie con conector RJ45 y, opcionalmente, de una segunda consola en formato micro USB tipo B. Ambos puertos pueden usarse indistintamente: son dos puertos de la misma conexión lógica.

El puerto de consola permite el acceso a todas la funcionalidades de gestión del Sistema Operativo (otras formas de acceso como el interfaz web solo permiten el acceso parcial), así como a las funcionalidades ofrecidas por el software de arranque del equipo (Boot ROM).

Algunas funciones, como la recuperación de las credenciales del equipo (procedimiento de *password recovery*) solo están disponibles desde el puerto de consola.

El uso del puerto de consola requiere proximidad física al equipo, con el objeto de conectar un terminal serie al mismo, aunque también es posible acceder remotamente a la consola si se utiliza un servidor de terminales (equipo que permite el acceso remoto a una o varias líneas serie a través de redes IP).

Como se ha mencionado, en el estado inicial de fábrica del equipo, la consola permite el acceso total a la gestión. Por tanto, la primera tarea a realizar debe ser la de crear y proteger con clave dos cuentas con niveles de privilegio de manager y de operador:

```
Aruba(config)# password operator user-name "operador" plaintext xxxx
Aruba(config)# password manager user-name "admin" plaintext yyyy
```

Desde el momento en que se ejecuta el primero de los comandos anteriores, el conmutador activa el control de acceso a la gestión del equipo y solo se permite el acceso con las credenciales de los dos usuarios definidos.

Por otro lado, es importante evitar que las sesiones establecidas en la consola del equipo se queden abiertas y puedan ser utilizadas con fines espurios. Por ello, es importante cerrar siempre las sesiones de gestión utilizando el comando *logout*, así como fijar un periodo máximo de inactividad para las sesiones, de forma que estas se cierren automáticamente transcurrido ese periodo de inactividad.

Para fijar el periodo de inactividad se utiliza el comando:

```
Aruba(config)# console idle-timeout <0-7200>
```

El periodo puede configurarse entre 1 y 7200 segundos. Un tiempo de 0 cancela la funcionalidad. Se recomienda un valor de unos 120-180 segundos.

El comando anterior (*console idle-timeout*) no solo afecta a las sesiones de consola, si no también a las sesiones SSH y TELNET. Existe además un comando más general que permite definir un tiempo de inactividad que afecta a todas las sesiones de gestión (consola, SSH, TELNET y WEB):

```
Aruba(config)# idle-timeout <tiempo-en-minutos>
```

Adicionalmente, se recomienda configurar un mensaje de bienvenida (banner) que indique que el acceso no autorizado a este equipo está prohibido. Dicho texto no debe proporcionar ninguna información acerca del sistema accedido que pueda ser utilizada por un atacante.

Para configurar el banner se utiliza el comando:

```
Aruba(config)# banner motd <caracter-delimitador> TEXTO <caracter-delimitador>
```

El carácter delimitador se utiliza para permitir introducir mensajes compuestos por múltiples líneas. No puede ser usado en el texto introducido. Una vez introducido el mensaje, los saltos de línea aparecen como “\n” al mostrar la configuración (*show running-config*).

Por ejemplo, usando el carácter % como delimitador:

```
Aruba(config)# banner motd %  
Enter TEXT message. End with the character '%'  
  
Este es un sistema privado. Abandone la conexión si no tiene autorización.%
```

En la configuración aparecerá como:

```
banner motd "Este es un sistema privado. \nAbandone la conexión si no tiene autorización\n\n"
```

Al igual que el valor del timer de inactividad, el mensaje de bienvenida aplica también a las sesiones establecidas por TELNET y SSH descritas más adelante.

Asimismo, conviene asignar un nombre al equipo mediante el comando *hostname*. Por ejemplo:

```
Aruba(config)# hostname aruba-sw  
aruba-sw(config)#
```

## 5.2.2 PUERTO USB

En los modelos que tienen un puerto USB tipo A (distinto de los puertos USB micro-B de consola), este puede ser usado como medio de almacenamiento para desplegar, analizar y copiar configuraciones o versiones del software.

El uso de este puerto puede habilitarse o deshabilitarse vía comandos de configuración:

```
aruba-sw(config)# [no] usb-port
```

Como regla general, se recomienda mantener deshabilitado el puerto USB, activándolo cuando su uso sea necesario y desactivándolo a posteriori una vez usado.

### 5.3 GESTIÓN REMOTA DEL EQUIPO

El equipo puede ser gestionado mediante una conexión remota a través de los puertos Ethernet que dispone. A través de protocolos basados en la arquitectura TCP/IP como Telnet, SSH, HTTP, HTTPS y a través de una API se puede acceder a la configuración del equipo y a la obtención de información de configuración y rendimiento de este.

En los siguientes apartados se verán más en detalle los requisitos para el acceso remoto a la gestión, así como los distintos servicios disponibles y las recomendaciones a seguir para no poner en riesgo la seguridad del equipo.

#### 5.3.1 REQUISITOS Y RECOMENDACIONES INICIALES

Para poder gestionar de forma remota el equipo es necesario asignarle previamente una dirección IP en alguna de las VLANs. Por ejemplo, el siguiente comando asigna la dirección 10.1.9.2/24 en la VLAN 100:

```
aruba-sw(config)# vlan 100
aruba-sw(vlan-100)# name Gestion
aruba-sw(vlan-100)# ip address 10.1.9.2 255.255.255.0
aruba-sw(vlan-100)# exit
```

Esta dirección permitirá a los sistemas con acceso a dicha VLAN acceder a la gestión del conmutador a través de alguno de los mecanismos descritos en las siguientes subsecciones basados en TCP/IP (SSH, web, SNMP, etc.).

Si el equipo tiene asignadas direcciones IP en otras VLANs, el acceso a la gestión será posible desde cualquiera de los equipos que tengan acceso a esas VLANs con dirección IP. Este hecho, muy habitual en el caso de que el equipo se configure a nivel 3 como router entre VLANs, puede plantear problemas de seguridad, ya que expone la gestión del equipo a múltiples equipos locales o incluso remotos.

Con el objeto de solventar este problema, ArubaOS ofrece la posibilidad de definir una VLAN de gestión segura, diseñada para restringir el acceso a la gestión del equipo únicamente a aquellos equipos que estén conectados a esa VLAN. Esto es, sólo los clientes que estén conectados a puertos que son miembros de la VLAN de gestión podrán tener acceso al equipo para gestión. El resto de direcciones IP asignadas a otras VLANs no servirán para acceder a la gestión.

Para configurar esta funcionalidad basta con ejecutar el comando siguiente, donde se especifica la VLAN que debe ser configurada como VLAN de administración, especificada mediante su identificador numérico o su nombre:

```
aruba-sw(config)# management-vlan <vid|vlan-name>
```

Existen algunas restricciones a tener en cuenta: solo una VLAN por equipo puede ser configurada como VLAN de gestión, la dirección IP de gestión ha de ser estática y no puede ser configurados en la VLAN de gestión nada relacionado con protocolos de encaminamiento o IGMP. Se recomienda no hacer uso de la VLAN creada por defecto (VLAN 1) para esta configuración, para evitar que durante la configuración de nuevos equipos se pueda dar acceso por error a la VLAN de gestión.

Además, es muy recomendable que la red de gestión se utilice exclusivamente para la gestión de equipos y ningún sistema ajeno a la gestión tenga conectividad con ella. Con este modelo (denominado gestión fuera de banda) mejoramos sensiblemente la seguridad del conmutador mediante la separación del tráfico de gestión del resto de tráfico de nuestras redes.

Adicionalmente, en los casos en los que la configuración de una VLAN segura de gestión sea muy restrictiva o en el caso de que queramos asegurar todavía más la VLAN segura, es posible restringir el acceso a la gestión mediante la definición de hasta 10 direcciones IP o rangos de direcciones IPs desde las cuales el equipo atenderá las peticiones de conexión. Para ello se utiliza el comando:

```
aruba-sw(config)#ip authorized-managers <direction-IP> < mascara> <operator|manager> access-method [all|ssh|telnet|web|snmp|tftp]
```

Este comando permite además definir conjuntos de direcciones distintos para los dos niveles de privilegios (manager y operador) y para los distintos protocolos de acceso a la gestión: ssh, telnet, web (el acceso tipo web también incluye el acceso mediante la API REST) snmp y tftp.

Por ejemplo, para permitir el acceso como manager a la dirección 10.1.9.55 y el acceso como operador desde la 10.1.9.56:

```
aruba-sw(config)# ip authorized-managers 10.1.9.55 255.255.255.255 access manager
aruba-sw(config)# ip authorized-managers 10.1.9.56 255.255.255.255 access operator
```

O para permitir el acceso mediante ssh desde 10.1.9.55 y mediante web desde la 10.1.9.56:

```
aruba-sw(config)# ip authorized-managers 10.1.9.55 255.255.255.255 access-method ssh
aruba-sw(config)# ip authorized-managers 10.1.9.56 255.255.255.255 access-method web
```

Tenga en cuenta que esta última medida no protege contra ataques en los que se falsifique la dirección IP (IP-spoofing). Es por ello que nunca se debe depender únicamente de esta medida, si no que se debe combinar con el resto de medidas propuestas (vlan segura, uso de contraseñas seguras, etc.).

### 5.3.2 TELNET

El acceso a la gestión mediante el protocolo TELNET se considera inseguro, ya que toda la información intercambiada en la sesión de gestión remota se envía sin cifrar. No se recomienda ni siquiera en los casos en los que se utilice una red de gestión aislada,

dada la facilidad de obtener información sensible (usuarios y claves, por ejemplo) si alguien consigue capturar el tráfico de gestión.

Dado que este servicio viene activado de fábrica, deberá ser desactivado mediante el comando siguiente:

```
aruba-sw(config)# no telnet-server
```

### 5.3.3 SSH

El protocolo recomendado para el acceso remoto a la gestión de un equipo en entornos sensible es SSH, dado que toda la información enviada es cifrada mediante algoritmos modernos considerados fiables.

Para configurar el acceso por SSH se deben ejecutar los siguientes comandos, para crear la clave del servidor de SSH y activar el protocolo:

```
aruba-sw(config)# crypto key generate ssh  
aruba-sw(config)# ip ssh
```

Es recomendable establecer un tiempo límite de inactividad en la sesión, tras el cual la sesión se cerrará. El comando `idle-timeout`, ya presentado en la sección 5.2.1, permite configurar ese tiempo límite en segundos:

```
aruba-sw(config)# console idle-timeout <0-7200>
```

### 5.3.4 HTTP/HTTPS

Es posible configurar y administrar el equipo de forma remota mediante un navegador web. El equipo se comportará como servidor web y será necesario el uso de un cliente web actualizado para acceder a él.

ArubaOS permite la gestión vía HTTP o HTTPS (HTTP sobre SSL/TLS o HTTP seguro). Por la misma razón que no se recomienda el uso de TELNET, se recomienda deshabilitar el uso de HTTP, ya que tampoco envía la información cifrada. En el caso de habilitar la gestión vía web, se debe utilizar obligatoriamente el acceso securizado vía HTTPS.

Por defecto, la configuración de fábrica del equipo tiene activado el acceso por http. Para deshabilitarlo se debe ejecutar el comando siguiente:

```
aruba-sw(config)# no web-management plaintext
```

Para habilitar el acceso por HTTPS es necesario crear o cargar un certificado en el switch para que se utilice en las conexiones SSL cifradas entre el switch y el navegador cliente. Existen dos tipos de certificados posibles: autofirmados o firmados por una autoridad de certificación externa.

### 5.3.4.1 Uso de certificados autofirmados

Los más sencillos de utilizar son los certificados autofirmados, ya que su creación puede realizarse ejecutando un único comando en el conmutador. Aunque plantean el problema de que no son reconocidos por los navegadores web, al no estar firmados por una autoridad de certificación en la que confíen los navegadores. Este hecho exigirá que tengamos que aceptar el certificado como una excepción al conectarnos por primera vez.

Para crear un certificado autofirmado en el conmutador, se debe ejecutar el siguiente comando:

```
aruba-sw(config)# crypto pki enroll-self-signed certificate-name <nombre-certificado> valid-start < MM/DD[/[YY]YY]> valid-end <MM/DD[/[YY]YY]> usage web
```

Por ejemplo:

```
aruba-sw(config)# crypto pki enroll-self-signed certificate-name aruba-https valid-start 4/1/2019 valid-end 4/1/2025 usage web
```

Para ver los certificados almacenados en el equipo:

```
aruba-sw(config)# show crypto pki local-certificate
```

Expiration	Parent / Profile	Name	Usage
-----	-----	-----	-----
aruba-https	Web	2025/04/01	default
...			

Una vez creado el certificado se puede activar el acceso a la gestión via HTTPS:

```
aruba-sw(config)# web-management ssl
```

A partir de ese instante ya será posible acceder a la gestión del equipo cargando la dirección (URL):

```
https://<nombre-o-dirección-IP-gestión>
```

Al igual que con otras formas de acceso a la gestión, conviene configurar un valor al timer de inactividad para que las sesiones web se cierren automáticamente transcurrido ese tiempo. Para ello:

```
aruba-sw(config)# web-management idle-timeout 300
```

El valor especificado con el comando “*web-management idle-timeout*” prevalece al valor especificado por el comando general “*idle-timeout*”.

Si por alguna causa se quiere borrar el certificado creado se puede usar el comando:

```
aruba-sw(config)# crypto pki clear certificate-name acme-http
```

### 5.3.4.2 Uso de certificados firmados por una autoridad externa

En caso de utilizar HTTPS con un certificado firmado por una autoridad de certificación (CA) externa, el procedimiento se complica, ya que hay que generar en el

conmutador una petición de firma de certificado (Certificate Signing Request o CSR) que se debe enviar a la CA para que la firme y nos devuelva el certificado ya firmado para instalarlo en el conmutador.

Los pasos a seguir en este caso son los siguientes:

- Establecer un perfil de identidad (Switch Identity Profile). Por ejemplo:

```
aruba-sw(config)# crypto pki identity-profile switch-id-profile subject common-name
aruba-sw country ES state Madrid locality Madrid org CCN org-unit CERT

Aruba-2930F-24G-PoEP-4SFPP(config)# show crypto pki identity-profile
Switch Identity:
  ID Profile Name      : switch-id-profile
  Common Name (CN)    : aruba-sw
  Org Unit (OU)       : CERT
  Org Name (O)        : CCN
  Locality (L)        : Madrid
  State (ST)          : Madrid
  Country (C)         : ES
```

- Crear un nuevo perfil de cadena de confianza (Trusted Anchor o TA) para el acceso web:

```
aruba-sw(config)# crypto pki ta-profile webprofile

aruba-sw(config)# show crypto pki ta-profile
Profile Name      Profile Status      CRL Configured  OSCP Configured
-----
webprofile       Pending Root Certificate In... No                No
```

- Copiar el certificado raíz de la CA (cacert.pem) desde un servidor externo al switch y asociarlo al perfil TA *webprofile* creado anteriormente:

```
aruba-sw(config)# copy sftp ta-certificate webprofile root@servidor cacert.pem
Attempting username/password authentication...
Enter root@10.1.0.1's password: *****
SFTP download in progress.
000M Transfer is successful
```

Una vez copiado, nos deberá aparecer que el estado del perfil ha cambiado:

```
aruba-sw(config)# show crypto pki ta-profile
Profile Name      Profile Status      CRL Configured  OSCP Configured
-----
webprofile       Root Certificate Installed No                No
```

- Generar la petición CSR en el switch vinculada al perfil TA *webprofile*:

```
aruba-sw(config)# crypto pki create-csr certificate-name aruba-sw ta-profile webprofile
usage web key-type rsa key-size 2048
-----BEGIN CERTIFICATE REQUEST-----
MIICpDCCAYwCAQAwXzERMA8GA1UEAxMIYXJ1eEtc3cxDALBgNVBAsTBENFU1QxD
...
O5IFaYvmWORIafg0supB1i/wsLWdmOBC2KJU5vGuec30PNbHHnZnJnFptntowE=
-----END CERTIFICATE REQUEST-----
```

- Una vez creada la petición se debe copiar a la CA y guardarla en un fichero (*aruba-sw.csr*) que debe incluir las líneas delimitadoras inicial y final. Esa petición debe ser

procesada por la CA para crear el certificado (*aruba-sw.pem*). Por ejemplo, si se utiliza *openssl* en Linux, el comando a utilizar sería:

```
openssl ca -in aruba-sw.csr -out aruba-sw.pem
```

- A continuación, se debe copiar e instalar el nuevo certificado en el conmutador:

```
aruba-sw(config)# copy sftp local-certificate root@servidor aruba-sw.pem
Attempting username/password authentication...
Enter root@10.1.0.1's password: *****
SFTP download in progress.
000M Transfer is successful
```

Una vez copiado, podemos comprobar que se ha instalado correctamente:

```
aruba-sw(config)# show crypto pki local-certificate
Name                Usage            Expiration        Parent / Profile
-----
aruba-sw            Web              2020/04/24       webprofile
```

Finalmente arrancamos el servicio de acceso por HTTPS con ese certificado:

```
aruba-sw(config)# web-management ssl
aruba-sw(config)# web-management idle-timeout 300
```

Una vez ejecutados los pasos anteriores, podremos ya acceder a la gestión del switch por HTTPS desde un navegador que confíe en la autoridad de certificación utilizada para firmar el certificado. En el ejemplo anterior el certificado se ha creado asociado al nombre “aruba-sw”, por lo que para que el navegador acepte el certificado deberemos conectarnos a:

```
https://aruba-sw
```

En caso de querer borrar todo lo relativo a la PKI interna del conmutador se puede utilizar el comando:

```
aruba-sw(config)# crypto pki zeroize
```

### 5.3.5 ACCESO MEDIANTE API REST

Adicionalmente a la gestión vía web, los switches Aruba proporcionan acceso a la gestión mediante una API RESTFull. Esto permite gestionar los equipos desde scripts o aplicaciones que llamen a las primitivas incluidas en al API.

La API REST utiliza el protocolo HTTP o HTTPS, por lo que su configuración es común a la configuración de acceso vía HTTP/HTTPS descrita en el apartado anterior.

El comando para activar o desactivar el API rest es el siguiente:

```
aruba-sw(config)# [no] rest-interface
```

Se puede comprobar si esta activado mediante:

```
aruba-sw(config)# show rest-interface
```

## REST Interface - Server Configuration

```

REST Interface          : Enabled
REST Operational Status : Up
REST Session Idle Timeout : 600 seconds
HTTP Access            : Disabled
HTTPS Access           : Enabled
SSL Port                : 443

```

Dado que el procedimiento de acceso requiere autenticación (usuario y password), es recomendable la creación de un usuario específicamente dedicado para los accesos a través de la API. Se recomienda que estas credenciales sólo sean usadas por la aplicación, para un mejor control del acceso.

Para más información sobre las primitivas disponibles a través del API y cómo utilizarlas se debe consultar la documentación de ArubaOS [4].

### 5.3.6 SNMP

SNMP es el protocolo de intercambio de información de gestión entre plataformas de gestión y los equipos gestionados.

Este protocolo tiene dos facetas principales, que se diferencian principalmente en qué motiva el intercambio de información, así como la naturaleza del mismo:

- Generación de alarmas o eventos, llamados *traps*, que se envían desde los dispositivos hacia una o varias estaciones gestoras de la red. Notifican eventos o cambios de estado producidos en un equipo o en su entorno (por ejemplo, la caída de un enlace o un exceso de temperatura)
- Dialogo o interrogación, que permite a las estaciones gestoras, con las correspondientes credenciales, interrogar o mandar órdenes al equipo (por ejemplo, configurar la dirección IP o consultar estadísticas sobre el valor de paquetes transmitidos por una interfaz).

Actualmente existen tres versiones del protocolo SNMP. Los equipos ArubaOS soportan las tres versiones, aunque la única que proporciona mecanismos de seguridad y control adecuados es SNMPv3, por lo que es la versión que se debe utilizar en caso de necesitar el uso de SNMP en un sistema.

Dado que SNMPv2 viene por defecto habilitado en la configuración inicial, se debe deshabilitar, así como borrar la comunidad SNMP por defecto:

```

aruba-sw(config)# no snmp-server community public
Aruba(config)# no snmp-server enable

```

En caso de que el gestor de la red sólo trabaje con SNMPv2 y solo se utilice para recabar información, se puede configurar el modo restringido, en el cual sólo se pueden hacer lecturas de información, pero no modificaciones de la configuración.

```

Aruba(config)# snmp-server enable
Aruba(config)# snmp-server community readonly_community restricted

```

De cualquier modo, la mejor recomendación es la de actualizar el gestor para que soporte SNMPv3.

La configuración de SNMPv3 se realiza en 4 pasos, consistentes en: habilitar el servicio de SNMPv3, configurar los usuarios permitidos, configurar las comunidades SNMPv3 y configurar los receptores de traps SNMP.

Se presentan a continuación los comandos más básicos relacionados con SNMPv3. Dado que la configuración puede ser compleja, para ampliar detalles se recomienda consultar la guía *“ArubaOS-Switch Management and Configuration Guide for 16.04”*.

1. Habilitar SNMPv3, mediante el comando:

```
aruba-sw(config)# snmpv3 enable
```

Al ejecutar ese comando, se presenta un asistente que va preguntando los valores de los diversos parámetros de configuración. Se recomienda el uso de SHA como protocolo de autenticación y AES como protocolo de privacidad:

```
aruba-sw(config)# snmpv3 enable
SNMPv3 Initialization process.
Creating user 'initial'
Authentication Protocol: MD5
Enter authentication password: *****
Privacy protocol is DES
Enter privacy password: *****
User 'initial' has been created
Would you like to create a user that uses SHA? [y/n] y
Enter user name: snmpv3user
Authentication Protocol: SHA
Enter authentication password: *****
Privacy protocol is AES
Enter privacy password: *****
User creation is done. SNMPv3 is now functional.
Would you like to restrict SNMPv1 and SNMPv2c messages to have read only
access (you can set this later by the command 'snmpv3 restricted-access')? [y/n] y
```

2. Para configurar los usuarios permitidos, es necesario primero crearlos y luego asignarlos a grupos. Los grupos son los que tienen los permisos. El comando para la creación de usuarios es:

```
aruba-sw(config)# [no] snmpv3 user <user_name> auth <sha> <authentication_password> priv
<aes> <privacy_password>
```

Un ejemplo del uso de este comando es:

```
aruba-sw(config)# snmpv3 user user auth sha 123 priv aes 123
```

3. El comando para la asignación de grupos es:

```
aruba-sw(config)# [no] snmpv3 group <nombre_grupo> user <nombre_usuario> sec-model <ver3>
```

Un ejemplo de este comando es:

```
aruba-sw(config)# snmpv3 group managerpriv user user sec-model ver3
```

Cabe destacar que existen los siguientes grupos ya creados:

```
aruba-sw(config)# show snmpv3 group
commanagerrw      Comunidad con manager y acceso de escritura restringido
commanagerr      Comunidad con manager y acceso de escritura no
comoperatorrw    Comunidad con operator y acceso de escritura restringido
comoperatorrrw   Comunidad con operator y acceso de escritura no restringido
managerauth      Requiere autenticación, puede acceder a todos los objetos.
managerpriv      Requiere privacidad y autenticación, puede accede a todos los objetos
operatorauth     Requiere privacidad y autenticación, acceso limitado a objetos
operatornoauth   No requiere autenticación, acceso limitado a objetos
```

- Para configurar los receptores de traps SNMPv3 es necesario indicar la dirección IP del receptor de traps y el tipo de usuario establecido para su acceso al equipo. Se recomienda dar los mínimos privilegios, por lo que se configurará como operator a no ser que las necesidades del usuario necesiten privilegios más elevados. Se configuran de la siguiente forma:

```
aruba-sw(config)# ip authorized-managers <snmpServer_ip> <snmpServerip_mask> access
operator access-method snmp
```

Para configurar los receptores de traps se utilizan los comandos:

```
aruba-sw(config)# [no] snmpv3 notify <notify_name> tagvalue <tag_name>
aruba-sw(config)# [no] snmpv3 targetaddress {<ipv4-addr|ipv6-addr>} <name>
aruba-sw(config)# [no] snmpv3 params <params_name> user <user_name>
```

Finalmente, es interesante configurar notificaciones de Seguridad para eventos SNMP:

```
aruba-sw(config)#[no] snmp-server enable traps [snmp-auth | password-change-mgr | login-
failure-mgr| port-security | auth-server-fail | dhcp-snooping | arp-protect | running-
configchange]
```

Los tipos de notificaciones disponibles pueden consultarse en la Tabla 1.

Se recomienda activar la notificación de todo tipo de eventos, ya sean eventos críticos, informativos, etc. de la manera siguiente:

```
aruba-sw(config)# snmp-server host <DirecciónIP_ServidorSNMP> trap-level all community
<ComunidadSNMP>
```

## 5.4 POLÍTICAS DE CALIDAD DE CONTRASEÑAS

Cuando la autenticación de un sistema se basa principalmente en nombres de usuarios y claves, la seguridad depende en gran medida de la calidad de las contraseñas utilizadas. En ArubaOS existen diversos comandos para el control de la calidad de las contraseñas que se usan para la gestión de conmutador y otras funciones de control de acceso de usuarios.

Uno de los parámetros que se puede configurar es la longitud mínima en caracteres que debe tener una clave. Para fijarla se utiliza el comando:

```
aruba-sw(config)# password minimum-length <0-64>
```

Tipo	Descripción
arp-protect	Traps para Dynamic ARP Protection.
auth-server-fail	Traps que informan de un servidor de autenticación que no se puede alcanzar
dhcp-server	Traps para servidor DHCP
dhcp-snooping	Traps para DHCP-Snooping.
dhcpv6-snooping	Configurar traps para DHCPv6 snooping.
dyn-ip-lockdown	Traps para Dynamic Ip Lockdown
dyn-ipv6-lockdown	Habilitar traps para Dynamic IPv6 Lockdown.
link-change	Traps para link-up y link-down.
login-failure-mgr	Traps para fallo en la interfaz de administración (management)
mac-count-notify	Traps para notificar el exceso de direcciones MAC aprendidas en un puerto
mac-notify	Traps para notificar el cambio en la tabla de direcciones MAC (aprendidas/eliminadas)
macsec	Configura traps para notificación sobre macsec
nd-snooping	Configura las traps para nd snooping.
password-change-mgr	Traps para cambio de contraseña en la interfaz de administración
port-security	Traps para fallos en la autenticación mediante port access
running-config-change	Traps para cambio en la configuración en curso
snmp-authentication	Selección de traps RFC-1157 (estandar) o HP-ICF-SNMP (extendido)
startup-config-change	Traps para cambios en la configuración de inicio
vsf	Habilita traps para la funcionalidad VSF

**Tabla 1: Tipos de notificaciones SNMP**

No se deben utilizar claves de menos de 12 caracteres. Es importante destacar que tanto los nombres de usuario como las passwords son campos sensibles a las mayúsculas.

Adicionalmente, existen otros comandos para configurar la **complejidad** de la clave, de manera que:

- no contenga tres caracteres seguidos repetidos:

```
aruba-sw(config)# password complexity repeat-char-check
```

- no coincida con una utilizada anteriormente:

```
aruba-sw(config)# password complexity repeat-password-check
```

- no contenga el nombre del usuario:

```
aruba-sw(config)# password complexity user-name-check
```

- todas las anteriores:

```
aruba-sw(config)# password complexity all
```

Asimismo, se puede controlar la composición de la contraseña, en términos del número de:

- letras minúsculas que debe contener (entre 2 y 15)

```
aruba-sw(config)# password composition lowercase <2-15>
```

- letras mayúsculas que debe contener (entre 2 y 15)

```
aruba-sw(config)# password composition uppercase <2-15>
```

- caracteres especiales que debe contener (entre 2 y 15)

```
aruba-sw(config)# password composition specialcharacters <2-15>
```

- cifras numéricas (entre 2 y 15)

```
aruba-sw(config)# password composition number <2-15>
```

El valor por defecto para cada uno de los comandos anteriores es 2.

El comportamiento del equipo respecto al uso de las contraseñas se puede particularizar también mediante estos comandos:

- Habilitar la comprobación de expiración de la clave:

```
aruba-sw(config)# password configuration aging
```

- Definir el periodo de expiración de la clave:

```
aruba-sw(config)# password configuration aging-period
```

- Definir el número de días previos a la expiración con que se genera el aviso:

```
aruba-sw(config)# password configuration alert-before-expiry
```

- Definir cuantos accesos (logines) adicionales se permiten tras la expiración:

```
aruba-sw(config)# password configuration expired-user-login
```

- Definir el número de horas a esperar antes de poder cambiar una clave:

```
aruba-sw(config)# password configuration update-interval-time <0-168>
```

- Habilitar la comparación de la nueva clave con las anteriores:

```
aruba-sw(config)# password configuration history
```

- Configurar el número de claves anteriores registradas:

```
aruba-sw(config)# password configuration history-record
```

- Configurar el que se oculte el resultado del comando “*show authentication last-login*”, que muestra detalles sobre las últimas sesiones de gestión:

```
aruba-sw(config)# password configuration log-on-details
```

- Definir el tiempo mínimo de espera, que debe aguardarse antes de permitir un cambio de password:

```
aruba-sw(config)# password configuration update-interval-time
```

Una vez configurados los parámetros de se debe activar la funcionalidad mediante el comando:

```
aruba-sw(config)# password configuration-control
```

Se recomienda la siguiente configuración:

```
password minimum-length 12
password composition lowercase 2
password composition uppercase 2
password composition specialcharacters 2
password composition number 2
password configuration history
password configuration history-record 3
password configuration aging-period 90
password configuration update-interval-time 0
password configuration-control
```

**Nota importante:** la funcionalidad de control de claves no es compatible con la gestión vía web o API REST.

#### 5.4.1 ALMACENAMIENTO Y CIFRADO DE CONTRASEÑAS

Las claves de usuario pueden almacenarse en el fichero de configuración o bien excluirse del mismo. Esto queda regulado a través del comando “*include-credentials*”.

Además, en caso de almacenarse en el fichero de configuración se puede obligar a que se guarden cifradas mediante el comando “*encrypt-credentials*”. Adicionalmente, si se ejecuta el comando “*password non-plaintext-sha256*”, todas las contraseñas a partir de ese momento deben tener sha256 como método de cifrado.

La configuración recomendada es la siguiente:

```
aruba-sw(config)# include-credentials
aruba-sw(config)# encrypt-credentials
aruba-sw(config)# password non-plaintext-sha256
```

#### 5.4.2 RECUPERACIÓN DE CONTRASEÑAS Y CONFIGURACIÓN DE FÁBRICA

En el panel frontal del equipo existen, entre otros, los botones *Reset* y *Clear* pensados para gestionar el rearranque y borrado de configuraciones del equipo. Para evitar una pulsación fortuita, estos botones han de ser pulsados con un elemento metálico (clip o

alambre). A continuación se detalla el funcionamiento de estos botones y el objetivo de uso que tienen:

- **Eliminación de contraseñas**

La pulsación del botón *Clear* durante al menos un segundo, provoca la eliminación de las claves (passwords) configuradas en el equipo. Esta funcionalidad puede ser útil en algunos entornos como los laboratorios de prueba de equipos, pero es muy peligrosa en entornos de producción.

Para evitar los problemas causados por la manipulación no autorizada de los botones *Clear* y *Reset*, el sistema operativo proporciona comandos para cambiar el comportamiento de dichos botones o directamente anular su funcionamiento.

En el caso de la funcionalidad de borrado de claves, es posible y totalmente recomendable configurar el equipo para que la pulsación de ese botón no elimine las claves. Para ello:

```
aruba-sw(config)# no front-panel-security password-clear
```

- **Restauración de la configuración de fábrica**

El procedimiento de restauración a la configuración de fábrica del equipo se realiza utilizando los botones *Clear* y *Reset* frontales mediante el procedimiento siguiente:

- Pulsar y mantener *Reset*.
- Pulsar y mantener *Clear*.
- Liberar el botón *Reset*.
- Cuando el LED Test parpadee, liberar *Clear*.

Transcurridos unos 30/50 segundos el equipo habrá arrancado de nuevo con la configuración de fábrica.

Es posible y recomendable inhabilitar este mecanismo para evitar que alguien teniendo acceso físico al equipo pueda obtener control de este mediante su reseteo. Para ello:

```
aruba-sw(config)# no front-panel-security factory-reset
```

Para activar de nuevo la funcionalidad de restauración a la configuración de fábrica:

```
aruba-sw(config)# front-panel-security factory-reset
```

- **Recuperación de contraseñas**

Está previsto que, en el caso que se hayan perdido las credenciales del equipo, sea posible obtener una contraseña de un solo uso (one-time password) contactando con el soporte de *Hewlett Packard Enterprise Aruba*.

Al igual que los mecanismos anteriores, este método puede también bloquearse mediante comandos de configuración si se desea. En tal caso, la pérdida de credenciales únicamente podría recuperarse restaurando el equipo de fábrica.

El procedimiento para evitar el uso de las claves de un solo uso obtenidas a través del soporte de HPE Aruba es el siguiente:

- Pulsar Clear.
- Dentro de los 60 segundos tras pulsar Clear, hemos de ejecutar el comando

```
aruba-sw(config)# no front-panel-security password-recovery
```

- Pulsar Y para verificar
- Tras ese punto, el equipo ya no permite la recuperación de claves.

En los equipos Aruba 3810M, además es posible seleccionar si la configuración de esta funcionalidad se incluye o no en la configuración a través de la ejecución del comando:

```
aruba-sw(config)# [no] front-panel-security display-in-config
```

## 6. SERVICIOS DE RED DEL EQUIPO

Para garantizar la seguridad de un equipo es imprescindible ser consciente de qué servicios de red tiene activados y tomar las medidas adecuadas para asegurarlos. Asimismo, se debe ser consciente de que mantener activo un servicio que no se utiliza puede convertirse en un problema de seguridad importante, ya que puede exponer vulnerabilidades que sean aprovechadas por atacantes. Por ello es muy importante desactivar todos aquellos servicios que no sean necesarios, sobre todo teniendo en cuenta que algunos de ellos suelen venir activados en las configuraciones por defecto.

### 6.1 LLDP Y CDP

El protocolo LLDP (Link Layer Discovery Protocol) estandarizado por el IEEE y el protocolo CDP (Cisco Discovery Protocol) propietario de CISCO son dos protocolos de nivel 2 que permiten compartir información entre equipos red (routers, conmutadores y otros) directamente conectados. Permiten a un equipo informar a sus vecinos de, por ejemplo, el nombre del equipo, el tipo o la versión del sistema operativo utilizada, así como proporcionar información sobre la configuración de las VLAN. Ambos protocolos consisten básicamente en el envío periódico de paquetes con la información mencionada hacia direcciones multicast que escuchan el resto de equipos. Los equipos ArubaOS tienen capacidad de utilizar tanto LLDP como CDP.

Ambos protocolos proporcionan información útil para la configuración y gestión de la red. Además, diversas herramientas y plataformas de gestión de red requieren su uso. Sin embargo, la información que proporcionan puede ser utilizada maliciosamente para conocer detalles sobre la red por parte de atacantes.

Por ello, si estos protocolos no son estrictamente necesarios para el funcionamiento de la red, se recomienda deshabilitarlos de forma global. En caso de que sean necesarios, se deben deshabilitar en los interfaces donde no se utilicen (p. ej., en los puertos de acceso a los que se conectan los equipos finales).

Para desactivar LLDP y CDP de forma global en un conmutador Aruba hay que usar los comandos:

```
aruba-sw(config)# no cdp run
aruba-sw(config)# no lldp run
```

Para desactivarlo selectivamente en un conjunto de puertos determinado se pueden utilizar los comandos:

```
aruba-sw(config)# no cdp enable <lista-de-puertos>
aruba-sw(config)# no lldp enable-notification <lista-de-puertos>
```

Siendo *<lista-de-puertos>* una sucesión separada por comas de números o rangos de puertos. Por ejemplo:

```
aruba-sw(config)# no lldp enable-notification 3,5-7,11-13
```

Finalmente, se puede obtener información sobre la configuración de LLDP y CDP mediante los comandos:

```
aruba-sw# show cdp
aruba-sw# show lldp config
```

## 6.2 ICMP

ICMP es el protocolo que define los mensajes de control de las redes IP, como, por ejemplo, las solicitudes/respuestas de eco utilizadas en ping o los mensajes de error enviados cuando se descarta un datagrama IP.

Dichos mensajes pueden ser utilizados por los atacantes para descubrir información de la red o para realizar ataques de denegación de servicio. Por ello, es muy recomendable desactivar todas aquellas opciones de uso de ICMP que no sean necesarias, tales como las respuestas a solicitudes de eco enviadas a direcciones de broadcast, los mensajes de redirección (pensados, por ejemplo, para escenarios con varios routers en una misma subred IP) y los mensajes de error de tipo “Destination unreachable” generados cuando el conmutador tiene activadas funciones de nivel 3 y descarta un paquete IP, por ejemplo, debido a que el TTL ha llegado a cero o no existe una ruta para encaminarlo.

Para desactivar las funciones de ICMP mencionadas:

```
aruba-sw(config)#no ip icmp unreachable
aruba-sw(config)#no ip icmp redirects
aruba-sw(config)#no ip icmp addrmask
aruba-sw(config)#no ip icmp echo broadcast-request
```

Existen además dos opciones adicionales para limitar el ritmo de envío de respuestas ICMP generadas por el conmutador y mitigar con ello posibles ataques de denegación de servicio.

Para activar la limitación de respuestas ICMP se debe utilizar el comando:

```
aruba-sw# ip icmp reply-limit
```

Y para definir el número máximo de paquetes ICMP enviados por segundo:

```
aruba-sw# ip icmp burst-normal <paquetes-icmp-por-segundo>
```

## 6.3 TRANSFERENCIA DE FICHEROS

Para transferir archivos a o desde el conmutador existen tres protocolos diferentes: TFTP, SFTP y SCP. El protocolo TFTP se considera inseguro, por no incluir capacidades de autenticación o cifrado. Por ello se recomienda que sea deshabilitado mediante los comandos:

```
aruba-sw(config)# no tftp server
aruba-sw(config)# no tftp client
```

Por lo tanto, se recomienda el uso de SFTP (Secure File Transfer Protocol) y SCP (Secure Copy Protocol) para el intercambio de ficheros. Para activarlo:

```
Aruba (config)# ip ssh filetransfer
TFTP and auto-TFTP are now disabled because they cannot be secured with SSH. TFTP can
be re-enabled with the 'tftp' command.
```

La activación de SSH desactiva por defecto, el protocolo TFTP como cliente y servidor, por lo que si se ejecuta este último comando no es necesario ejecutar los anteriores.

La copia de ficheros se realiza mediante el comando “*copy*”. Por ejemplo, para copiar la configuración de arranque del conmutador a un servidor usando sftp:

```
copy startup-config sftp admin@10.1.1.10 startup-config
```

## 6.4 SINCRONIZACIÓN DE TIEMPO

Los equipos ArubaOS soportan la sincronización del reloj interno con servidores de hora externos mediante los protocolos TimeP, NTP (Network Time Protocol) y SNTP (Simple NTP). Se recomienda usar NTP por su mayor precisión y su capacidad de gestionar más de una fuente de sincronización.

Para evitar posibles ataques relacionados con el protocolo NTP (p. ej., aquellos que buscan cambiar la hora de los equipos para realizar ataques del tipo reply o simples ataques DoS), es muy recomendable que los servidores de NTP utilizados en la organización tengan activada la opción de autenticación. De esta forma se pueden autenticar los mensajes intercambiados entre clientes y servidores, dificultando con ello la mayoría de los ataques.

Para configurar que el equipo forme parte del servicio NTP de la red es necesario: configurar el modo de trabajo, broadcast o unicast; configurar las claves de autenticación; configurar el número máximo de asociaciones de terceros equipos; configurar los servidores NTP desde los cuales sincronizarse; y, por último, habilitar NTP.

Los pasos que seguir para configurar que el reloj del equipo se sincronice con uno o varios servidores NTP externos son los siguientes:

1. Configurar si NTP funcionará en modo unicast o broadcast (depende de la configuración del servidor):

```
aruba-sw#(config)# ntp [unicast|broadcast]
```

2. Configurar la clave de autenticación, así como su identificador y el tipo de cifrado (se recomienda usar SHA1):

```
aruba-sw(config)# ntp authentication key-id 1 authentication-mode <sha1> key-value <key>
```

3. Configurar el número máximo de servidores que pueden estar asociados a este cliente (entre 1 y 8):

```
aruba-sw(config)# ntp max-association
```

4. Configurar la dirección IP del servidor NTP (repetir para cada servidor):

```
aruba-sw(config)# ntp server <NTPserverIP>
```

5. Se define el protocolo de sincronización utilizados (NTP):

```
aruba-sw(config)# timesync ntp
```

6. Se habilita el servicio NTP cliente:

```
aruba-sw(config)# ntp enable
```

Finalmente, se puede comprobar la configuración en el conmutador de NTP mediante:

```
aruba-sw(config)# show ntp associations
NTP Associations Entries
Remote
St T When Poll Reach Delay Offset Dispersion
-----
10.1.3.6 3 u 10234 8 37 0.000 0.000 15.56394
```

Y se comprueba que el reloj del equipo se ha sincronizado correctamente:

```
aruba-sw(config)# show time
Tue Apr 17 11:22:36 2018
```

## 7. CONTROL DE TRÁFICO Y SEGURIDAD EN LOS PUERTOS DE RED

En este capítulo se describirán una serie de medidas de seguridad que pueden aplicarse a los equipos con ArubaOS para mejorar la seguridad de los puertos de red de los conmutadores, así como controlar el tráfico que gestiona el equipo.

### 7.1 APAGADO DE PUERTOS

Una medida de seguridad básica consiste en deshabilitar todos aquellos puertos del conmutador que no se estén utilizando, con el objeto de evitar que alguien pueda conectar equipos no autorizados a esos puertos.

Para bloquear o cerrar los puertos no utilizados:

```
aruba-sw(config)# interface <Puerto> disable
```

Para conocer el estado de los puertos se puede utilizar el comando:

```
aruba-sw# show interfaces brief
```

### 7.2 MEDIDAS DE PROTECCIÓN DE PUERTOS

ArubaOS proporciona medidas específicas para poder configurar la seguridad a nivel de puerto de una forma precisa, definiendo qué equipos (direcciones MAC) pueden conectarse o limitando el número máximo de equipos que se conectan a cada puerto. Permite, además, configurar la generación de alarmas o incluso el bloqueo de los puertos en caso de que se detecten accesos no permitidos.

Para poder configurar estas funcionalidades es necesario tener un inventario de las direcciones físicas de los equipos que se van a conectar o una estimación de cuantos equipos se conectan en cada puerto. Asimismo, es necesario introducir algunos conceptos previos para conocer cuál de las configuraciones posibles se adapta mejor a nuestras necesidades.

Cuando se activa el control de seguridad en un puerto, se puede configurar:

- El número máximo de direcciones MAC permitidas:

```
aruba-sw(config)# port-security <num-puerto> address-limit <número-entradas>
```

- El modo en el que el puerto conoce las direcciones permitidas:
  - *Continuous*: las direcciones se aprenden de la forma estándar, siguiendo el algoritmo de aprendizaje-hacia-atrás (backward learning) estándar de los conmutadores Ethernet. Este es el modo por defecto.
  - *Limited-continuous*: las direcciones se aprenden de la forma estándar, pero se limita el número máximo de direcciones (1-32).

- *Static*: se establece un límite máximo de direcciones y se especifican de forma estática las direcciones permitidas. Si se especifican menos direcciones que el máximo, se permite aprender direcciones adicionales hasta el máximo fijado.
- *Configured*: solo se permiten direcciones MAC configuradas. Se deshabilita el aprendizaje.
- *Port-access*: se permiten solo las direcciones MAC de los dispositivos autenticados mediante 802.1X.

El modo de aprendizaje se configura mediante el comando:

```
aruba-sw(config)# port-security <num-puerto> learn-mode  
continuous|static|configured|port-access|limited-continuous
```

Y las direcciones MAC permitidas se configuran mediante el comando:

```
aruba-sw(config)# port-security <num-puerto> mac-address <dirección-mac>
```

Es importante tener en cuenta que las direcciones MAC aprendidas de forma dinámica se almacenan en la tabla de filtrado del switch durante un tiempo máximo (unos minutos típicamente) siguiendo el algoritmo estándar de los conmutadores Ethernet. Además, si el conmutador se reinicia se eliminan. Por el contrario, las direcciones configuradas de forma estática se añaden a la configuración del conmutador, por lo que se conservan en la tabla de filtrado tras un rearranque del equipo.

Asimismo, es necesario remarcar que existe un número máximo de direcciones MAC seguras que podemos configurar, y que viene fijado por el número máximo de direcciones MAC que el sistema es capaz de gestionar. Se pueden realizar ataques aprovechando esta capacidad máxima de retención de direcciones MAC, se verá más en detalle en el apartado de prevención de ataques.

Una vez activada la seguridad en un puerto del conmutador pueden producirse situaciones que violen los límites de seguridad definidos, bien porque el número máximo de direcciones MAC seguras en un puerto se ha alcanzado, o bien porque una dirección aprendida en un interfaz se ha detectado en otro interfaz de la misma VLAN.

Para detectar este tipo de situaciones, denominadas intrusiones, los conmutadores Aruba permiten definir una política de reacción frente a este tipo de eventos mediante el parámetro *action* del comando *port-security*:

```
aruba-sw(config)# port-security 4 action <opción>
```

Donde *<opción>* define el modo de proceder cuando se detecte un evento de las características descritas:

- *none*: no se realiza ninguna acción. Es el valor por defecto.
- *send-alarm*: se envía una trap SNMP cuando se detecta la intrusión.

- *send-disable*: se envía una trap SNMP y deshabilita el puerto en el que se ha detectado la intrusión.

Finalmente, para conocer la configuración de seguridad de los puertos del equipo se puede usar el comando:

```
aruba-sw# show port-security
aruba-sw# show port-security <port-number>
```

### 7.3 LIMITACIÓN DEL TRÁFICO DE BROADCAST Y OTROS TIPOS

Una tormenta de broadcast consiste en la presencia de un volumen de tráfico excesivo enviado a la dirección de difusión (broadcast) de la LAN. Dado que por definición el tráfico enviado a broadcast debe ser replicado en todos los puertos del conmutador, esto puede provocar un consumo excesivo de recursos del equipo y de ancho de banda de la red para hacer frente a este tráfico. En casos graves, incluso puede causar el colapso total del conmutador y de la red.

El origen de las tormentas de broadcast puede ser variado, aunque principalmente se producen por dos motivos: que existan bucles en la red y que no se tenga activado el protocolo Spanning Tree para gestionarlos; o que existan dispositivos finales que generen mucho tráfico de difusión o que retransmitan el tráfico enviado a la dirección de difusión que reciben de vuelta a la red.

En este último caso, el origen del tráfico puede deberse a la configuración errónea de un dispositivo o a un ataque de denegación de servicio, como, por ejemplo, los ataques conocidos como ataques Fraggle o Smurf.

En estos ataques, se generan paquetes de tipo UDP o ICMP suplantando la dirección de origen de la víctima y enviados a la dirección de broadcast de una subred IP. Cuando todas las máquinas conectadas a ella responden, generan una tormenta de respuestas hacia la víctima, consumiendo el ancho de banda de la red y congestionando el sistema atacado.

Para prevenir este tipo de ataques o los efectos de una configuración errónea, es importante implementar medidas que limiten el tráfico de tipo broadcast, de forma que nunca llegue a consumir todos los recursos de la red y quede un margen para que el equipo pueda seguir teniendo conectividad y no se quede aislado.

ArubaOS proporciona comandos para limitar el ancho de banda consumido por el tráfico de difusión. La configuración se realiza para cada puerto del conmutador, por lo que se pueden implementar políticas de limitación distintas para cada tipo de interface (acceso, inter-switch, etc.).

Para limitar el ancho de banda consumido por el tráfico de difusión en un puerto:

```
aruba-sw(config)# interface <Puerto>
aruba-sw(eth-1)# rate-limit bcast <in|out> <kbps|percent> <0-100000000|0-100>
```

Por ejemplo, para limitar al 10% el tráfico de difusión que entra por el interfaz 1:

```
aruba-sw(config)# interface 1
aruba-sw(eth-1)# rate-limit bcast in percent 10
```

No existe una recomendación general sobre el valor límite de tráfico de difusión, ya que depende de cada entorno particular. Sin embargo, un valor entre el 5 y 10% puede ser razonable para la mayoría de los casos.

Es posible con este mismo comando controlar otros tipos de tráfico, como el tráfico multicast, el tráfico ICMP o todo el tráfico del puerto:

```
aruba-sw(eth-1)# rate-limit
all          Set a rate limit for all traffic.
bcast       Set a rate limit for broadcast traffic.
icmp        Set a rate limit for ICMP traffic.
mcast       Set a rate limit for multicast traffic.
queues      Set a rate limit for each traffic queue.
```

## 7.4 USO DE VLAN COMO MEDIDA DE AISLAMIENTO

Las LAN virtuales o VLAN constituyen la funcionalidad básica de los conmutadores actuales para crear redes separadas dentro del mismo conmutador. A grandes rasgos, una VLAN es una forma de virtualizar un conmutador, creando múltiples redes lógicas sobre un mismo conmutador físico.

Cada puerto de un conmutador debe pertenecer a al menos una VLAN. El tráfico enviado por los sistemas conectados a cada VLAN queda confinado en esa VLAN: cada VLAN constituye un dominio de broadcast diferente. En este sentido, el uso de VLAN permite reducir el tráfico de difusión en las redes, confinando a cada VLAN los efectos de los problemas tales como las tormentas de broadcast.

Un puerto puede pertenecer a varias VLAN distintas, en cuyo caso se debe configurar en modo etiquetado. Es el caso habitual de los enlaces entre conmutadores o de los puertos conectados a servidores. Los puertos conectados a sistemas de usuarios suelen pertenecer a una única VLAN y estar configurados en modo no etiquetado.

Las VLAN pueden expandirse a varios conmutadores, de forma que sistemas conectados a conmutadores distintos pueden pertenecer a la misma VLAN.

La conexión entre VLAN debe realizarse a nivel 3, mediante routers externos o servicios de encaminamiento implementados en los propios conmutadores.

En ArubaOS, la configuración inicial del equipo consta de una única VLAN ya creada (VLAN 1), a la que pertenecen por defecto todos los interfaces. Para evitar posibles problemas en la red cuando se interconectan switches, es recomendable no utilizar dicha VLAN y crear una nueva VLAN en la que se agrupen todos los puertos inactivos.

Para crear una VLAN basta con realizar:

```
aruba-sw(config)# vlan <VLAN_ID>
```

Donde VLAN\_ID debe ser un identificador no usado del rango de VLAN permitido (2-4094). Para que un puerto pertenezca a una vlan en el modo etiquetado (*tagged*) o no etiquetado (*untagged*) se utiliza el siguiente comando ejecutado desde el contexto de la VLAN:

```
aruba-sw(config)# vlan <VLAN_ID>
aruba-sw(vlan-<VLAN_ID>)# <tagged|untagged> <Lista-de-puertos>
```

Cuando se configuran las VLANs y los puertos que pertenecen a ella hay que tener especial cuidado en introducir estrictamente los puertos que se hayan considerado de uso necesario.

Para conocer las VLAN definidas en un conmutador y sus características se puede utilizar el comando:

```
aruba-sw# show vlan
```

Existen soluciones que permiten automatizar la distribución de la información sobre las VLAN creadas en cada conmutador y evitar la configuración manual de la pertenencia a las VLAN creadas en los enlaces troncales entre conmutadores.

El protocolo GVRP (GARP VLAN Registration Protocol) o su sucesor MVRP (Multiple VLAN Registration Protocol) se utilizan para esta función. Su funcionamiento se basa en el intercambio de información entre conmutadores a través de los enlaces en los que están activados.

En ArubaOS, cuando se activa GVRP mediante el comando *gvrp*, por defecto se activa el funcionamiento de GVRP en todos sus interfaces. Por ejemplo:

```
aruba-sw(config)# gvrp
aruba-sw(config)# sh gvrp

GVRP support

Maximum VLANs to support [256] : 256
Primary VLAN : VLAN1
GVRP Enabled [No] : Yes

Port      Type      | Unknown VLAN Join  Leave  Leaveall
-----+-----
1         100/1000T | Learn      20    300    1000
2         100/1000T | Learn      20    300    1000
3         100/1000T | Learn      20    300    1000
```

Este hecho puede constituir un problema de seguridad, ya que el conmutador propaga la información sobre sus VLAN a través de todos sus puertos y esa información podría ser utilizada para conocer los identificadores de VLAN utilizados. Por ello, es importante deshabilitar el funcionamiento de GVRP en todos los puertos que no sean troncales o en los que no se necesite su uso.

Para deshabilitar GVRP, por ejemplo, en el interfaz 1 se debe usar el comando:

```
aruba-sw(config)# interface 1
aruba-sw(eth-1)# unknown-vlans disable
```

Si se utiliza MVRP (recomendado frente a GVRP), cuando se activa el protocolo con el comando `gvrp enable`, su funcionamiento no se activa en los enlaces, por lo que no es necesario desactivarlos:

```
aruba-sw(config)# mvrp enable
aruba-sw(config)# sh mvrp config

Configuration and Status - MVRP

Global MVRP status : Enabled

Port      Status   Periodic Registration  Join  Leave  LeaveAll  Periodic
-----  -
Timer     Type    Time   Timer   Timer   Timer     Timer
-----  -
1         Disabled Enabled Normal   20    300    1000    100
2         Disabled Enabled Normal   20    300    1000    100
3         Disabled Enabled Normal   20    300    1000    100
```

Para activar MVRP en un enlace es necesario ejecutar el comando:

```
aruba-sw(config)# interface 1 mvrp enable
```

Finalmente, si se quiere excluir el funcionamiento de un puerto en una VLAN, puede ser configurada la no pertenencia de un puerto a dicha VLAN a través de:

```
Aruba(eth-<Puerto>)# forbid vlan <VLAN>
```

## 7.5 VLAN PRIVADAS

Las VLAN privadas o PVLAN constituyen una manera de limitar la conectividad entre los sistemas pertenecientes a una VLAN, segregándolos en varios conjuntos de puertos o puertos aislados y definiendo qué tráficos están permitidos entre ellos.

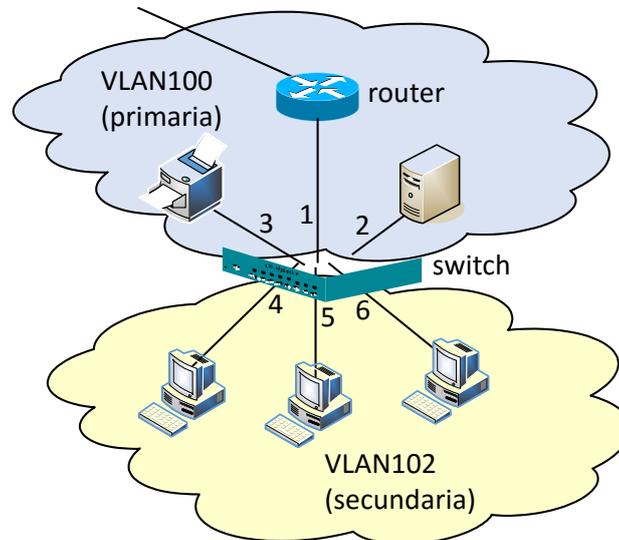
En una PVLAN existen tres tipos de puertos:

- **Puertos promiscuos (promiscuous)**, que pueden comunicarse con cualquier otro puerto de la VLAN. Son los puertos asignados típicamente a los servidores o routers.
- **Puertos aislados (isolated)**, que solo pueden comunicarse con puertos promiscuos.
- **Puertos de comunidad (community)**, que pueden comunicarse con los puertos promiscuos y con los puertos de su misma comunidad (se pueden crear múltiples comunidades de puertos).

Para crear una VLAN privada en ArubaOS se utilizan varias VLAN de dos tipos distintos:

- **VLAN primaria**, que es la responsable de distribuir el tráfico desde los puertos promiscuos a las VLAN secundarias en las que residen los puertos aislados o de comunidad. Los puertos promiscuos deben pertenecer a esta VLAN.
- **VLAN secundarias**, que son aquellas a las que pertenecen los puertos o conjuntos de puertos que no deben tener conectividad entre ellos. Típicamente se crea una

VLAN secundaria para los puertos aislados y otra por cada comunidad que se utilice. Para poder usar una VLAN secundaria es necesario asociarla a una VLAN primaria.



**Figura 3: ejemplo de PVLAN**

La configuración en ArubaOS de una PVLAN sencilla como la representada en la Figura 3 se realiza de la siguiente forma:

- Paso 1: crear una VLAN privada primaria:

```
aruba-sw(config)# vlan 100
aruba-sw(vlan-100)# name PVLAN-A-primary
aruba-sw(vlan-100)# private-vlan primary
```

- Paso 2: asignar los puertos promiscuos a la VLAN primaria:

```
aruba-sw(vlan-100)# untagged 1,2,3
```

- Paso 3: crear la VLAN secundaria y asociarla a la VLAN primaria creada anteriormente:

```
aruba-sw(config)# vlan 102
aruba-sw(vlan-102)# name PVLAN-A-isolated
aruba-sw(vlan-102)# vlan 100
aruba-sw(vlan-100)# private-vlan isolated 102
```

- Paso 4: asignar los puertos aislados a la VLAN secundaria:

```
aruba-sw(config)# vlan 102
aruba-sw(vlan-102)# untagged 4,5,6
```

El resumen de la configuración sería el siguiente:

```
vlan 100
 name "PVLAN-A-primary"
 private-vlan primary
 private-vlan isolated 102
 untagged 1,2,3
```

```
exit
vlan 102
  name "PVLAN-A-isolated"
  untagged 4,5,6
exit
```

Con esta configuración los ordenadores personales en los puertos 4, 5 y 6 podrían comunicarse con el servidor, la impresora y el router, pero no tendrían comunicación entre ellos. Para configuraciones más avanzadas de VLAN privadas se puede consultar la sección “Private VLANs” del manual [3].

## 7.6 PROTECCIÓN FRENTE A ENVÍO DE MENSAJES DE CONTROL STP

Cuando se utiliza el protocolo Spanning Tree (STP) en una red compuesta por varios conmutadores es muy importante protegerse frente al posible envío de mensajes de control (BPDU) falsos generados desde los puertos de acceso a la red.

Existen múltiples ataques documentados que mediante el envío de estas BPDUs falsas permiten redirigir la información hacia sistemas fraudulentos (ataques man-in-the-middle) o simplemente interrumpir el servicio (ataques de denegación de servicio).

Es por ello imprescindible proteger los conmutadores para que solo acepten BPDUs procedentes de los puertos que los conectan con otros conmutadores y descartar todas aquellas BPDUs recibidas a través de puertos de acceso.

Para activar la protección frente al envío de BPDUs en un conjunto de puertos, se debe utilizar el comando:

```
aruba-sw(config)# spanning-tree <lista-de-puertos> bpdu-protection
```

Una vez activado el comando, si se recibe una BPDU por alguno de los puertos especificados, se descartará la BPDU y se deshabilitará el puerto.

Adicionalmente, es posible generar una alarma (trap) de SNMP para avisar al gestor de red del evento detectado. Para ello hay que ejecutar el comando:

```
aruba-sw(config)# spanning-tree trap errant-bpdu
```

Para ver la configuración relativa a la protección frente a envío de BPDUs se puede ejecutar el comando:

```
aruba-sw# show spanning-tree bpdu-protection
```

## 7.7 LISTAS DE ACCESO IP

Las listas de acceso (Access Control List o ACL) son un método de filtrado de flujos de datos. Pueden ser usadas para restringir el acceso a la gestión del equipo de una forma más precisa que con el comando “*ip authorized-managers*” y para realizar el filtrado de tráfico que atraviesa el equipo. Aplicadas a los conmutadores, las ACLs se basan en un conjunto de reglas básicas (Access Control Entries o ACE) que determinan la autorización

(reglas de tipo *permit*) o denegación (reglas de tipo *deny*) del tráfico en función de campos de las cabeceras de los paquetes tales como las direcciones IP y puertos, tanto origen como destino, o el protocolo.

Por cada paquete recibido, las reglas que componen una ACL se evalúan línea a línea hasta que se encuentra una coincidencia. Por esta razón, hay que definir primero las reglas más específicas y posteriormente las más generales. Es importante recordar que las listas de acceso tienen una denegación implícita al final (todo el tráfico se filtra por defecto salvo que se permita explícitamente). Por ello, en el caso en que queramos restringir un determinado tráfico, debemos asegurarnos de permitir el resto de tráfico en una regla final.

En ArubaOS las ACL pueden aplicarse, entre otros, a puertos específicos o a VLANs completas. También es posible definir ACLs asociadas a usuarios o grupos de usuarios concretos que se cargan dinámicamente desde servidores RADIUS tras el proceso de autenticación. Por ello, es importante tener en cuenta que varias ACLs pueden estar aplicándose simultáneamente a un determinado tráfico.

ArubaOS permite crear listas de acceso tanto para IPv4 como IPv6. Aunque solo se muestran en este apartado los principales comandos para crear listas para IPv4, su utilización para IPv6 es inmediata, sin más que cambiar el formato de las direcciones.

### 7.7.1 TIPOS DE ACLS

Existen dos tipos de ACL según el procedimiento usado para definir las: **estáticas** o **dinámicas**.

- Las **ACLs estáticas** se configuran mediante comandos en el propio equipo. Pueden ser usadas para asignarlas a puertos o a VLANs con el objeto de restringir el tráfico entrante o saliente.
- Por el contrario, una **ACL dinámica** no se define en el propio equipo, sino que se configura en un servidor RADIUS y se carga dinámicamente en el conmutador como resultado de un proceso de autenticación. La existencia de las ACL dinámicas está ligada a la sesión de autenticación: una vez el cliente cierra la sesión, se libera la ACL.

En este apartado nos centramos únicamente en las ACL estáticas.

Según su aplicación, podemos distinguir también varios tipos de ACL: **Port ACL** o **PAACL**, cuando se aplica a un puerto; **VLAN ACL** o **VACL** cuando se aplica al tráfico interno de una VLAN; o **Routed ACL** o **RAACL**, cuando se aplica al tráfico que entra o sale de una VLAN cuando es el propio equipo el que realiza el encaminamiento a nivel 3.

Según cómo se especifica el tráfico a filtrar, existen dos tipos de listas de acceso:

- **ACL standard**, que filtran el tráfico únicamente en función de las direcciones IP origen. Se identifican mediante un nombre o un número en el rango 1-99.

- **ACL extendida**, que permiten filtrar el tráfico basándose en un número de campos mucho más amplio, principalmente las direcciones IP y los puertos, tanto origen como destino, así como los protocolos y puertos utilizados. También se identifican mediante un nombre o un número en el rango 100-199.

Dada su mayor expresividad y potencial se recomienda siempre utilizar listas de acceso extendidas.

### 7.7.2 CREACION DE ACLS

Para crear una nueva lista de acceso se debe usar el comando:

```
aruba-sw(config)# ip access-list <standard|extended> <Nombre-ACL|Número>
```

Con el parámetro *<standard|extended>* se selecciona el tipo de lista: estándar o extendida. Con el parámetro *<Nombre-ACL|Número>* se define el identificador de la lista de acceso, que puede ser un nombre o un número (1-99 para lista estándar y 100-199 para listas extendidas).

Se recomienda identificar las listas con nombres y que estos sean descriptivos (p.ej.: DENY-PORT-23, DENY-HOST-X, etc.).

En el caso de las **listas estándar**, el formato de las distintas entradas (ACE) que la componen es el siguiente:

```
permit|deny <origen> [log]
```

Donde *<origen>* es el parámetro que especifica el conjunto de direcciones IP origen para las cuales se aplicará la regla. Este se puede especificar mediante:

- *any*, que representa cualquier dirección.
- *host <SA>*, que especifica una determinada dirección IP. Por ejemplo: *host 10.1.0.12*.
- *<SA> <mask>/<SA>/<mask-length>*, que especifica un rango de direcciones, descrito con el formato clásico (*prefijo máscara invertida*) o con el formato compacto (*prefijo/longitud*). Por ejemplo: *10.1.0.0 0.0.255.255* o *10.1.0.0/16*.

El parámetro opcional [log] permite especificar si se quiere que se guarde una entrada en el log del equipo cada vez que se aplique esa regla.

Con ello, el formato completo del comando para crear una ACE de una ACL estándar es el siguiente:

```
aruba-sw(config-std-nacl)# permit|deny <any|host <SA>|<SA> <mask>|<SA>/<mask-length>> [log]
```

Por ejemplo, para crear una ACL que filtre todos los paquetes procedentes de la dirección IP 10.1.0.12:

```
aruba-sw(config)# ip access-list standard DENY-HOST12
aruba-sw(config-std-nacl)# deny host 10.1.0.12
```

```
aruba-sw(config-std-nacl)# permit any
aruba-sw(config-std-nacl)# exit
```

En el caso de las **listas extendidas**, el formato de las ACE es el siguiente:

```
permit|deny <protocolo> <origen> <destino> [parametros adicionales] [log]
```

Donde: *<protocolo>* especifica el valor el protocolo al que aplica la regla; *<origen>* y *<destino>* especifican los rangos de direcciones origen y destino siguiendo el mismo formato que el mencionado para las listas de acceso estándar; *[parametros adicionales]* especifican algunos campos adicionales de la cabecera IP (precencia y tipo de servicio); y *[log]* especifica si se debe crear entradas en el log del equipo.

Algunos valores posibles para el campo *<protocolo>* son: *ip*, para cualquier paquete IP; *tcp* o *udp*, para cualquier paquete IP que transporte un segmento TCP o UDP respectivamente; o *icmp*, para mensajes ICMP.

Con ello, el formato completo del comando para crear una ACE de una ACL extendida es el siguiente:

```
aruba-sw(config-ext-nacl)# permit|deny <ip|ip-protocol|ip-protocol-nbr> <any|host>
<SA>|SA|mask-length|SA <mask>> <any|host> <DA>|DA|mask-length|DA <mask>> [precedence] [tos]
[log]
```

Por ejemplo, para crear una ACL que filtre todos los paquetes procedentes de la dirección IP 10.1.0.12:

```
aruba-sw(config)# ip access-list extended EXT-DENY-HOST12
aruba-sw(config-ext-nacl)# deny ip host 10.1.0.12 any
aruba-sw(config-ext-nacl)# permit ip any any
aruba-sw(config-ext-nacl)# exit
```

O para filtrar cualquier tráfico dirigido hacia el puerto de TELNET (23):

```
aruba-sw(config)# ip access-list extended EXT-DENY-TELNET
aruba-sw(config-ext-nacl)# deny tcp any any eq 23
aruba-sw(config-ext-nacl)# permit tcp any any
aruba-sw(config-ext-nacl)# exit
```

Para mostrar las listas de acceso configuradas en un conmutador o los detalles de alguna de ellas se pueden utilizar los comandos:

```
aruba-sw# show access-list
aruba-sw# show access-list <ACL>
```

Existen comandos adicionales para modificar las listas de acceso e introducir nuevas entradas. Consulte el manual [2] para conocerlas.

### 7.7.3 APLICACION DE LISTAS DE ACCESO

Las listas de acceso pueden aplicarse directamente a puertos del conmutador, definiendo si aplican al tráfico entrante o saliente:

```
aruba-sw(config)# interface <Puerto>
```

```
aruba-sw(eth-X)# ip access-group <ACL> in|out
```

Donde <ACL> es el nombre o identificador numérico de la lista de acces a aplicar, y *in/out* especifica si la ACL se aplica al tráfico entrante en el conmutador (*in*) o saliente (*out*).

Por ejemplo, para aplicar la ACL EXT-DENY-HOST12 al tráfico entrante por el puerto 7 de un conmutador:

```
aruba-sw(config)# interface 7
aruba-sw(eth-7)# ip access-group EXT-DENY-HOST12 in
```

También es posible aplicar las listas de acceso a una VLAN completa (equivale a aplicar la lista a todos los puertos que pertenecen a dicha VLAN). Para ello:

```
aruba-sw(config)# vlan <id-vlan>
aruba-sw(eth-X)# ip access-group <ACL> in|out|vlan-in|vlan-out
```

En este caso la especificación del último parámetro admite cuatro valores posibles:

- *vlan-in*: la ACL se aplicará al tráfico entrante en la VLAN procedente de los puertos pertenecientes a esa VLAN o de los interfaces de nivel 3 del conmutador en esa VLAN.
- *vlan-out*: la ACL se aplicará al tráfico saliente de la VLAN hacia sistemas conectados a los puertos pertenecientes a esa VLAN o hacia los interfaces de nivel 3 del conmutador en esa VLAN.
- *in*: la ACL se aplicará al tráfico que el conmutador ha encaminado a nivel 3 desde esta VLAN hacia otras VLAN.
- *out*: la ACL se aplicará al tráfico que el conmutador ha encaminado a nivel 3 desde otras VLAN hacia esta VLAN.

Por ejemplo, para aplicar la ACL EXT-DENY-TELNET al tráfico entrante en la VLAN 2:

```
aruba-sw(config)# vlan 2
aruba-sw(vlan-2)# ip access-group EXT-DENY-TELNET vlan-in
```

Es posible conocer estadísticas sobre el número de veces que se aplican las distintas reglas de una ACL. Por ejemplo, para conocer esos datos para el ejemplo anterior:

```
aruba-sw# show statistics aclv4 EXT-DENY-TELNET vlan 2 vlan-in
```

Más información sobre las listas de acceso IP puede encontrarse en [2].

## 7.8 LISTAS DE ACCESO MAC

De forma análoga a la listas de acceso IP descritas en el apartado anterior, ArubaOS permite filtrar el tráfico en función de campos de las cabeceras de nivel 2 de las tramas

Ethernet: direcciones MAC origen y destino, protocolo (Ethertype), identificador de VLAN ID y clase de servicio.

La configuración de las listas de acceso de nivel 2 es muy similar a las listas de acceso IP, por lo que los conceptos y pasos a seguir son prácticamente los mismos: es posible crear listas de acceso estandar y extendidas; las listas pueden ser identificadas por nombre o por número; y también pueden ser asignadas a un puerto o a una VLAN completa.

El formato de las entradas en las listas extendidas es el siguiente:

```
SEQ-NUM <permit|deny> <any | host SRC-MAC | SRC-MAC SRC-MAC-MASK> <any | host DST-MAC | DST-MAC DST-MAC-MASK <any | ETHERTYPE> [cos COS] [log]
```

Destacar que a la hora de especificar la dirección MAC origen o destino se puede especificar una máscara, por lo que es posible filtrar en función de prefijos. Por ejemplo, se podrían permitir o filtrar tramas procedentes de dispositivos de un determinado fabricante.

Por ejemplo, una sencilla lista extendida que nos permita filtrar todas las tramas procedentes de una determinada dirección MAC (02:fd:00:05:04:01) sería la siguiente:

```
aruba-sw(config)# mac-access-list extended FILTER-MAC1
aruba-sw(config-ext-mac1)# deny host 02:fd:00:05:04:01 any any
aruba-sw(config-ext-mac1)# permit any any any
```

Para aplicar esa lista a un puerto o una VLAN utilizaríamos los comandos:

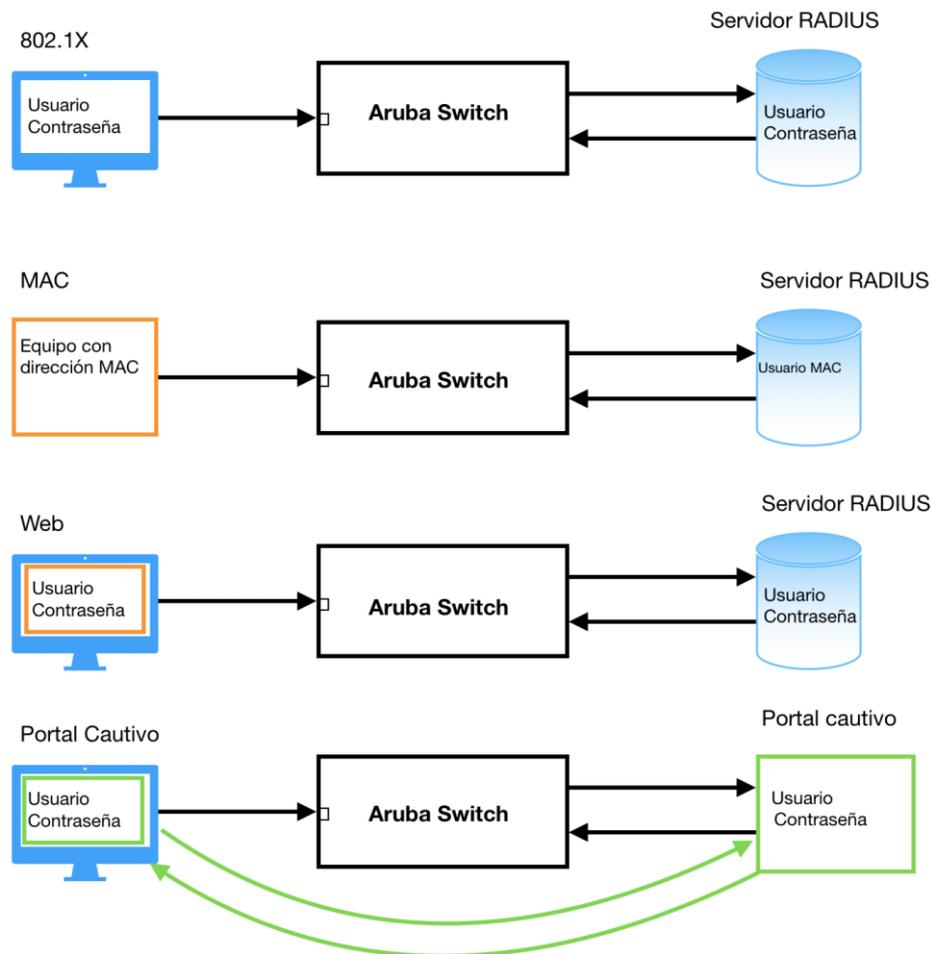
```
aruba-sw(config)# interface|vlan <puerto|VLANid>
aruba-sw(config)# mac-access-group FILTER-MAC1 in|out
```

Donde el parámetro *in* indicaría que la lista de acceso debe aplicarse a las tramas entrantes al puerto o VLAN, y *out* a las salientes.

Más información sobre las listas de acceso MAC puede encontrarse en [2].

## 8. SISTEMAS DE CONTROL DE ACCESO

ArubaOS proporciona distintos modos de controlar el acceso a la red de los equipos que se conectan directamente a ella, tal como se representa en la Figura 4. El equipo está preparado para usar el estándar 802.1X, tanto con usuarios locales al conmutador como con un servidor RADIUS externo. También es posible autenticar en función de la dirección MAC de los equipos que deseen acceder a la red, centralizando el registro de las direcciones MAC en un servidor RADIUS. Finalmente ofrece la posibilidad de realizar una autenticación mediante una página web de login alojada en el propio equipo o mediante el uso de un portal cautivo externo.



**Figura 4: métodos de control de acceso en ArubaOS**

Se describen a continuación las distintas formas de control de acceso soportadas. Para cualquiera de las modalidades, es recomendable configurar el registro de accesos a la red mediante:

```
aruba-sw(config)#[no] aaa accounting update periodic <time>
aruba-sw(config)#[no] aaa accounting network start-stop radius server-group <NombreGrupo>
```

Ejemplo de configuración:

```
aruba-sw(config)# aaa accounting update periodic 10
aruba-sw(config)# aaa accounting network start-stop radius server-group "RAD-TEST"
```

## 8.1 CONTROL DE ACCESO MEDIANTE 802.1X

802.1X es una norma del IEEE para el control de acceso a red utilizada tanto en redes Ethernet inalámbricas como cableadas. Permite autenticar los dispositivos que se conectan a la red con el objeto de autorizar o no el acceso a la red. La autenticación se lleva a cabo mediante un servidor RADIUS externo, aunque también puede configurarse localmente.

Para usar 802.1X en ArubaOS se debe activar la autenticación de acceso a los puertos del conmutador e indicar que se realizará mediante el servidor RADIUS:

```
aruba-sw(config)# aaa port-access authenticator active
aruba-sw(config)# aaa authentication port-access eap-radius server-group <nombre-grupo>
```

Siendo *<nombre-grupo>* el nombre de grupo de servidores RADIUS configurado. Por ejemplo:

```
aruba-sw(config)# aaa port-access authenticator active
aruba-sw(config)# aaa authentication port-access eap-radius server-group "RAD-TEST"
```

A continuación, hay que indicar qué puertos del conmutador tendrán autenticación 802.1X y configurar el límite de MACs por puerto mediante:

```
aruba-sw(config)# aaa port-access authenticator <lista-de-puertos>
aruba-sw(config)# aaa port-access authenticator <lista-de-puertos> client-limit <num-macs>
```

Siendo *<num-macs>* el número máximo de MACs permitidas por puerto (de 1 a 32). Si no se configura el límite es 1.

Ejemplo de configuración:

```
aruba-sw(config)# aaa port-access authenticator 3-4
aruba-sw(config)# aaa port-access authenticator 3 client-limit 2
aruba-sw(config)# aaa port-access authenticator 4 client-limit 4
```

Para comprobar que el control de acceso a puertos se ha configurado correctamente contra el servidor RADIUS, se puede utilizar el comando:

```
aruba-sw# show authentication

Status and Counters - Authentication Information
Authorized enabled as backup for secondary login are preceded by *

Login Attempts : 3
Lockout Delay : 0
Respect Privilege : Disabled
Bypass Username For Operator and Manager Access : Disabled

Access Task      | Login      Login      Login
                  | Primary    Server Group Secondary
-----+-----+-----+-----
```

```
...
Port-Access | EapRadius radius-grp None
...
```

Para ver los detalles sobre la configuración del control de acceso a puertos se pueden utilizar, por ejemplo, los siguientes comandos (ver más opciones en la ayuda):

```
aruba-sw# show port-access authenticator
aruba-sw# show port-access config
aruba-sw# show port-access clients
```

Para más información sobre la autenticación basada en 802.1x puede consultarse la sección “Configuring Port and User-Based Access Control (802.1X)” de [2].

## 8.2 AUTENTICACION BASADA EN DIRECCIONES MAC

Otra forma de controlar el acceso a los puertos del conmutador es mediante la autenticación basada en direcciones MAC y servidores RADIUS. Este método se puede utilizar para clientes y dispositivos que no son capaces de utilizar 802.1x. Es importante destacar que las direcciones MAC son fácilmente falsificables, por lo que este método no se debe utilizar en redes que requieran un control de acceso seguro.

A diferencia de la protección basada en direcciones MAC presentada en el apartado 7.2, en este caso se centraliza en el servidor RADIUS el registro de las direcciones MAC permitidas. Por defecto los puertos configurados con control de acceso por MAC están bloqueados.

Cuando se recibe tráfico en ellos, el conmutador envía una solicitud de autenticación al servidor RADIUS con la dirección MAC del cliente como nombre de usuario y clave. Si la dirección está registrada en el RADIUS, el puerto se desbloqueará.

Además de autenticar, es posible configurar la VLAN asignada al puerto tras la autenticación en función de la dirección MAC. En ArubaOS existen varias posibilidades:

- Configurar manualmente la VLAN del puerto utilizando los comandos estándar.
- Configurar la VLAN a asignar mediante un atributo en RADIUS asociado a la MAC.
- Configurar localmente la VLAN que se asignará a cada puerto una vez autenticado. Además, es posible configurar una VLAN en la que estarán los puertos que no se hayan autenticado.

Para configurar el control de acceso basado en direcciones MAC, primero se debe configurar la autenticación basada en MAC contra el servidor RADIUS:

```
aruba-sw(config)# aaa authentication mac-based chap-radius server-group <NombreGrupo>
```

Por ejemplo:

```
aruba-sw(config)# aaa authentication mac-based chap-radius server-group "RAD-TEST"
```

A continuación, se indicarán los puertos del conmutador en los que se quiere activar la autenticación basada en MAC802.1X, así como configurar el límite de MACs por puerto mediante:

```
aruba-sw(config)# aaa port-access mac-based <lista-de-puertos>
aruba-sw(config)# aaa port-access mac-based <lista-de-puertos> addr-limit <numMACs>
```

Opcionalmente, se pueden configurar las VLAN a asignar a los puertos autenticados y no autenticados respectivamente:

```
aruba-sw(config)# aaa port-access mac-based <lista-de-puertos> auth-vid <vlan>
aruba-sw(config)# aaa port-access mac-based <lista-de-puertos> unauth-vid <vlan>
```

Por ejemplo, para configurar el puerto 3 para que autentique en función de MAC, con un límite de cuatro direcciones y que se configure en la VLAN 100 en caso de autenticar y en la VLAN 99 en caso de no hacerlo:

```
aruba-sw(config)# aaa port-access mac-based 3
aruba-sw(config)# aaa port-access mac-based 3 client-limit 4
aruba-sw(config)# aaa port-access mac-based 3 auth-vid 100
aruba-sw(config)# aaa port-access mac-based 3 unauth-vid 99
```

Para ver las direcciones autenticadas y en uso en cada enlace se puede utilizar el comando:

```
aruba-sw# show port-access clients
```

Para comprobar la VLAN en la que está un puerto se puede utilizar el comando:

```
aruba-sw# show vlan port <num-puerto>
```

Para más información sobre la autenticación basada en 802.1x puede consultarse la sección "Port-based MAC authentication" de [2].

### 8.3 AUTENTICACIÓN BASADA EN WEB

En este modo el switch permite que los usuarios se autenticuen manualmente a través de una sencilla página web de login proporcionada por el propio conmutador.

En los puertos en los que se active la autenticación web, inicialmente el conmutador proporciona un servidor DHCP para que los clientes conectados al puerto obtengan una dirección IP con la que poder acceder a la página de autenticación.

Por defecto, el rango de direcciones a asignar corresponde a la subred 192.168.0.0/24 y el servidor web para la autenticación está en la dirección 192.168.0.1. Es posible cambiar ese rango de direcciones mediante un comando.

Una vez autenticado el usuario a través de la página mediante un nombre de usuario y clave que esté registrado en el RADIUS, el conmutador desbloqueará el puerto, moviéndolo a la VLAN que se haya configurado.

Para realizar la configuración de autenticación WEB de un usuario conectado en el puerto se introducirán los comandos siguientes:

```
aruba-sw(config)#[no] aaa port-access web-based <lista-de-puertos>
aruba-sw(config)#[no] aaa port-access web-based <lista-de-puertos> client-limit <numMACs>
aruba-sw(config)#[no] aaa port-access web-based <lista-de-puertos> unauth-vid <vlan>
aruba-sw(config)#[no] aaa port-access web-based <lista-de-puertos> auth-vid <vlan>
```

Si no se configura alguna de las VLANs (unauth-vid y auth-vid), se utilizará la asignada al puerto.

Ejemplo de configuración de autenticación web en el puerto 3:

```
aruba-sw(config)# aaa port-access web-based 3
aruba-sw(config)# aaa port-access web-based 3 client-limit 2
aruba-sw(config)# aaa port-access web-based 3 unauth-vid 205
aruba-sw(config)# aaa port-access web-based 3 auth-vid 200
```

Para cambiar el rango de direcciones asignadas por DHCP, por ejemplo al rango 10.11.12.0/24, se puede utilizar el comando:

```
aruba-sw(config)# aaa port-access web-based dhcp-addr 10.11.12.0/24
```

Adicionalmente, es posible especificar una URL a la que se redirija el navegador tras una autenticación con éxito. Para ello hay que utilizar el comando:

```
aruba-sw(config)# aaa port-access web-based <port-list> [redirect-url <url>]
```

Para más información sobre la autenticación basada en 802.1x puede consultarse la sección “Web and MAC Authentication” de [2].

## 8.4 AUTENTICACIÓN MEDIANTE PORTAL CAUTIVO EXTERNO

Adicionalmente, es posible configurar el uso de un portal cautivo externo al conmutador como método de autenticación de acceso a la red. Para ello, ArubaOS proporciona un conjunto de comandos específico (*aaa authentication captive-portal*).

Dada la complejidad de poner en marcha una solución de ese tipo, se recomienda acudir a la documentación específica sobre ese tema de Aruba (por ejemplo, [5]).

Si no se va a hacer uso del portal cautivo externo, se recomienda desactivarlo mediante:

```
aruba-sw(config)# aaa authentication captive-portal disable
```

Se puede comprobar el estado de la configuración del portal cautivo mediante:

```
aruba-sw(config)# show captive-portal
```

En [5] puede encontrarse más información sobre como configurar un portal cautivo con ClearPass.

## 8.5 POLÍTICAS DE ACCESO BASADAS EN ROLES

En los equipos ArubaOS-Switch es posible, además de regular el acceso a la red mediante la consulta de credenciales a un servidor RADIUS, el asignarles una política de acceso a la red. Con ello se consigue administrar las conexiones del equipo de una manera más segura y eficiente: cuando un usuario se autentica y se conecta correctamente a la red se le asigna automáticamente un perfil de acceso que le permitirá realizar las acciones que el administrador del equipo haya definido.

Este método de definición de usuarios, denominado definición de roles o perfiles, puede resultar interesante ya que permite aislar partes de la red a personas no autorizadas, creando para ello roles de invitado, de empleado, de administradores, etc.

El despliegue de una política basada en roles requiere una cierta planificación de los diversos tipos de usuarios y permisos que existirán. Los pasos a seguir para configurar esta funcionalidad son los siguientes:

- Activar el protocolo 802.1X en los puertos deseados según el procedimiento del apartado 8.1.
- Definición de los roles de usuario en el conmutador.
- Configuración del atributo Role en RADIUS.
- Activación de la funcionalidad

Para conocer más en detalle los parámetros de configuración de políticas de roles, así como la configuración de listas de acceso se recomienda mirar el manual de configuración correspondiente de Aruba-OS.

### 8.5.1 DEFINICIÓN DE ROLES DE USUARIO

Se realiza mediante los siguientes pasos:

1. Definir una lista de acceso para el rol deseado.

Se identifican a través de un literal o cadena de texto entrecomillado. Existen las siguientes configuraciones en la definición de una clase de servicio:

- match – Crea una regla para clasificar el tráfico que se define.
- Ignore – Crea una regla para ignorar el tráfico que se define
- Remark – Añade un comentario a la definición, para documentación.

```
class ipv4/ipv6/mac/resequence "<nombre_ACL>"
    <numero_secuencia> match/ignore/remark ip/tcp/ospf.../ <Direccion_origen> <máscara>
    <direccion_destino> < mascara>
```

Ejemplo: lista de acceso que permite cualquier tráfico

```
class ipv4 "IP-ANY-ANY"
    10 match ip any any
```

Se pueden ver las clases configuradas mediante el comando:

```
aruba-sw# show class config
```

2. Crear una política de tráfico que asigne la ACL creada con anterioridad a un usuario determinado.

Estas políticas pueden incluir acciones que se deben llevar a cabo tales como:

- Class – Asocia una clase de servicio definida, con una acción de QoS
- Remark - Añade un comentario a la definición, para documentación

```
policy user "<nombre_politica>"
  <num_secuencia> <class|remark> <ipv4|ipv6> <nombre_ACL> action <deny|dscp|ip-
precedence|permit|priority|rate-limit|redirect>
exit
```

Ejemplo:

```
policy user "empleado"
  10 class ipv4 "IP-ANY-ANY" action dscp default action permit
Exit
```

Se pueden ver las políticas configuradas mediante el comando:

```
aruba-sw# show policy config
```

3. Crear el rol de usuario asignando la política de acceso creada con anterioridad.

En este paso al rol de usuario se le asigna la política creada, un periodo para volver a realizar la autenticación y además la VLAN que se le asignará al usuario si la autenticación se lleva a cabo con éxito.

```
aaa authorization user-role name "<nombre_rol>"
  policy "<nombre_politica>"
  reauth-period <tiempoSegundos>
  vlan-name "<nombre_VLAN>"
exit
```

Ejemplo:

```
aaa authorization user-role name "empleado"
  policy "empleado"
  reauth-period 28800
  vlan-name "Servicio"
exit
```

Se pueden ver los roles de usuario configurados mediante el comando:

```
aruba-sw# show user-role
aruba-sw# show user-role empleado
```

## 8.5.2 CONFIGURACIÓN DEL ATRIBUTO ROLE EN RADIUS

Para poder utilizar los roles de usuario es necesario disponer de un servidor RADIUS que realice la autenticación de los usuarios.

Además, para poder realizar la asociación entre un usuario y el rol correspondiente en el conmutador es necesario definir el atributo *Aruba-user-role* en el servidor, el cual especificará el rol en concreto asignado al usuario.

Como ejemplo se puede utilizar un servidor freeRADIUS en el que se definirá esta configuración en el fichero *users*:

```
Bob cleartext-password:="bob", Aruba-User-Role:="Employee"
```

Existen más ajustes disponibles como *Aruba-User-Vlan*, *Aruba-Admin-Role*, etc. especificados en el fichero de diccionario del dispositivo disponible en la web de soporte de Aruba.

### 8.5.3 ACTIVACIÓN DE LA FUNCIONALIDAD

Para que la funcionalidad pueda ser usada es necesario activarla mediante el comando:

```
aruba-sw# aaa authorization user-role enable
```

Este comando activa el rol creado por defecto en el conmutador de fábrica, por lo que si se desea usar el nuevo rol configurado bastará con realizar la siguiente acción:

```
aruba-sw# aaa authorization user-role initial-role "<nombre-rol>"
```

Con esto se habrá cambiado el rol de inicio y podremos hacer uso de la configuración de roles. Con ella, cuando el usuario se autentique en RADIUS, se obtendrá el role que tiene asignado y se le aplicarán las políticas definidas en el perfil que le corresponda (calidad de servicio, lista de acceso, vlan, etc.).

## 9. PROTECCIÓN FRENTE ATAQUES

Existen en la actualidad multitud de ataques posibles encaminados a comprometer la seguridad de nuestros equipos de red, con el objetivo de hacerse con el control de su gestión, obtener copias de la información que viaja por la red o simplemente paralizar el servicio que ofrecen.

Muchos de estos ataques clásicos a los conmutadores están documentados y los fabricantes ofrecen ya medidas efectivas que pueden mitigarlos, impidiendo que ocurran. Incluso para los ataques desconocidos (zero day), existen medidas que pueden ayudar a detectarlos de forma temprana y poder tomar medidas para paliarlos.

A continuación, se detallan algunos de estos ataques y las medidas que se pueden tomar para securizar un conmutador ArubaOS frente a ellos.

### 9.1 DHCP SNOOPING

El ataque conocido como *DHCP Spoofing* se aprovecha de la simplicidad del protocolo DHCP y de la falta de mecanismos estándar que permitan asegurar la autoconfiguración de los sistemas finales mediante DHCP.

El ataque se basa en el despliegue de servidores DHCP maliciosos que asignan parámetros de configuración (direcciones IP, dirección del router, etc.) falsos a los sistemas finales, con varios objetivos posibles:

- Comprometer la disponibilidad de la red (ataque *DoS*), asignando, por ejemplo, direcciones IP incorrectas a los equipos para que pierdan la conectividad con el resto de la red.
- Redirigir el tráfico de los sistemas finales para que atraviesen otros sistemas maliciosos (ataque *man-in-the-middle*) para interceptar los mensajes enviados y poder realizar acciones como el robo de datos y credenciales.

El principal problema que presenta DHCP se debe a la incapacidad de distinguir los servidores legítimos de los que no lo son, que permite el fácil despliegue de servidores DHCP falsos en los sistemas de usuarios conectados a la red.

Para evitar ataques del tipo DHCP spoofing, ArubaOS incluye la funcionalidad *DHCP Snooping*, que permite informar al conmutador de cuales son los puertos y direcciones de los servidores DHCP legítimos, con el objetivo de filtrar todos los mensajes DHCP procedentes de servidores no autorizados.

Para activar la funcionalidad de DHCP snooping globalmente se debe ejecutar:

```
aruba-sw(config)# dhcp-snooping
```

Posteriormente, para activar DHCP snooping en una determinada VLAN:

```
aruba-sw(config)# dhcp-snooping vlan <vid>
```

Una vez activado en una VLAN, todos los puertos participantes pasan a ser no confiables y cualquier mensaje DHCP de los enviados por los servidores será filtrado.

Para declarar como confiable uno o varios puertos del conmutador donde se encuentran conectados los servidores DHCP:

```
aruba-sw(config)# dhcp-snooping trust <lista-de-puertos>
```

Una vez declarado un puerto como confiable, se aceptarán los mensajes DHCP de servidor procedentes de él, independientemente de la dirección IP de la que procedan.

Para mejorar todavía más la seguridad, se pueden definir las direcciones IP de los servidores confiables ejecutando el comando siguiente para cada servidor:

```
aruba-sw(config)# dhcp-snooping authorized-server <dir-IP-servidor-DHCP>
```

Una vez ejecutado el comando anterior, el conmutador solo aceptará mensajes de los servidores registrados.

Por ejemplo, los comandos a ejecutar para activar DHCP snooping en la VLAN 100, estando el servidor de DHCP conectado al puerto 7 y con dirección 10.1.0.11 serían los siguientes:

```
aruba-sw(config)# dhcp-snooping vlan 100
aruba-sw(config)# dhcp-snooping trust 7
aruba-sw(config)# dhcp-snooping authorized-server 10.1.0.11
```

Para ver la configuración de DHCP snooping se puede usar el comando:

```
aruba-sw# show dhcp-snooping

DHCP Snooping Information

DHCP Snooping           : Yes
Enabled VLANs           : 100
Verify MAC address      : Yes
Option 82 untrusted policy : drop
Option 82 insertion     : Yes
Option 82 remote-id     : mac
Store lease database     : Not configured

Authorized Servers
-----
10.1.0.11

Port Trust Max Current Bindings
-----
1 Yes - - -
Static Dynamic
-----

Ports 2,4-10 are untrusted
```

Cuando se activa DHCP snooping, el conmutador guarda información sobre las direcciones asignadas a cada sistema. Esta información se puede consultar mediante:

```
aruba-sw# show dhcp-snooping binding
```

Asimismo, se pueden obtener estadísticas interesantes sobre los mensaje DHCP recibidos con el comando:

```
aruba-sw# show dhcp-snooping stats
```

Finalmente mencionar que, aunque todos los comandos anteriores aplican al protocolo IPv4, existen los mismos comandos para asegurar servidores de DHCP para IPv6 (comandos *dhcpx6-snooping*). Para más información sobre DHCP Snooping en ArubaOS puede consultarse [2].

## 9.2 ARP SNOOPING

El protocolo de resolución de direcciones ARP (Address Resolution Protocol) es un sencillo protocolo que permite averiguar dinámicamente la dirección MAC que corresponde a una determinada dirección IP en una LAN. Se basa en el envío de mensajes de solicitud (ARP Request), normalmente a la dirección de difusión de la LAN, que son contestados por los sistemas aludidos mediante respuestas (ARP Reply). La información obtenida mediante ARP se almacena en las tablas ARP que todo sistema IP mantiene.

Al igual que DHCP, la sencillez del protocolo lo hace vulnerable a los ataques de falsificación de mensajes ARP o *ARP Spoofing*. El ataque más típico consiste en enviar mensajes ARP falsificados a una LAN para conseguir atraer el tráfico de otros sistemas hacia un sistema malicioso que posteriormente los redirige hacia su destino original. De esta forma se consigue realizar un ataque del tipo man-in-the-middle.

El ataque se realiza “envenenando” las tablas de ARP de los sistemas de la red, mediante el envío de paquetes ARP Reply en los que se asocian las direcciones IP por las que preguntan otros sistemas con la dirección MAC del sistema malicioso. De esta forma los sistemas “envenenados” enviarán el tráfico hacia el sistema malicioso.

ArubaOS proporciona una funcionalidad para proteger el funcionamiento del protocolo ARP denominada *ARP protect*, que funciona de manera combinada con la funcionalidad DHCP snooping.

A grandes rasgos, el funcionamiento de esta protección es el siguiente:

- El conmutador guarda en una tabla interna las asociaciones entre direcciones IP y MAC, que consigue escuchando los mensajes DHCP.
- Cuando recibe un mensaje de respuesta ARP (ARP Reply), compara su contenido con la tabla interna de asociaciones IP-MAC: si coincide con alguna entrada en la tabla, deja pasar el mensaje hacia su destino; si no, descarta el mensaje.
- Es posible declarar puertos fiables, de manera que el conmutador confíe en las respuestas ARP recibidas por esos puertos, independientemente de que aparezcan en la tabla de asociaciones.

- También es posible añadir entradas estáticas a la tabla de asociaciones IP-MAC.

En resumen, con esta protección activada el conmutador confiará y distribuirá las respuestas ARP recibidas por los puertos declarados como confiables y las recibidas por los puertos no confiables que aparezcan en la tabla de asociaciones IP-MAC, ya sean procedentes de una autoconfiguración DHCP anterior o de una entrada configurada estáticamente. El resto de respuesta ARP se descartarán.

Para activar globalmente el mecanismo de protección frente ataques de ARP es necesario ejecutar:

```
aruba-sw(config)# arp-protect
```

Posteriormente, para que el mecanismo esté operativo es necesario activarlo en las VLAN que se desee:

```
aruba-sw(config)# arp-protect vlan [rango-vlans]
```

Nota: dado que este mecanismo de protección de ARP funciona de forma combinada con la funcionalidad de DHCP snooping, ésta debe estar activada en las VLANs en las que se active la protección de ARP.

Una vez activado el mecanismo, todos los puertos participantes en dicha VLAN se declaran como puertos no confiables, tal como se puede apreciar con el siguiente comando:

```
aruba-sw(config)# sh arp-protect

ARP Protection Information

ARP Protection Enabled : Yes
Protected Vlans   : 100
Validate         :

Port  Trust
-----
Ports 1-10 are untrusted
```

Esto significa que todas las respuestas ARP que se envíen desde esos puertos serán comprobadas por el conmutador, comparándolas con la tabla de asociaciones IP-MAC (visible con *show dhcp-snooping binding*). En caso de que la respuesta no coincida con una entrada de esta tabla, se descartará.

Es posible declarar qué puertos son confiables mediante el comando:

```
aruba-sw(config)# arp-protect trust <lista-de-puertos>
```

Ello provocará que el conmutador confíe en las respuestas ARP generadas por los sistemas conectados a dichos puertos.

Para configurar asignaciones estáticas de IP-MAC se utilizará el comando:

```
aruba-sw(config)# ip source-binding <vlan-id> <dir-ip> <dir-mac> <puerto>
```

De esta forma, las respuestas ARP que contengan esa asociación entre dirección IP y MAC se aceptarán aunque procedan de puertos no confiables.

Adicionalmente se pueden forzar validaciones adicionales sobre la corrección de las respuestas ARP mediante el comando:

```
aruba-sw(config)# arp-protect validate <[src-mac]|[dest-mac]|[ip]>
```

Este comando hará que se compruebe que:

- *src-mac*: que la dirección MAC origen de la trama Ethernet coincide con el campo “Sender MAC” del mensaje ARP.
- *dst-mac*: que la dirección MAC destino de la trama Ethernet coincide con el campo “Target MAC” del mensaje ARP.
- *ip*: que la dirección IP del campo “sender IP” del mensaje no es falsa.

Por ejemplo, el siguiente comando activará las tres comprobaciones anteriores:

```
aruba-sw(config)# arp-protect validate src-mac dest-mac ip
```

Es posible ver estadísticas sobre el funcionamiento de ARP protect con:

```
aruba-sw# show arp-protect statistics
```

Existe una funcionalidad adicional denominada *IP Lockdown*, que permite prevenir el envío de paquetes IP con la dirección IP origen falseada. Cuando se activa esta funcionalidad en un puerto, el conmutador solo reenviará los paquetes IP recibidos si la asociación entre dirección IP y MAC origen aparece en la tabla de asociaciones IP-MAC.

Para activar IP lockdown en uno o varios enlaces para IPv4 o IPv6:

```
aruba-sw(config)# ip source-lockdown <lista-de-puertos>
```

Para más información sobre ARP Protect en ArubaOS puede consultarse la sección “Configuring Advanced Threat Protection” de [2].

### 9.3 INUNDACIÓN MAC

Todo conmutador Ethernet mantiene una Tabla de Filtrado que utilizan para realizar la función básica de encaminamiento de las tramas que recibe. Esta tabla almacena la asociación entre las direcciones MAC de los equipos conectados a sus puertos y los identificadores de esos puertos.

Cuando el conmutador recibe una trama por alguno de sus puertos, extrae la dirección destino y la compara con las entradas de su tabla de filtrado (almacenada en una CAM o Content-Addressable Memory). Si la dirección de destino se encuentra en la tabla, reenvía el paquete a través del puerto asociado en la tabla CAM con esa dirección. En caso contrario, si la dirección no está registrada en la tabla, se envía la trama por todos los puertos (difusión).

Las entradas en la tabla de filtrado se autoconfiguran automáticamente mediante el algoritmo de aprendizaje hacia atrás: cada vez que llega una trama se aprende de su dirección MAC origen, registrando en la tabla CAM dicha dirección asociada al puerto por el que llegó la trama.

Las tabla CAM tienen una capacidad limitada, que típicamente se especifica en la hoja de datos de cada equipo. Cuando esta tabla se llena, el conmutador deja de aprender direcciones y, por tanto, para todas aquellas direcciones destino que no aparezcan en su tabla se verá obligado a hacer difusión de las tramas a través de todos los puertos de la VLAN.

Esta situación puede darse en casos en que el número de usuarios conectados a la red supere la capacidad de los conmutadores. Aunque es más frecuente que la causa sean los ataques de Denegación de Servicio (DoS) que tratan de explotar esta vulnerabilidad.

Típicamente, estos ataques consisten en enviar un número muy grande de tramas Ethernet con direcciones MAC origen aleatorias, de manera que se llenen las tablas CAM del equipo. Este hecho provocará que el conmutador realice difusión de una parte del tráfico que conmuta, provocando un aumento de la carga de la red o incluso su colapso. Además, provocará un problema añadido de confidencialidad de la información, ya que tramas que deberían enviarse por un puerto concreto se están distribuyendo a todos los puertos de la VLAN.

Uno de los mecanismos de prevención consiste en utilizar las medidas ya citadas en la sección 7.2 para limitar el número de direcciones MAC que el conmutador puede aprender en cada puerto.

Esta limitación se debe aplicar a los puertos de acceso que conectan equipos de usuario, para evitar que un posible atacante inunde la tabla de filtrado. En los puertos troncales no se suele incluir esta limitación. O si se incluye, se debe tener en cuenta que en esos puertos el número de direcciones asociadas puede ser grande.

Para configurar esta funcionalidad, se deben ejecutar los siguientes comandos:

```
aruba-sw(config)# port-security <lista-de-puertos> learn-mode static
aruba-sw(config)# port-security <lista-de-puertos> address-limit <número-máximo-de-entradas>
```

Se puede consultar la tabla de filtrado del conmutador (direcciones MAC asociadas a cada puerto) mediante los comandos siguientes:

```
aruba-sw(config)# show mac-address detail
aruba-sw(config)# show mac-address <lista-de-puertos>
aruba-sw(config)# show mac-address vlan <vlan-id>
```

## 9.4 VIRUS THROTTLING

Con el objeto de detectar y mitigar los ataques que hacen uso de técnicas intensivas de escaneo de puertos, los equipos ArubaOS incluyen una funcionalidad para controlar

el número de conexiones por segundo que se establecen a través de los puertos de un conmutador.

Esta funcionalidad del conmutador, denominada *connection-rate-filter*, permite configurar cada puerto de manera que, cuando se detecte una actividad sospechosa en términos de un número anormal de solicitudes de conexión procedentes de un determinado host, reaccione de las siguientes formas posibles:

- **Notify-only:** únicamente se genera una notificación en el log y, opcionalmente, se manda un trap si SNMP está activado.
- **Throttle:** se bloquea el puerto durante un periodo de tiempo configurable, además de generarse la correspondiente notificación.
- **Block:** se genera la notificación y el puerto se bloquea indefinidamente. El administrador debe posteriormente rehabilitar el puerto manualmente.

Además, es posible configurar la sensibilidad de la detección de la situación anómala que dispara las acciones anteriores. Existen cuatro niveles definidos con los umbrales de disparo del mecanismo que se mencionan a continuación:

- **Low:** se permiten solicitudes de hasta 54 destinos distintos en menos de 0.1 segundos y se establece la penalización de 30 segundos si se supera cuando el puerto está en modo Throttle.
- **Medium:** se permiten hasta 37 destinos distintos en menos de 1 segundo y se establece una penalización de 30 a 60 segundos en modo Throttle.
- **High:** se permiten hasta 22 destinos en menos de 1 segundo y se establece una penalización de 60 a 90 segundos en modo Throttle.
- **Aggressive:** se permiten hasta 15 destinos en menos de 1 segundo y se establece una penalización de 90 a 120 segundos en modo Throttle.

La funcionalidad *connection-rate-filter* se debe activar de forma global estableciendo la sensibilidad de detección y, posteriormente, se debe activar en los puertos concretos que se quieran proteger, definiendo la forma de reacción.

Se recomienda activar esta funcionalidad en los puertos de acceso al conmutador y no tanto en los enlaces troncales, en los que puede ser normal un número alto de solicitudes de conexión.

Para configurar Virus Throttling o *connection-rate-filter* de forma global se debe ejecutar el comando:

```
aruba-sw(config)# connection-rate-filter sensitivity <low|medium|high|aggressive>
```

Para aplicar la funcionalidad a un conjunto de puertos se debe utilizar el comando:

```
aruba-sw(config)#filter connection-rate <port-list> {notify-only|throttle|block}
```

El comando anterior puede ejecutarse varias veces sobre puertos distintos, para poder particularizar la forma de reacción de cada puerto concreto.

Para consultar la configuración de esta funcionalidad se puede utilizar el comando:

```
aruba-sw# show connection-rate-filter
```

```
Connection Rate Filter Configuration
```

```
Global Status:    Enabled
Sensitivity:      Low
```

Port	Filter Mode
3	NOTIFY-ONLY
5	THROTTLE
7	BLOCK

Y los host bloqueados en cada momento se pueden consultar con:

```
aruba-sw# show connection-rate-filter <all-hosts | blocked-hosts | throttled-hosts>
```

En caso de bloqueo de un enlace, éste se puede desbloquear mediante el comando:

```
aruba-sw(config)# connection-rate-filter unblock <all | host <dir-ip> | <prefijo/longitud>>
```

Adicionalmente, se puede mejorar la funcionalidad mediante el uso de ACLs específicas de esta funcionalidad, que permiten afinar los criterios de computo de las tasas que disparan la detección. Para conocer más detalles, recomendamos acceder a la sección “Virus Throttling” dela guía [2].

## 9.5 MEDIDAS DE PROTECCIÓN DE ENLACES TRONCALES

### 9.5.1 MACSEC

Media Access Control Security (MACsec) es un estándar del IEEE (802.1AE) que describe como securizar un enlace entre dos equipos a nivel de la capa de enlace. Proporciona confidencialidad e integridad en la transmisión de información entre sistemas finales y conmutadores en enlaces de acceso o entre conmutadores en enlaces troncales, mediante el cifrado de la información transmitida utilizando algoritmos de cifrado simétrico muy eficientes que mantienen el rendimiento y prestaciones de los puertos implicados.

MACsec está pensado para entornos cableados; los entornos WLAN tienen su propia solución de cifrado. Es un protocolo con demanda en auge para garantizar la confidencialidad de la comunicaciones cableadas. MACsec actualmente se soporta en las familias de conmutadores Aruba 5400R, 3810M y 2930M.

El protocolo MACsec proporciona:

- Integridad de datos. Cada trama MAC transporta un campo separado de verificación de integridad.
- Autenticidad del origen de la trama. Se garantiza que cada trama MAC ha sido enviada por una estación autorizada MACSEC.
- Confidencialidad. Cada trama es cifrada.
- Protección contra la redifusión. Las tramas capturadas, no pueden ser insertadas de nuevo al medio sin ser detectadas.
- Se mejora la seguridad en las comunicaciones switch-to-switch con el protocolo MACsec Key Agreement (MKA) y el modo Static Connectivity Association Key (CAK).

Los pasos para la activación de la funcionalidad MACSEC son: creación y configuración de una política MACsec; aplicación de una política MACSEC a los puertos deseados; y configuración de los parámetros MKA en los puertos.

Para crear una política MACsec se debe utilizar los comandos siguientes:

```
aruba-sw(config)# macsec policy <policy-name>
aruba-sw(Policy-<policy-name>)# mode pre-shared-key ckn <CKN> [cak|encrypted-cak] <CAK-value>
aruba-sw(Policy-<policy-name>)# [no] confidentiality
aruba-sw(Policy-<policy-name>)# [no] replay-protection <replaywindowsize>
aruba-sw(Policy-<policy-name>)# [no] include-sci-tag
```

Para configurar la clave puede optarse por hacerlo en texto plano o ya cifrada. Para introducirla cifrada debe haberse obtenido desde un dispositivo compatible.

Se recomienda el uso de el comando *confidentiality* ya que activa el cifrado de la trama MAC. Si se desactiva no se cifra pero si se valida la integridad de los datos.

El comando *replay-protection* verifica el campo IP number, y asegura que el paquete no haya llegado retrasado un número de paquetes mayor que el valor *<replaywindowsize>*. En tal caso descarta la trama. Con el valor 0 aseguramos que todas las tramas lleguen en orden estricto, evitando, reinserciones de tramas.

El comando *include-sci-tag* incluye la etiqueta Secure Channel Identifier (SCI) en el campo Secure Tag.

Para aplicar una política ya creada a un puerto concreto:

```
aruba-sw(config)# macsec apply policy <policy-name> ethernet <lista-de-puertos>
```

Se puede consultar información sobre la configuración de MACsec en un conmutador mediante el comando:

```
aruba-sw# show macsec policy
```

Para obtener más información sobre la configuración de MACsec en ArubaOS se puede consultar el capítulo “Infrastructure MACsec” de [AccSecGuide].

## 9.5.2 PROTECCIÓN DEL PROTOCOLO DLDP

El protocolo **DLDP (Device Link Detection Protocol)** se utiliza para monitorizar el estado de los enlaces y detectar fallos de conectividad que se producen en un único sentido de transmisión.

Cuando se detecta un fallo de este tipo, el equipo reacciona desconectando la interfaz automáticamente o mostrando un aviso al usuario para que pueda desactivarla manualmente. Esto puede ser utilizado por un atacante para averiguar información sobre los equipos conectados en la red o para provocar artificialmente caídas en los enlaces. Por ello, si no es estrictamente necesario utilizar el protocolo, se recomienda desactivar su uso de manera global mediante:

```
aruba-sw(config)# dldp disable
```

Alternativamente, en caso de que sea necesario utilizar DLDP, se deberá activar el protocolo solo en los puertos donde se requiera su uso y configurar un modo de autenticación para evitar la introducción de mensajes maliciosos mediante:

```
aruba-sw(config)# dldp authentication-mode md5
aruba-sw(config)# dldp authentication-password simple <contraseña>
aruba-sw(config)# interface <puerto>
aruba-sw(eth-puerto)# dldp enable
```

## 10. RESUMEN DE RECOMENDACIONES

Se resumen a continuación las principales recomendaciones de seguridad presentadas en esta guía para la configuración de un equipo desde cero.

### 10.1 PROCEDIMIENTO DE CONFIGURACION INICIAL DE UN EQUIPO

- Arrancar el equipo desconectado de la red y acceder a la gestión a través de la consola. Si el equipo tiene una configuración anterior, se recomienda borrar la configuración mediante el procedimiento de reseteo a fabrica o con el comando *“erase startup-config”*.
- Configurar nombre del equipo, asignar nombre de usuario y contraseña a usuarios locales de nivel operador y manager, usar sha256 para almacenar claves, configurar timer de inactividad en consola, el mensaje del día y deshabilitar usb:

```
hostname aruba-sw
password operator user-name operator plaintext <clave-segura-operador>
password manager user-name admin <clave-segura-manager>
password non-plaintext-sha256
console idle-timeout 180
banner motd %
Este es un sistema privado. Abandone la conexión si no tiene autorización.%
no usb-port
```

- Decidir si se quiere almacenar las claves en la configuración o no. Si se decide incluirlas, forzar el cifrado de las claves en la configuración:

```
include-credentials
encrypt-credentials
```

- Configurar la gestión de la calidad de las claves (si las claves configuradas anteriormente no cumplen los criterios deberán ser actualizadas):

```
password minimum-length 12
password composition lowercase 2
password composition uppercase 2
password composition specialcharacters 2
password composition number 2
password configuration history
password configuration history-record 3
password configuration aging-period 90
password configuration update-interval-time 0
password configuration-control
```

- Deshabilitar servicios no deseados:

```
no telnet-server
no web-management plaintext
no rest-interface
no snmp-server community public
no snmp-server enable
no cdp run
no lldp run
no tftp server
no tftp client
```

- Limitar mensajes ICMP mediante:

```
no ip icmp unreachable
no ip icmp redirects
no ip icmp addrmask
no ip icmp echo broadcast-request
ip icmp reply-limit
ip icmp burst-normal 5000 burst-max 10000 lockup 300
```

- Crear una VLAN específica para la gestión (por ejemplo, la 100), asignar una dirección IP para la gestión remota del conmutador (p.ej.: 10.1.1.1) y restringir el acceso a la gestión al rango de direcciones deseado (p.ej.: 10.1.1.0/24):

```
vlan 100
name "Gestion"
ip address 10.1.1.1/24
exit
management-vlan 100
ip authorized-managers 10.1.1.0 255.255.255.0 access manager
ip authorized-managers 10.1.1.0 255.255.255.0 access operator
```

- Habilitar acceso y transferencia de ficheros por SSH:

```
crypto key generate ssh
ip ssh
ip ssh filetransfer
```

- Habilitar acceso a la gestión por HTTPS (no recomendado si no es estrictamente necesario). Por ejemplo, si se usa un certificado autofirmado:

```
crypto pki identity-profile switch-id-profile subject common-name aruba-sw country ES
state Madrid locality Madrid org CCN org-unit CERT
crypto pki enroll-self-signed certificate-name aruba-https valid-start 4/1/2019 valid-end
4/1/2025 usage web
web-management ssl
web-management idle-timeout 300
```

Nota: el acceso por web es incompatible con la gestión de la calidad de las claves.

- Configurar la sincronización del reloj por NTP, a ser posible contra un servidor interno autenticado:

```
ntp [unicast|broadcast]
ntp authentication key-id 1 authentication-mode <sha1> key-value <key>
ntp max-association
ntp server <NTPserverIP>
timesync ntp
ntp enable
```

- Crear una VLAN para enlaces inactivos, mover todos los interfaces a ella y deshabilitar todos los puertos que no se vayan a utilizar:

```
vlan 5000
name "Inactiva"
unatagged <todos-los-puertos>
exit
interface X disable
interface Y disable
...
```

- Actualizar el firmware del equipo si es necesario. Para ello se puede utilizar la dirección IP de gestión ya configurada. Será necesario configurar uno de los puertos en la VLAN de gestión (100) y conectarlo a una red en la que haya un sistema desde el que copiar la imagen. Por ejemplo:

```
vlan 100
untagged 1
exit
copy sftp flash usuario@10.1.1.10 K_15_10_0001.swi secondary
```

- Si la configuración de los sistemas de red se realiza mediante DHCP, habilitar DHCP snooping para protegerlo en las VLAN en que se necesite.

```
dhcp-snooping
dhcp-snooping vlan <vid>
dhcp-snooping vlan <vid>
...
dhcp-snooping trust <lista-de-puertos>
dhcp-snooping authorized-server <dir-IP-servidor-DHCP>
```

- Configurar, si se considera necesario y si se usa DHCP snooping, la protección de ARP en las VLAN en las que se necesite, declarando los puertos fiables y configurando las asociaciones IP-MAC necesarias:

```
arp-protect
arp-protect vlan [rango-vlans]
arp-protect trust <lista-de-puertos>
ip source-binding <vlan-id> <dir-ip> <dir-mac> <puerto>
```

- Aplicar las medidas de protección de puertos (sección 7.3) que se consideren necesarias (limitación número de MACs, filtrado por MAC, limitar el tráfico broadcast, etc.):

```
port-security <num-puerto> address-limit <número-entradas>
port-security <num-puerto> mac-address <dirección-mac>
interface <num-puerto> rate-limit bcast in percent 10
```

- Si se usan los protocolos GVRP o MVRP (recomendado), deshabilitar su funcionamiento en enlaces de acceso (sección 7.4).
- Aplicar la protección frente al envío de BPDUs en los puertos de acceso:

```
spanning-tree <Lista-de-puertos> bpdu-protection
```

- Configurar las listas de acceso IP y MAC que se consideren necesarias (sección 7.7 y 7.8). En particular, se recomienda tener probadas e incluso preconfiguradas listas para filtrar el tráfico procedente de una determinada dirección IP y dirección MAC, de manera que, si a consecuencia de algún incidente se necesita filtrar un determinado sistema, se pueda llevar a cabo rápidamente.
- A partir de este momento, se puede desplegar con seguridad el equipo en la red y acceder a la gestión de este de forma remota.

## 10.2 OTRAS RECOMENDACIONES ADICIONALES

- Aplicar, si se considera necesario y en los puertos de acceso, la funcionalidad Virus throttling que limita el número de conexiones por segundo (sección 9.5).
- Si el equipo va a ser gestionado por múltiples personas, conviene crear cuentas individuales para cada una de ellas. Si, además, cada una debe realizar unas tareas de gestión concretas, puede ser interesante utilizar el sistema de roles (RBAC, sección 4.3.1) para particularizar los comandos a los que cada una tiene acceso.
- Se recomienda el uso de servidores RADIUS externos para centralizar la autenticación de usuarios (sección 4.2.1.2). Se debe configurar la autenticación local como segunda opción para poder acceder al equipo en caso de que el RADIUS no esté disponible. Si es posible, se deben tener al menos dos servidores para mejorar la fiabilidad.
- Configurar SNMPv3 si se va a utilizar para gestionar el equipo desde una plataforma de gestión (sección 5.3.6).
- Configurar, si se considera necesario, algún sistema de control de acceso a los puertos de la red de los descritos en el capítulo 8. Si se utilizan, configurar el registro de accesos mediante los comandos *aaa accounting*.

## 11. REFERENCIAS

Documentación disponible en: <https://www.arubanetworks.com/documentation>

- [1] ArubaOS-Switch Basic Operation Guide.
- [2] ArubaOS-Switch Access Security Guide.
- [3] ArubaOS-Switch Advanced Traffic Management Guide.
- [4] ArubaOS-Switch REST API and JSON Schema Reference Guide.
- [5] ArubaOS-Switch Management and Configuration Guide.