Chapter 7: Network Management

CCNP SWITCH: Implementing Cisco IP Switched Networks

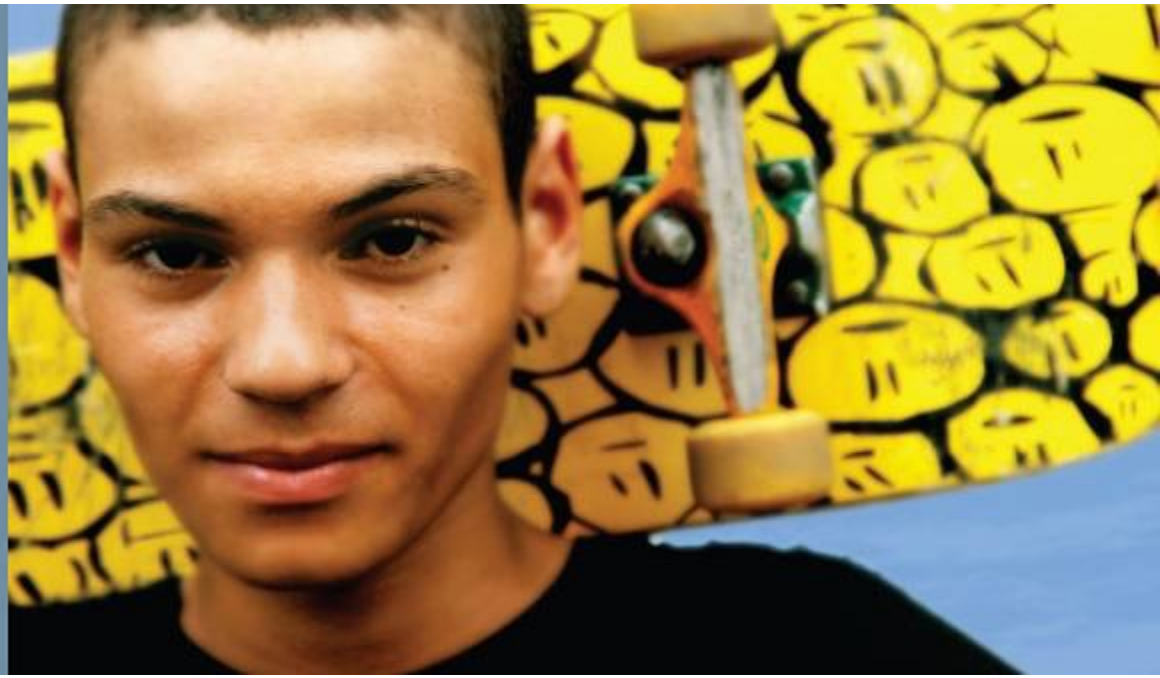
Cisco | Networking Academy®
Mind Wide Open™

Chapter 7 Objectives

This chapter covers the following topics related to network management and mobility:

- AAA
- Identity-based networking 802.1X
- NTP
- SNMP

AAA



AAA

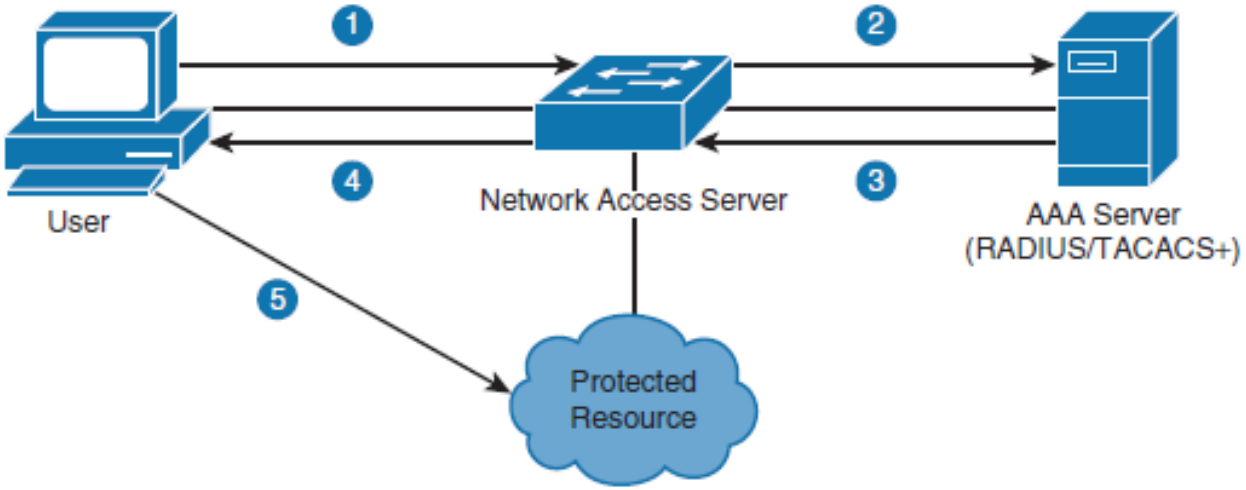
- ■ **Authentication**
  - Authentication is the process of identifying a user before that user is allowed access to a protected resource.
- ■ **Authorization**
  - After the user gains access to the network, authorization is performed.
  - Authorization allows you to control the level of access users have.
- ■ **Accounting**
  - Accounting is performed after authentication. Accounting enables you to collect information about the user activity and resource consumption.

AAA Benefits

- **Increased flexibility and control of access configuration**
  - AAA offers additional authorization flexibility on a per-command or per-interface level, which is unavailable with local credentials.
- **Scalability**
  - As the network grows, managing a large number of users on multiple devices becomes highly impractical and error-prone, with a lot of administrative burden.
- **Standardized authentication methods**
  - AAA supports the RADIUS protocol, which is an **industry open standard**. This ensures interoperability and allows flexibility because you can mix and match **different vendors**.
- **Multiple backup systems**
  - You may specify multiple servers when configuring authentication options on the method list, combining them in a server group.

RADIUS and TACACS+ Overview

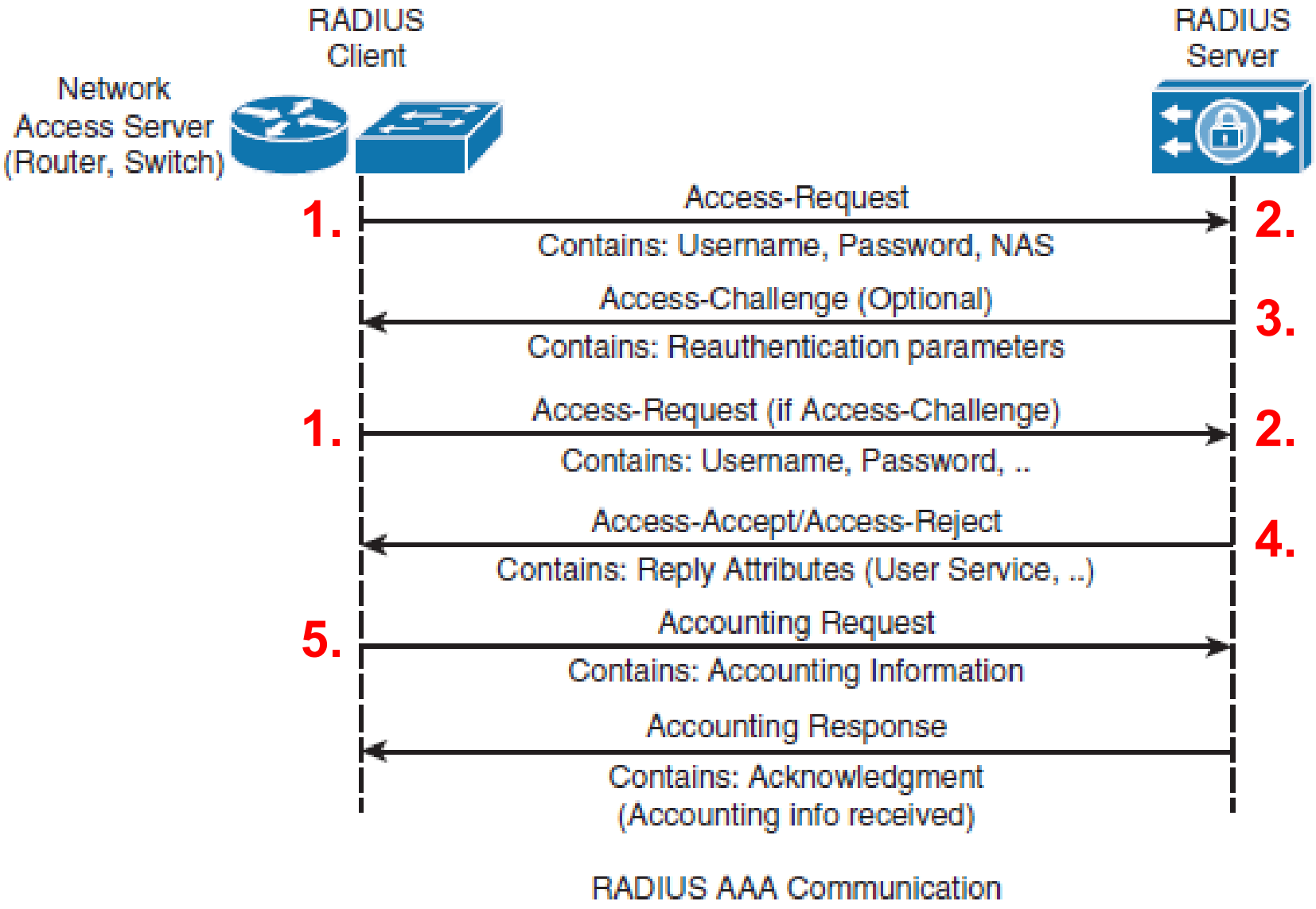
- RADIUS and TACACS+ are AAA protocols.
- Both use the client/server model.
- As shown in Step 1, a user or machine sends a request to a networking device such as a router that acts as a network access server when running AAA.
- The network access server then communicates (2, 3) with the server exchanging RADIUS or TACACS+ messages.
- If authentication is successful, the user is granted (4) an access to a protected resource (5), such as a device CLI, network, and so on.



TACACS+ Versus RADIUS

Feature	RADIUS	TACACS+
Developer	Livingston Enterprise (now industry standard)	Cisco (proprietary)
Transport protocol	UDP ports 1812 and 1813	TCP port 49
AAA support	Combines authentication and authorization and separates accounting	Uses the AAA model and separates all three services
Challenge response	One-way, unidirectional (single challenge response)	Two-way, bidirectional (multiple challenge responses)
Security	Encrypts only the password in the packet	Encrypts the entire packet body

RADIUS Authentication Process



RADIUS AAA Communication

RADIUS Authentication Process

1. **RADIUS** authentication process between the NAS and RADIUS server starts when a client sends a **login request** in the form of an **Access-Request packet**. This packet contains a **username**, **encrypted password**, **the NAS IP address**, and the **NAS port number**.

2. **When the RADIUS** server receives the query, it **first compares the shared secret key** sent in the request packet with the value configured on the server. When shared secrets are not identical, the server silently drops the packet. This ensures that only authorized clients can communicate with the server. If shared secrets are identical, the packet is further processed, **comparing the username and password inside the packet** with those found in the database.

RADIUS Authentication Process

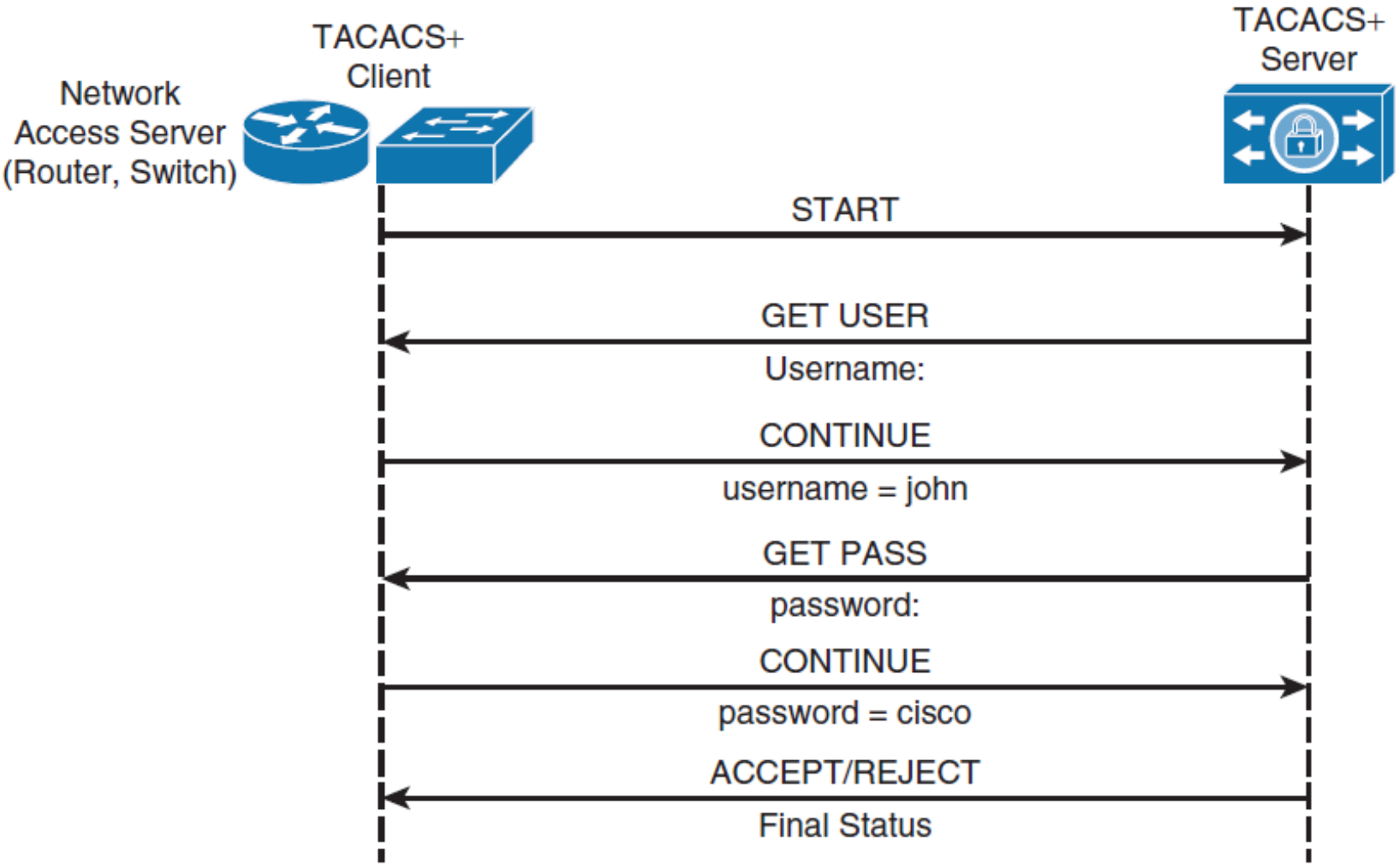
3. **During** the authentication and authorization phase, an **optional Access-Challenge** message may be requested by the RADIUS server with the purpose of collecting additional data (PIN, token card, and so on), further verifying the client's identity.

4. **If** a match is found, the server returns an **Access-Accept packet** with a list of attributes to be used with this session in the form of AV pairs (IP address, access control list [ACL] for NAS). If a match is not found, however, the RADIUS server returns an Access-Reject packet. It is important to notice that **authentication and authorization phases are combined in a single Access-Request packet**, unlike TACACS+.

RADIUS Authentication Process

5. **Moreover**, the accounting phase is realized separately after the authentication and authorization phases, using **Accounting-Request** and **Accounting-Response** messages.

TACACS+ Authentication Process



TACACS+ Authentication Communication

TACACS+ Authentication Process

1. TACACS+ communication between the NAS and the TACACS+ server **starts with a TCP communication**, unlike RADIUS (which uses UDP).
2. Next, the NAS contacts the TACACS+ server to obtain a **username prompt**, which is then displayed to the user. The username entered by the user is forwarded to the server.
3. The server prompts the user again, this time for a **password**. The password is then sent to the server, where it is validated against the database (local or remote).
4. If a match is found, the TACACS+ server sends an ACCEPT message to the client, and the authorization phase may begin (if configured on the NAS). If a match is not found, however, the server responds with a REJECT message, and any further access is denied.

Configuring AAA

- To enable AAA, the first step is to configure the **`aaa new-model`** command in global configuration mode.
- This step essentially enables AAA capability.
- In addition, until this command is enabled, all other AAA commands are hidden.
- The **`aaa new-model`** command immediately applies local authentication to all lines and interfaces (except console line con 0).
- To avoid being locked out of the router, it is a best practice to define a local username and password before starting the AAA configuration:

```
Switch(config)# username User123 secret Secretpwd
```

Configuring AAA

- **Warning**: The **aaa new-model** command immediately applies local authentication to all lines and interfaces (except console line **line con 0**).
- If a telnet session is opened to the router after this command is enabled (or if a connection times out and has to reconnect), then the user has to be authenticated with the local database of the router.
- It is recommended to define a username and password on the access server before you start the AAA configuration, so you are not locked out of the router. See the next code example:

```
Router(config)#username xxx password yyy
```

Configuring AAA

- **Tip**: Before you configure your AAA commands, save your configuration.
- You can save the configuration again only after you have completed your AAA configuration (and are satisfied that it works correctly).
- This allows you to recover from unexpected lockouts as you can roll back any change with a reload of the router.

Configuring the External Radius AAA Server

- Switch(config)# **radius server** *configuration-name*
- Switch(config-radius-server)# **address ipv4** *hostname* [**auth-port** *int*] [**acct-port** *int]*
- Switch(config-radius-server)# **key {0** *string* **| 7** *string* **|** *string* **}**

```
R1(config)#radius server RADIUS_SERVER1
R1(config-radius-server)#address ipv4 192.168.1.10
R1(config-radius-server)#key STUDY_CCNA1
```

Configuring the External Radius AAA Server

- we can use more external radius servers at the same time:

```
R1(config)#radius server RADIUS_SERVER1
R1(config-radius-server)#address ipv4 192.168.1.10
R1(config-radius-server)#key STUDY_CCNA1
R1(config)#radius server RADIUS_SERVER2
R1(config-radius-server)#address ipv4 192.168.1.11
R1(config-radius-server)#key STUDY_CCNA2
```

- in this case we can specify a subset of RADIUS servers by the **aaa group server** command.
- For example, use the **aaa group server** command to first define the members of **STUDY_CCNA:**

```
R1(config-radius-server)#aaa group server radius STUDY_CCNA
R1(config-sg-radius)#server name RADIUS_SERVER1
R1(config-sg-radius)#server name RADIUS_SERVER2
```

Authentication Configuration

- Authentication verifies users before they are allowed access to the network and network services (which are verified with authorization).
- To configure AAA authentication:
  1. First define a named list of authentication methods (in global configuration mode).
  2. Apply that list to one or more interfaces (in interface configuration mode).
- The only exception is the ***default*** method list (which is named **default**).
- The default method list is automatically applied to all interfaces except those that have a named method list explicitly defined.
- A defined method list overrides the default method list.

Authentication Configuration

- The Cisco IOS software uses the first method listed to authenticate users.
- If that method fails to respond (indicated by an ERROR), the Cisco IOS software selects the next authentication method listed in the method list.
- This process continues until there is successful communication with a listed authentication method, or all methods defined in the method list are exhausted.
- It is important to note that the Cisco IOS software attempts authentication with the next listed authentication method only when there is no response from the previous method. If authentication fails at any point in this cycle, that is, if the AAA server or local username database responses are to deny the user access (indicated by a FAIL), the authentication process stops, and no other authentication methods are attempted.

Authentication Configuration – Login auth

- You can use the **aaa authentication login** command to authenticate users who want Exec Access into the access server (tty, vty, console and aux).

Example 1: *Exec Access with Radius then Local*

- `Router(config)# aaa authentication login` **`default group radius local`**
  - the named list is the default one (**`default`**).
  - there are two authentication methods (**`group radius`** and **`local`**).

Authentication Configuration – Login auth

- So, we can have two authentication methods simultaneously.
- Command example with earlier defined radius servers group:

```
R1(config-sg-tacacs+)#aaa authentication login default group STUDY_CCNA local
```

  - ***aaa authentication login*** – enable authentification
  - ***default*** – define where the authentification will be used
  - ***group STUDY_CCNA*** – defined groupe of radius servers
  - ***local*** – means the local account

- All users are authenticated using the Radius server (the first method - *group STUDY_CCNA*).
- If the Radius server doesn't respond, then the router's local database is used (the second method - local).

Authentication Configuration – Login auth

- Because the list ***default*** in the **aaa authentication login** command is used, login authentication is automatically applied for all login connections (such as tty, vty, console and aux).
- If you use the previous example, without **local** keyword, the result is:

```
Router(config)#aaa authentication login
default group STUDY-CCNA
```

- If the AAA server does not reply to the authentication request, the authentication fails (since the router does not have an alternate method to try).
- The **group** keyword provides a way to group current server hosts. The feature allows the user to select a subset of the configured server hosts and use them for a particular service.

Authentication Configuration – Login auth

Example 2: *Console Access used with Line Password*

- Expand the configuration from Example 1 so that console login is only authenticated by the password set on line con 0.
- The list CONSOLE is defined and then applied to line con 0:

```
Router(config)#aaa authentication login CONSOLE line
```

  - the named list is CONSOLE.
  - there is only one authentication method (line).

- When a named list (in this example, CONSOLE) is created, it must be applied to a line or interface before it executes.

Authentication Configuration – Login auth

- When a named list (in this example, CONSOLE) is created, it must be applied to a line or interface before it executes.
- This is done with the login authentication <list_name> command:

```
Router(config)#line con 0
Router(config-line)#exec-timeout 0 0
Router(config-line)#password cisco
Router(config-line)#login authentication CONSOLE
```

- The CONSOLE list overrides the default method list default on line con 0.
- After this configuration on line con 0, you need to enter the password **cisco** to get console access.
- The default list is still used on tty, vty, and aux.

Authentication Configuration – Login auth

- To have console access authenticated by a **local username** and **password**, use the next code example:

```
Router(config)#aaa authentication login CONSOLE
local
```

- In this case, a username and password have to be configured in the local database of the router. The list must also be applied to the line or interface.
- To have no authentication, use the next code example:

```
Router(config)#aaa authentication login CONSOLE
none
```

- In this case, there is no authentication to get to the console access. The list must also be applied to the line or interface.

Authentication Configuration – Login auth

Example 3: *Enable Mode Access used with External AAA Server*

- You can issue authentication to get to enable mode (privilege 15):

```
Router(config)#aaa authentication enable default
group radius enable
```

- Only the password can be requested, the username is $enab15$.
- Hence the username $enab15$ must be defined on the AAA server.
- If the Radius server does not reply, the enable password configured locally on the router can have to be entered.

Authentication Configuration - example

- Switch(config)# **aaa authentication login radius_list group Mygroup2 local**
- Switch(config)# **line vty 0**
- Switch(config-line)# **login authentication radius_list**

```
Switch(config)# radius server myRadius
Switch(config-radius-server)# address ipv4 172.16.1.1
Switch(config-radius-server)# key cisco456
Switch(config)# aaa group server radius Mygroup2
Switch(config-sg-radius)# server name myRadius
```

```
Switch(config)# username User123 secret Secretpwd
```

Configuring TACACS+ for Console and vty Access

- Switch(config)# **tacacs server** *configuration-name*
- Switch(config-server-tacacs)# **address ipv4** *hostname*
- Switch(config-server-tacacs)# **port** *integer*
- Switch(config-server-tacacs)# **key** *string*
- Switch(config)# **aaa group server tacacs+** *group-name*
- Switch(config-sg-tacacs+)# **server name** *configuration-name*

```
Switch(config)# tacacs server myTacacs
Switch(config-server-tacacs)# address ipv4 192.168.1.1
Switch(config-server-tacacs)# key cisco123
Switch(config)# aaa group server tacacs+ Mygroup1
Switch(config-sg-tacacs+)# server name myTacacs
```

```
Switch(config)# aaa authentication login default group Mygroup1 local
Switch(config)# aaa authorization exec default group Mygroup1 local
```

AAA Authorization

- Authorization is the process by which you can control what a user can do.
- AAA authorization has the same rules as authentication:
  1. First define a named list of authorization methods.
  2. Apply that list to one or more interfaces (except for the default method list).
- The first listed method is used.
- If it fails to respond, the second one is used, and so on.
- Method lists are specific to the authorization type requested.
- This document focuses on the Exec and Network authorization types.

AAA Authorization

To configure authorization, complete the following steps:

- **Step 1.** Define a named list of authorization methods.
- **Step 2.** Apply that list to one or more interfaces (except for the default method list).
- **Step 3.** The first listed method is used. If it fails to respond, the second one is used, and so on until all listed methods are exhausted. Once the method list is exhausted, a failure message is logged.

- `Switch(config)# aaa authorization authorization-type list-name method-list`
- `Switch(config)# line line-type line-number`
- `Switch(config)# authorization { arap | commands level | exec | reverse-access } list-name`

AAA Authorization

Exec Authorization

- The aaa authorization exec command determines if the user is allowed to run an EXEC shell. This facility can return user profile information such as auto command information, idle timeout, session timeout, access-list and privilege and other per-user factors.
- Exec authorization is only carried out over vty and tty lines.

Example 1: *Same Exec Authentication Methods for All Users*

```
Router(config)#aaa authentication login default
group radius local
```

- All users who want to log in to the access server have to be authorized with Radius (first method) or local database (second method).

```
Router(config)#aaa authorization exec default
group radius local
```

AAA Accounting

- The AAA accounting feature enables you to track the services that users access and the amount of network resources that they consume.
- AAA accounting has the same rules as authentication and authorization:
  1. You must first define a named list of accounting methods.
  2. Apply that list to one or more interfaces (except for the default method list).
- The first listed method is used, if it fails to respond, the second one is used and so on.
- Network accounting provides information for all PPP, Slip and AppleTalk Remote Access Protocol (ARAP) sessions: packet count, octects count, session time, start and stop time.
- Exec accounting provides information about user EXEC terminal sessions (a telnet session for instance) of the network access server: session time, start and stop time.

AAA Accounting

AAA accounting has the same rules and configuration steps as authentication and authorization:

- **Step 1.** You must first define a named list of accounting methods.
- **Step 2.** Apply that list to one or more interfaces (except for the default method list).
- **Step 3.** The first listed method is used; if it fails to respond, the second one is used, and so on.

- `Switch(config)# aaa accounting accounting-type list-name { start-stop | stop-only | none } method-list`
- `Switch(config)# interface interface-type interface-number`
- `Switch(config-if)# ppp accounting list-name`

Limitations of TACACS+ and RADIUS

RADIUS may not be the optimal choice in the following situations:

- **Device-to-device situations**
  - RADIUS does not offer two-way authentication.
- **Networks using multiple service**
  - RADIUS generally binds a user to a single service model (character and PPP mode).

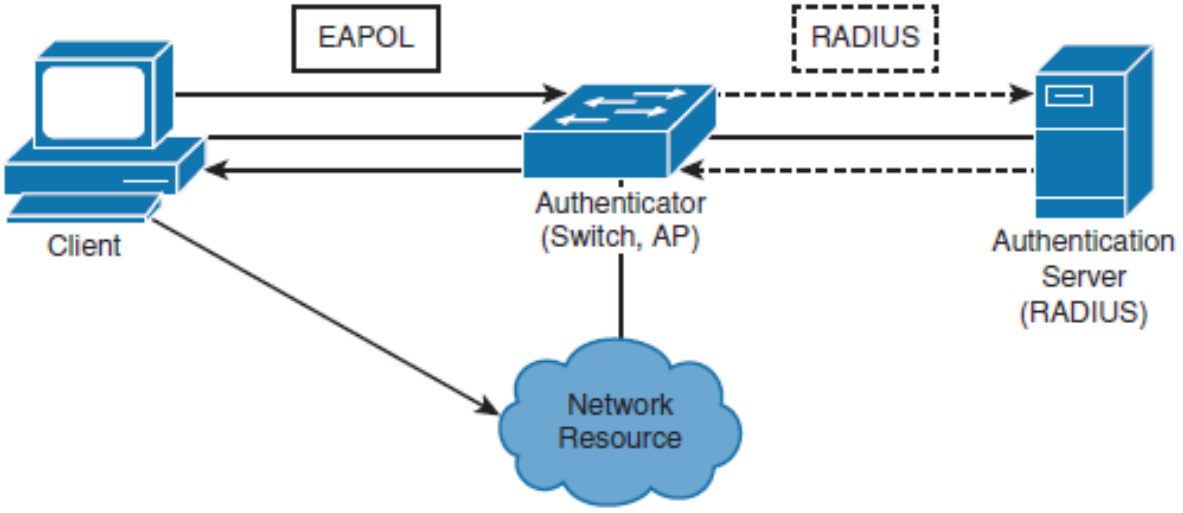
TACACS+ may not be the optimal choice in the following situations:

- **Multivendor environment**
  - TACACS+ is a Cisco proprietary protocol
- **When speed of response from the AAA services is of concern**
  - TACACS+ uses TCP as a transport protocol mechanism.

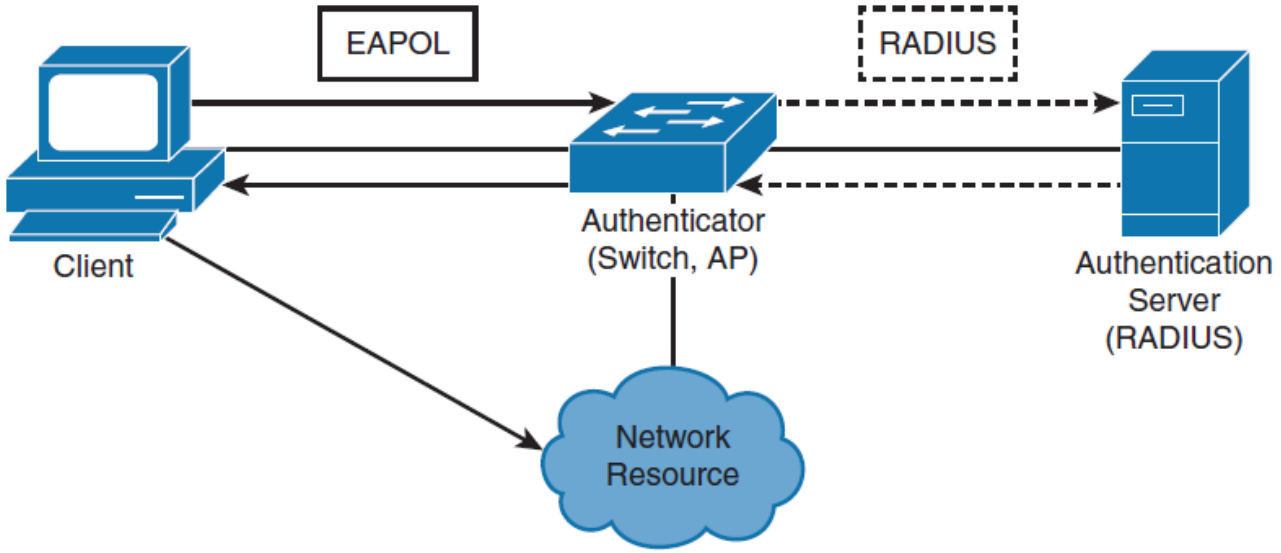
Identity-Based Networking

Identity-Based Networking

- Identity-based networking is a concept that unites several features to include authentication, access control, mobility, and user policy components with the aim to provide and restrict users with the network services that they are entitled to.
- From a switch perspective, identity-based networking allows you to verify users once they connect to a switch port.



IEEE 802.1X Port-Based Authentication Overview

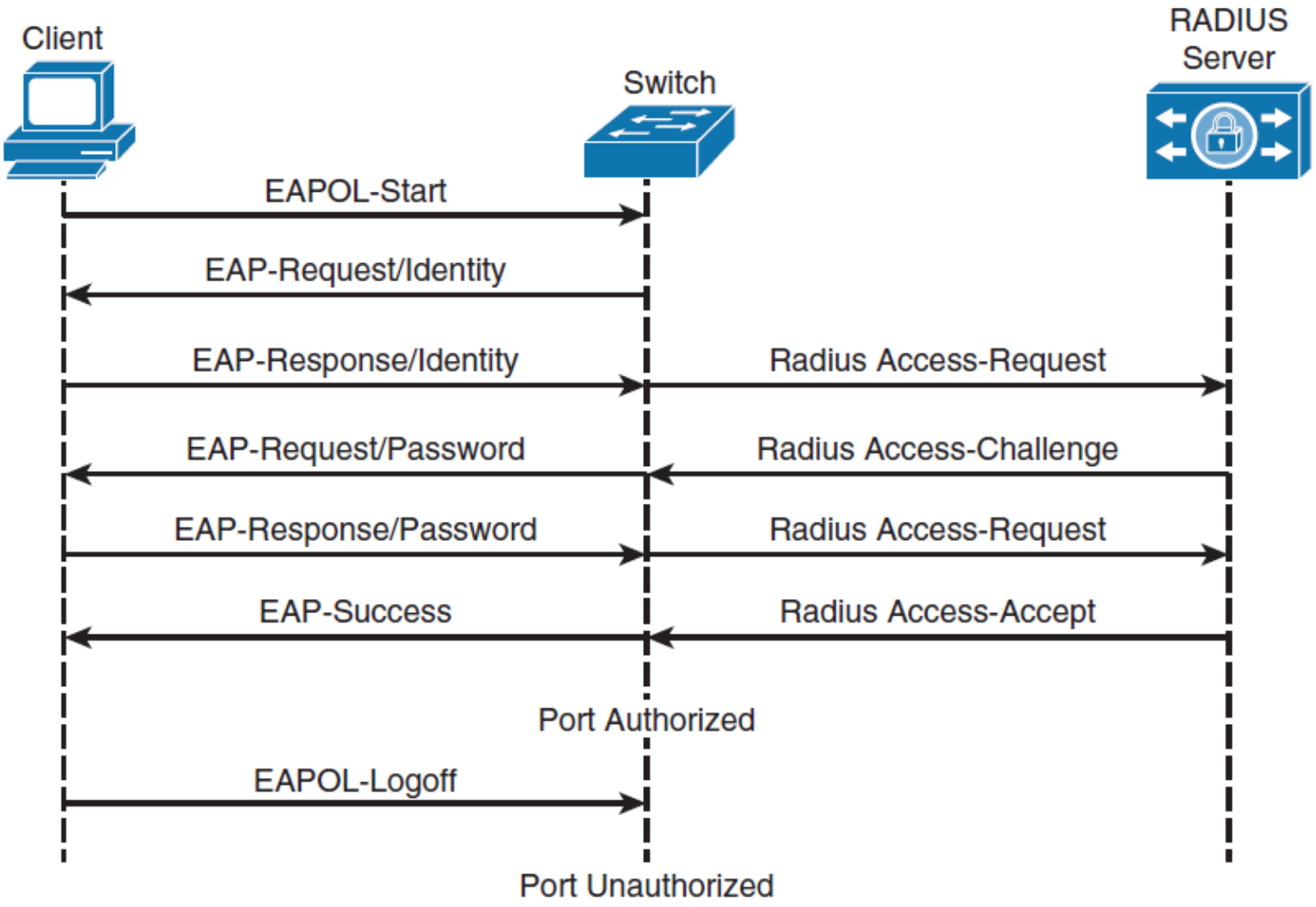


- Until the client is authenticated, 802.1X access control allows only EAPoL, Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic to pass through the port to which the client is connected. After authentication is successful, normal traffic can pass through the respective port.

802.1X Client/Server Model

- **Client**
  - Usually, a workstation or laptop with 802.1X-compliant client software.
  - Most modern operating systems include native 802.1X support.
  - The client is also referred to as a *supplicant* in 802.1X terminology.
- **Authenticator**
  - Usually, an edge switch or wireless access point (AP), the authenticator controls the physical access to the network based on the authentication status of the client.
  - Authenticator includes a RADIUS client, which is responsible for encapsulation and decapsulation of Extensible Authentication Protocol (EAP) frames and interaction with the authentication server.
- **Authentication server**
  - A server that performs the actual authentication of the client.
  - Currently, a RADIUS server with EAP extensions is the only supported authentication server.

802.1X Port-Based Authentication Overview



802.1X Configuration Example

```
Switch(config)# aaa new-model
Switch(config)# radius server host 172.16.1.1 key cisco456
Switch(config)# aaa group server radius Mygroup3
Switch(config-sg-radius)# server 172.16.1.1
Switch(config)# aaa authentication dot1x default group Mygroup3
Switch(config)# dot1x system-auth-control
Switch(config)# interface GigabitEthernet0/2
Switch(config-if)# dot1x port-control auto
```

- You will not be able to issue **dot1x** commands on the interface if it is not set to **switchport mode access** prior.
- The default state of switch ports varies between switches, but it is not commonly set to the access mode.

Network Time Protocols

The Need for Accurate Time

- The need for accurate time is increasing year by year.
- Coordinating events, marking logs, and kicking-off scripts all run based on a system clock.
- Therefore, in today's network, coordination of system clocks and their accuracy is increasing in importance.
- From a best practice perspective, it is recommended to set clocks on all network devices to UTC regardless of their location, and then configure the time zone to display the local time if desired. In this manner, global operations can fall back to UTC time for relative time.

Configuring the System Clock Manually

```
Switch# show clock
10:10:03.979 UTC Thu Feb 22 2001
! Shows what the device thinks is the current time
Switch# clock set 12:13:00 10 January 2015
! Manual system clock reconfiguration
Switch# show clock detail
12:13:03.487 UTC Sat Jan 10 2015
Time source is user configuration
! Verification of how system clock has changed. Adding the detail keyword will tell
  you what was the source of clock configuration
```

```
Switch(config)# clock timezone EDT -5
Switch(config)# clock summer-time EDT recurring
! Changes timezone and enables daylight savings time. In this example, EDT is used.
Switch# show clock detail
07:44:12.370 EDT Sat Jan 10 2015
Time source is user configuration
Summer time starts 02:00:00 EDT Sun Mar 8 2015
Summer time ends 02:00:00 EDT Sun Nov 1 2015
! Verifies how clock settings now reflect local time
```

Setting Summertime

- **clock summer-time** *zone* **recurring** [*weekday month hh:mm weekday month hh:mm* [offset]]

- **clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]

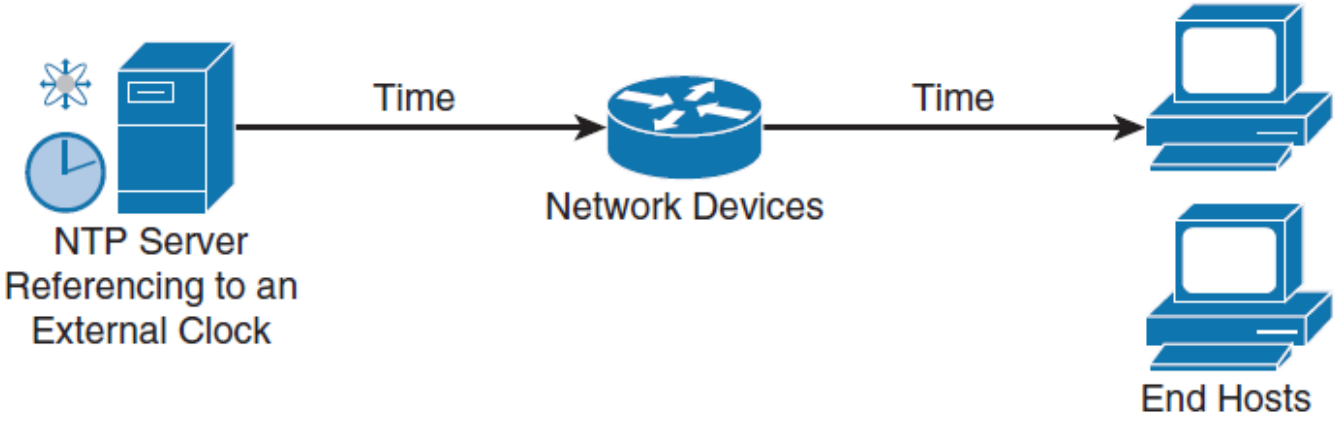
- **clock summer-time** *zone* **date** *month date year hh:mm month date year hh:mm* [*offset*]

Setting Summertime

Parameter	Description
zone	Name of the time zone (for example, PDT) to be displayed when summer time is in effect
recurring	Indicates that summer time should start and end on the corresponding specified days every year
date	Indicates that summer time should start on the first specific date that is listed in the command and end on the second specific date in the command
week	(Optional) Week of the month (1 to 5 or last).
day	(Optional) Day of the week (Sunday, Monday, and so on)
date	Date of the month (1 to 31)
month	(Optional) Month (January, February, and so on)
year	Year (1993 to 2035)
hh:mm	(Optional) Time (military format) in hours and minutes
offset	(Optional) Number of minutes to add during summer time (default = 60)

Network Time Protocol Overview

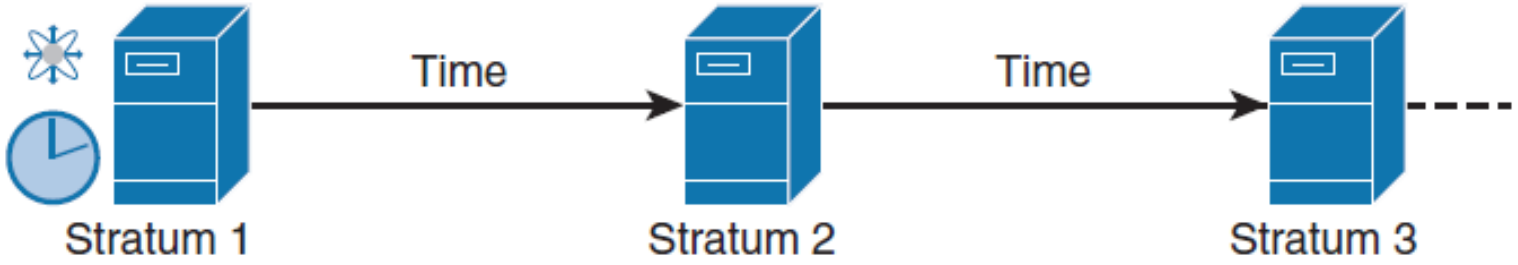
- Manually setting the clocks of any network device is neither accurate nor scalable.
- The best practice is to use Network Time Protocol (NTP), Simple NTP (SNTP), or Precision Time Protocol (PTP)
- NTP is designed to synchronize the time throughout an entire network infrastructure, including servers, switches, routers, host machines, wireless access points, uninterruptible power supply (UPS), and so on.
- NTP leverages UDP port 123 for both the source and destination by default.



Network Time Protocol Overview

- An NTP network usually gets its reference time from an authoritative time source, such as a radio clock, GPS, or an atomic clock attached to an NTP time server somewhere in the network.
- NTP then distributes this time across the network.
- Accurate timekeeping is made possible by exchanging NTP messages between each pair of machines (server/client) with an association.
- However, in a LAN environment, NTP can be configured to use IP broadcast messages instead.
- To keep accuracy of time, NTP uses the concept of a stratum to describe how many NTP hops away a machine is from an authoritative time source.
- A machine running NTP automatically chooses the machine with the lowest stratum number

NTP: Stratum



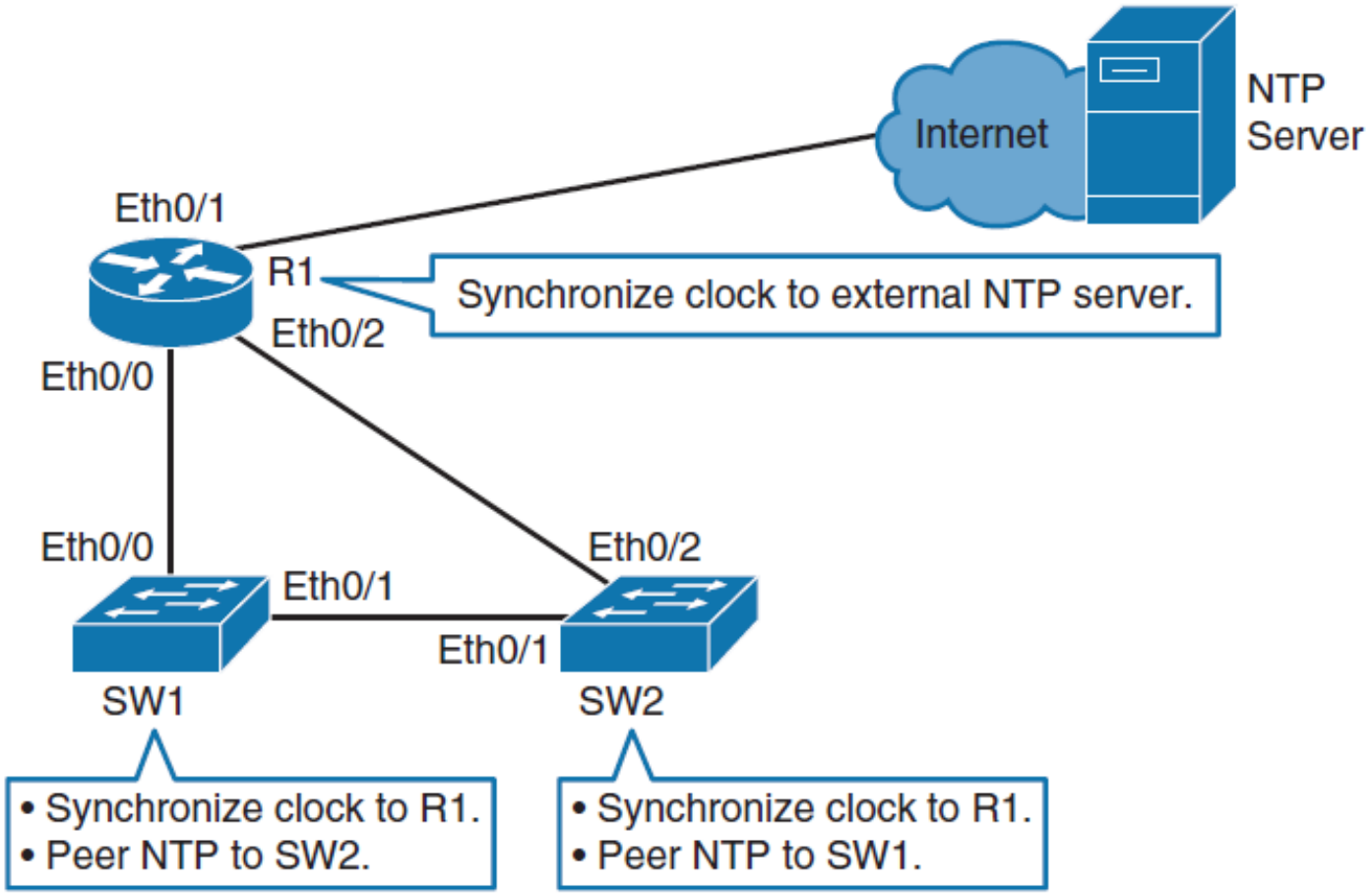
- NTP avoids in two ways synchronizing to a machine whose time may not be accurate.
  - NTP never synchronizes to a machine that is not synchronized itself.
  - NTP compares the time that is reported by several machines and will not synchronize to a machine whose time differs significantly from the others, even if its stratum is lower.

NTP Modes

- A device may take on more than one role at a time.
- **Server**
  - Provides accurate time information to clients on the network.
- **Client**
  - Synchronizes its time to an NTP server. This mode is most suited for file server and workstation clients that are not required to provide any form of time synchronization to other local clients. It can also provide accurate time to other devices.
- **Peers**
  - Peers only exchange time synchronization information.
- **Broadcast/multicast**
  - Special “push” mode of NTP server where the local LAN is flooded with updates; used only when time accuracy is not an issue.

NTP Example

```
ntp server 209.165.200.187
```



Verify NTP

```
R1# show ntp status
Clock is synchronized, stratum 2, reference is 209.165.200.187
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**10
ntp uptime is 1500 (1/100 of seconds), resolution is 4000
reference time is D67E670B.0B020C68 (05:22:19.043 PST Mon Jan 13 2014)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 630.22 msec, peer dispersion is 189.47 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 64, last update was 5 sec ago.
```

```
R1# show ntp associations

  address         ref clock       st   when   poll reach  delay  offset   disp
*~209.165.200.187 .LOCL.           1     24     64    17  1.000  -0.500  2.820
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
```

Setting and Verifying the Clock Time Zone and Daylight Savings Time

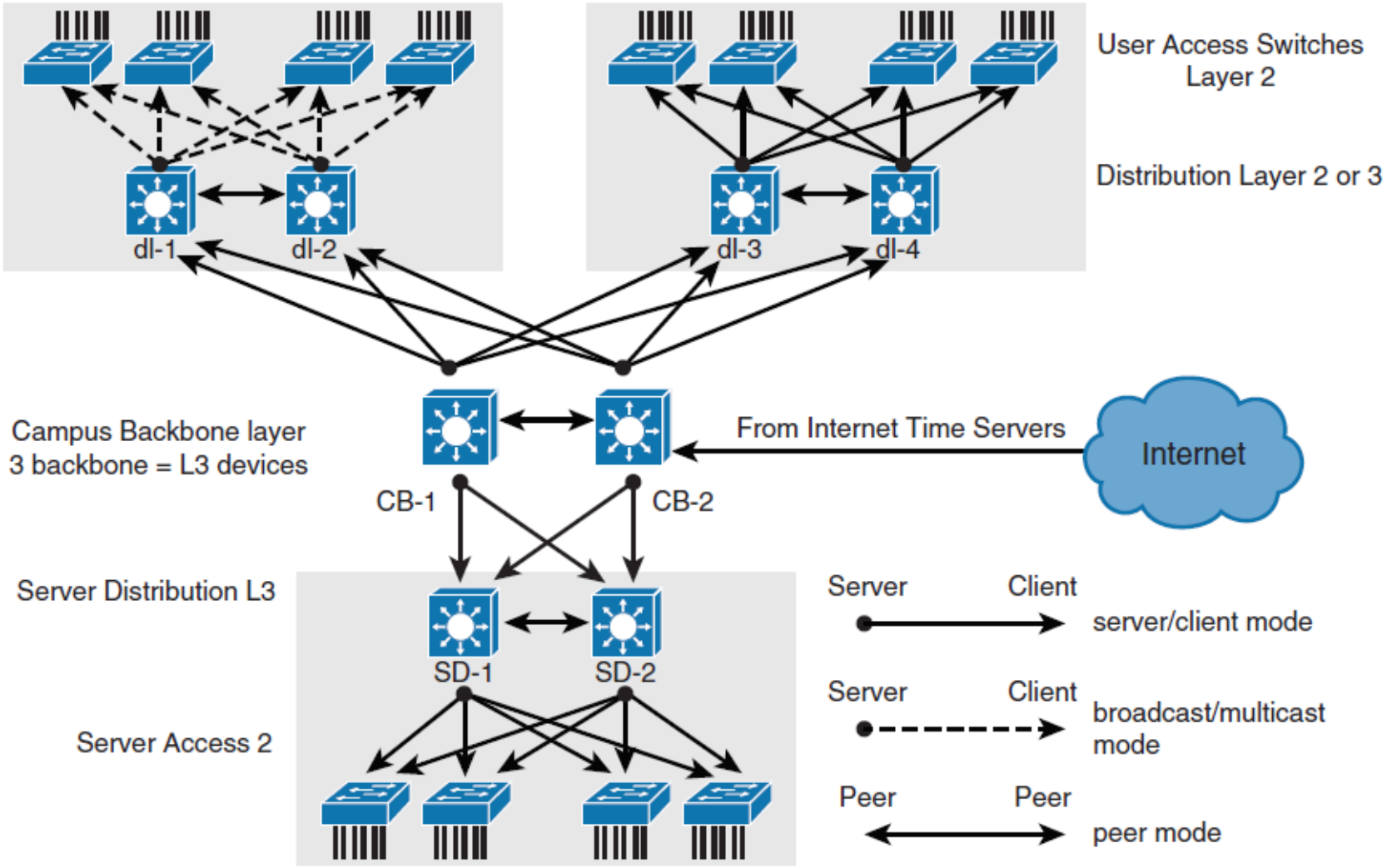
```
R1(config)# clock timezone EDT -5
R1(config)# clock summer-time EDT recurring

R1# show clock detail
08:01:54.470 EDT Tue Jan 14 2014
Time source is NTP
Summer time starts 02:00:00 EDT Sun Mar 9 2014
Summer time ends 02:00:00 EDT Sun Nov 2 2014
```

Downstream NTP Example

```
SW1(config)# ntp server 10.0.0.1
SW1(config)# clock timezone EDT -5
SW1(config)# clock summer-time EDT recurring
```

```
SW1# show ntp status
Clock is synchronized, stratum 3, reference is 10.0.0.1
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**18
reference time is D67FD8F2.4624853F (10:40:34.273 EDT Tue Jan 14 2014)
clock offset is 0.0053 msec, root delay is 0.00 msec
root dispersion is 17.11 msec, peer dispersion is 0.02 msec
```

NTP Design Principles



Securing NTP

NTP authentication steps:

- **Step 1.** Define NTP authentication key or keys with **`ntp authentication-key`** command. Every number specifies a unique NTP key.
- **Step 2.** Enable NTP authentication using the **`ntp authenticate`** command.
- **Step 3.** Tell the Cisco device which keys are valid for NTP authentication using the **`ntp trusted-key`** command. The only argument to this command is the key that you defined in the first step.
- **Step 4.** Specify the NTP server that requires authentication by using the **`ntp server`** *`ip-address`* **`key`** *`key-number`* command. You can similarly authenticate NTP peers by using the **`ntp peer`** *`ip-address`* **`key`** *`key-number`* command.

NTP Authentication Example

```
NTPServer(config)# ntp authentication-key 1 md5 MyPassword
NTPServer(config)# ntp authenticate
NTPServer(config)# ntp trusted-key 1
NTPClient(config)# ntp authentication-key 1 md5 MyPassword
NTPClient(config)# ntp authenticate
NTPClient(config)# ntp trusted-key 1
NTPClient(config)# ntp server 10.0.1.22 key 1
```

NTP ACL's

For NTP, you can configure the following four restrictions through access lists:

- **Peer**
  - Time synchronization requests and control queries are allowed. The device is allowed to synchronize itself to remote systems that pass the access list.
- **Server:**
  - Time synchronization requests and control queries are allowed. The device is *not* allowed to synchronize itself to remote systems that pass the access list.
- **Server-only**
  - Only allows synchronization requests.
- **Query-only**
  - Only allows control queries.

NTP Access List Example

```
Router(config)# access-list 1 permit 10.0.1.0 0.0.255.255
Router(config)# ntp access-group peer 1
```

```
Router(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Router(config)# ntp access-group server-only 1
```

NTP Source Address

- The source of the NTP packet will be the same as the interface the packet was sent out on.
- When implementing authentication and access lists, it is good to have a specific interface set to act as the source interface for NTP.
- It would be wise of you to choose a loopback interface to use as the NTP source.
- This is because the loopback will never be down like physical interfaces.
- If you configured loopback 0 to act as the NTP source for all communication and that interface has, for example, an IP address of 192.168.12.31, you can write up just one access list that will allow or deny based on one single IP address of 192.168.12.31.

NTP Versions

- NTPv4 is an extension of NTP Version 3.
- NTPv4 supports both IPv4 and IPv6 and is backward compatible with NTPv3.

NTPv4 adds the following capabilities:

- Support for IPv6
- Better security
- Leverages multicast over broadcast for push modes

SNMP

SNMP

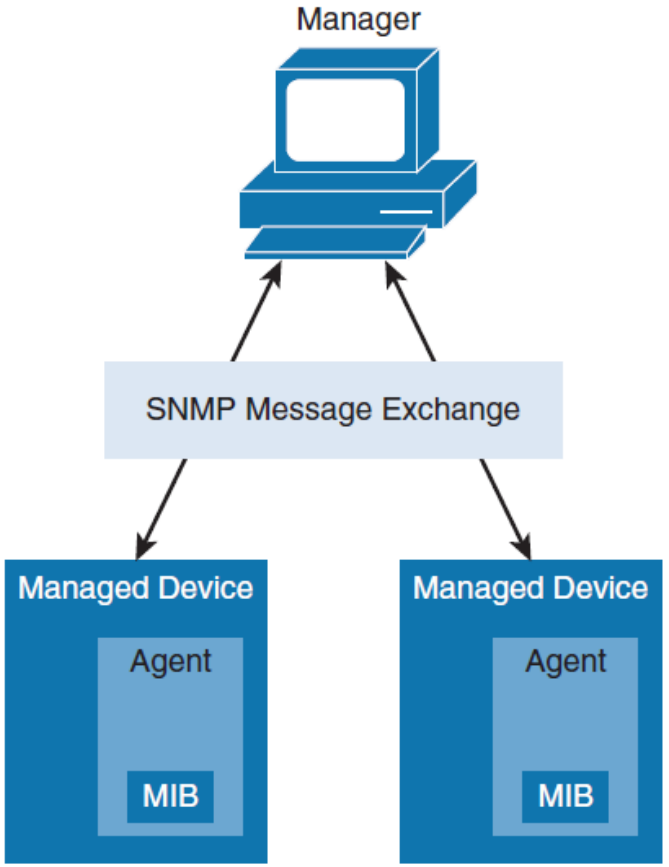
This subsection covers the following topics related to SNMP:

- The role of SNMP
- Different SNMP versions
- Recommended practices for setting up SNMP
- Configuration examples for SNMP Version 3
- Verifying SNMP configurations

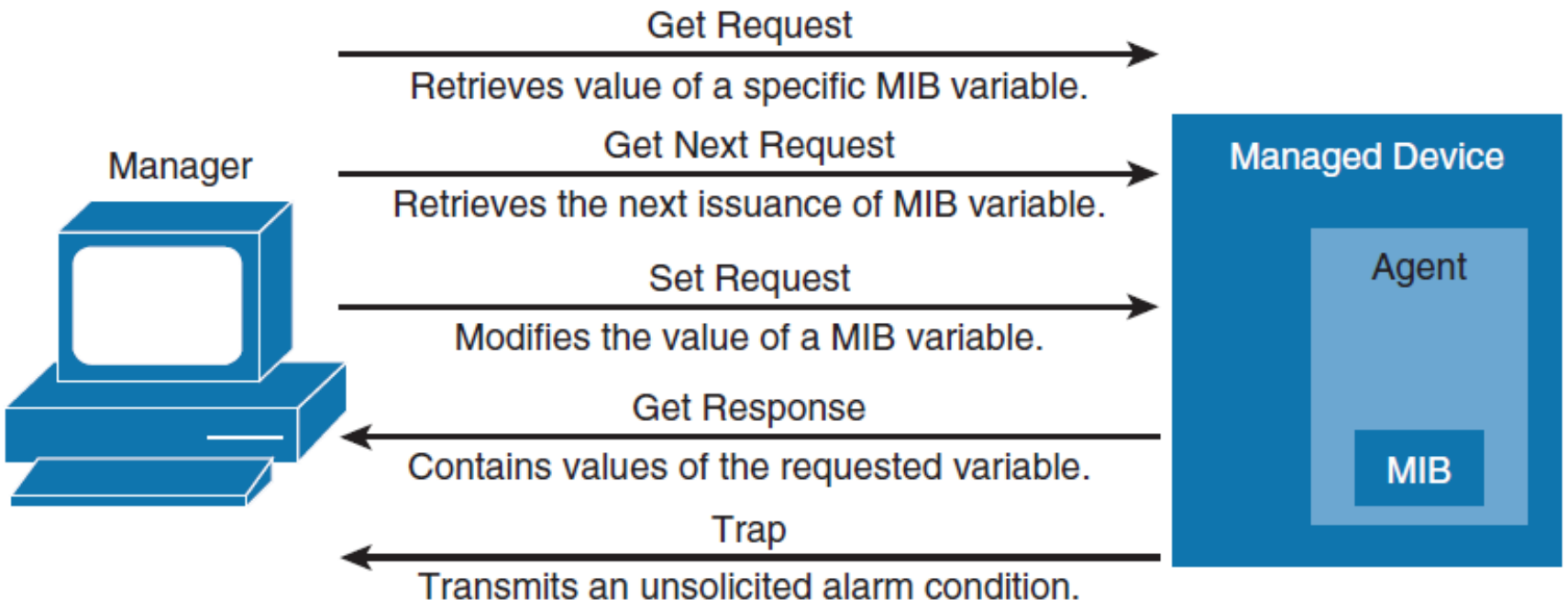
SNMP Overview

SNMP systems consist of two components, as follows:

- The **SNMP manager** that periodically polls the SNMP agents on managed devices by querying the device for data. Periodic polling has a disadvantage: A delay occurs between an actual event occurrence and the time the SNMP manager polls the data.
- **SNMP agents** on managed devices collect device information and translate it into a compatible SNMP format according to the MIB. MIBs are collections of definitions of the managed objects. SNMP agents keep the database of values for definitions written in the MIB.



SNMP Process



SNMP Versions

- **Version 1**
  - Introduced five message types
    - Get Request,
    - Get Next Request
    - Set Request
    - Get Response
    - Trap.
  - This version is rarely used nowadays.
- **Version 2**
  - Introduced two new message types
    - Get Bulk Request to poll large amounts of data,
    - Inform Request, a type of trap with expected acknowledgment on receipt.
  - Version 2 added 64-bit counters to accommodate faster network interfaces.
  - Added a complex security model, which was never widely accepted.
- ***Version 2c***
  - Community-based SNMP Version 2, is wide accepted
  - Community-based version of SNMP is very unsecure.
- **Version 3**
  - Methods to ensure the secure transmission of critical data between the manager and agent were added.

SNMPv3 Security

SNMPv3 supports the following three levels of security:

- **noAuthNoPriv**
  - No authentication is required, and no privacy (encryption) is provided.
- **authNoPriv**
  - Authentication is based on Hashed Message Authentication Code (HMAC), MD5, or Secure Hash (SHA). No encryption is provided.
- **authPriv**
  - In addition to authentication, cipher block chaining - Data Encryption Standard (CBC-DES) encryption is used.

SNMP Best Practices

- Restrict access to read-only.
- Use write access with separate credentials and careful consideration.
- Set up SNMP views to restrict manager to only access needed sets of MIBs.
- Configure ACLs to restrict SNMP access only by known managers.
- Use SNMPv3 authentication, encryption, and integrity where possible, including upgrading devices to support SNMPv3 if necessary.

SNMPv3 Configuration Steps

- **Step 1.** Configure an access list to be used to restrict subnets for SNMP access.
- **Step 2.** Configure the SNMPv3 views to limit access to specific MIBs.
- **Step 3.** Configure the SNMPv3 security groups.
- **Step 4.** Configure the SNMPv3 users.
- **Step 5.** Configure the SNMPv3 trap receivers.
- **Step 6.** Configure ifindex persistence to prevent ifindex changes.

SNMPv3 Best Practice Configuration

```
Switch(config)# access-list 99 permit 10.1.1.0 0.0.0.255
Switch(config)# snmp-server view OPS sysUpTime included
Switch(config)# snmp-server view OPS ifDescr included
Switch(config)# snmp-server view OPS ifAdminStatus included
Switch(config)# snmp-server view OPS ifOperStatus included
Switch(config)# snmp-server user userZ groupZ v3 auth sha secretpwd2 priv aes 256
  secondsecretpwd2
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host 10.1.1.50 traps version 3 priv userZ cpu port-
  security
Switch(config)# snmp-server ifindex persist
```

SNMP Command Reference

Command	Description
snmp-server enable traps [*notification-type*]	Enables SNMP notification types that are available on your system
snmp-server group *group-name* {v1 | v2c | v3 {auth | noauth | priv}} [context *context-name*] [read *read-view*][write *write-view*] [notify *notify-view*][access [*acl-number* | *acl-name*]]	Configures a new SNMP group with specified authentication and optionally with the specified associated SNMP context, read, write, notify view, and associated ACL
snmp-server host {*ip-address*} [informs | traps| version{1 | 2c | 3 {auth | noauth}}]	Specifies the recipient of an SNMP notification operation
snmp-server ifindex persist	Enable interface index persistence
snmp-server user *username group-name* {v1 | v2c | v3 [encrypted][atuh {md5 | sha} *auth-password*]} [access [priv {des | 3des | aes {128 | 192 | 256}} *privpassword*]] {*acl-number* | *acl-name*}]	Configures a new user to an SNMP group
snmp-server view *view-name oid-tree*	Creates a view entry

Chapter 7 Summary

- The AAA features include authentication, authorization, and accounting. The use of AAA is required in nearly all campus networks because it secures and provides administrative control and logging of user access to network devices and to the network itself.
- Identity-based networking leverages protocols such as 802.1X to support mobility, security, authentication, and authorization of users to network resources.
- Accurate time is essential for time logging services in campus networks, as are many security features like encryption.
- All Cisco Catalyst switches support NTP for time synchronization.
- NTP generally achieves millisecond accuracy in LAN networks.
- SNMP is a lightweight protocol that not only monitors and controls devices but also supports alerting of events.
- SNMPv3 is the best practice recommendation for SNMP; avoid using SNMPv2 (or v1) if it all possible (because of its lack of security features).
- Security around SNMP must be considered as part of any implementation plan. At a minimum, use authentication and encryption along with restricted write access and IP ACLs to restrict network access.

Chapter 7 Labs

- **CCNPv7.1 SWITCH Lab7.1 NTP**
- **CCNPv7.1 SWITCH Lab7.2 SNMP**

Acknowledgment

- *Some of the images and texts are from Implementing Cisco IP Switched Networks (SWITCH) Foundation Learning Guide: (CCNP SWITCH 300-115)* by Richard Froom and Erum Frahim (1587206641)
- Copyright © 2015 – 2016 Cisco Systems, Inc.
- Special Thanks to *Bruno Silva*