Kapitola 7: Network Management

CCNP SWITCH: Implementing Cisco IP Switched Networks

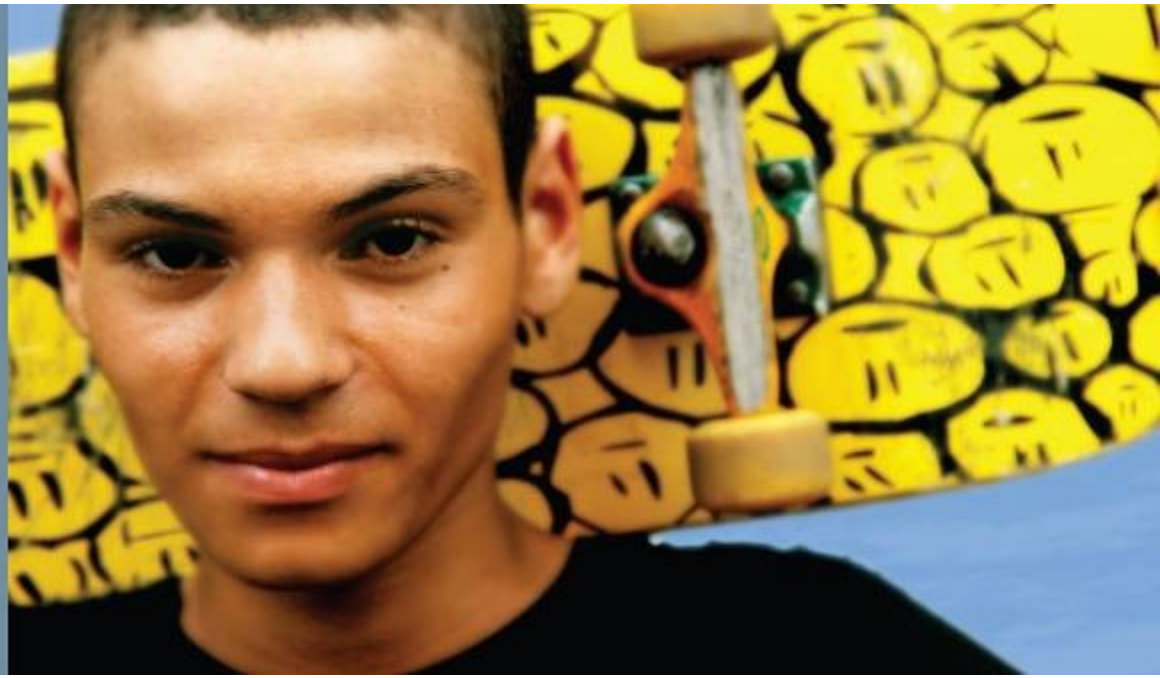
Cisco | Networking Academy®
Mind Wide Open™

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

- **Autentizace**

  Autentizace je proces identifikace uživatele před tím, než je mu povolen přístup k chráněnému zdroji.

- **Autorizace**

  Poté, co uživatel získá přístup k síti, provede se autorizace.
  Autorizace umožňuje kontrolovat úroveň uživatelů přístupu

- Účetnictví (**Accounting)**

  Účetnictví se provádí po autentizaci. Účetnictví umožňuje shromažďovat informace o činnosti uživatele a spotřebě zdrojů.

  .

Výhody AAA

- **Vyšší flexibility a řízení konfigurace přístupu**

  AAA nabízí dodatečnou pružnost autorizace na úrovni příkazů či rozhraní, která není k dispozici u lokálních pověření.

- **Škálovatelnost**

  Jak síť roste, správa velkého počtu uživatelů na více zařízeních se stává vysoce nepraktickou a náchylnou k chybám, s velkým množstvím administrativní zátěže.

- **Standardizované autentizační metody**

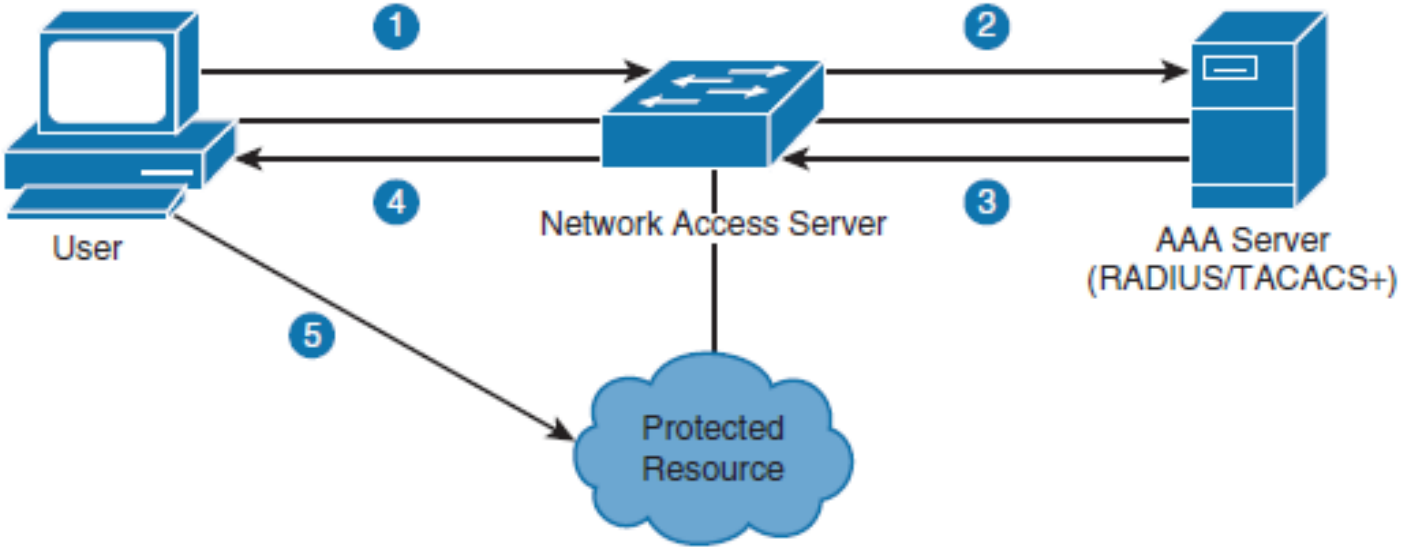
  AAA podporuje protokol RADIUS, což je průmyslově otevřený standard. To zajišťuje interoperabilitu a umožňuje flexibilitu, protože lze kombinovat různé dodavatele.

- **Více zálohovacích (backup) systémů**

  Při konfiguraci autentizace lze určit více serverů a ty kombinovat v rámci skupiny.

RADIUS a TACACS+ – přehled

- RADIUS a TACACS+ jsou AAA protokoly.
- Oba používají model klient/server.
- Uživatel resp. počítač odešle požadavek (1) na síťové zařízení, jako je například směrovač, který při spuštění AAA funguje jako server pro přístup k síti.
- Server pro přístup k síti pak komunikuje (2, 3) se serverem, který si vyměňuje zprávy RADIUS nebo TACACS +.
- Pokud je ověření úspěšné, je uživateli uděleno (4) přístup k chráněnému zdroji (5), např. k CLI zařízení, síti atd.

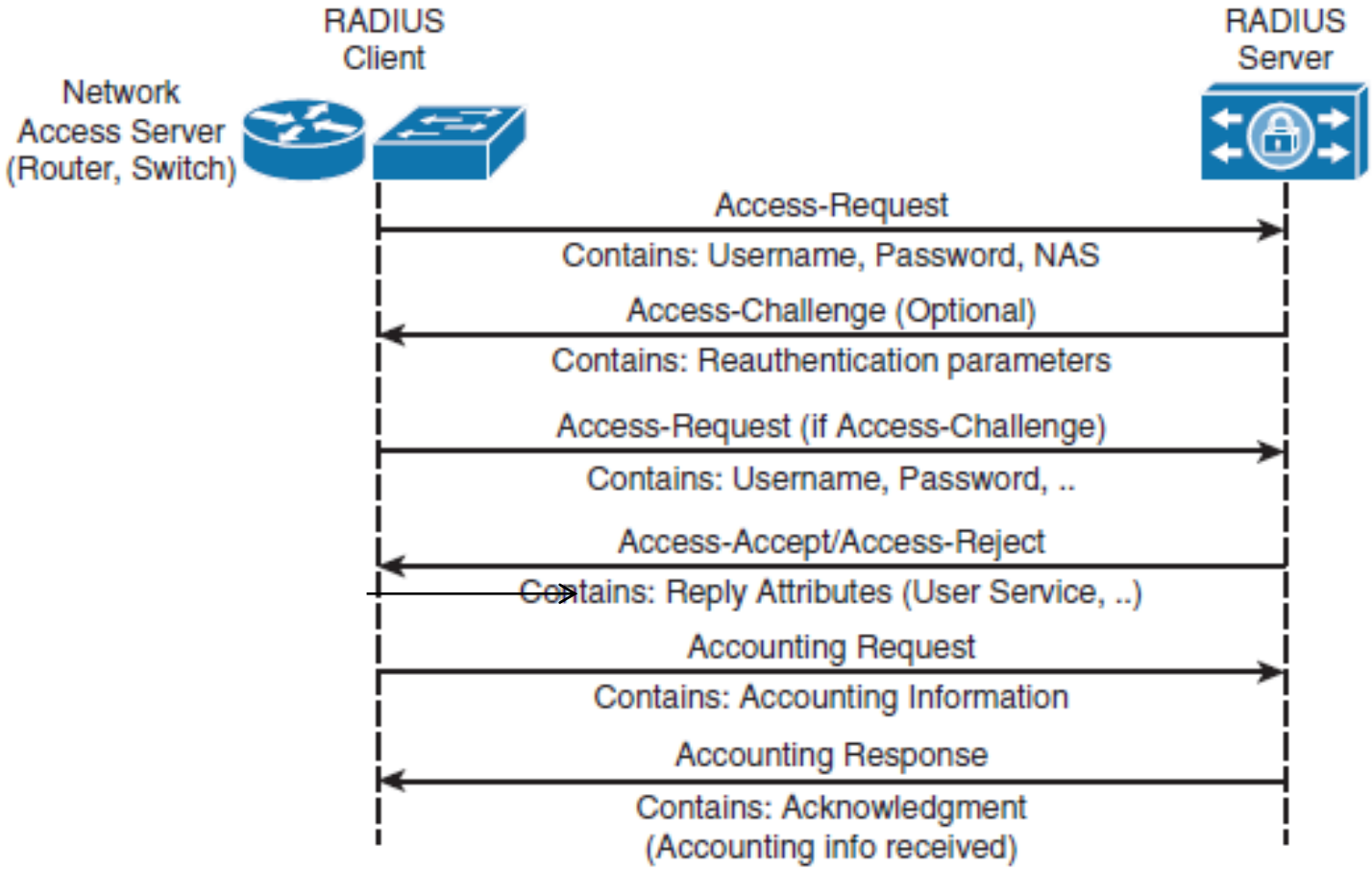


TACACS+ versus RADIUS

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 – Autentizační proces (nad UDP)

Autentizační a autorizační údaje jsou kombinovány v jediném paketu Access-Request.
Tento paket obsahuje *uživatelské jméno, šifrované heslo, adresu IP serveru NAS a číslo portu NAS.*



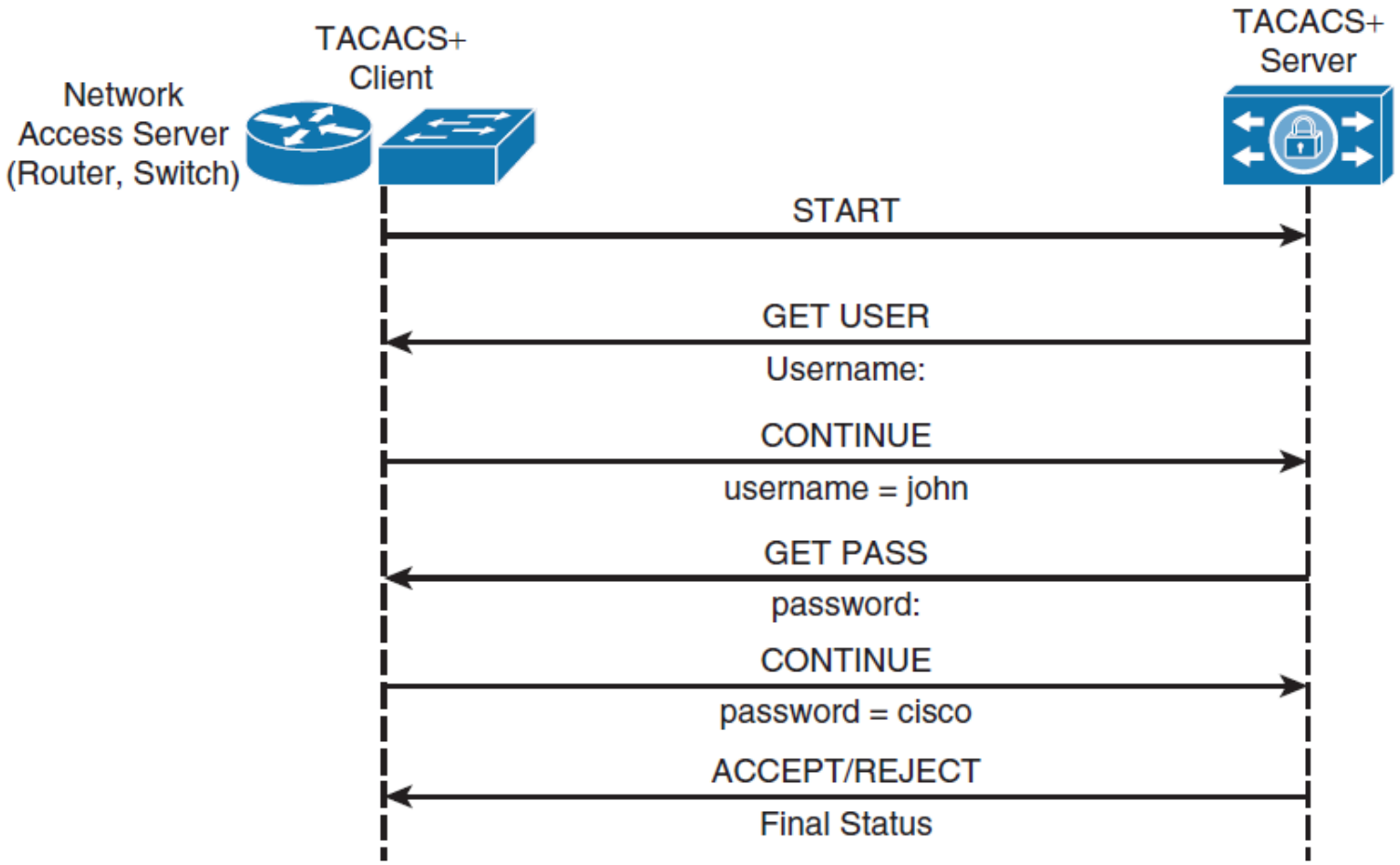
RADIUS AAA Communication

Páry AV attribut – value

https://www.cisco.com/en/US/products/sw/secursw/ps2086/products_user_guide_chapter09186a008007e364.html

https://www.iana.org/assignments/radius-types/radius-types.xhtml

TACACS+ – Autentizační proces (nad TCP)



TACACS+ Authentication Communication

Konfigurace AAA

- Pro povolení AAA je prvním krokem konfigurace příkazu **aaa new-model** v režimu globální konfigurace.
- Tento krok v podstatě nastavuje možnosti AAA.
- Pokud není tento příkaz povolen, jsou všechny ostatní příkazy AAA skryty.

- Příkaz **aaa new-model** okamžitě aplikuje lokální ověřování na všechny řádky a rozhraní (kromě řádku konzoly con 0).
- Chcete-li se vyhnout zablokování směrovače, je nejlepším postupem definovat lokální uživatelské jméno a heslo před spuštěním konfigurace AAA.

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

Konfigurace RADIUS přístupu

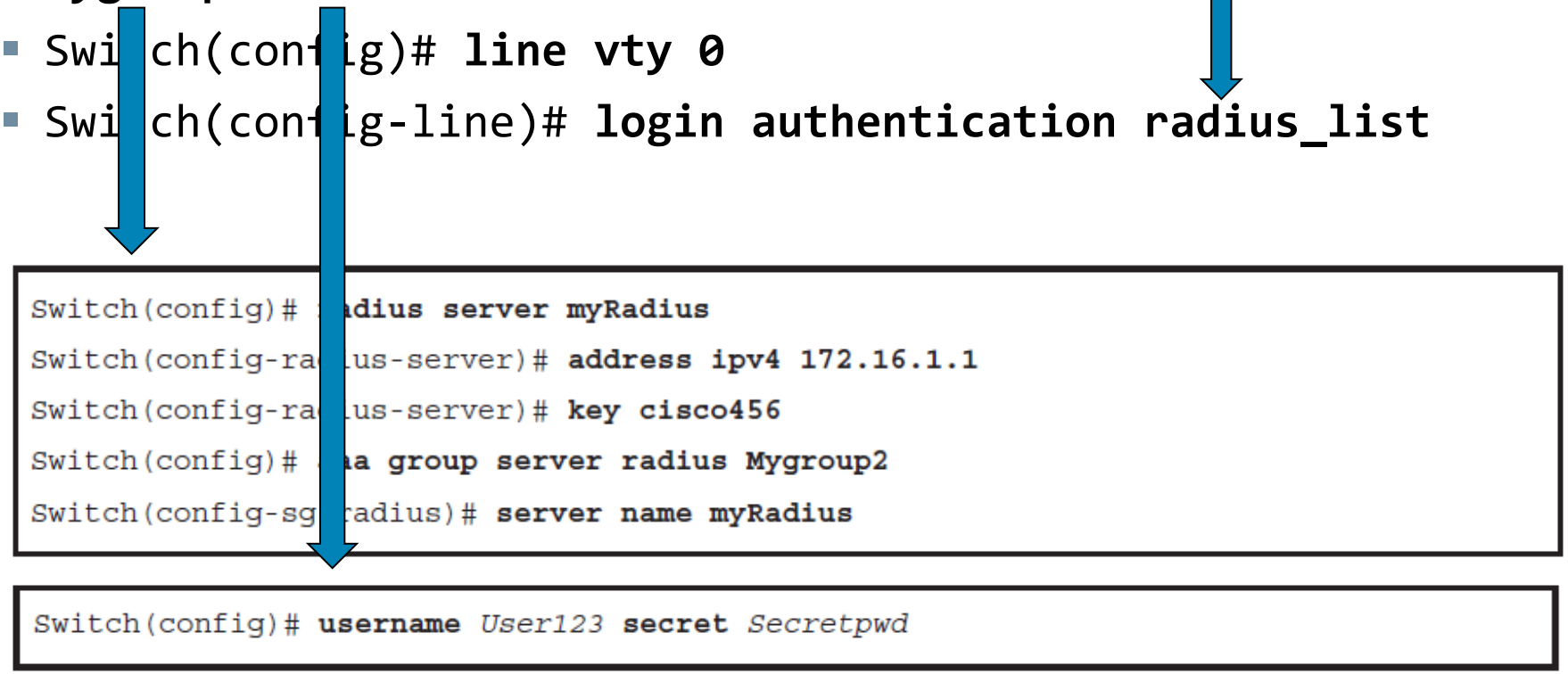
- Switch(config)# **radius server** *configuration-name*
- Switch(config-radius-server)# **address ipv4** *hostname* [**auth-port** *integer*] [**acct-port** *integer]*
- Switch(config-radius-server)# **key** *string*
- Switch(config)# **aaa group server radius** *group-name*
- Switch(config-sg-radius)# **server name** *configuration-name*

```
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
```

Specifikace zákaznického portu je volitelná

Aplikace metod RADIUSu na vty

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



Konfigurace TACACS+ pro konzolový a vty přístup

- 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 autorizace

Kroky konfigurace autorizace:

- **Krok 1.** Uveďte seznam autorizačních metod.
- **Krok 2.** Aplikujte seznam těchto metod na jedno či více rozhraní (vyjma seznamu defaultních metod).
- **Krok 3.** Je použita první uvedená metoda. Pokud neodpoví, použije se druhá a tak dále, dokud nejsou vyčerpány všechny uvedené metody. Jakmile je seznam metod vyčerpán, je zaznamenána zpráva o poruše.

- `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 Accounting

- Účetnictví AAA má stejná pravidla a konfigurační kroky jako autentizace a autorizace:
  Krok 1. Nejprve musíte definovat pojmenovaný seznam účetních metod.
  Krok 2. Použijte tento seznam na jedno nebo více rozhraní (kromě defaultního seznamu metod).
  Krok 3. Je použita první uvedená metoda; pokud neodpoví, použije se druhá a tak dále.

- `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`

Omezení pro TACACS+ a RADIUS

- RADIUS nemusí být optimální v následujících situacích:
  - Situace komunikace zařízení – zařízení
    (RADIUS nenabízí obousměrné ověřování).
  - Sítě využívající více služeb
    (RADIUS obecně váže uživatele na jeden model služby).

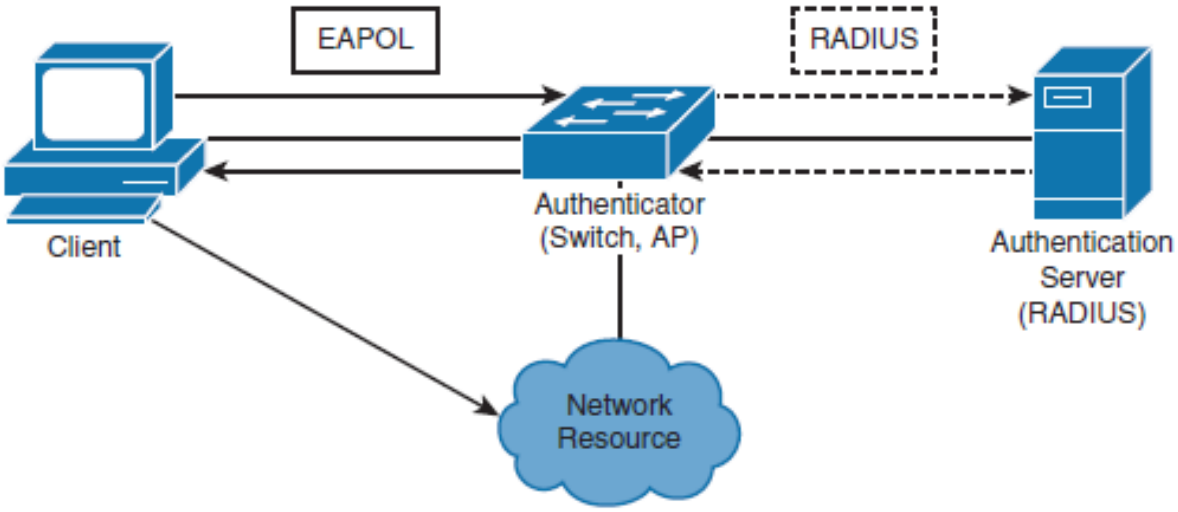
- TACACS+ nemusí být optimální v následujících situacích:
  - Multivendorové prostředí
    (TACACS+ je proprietární protokol společnosti Cisco)
  - Když se jedná o rychlost reakce ze služeb AAA (TACACS+ používá TCP jako mechanismus transportního protokolu).

Identity-Based Networking

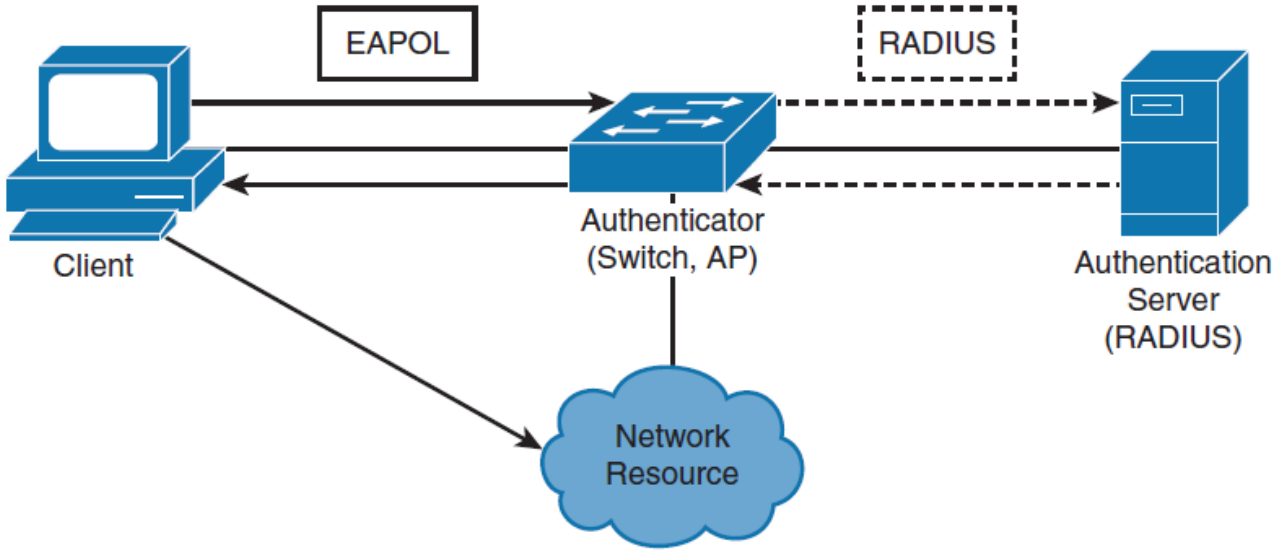
Identity-Based Networking

- Identity-based networking (síť založená na identitě) je koncept, který spojuje několik funkcí, které zahrnují autentizaci, řízení přístupu, mobilitu a součásti zásad uživatelů s cílem poskytovat a omezovat uživatele se síťovými službami, na které mají nárok.

- Z pohledu přepínače vám síť založená na identitě umožňuje ověřit uživatele po připojení k portu přepínače.



IEEE 802.1X Port-Based Autentizace

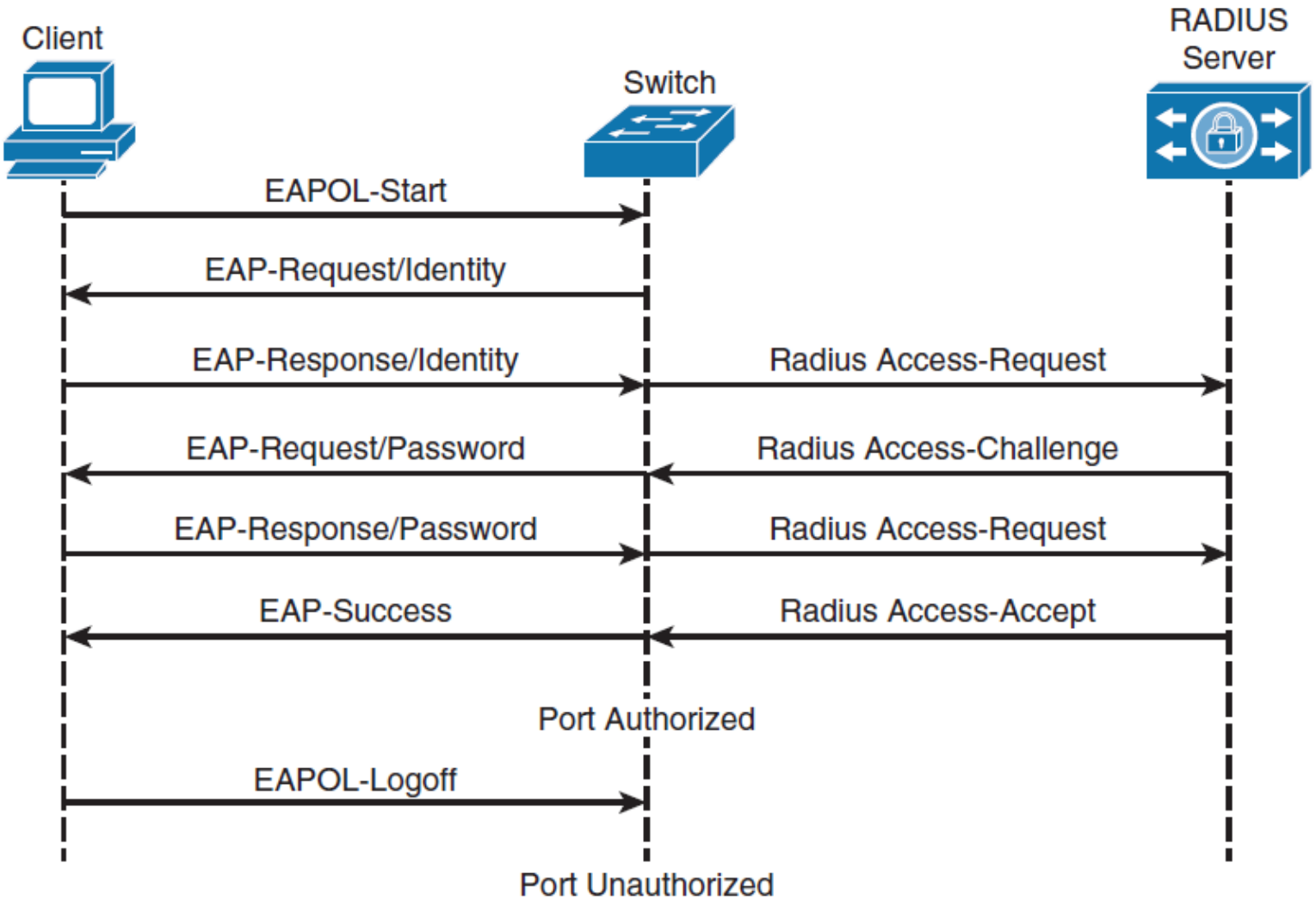


- Dokud není klient autentizován, řízení přístupu 802.1X umožňuje, aby pouze provoz EAPOL, CDP a STP procházel portem, ke kterému je klient připojen. Po úspěšném ověření může příslušným portem projít běžný provoz.

Model klient/server protokolu 802.1X

- **Klient**
  - Obvykle pracovní stanice nebo notebook s klientským softwarem kompatibilním s 802.1X.
  - Většina moderních operačních systémů zahrnuje nativní podporu 802.1X.
  - Klient je v terminologii 802.1X označován také jako žadatel.
- **Autentikátor**
  - Obvykle hraniční přepínač nebo bezdrátový přístupový bod (AP), autentizátor řídí fyzický přístup k síti na základě ověřovacího statusu klienta.
  - Authentikátor obsahuje klienta RADIUS, který je zodpovědný za zapouzdření a dekomulaci rámců protokolu EAP (Extensible Authentication Protocol) a interakci s ověřovacím serverem.
- **Autentizační server**
  - Server, který provádí skutečné ověření klienta.
  - Server RADIUS s příponami (extensions) EAP je v současné době jediným podporovaným autentizační serverem.

802.1X Port-Based Authentication



802.1X – konfigurační příklad

```
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
```

- Nelze na rozhraní vydávat příkazy dot1x, pokud předtím není nastaven switchport mode.
- Defaultní stav portů přepínačů se liší mezi přepínači, ale není běžně nastaven na režim access.

Network Time Protocols

Proč přesný čas

- Potřeba přesného času se každým rokem zvyšuje.
- Koordinační události, logování značek a skripty, které jsou založeny na systémových hodinách.
- Proto se v dnešní síti zvyšuje význam koordinace systémových hodin a jejich přesnosti.
- Z hlediska nejlepší praxe se doporučuje nastavit hodiny na všech síťových zařízeních na UTC bez ohledu na jejich umístění a poté nakonfigurovat časové pásmo, aby se v případě potřeby zobrazil místní čas.

Manuální konfigurace systémových hodin

```
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
```

Nastavení letního času

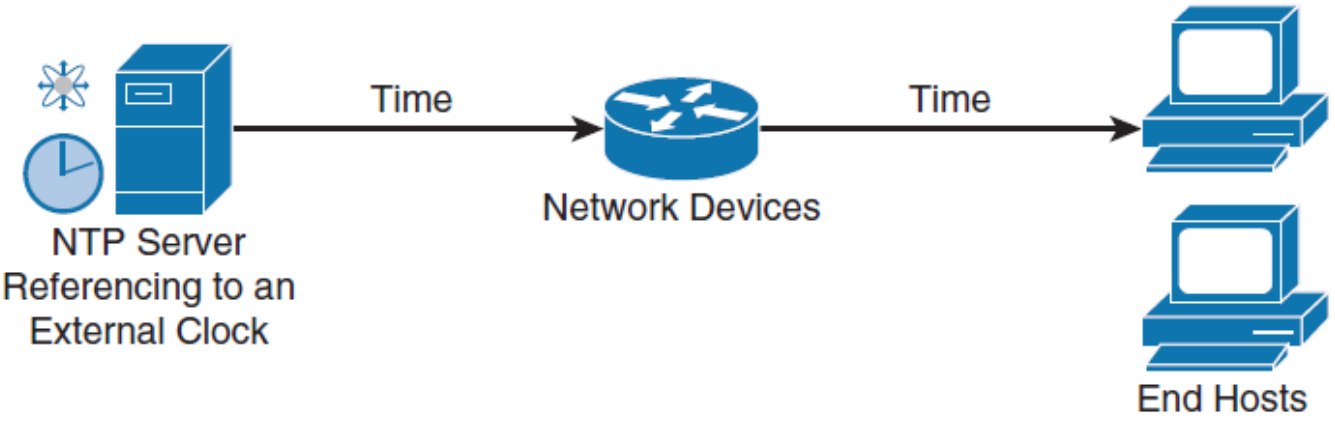
- **clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [offset]]
- **clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]
- **clock summer-time** *zone* **date** *month date year hh:mm month date year hh:mm* [*offset*]
- https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4000/8-2glx/configuration/guide/ntp.html

Nastavení letního času

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 – přehled

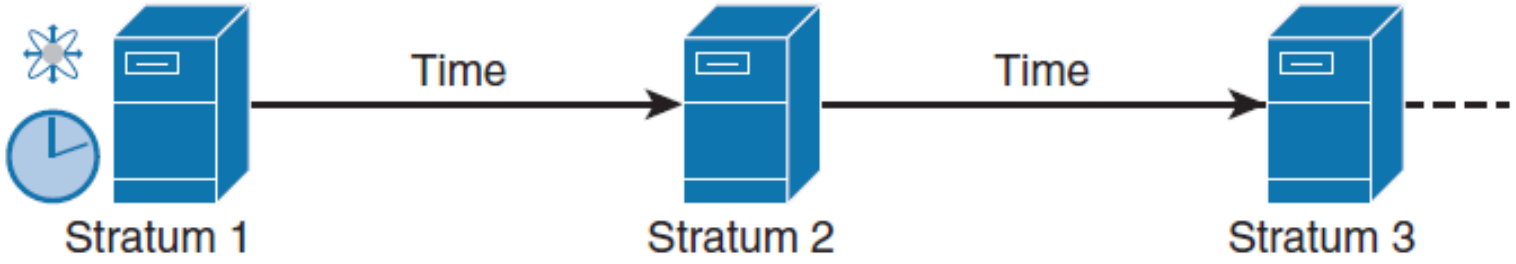
- Ruční nastavení hodin libovolného síťového zařízení není ani přesné ani škálovatelné.
- Nejlepším postupem je použití protokolu NTP (Network Time Protocol), jednoduchého protokolu NTP (SNTP) nebo protokolu PTP (Precision Time Protocol).
- NTP je navržen tak, aby synchronizoval čas v celé síťové infrastruktuře, včetně serverů, přepínačů, směrovačů, hostitelských počítačů, bezdrátových přístupových bodů, zdroje nepřerušitelného napájení (UPS) a tak dále.
- NTP standardně využívá UDP port 123 pro zdroj i cíl.



Network Time Protocol – přehled

- Síť NTP obvykle získá svůj referenční čas z autoritativního zdroje času, jako jsou rádiové hodiny, GPS nebo atomové hodiny připojené k časovému serveru NTP někde v síti. NTP pak tento čas distribuuje po síti.
- Přesné měření času je umožněno výměnou zpráv NTP mezi jednotlivými dvojicemi strojů (server / klient).
- V prostředí sítě LAN však může být služba NTP nakonfigurována tak, aby místo ní použila broadcasty IP.
- Chcete-li zachovat přesnost času, NTP používá **koncept stratum** k popisu, kolik NTP skoků je od autoritativního zdroje času.
- **Stroj s NTP automaticky vybere stroj s nejnižším číslem vrstvy**.

NTP: Stratum



NTP se vyhne dvěma způsoby synchronizace se strojem, jehož čas nemusí být přesný.

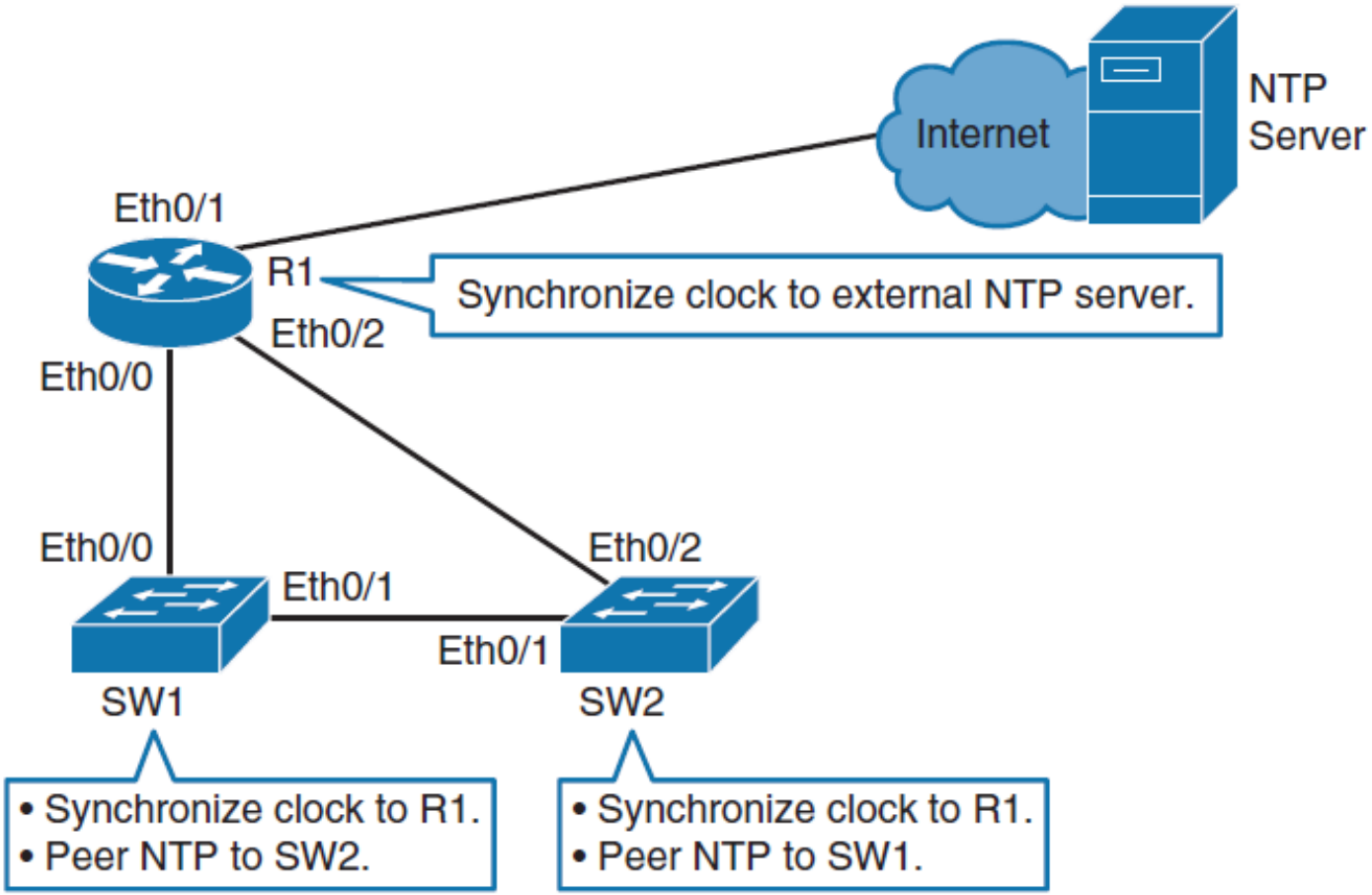
- NTP se nikdy nesynchronizuje se strojem, který není sám synchronizován.
- NTP porovnává čas, který je hlášen několika stroji, a nebude synchronizován se strojem, jehož čas se výrazně liší od času ostatních, i když je jeho vrstva nižší.

Role NTP

- Zařízení může mít současně více než jednu roli.
- **Server**
  - Poskytuje přesné časové informace klientům v síti.
- **Klient**
  - Synchronizuje svůj čas se serverem NTP. Tento režim je nejvhodnější pro klienty souborových serverů a pracovních stanic, kteří nejsou povinni poskytovat žádnou formu synchronizace času ostatním lokálním klientům. Může také poskytovat přesný čas dalším zařízením.
- **Peers**
  - Pouze si vyměňují informace sloužící pro symchronizaci času. Ten z nich, který má nejpřesnější čas, je server a ostatní klienti.
- **Broadcast/multicast**
- Speciální „push“ režim NTP serveru, který se používá pokud je místní LAN zaplavena aktualizacemi. Používá se pouze tehdy, není-li časová přesnost problém.

Příklad NTP

```
ntp server 209.165.200.187
```



Dvě fáze procesu postupné konvergence

- V případě velkých rozdílů (nad 128 ms) dochází k tzv. skokové (stepping) synchronizaci. Ta zaručí okamžité srovnání času-
- V případě menších rozdílů probíhá tzv. slewing (přibližování).
- Když je rozdíl větší než cca 17 minut, výsledkem je konec aplikace na straně klienta a chybové hlášení informující o tom, že není něco v pořádku.

Verifikace 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
```

Nastavení a verifikace Clock Time Zone a 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 – příklad

```
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
```

Plochá (flat) struktura

- S plochou strukturou jsou všechny směrovače nakonfigurovány tak, aby se navzájem sledovaly jako NTP peers. Každý router bude fungovat jako klient i server s každým dalším směrovačem. Dva nebo tři směrovače by měly být nakonfigurovány tak, aby synchronizovaly svůj čas s externími časovými servery a tím je zajištěna redundance externích časových serverů.

- Tento model je velmi stabilní, protože každé zařízení se synchronizuje s každým dalším zařízením v síti. Nevýhodami jsou obtíže při administraci, pomalé konvergence a špatná škálovatelnost.

- Pokud přidáte zařízení do sítě, může vám dlouho trvat, než identifikujete všechna ostatní zařízení a nahlédnete do nového zařízení. Vzhledem k tomu, že všechna zařízení ve vztahu peer-to-peer mají slovo při výběru nejlepšího času, může chvíli trvat, než se směrovače shodnpu na přesném čase.

- **Závěr:** plochá struktura NTP se nedoporučuje pro velké sítě.

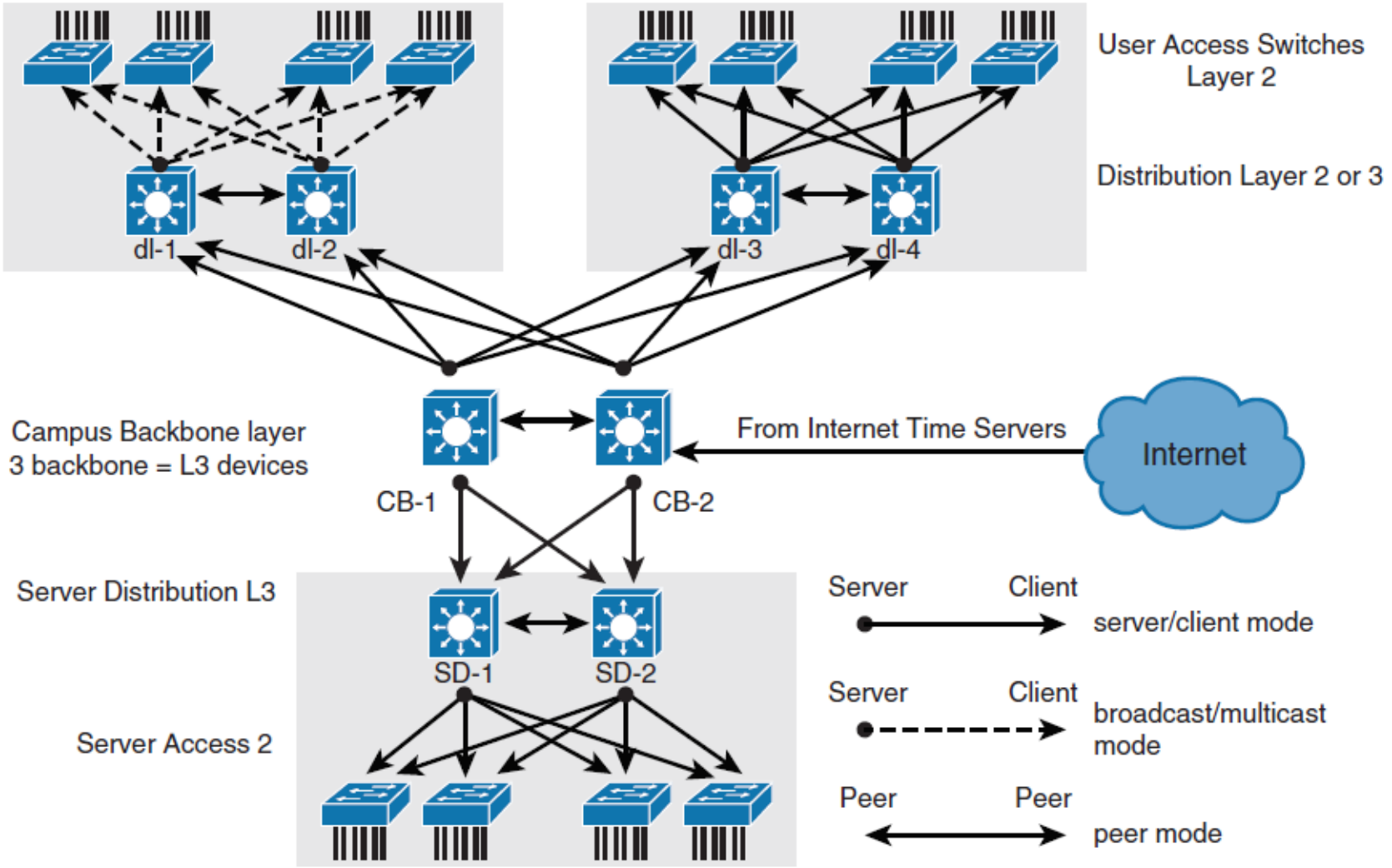
Hierarchická implementace NTP

- Pro velké sítě je nejlepším postupem implementovat NTP hierarchicky.
- Poskytovatelé internetových služeb (ISP) používají tento typ hierarchického modelu pro škálování časové synchronizace.
- Každý ISP má několik serverů úrovně stratum 1, které jsou synchronizovány s jinými zařízeními ISP, a poskytovatel služeb Internetu nakonec poskytuje služby synchronizace času pro zařízení zákazníků na okraji sítě. Okrajové zákaznické zařízení pak poskytuje časovou synchronizaci zbytku interní zákaznické sítě.
- Výsledkem je, že tento odstupňovaný model spotřebovává méně administrativní režie a časová konvergence je minimalizována, protože každý zákaznický okrajový směrovač není spojen s každým jiným zákaznickým okrajovým směrovačem. Pro velké podnikové sítě má smysl implementovat podobnou hierarchii synchronizace NTP.

Hybridní model

- Hybridní model je označován jako struktura hvězd, kde všechna zařízení v síti mají vztah s několika časovými servery v jádru.

Ilustrace návrhových principů NTP



Zabezpečení NTP

Kroky ověřování NTP:

Krok 1. Definujte NTP ověřovací klíč nebo klíče příkazem **ntp autentication-key**. Každé číslo určuje unikátní klíč NTP.

Krok 2. Povolte autentizaci NTP pomocí příkazu **ntp authenticate**.

Krok 3. Řekněte zařízení Cisco, které klíče jsou platné pro ověřování NTP pomocí příkazu **ntp trusted-key**. Jediným argumentem tohoto příkazu je klíč, který jste definovali v prvním kroku.

Krok 4. Určete server NTP, který vyžaduje ověření pomocí příkazu **ntp server *ip-address* key *key-number***. Podobně můžete ověřit totožnost NTP pomocí příkazu **ntp peer ip-key *key-number***.

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
```

Čtyři omezení pomocí NTP ACL

- **Peer**
  - Jsou povoleny požadavky na synchronizaci času a kontrolní dotazy. Zařízení se může synchronizovat se vzdálenými systémy, které vyhovují ACL.
- **Server**
  - Jsou povoleny požadavky na synchronizaci času a kontrolní dotazy. Zařízení není dovoleno synchronizovat se se vzdálenými systémy, které splňují ACL.
- **Server-only**
  - Pouze povoluje synchronizační požadavky.
- **Query-only**
  - Umožňuje pouze kontrolní dotazy.

Příklad NTP ACL

```
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 – zdrojová adresa

- Zdroj paketu NTP bude stejný jako rozhraní, na kterém byl paket odeslán.
- Při implementaci ověřovacích a přístupových seznamů je dobré mít specifické rozhraní, které bude fungovat jako zdrojové rozhraní pro NTP.
- Bylo by moudré zvolit rozhraní feedbacku pro použití jako zdroj NTP. Je to proto, že feedback nikdy nebude jako fyzická rozhraní.
- Pokud jste nakonfigurovali loopback 0, aby fungoval jako zdroj NTP pro veškerou komunikaci a toto rozhraní má například IP adresu 192.168.12.31, můžete zapsat pouze jeden přístupový seznam, který povolí nebo zakáže na základě jedné IP adresy.

Verze NTP

- NTPv4 je rozšíření NTP Version 3. NTPv4 podporuje IPv4 a IPv6 a je zpětně kompatibilní s NTPv3.

NTPv4 přidává následující možnosti:

- Podpora IPv6
- Vyšší úroveň bezpečnosti
- Preferuje multicast před broadcastem pro push módy

NTPv4 poskytuje následující možnosti:

- NTPv4 podporuje IPv6, což umožňuje synchronizaci času NTP přes IPv6.

```
Switch (config) # ntp server 2001: DB8: 0: 0: 8: 800: 200C: 417A version 4
```

- Bezpečnost je zlepšena oproti NTPv3. Protokol NTPv4 poskytuje celý bezpečnostní rámec založený na kryptografii veřejného klíče a standardních certifikátech X509.

- Pomocí specifických skupin multicast vysílání může NTPv4 automaticky vypočítat svou hierarchii časového rozdělení prostřednictvím celé sítě.

- NTPv4 automaticky konfiguruje hierarchii serverů tak, aby bylo dosaženo nejlepší časové přesnosti při nejnižších nákladech na šířku pásma. Tato funkčnost využívá lokální IPv6 multicast adresy.

SNMP

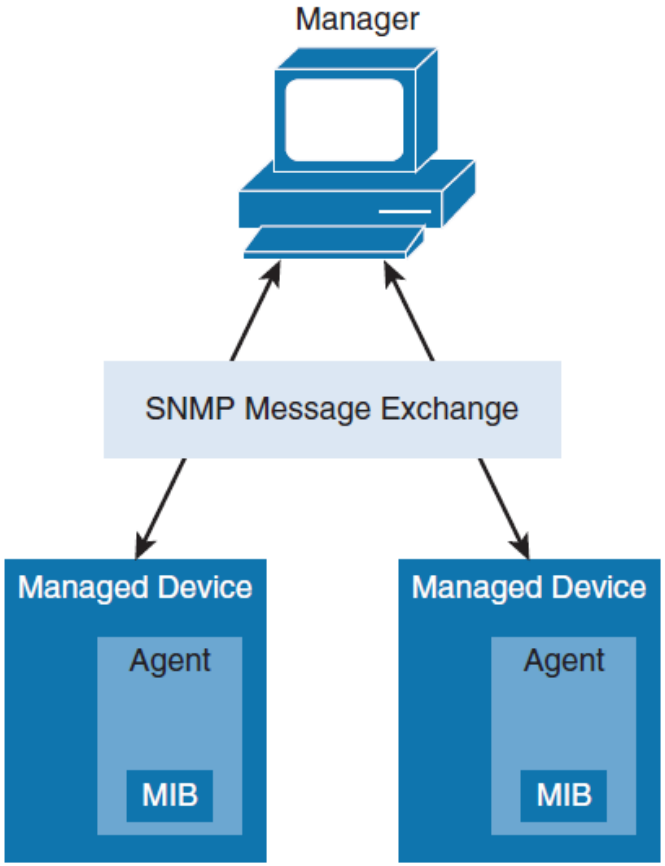
Obsah kapitoly SNMP

- Role SNMP
- Rozdíly mezi verzemi SNMP
- Praktická doporučení pro nastavení SNMP
- Konfigurační příkazy pro SNMP Version 3
- Verifikace konfigurace SNMP

Přehled SNMP

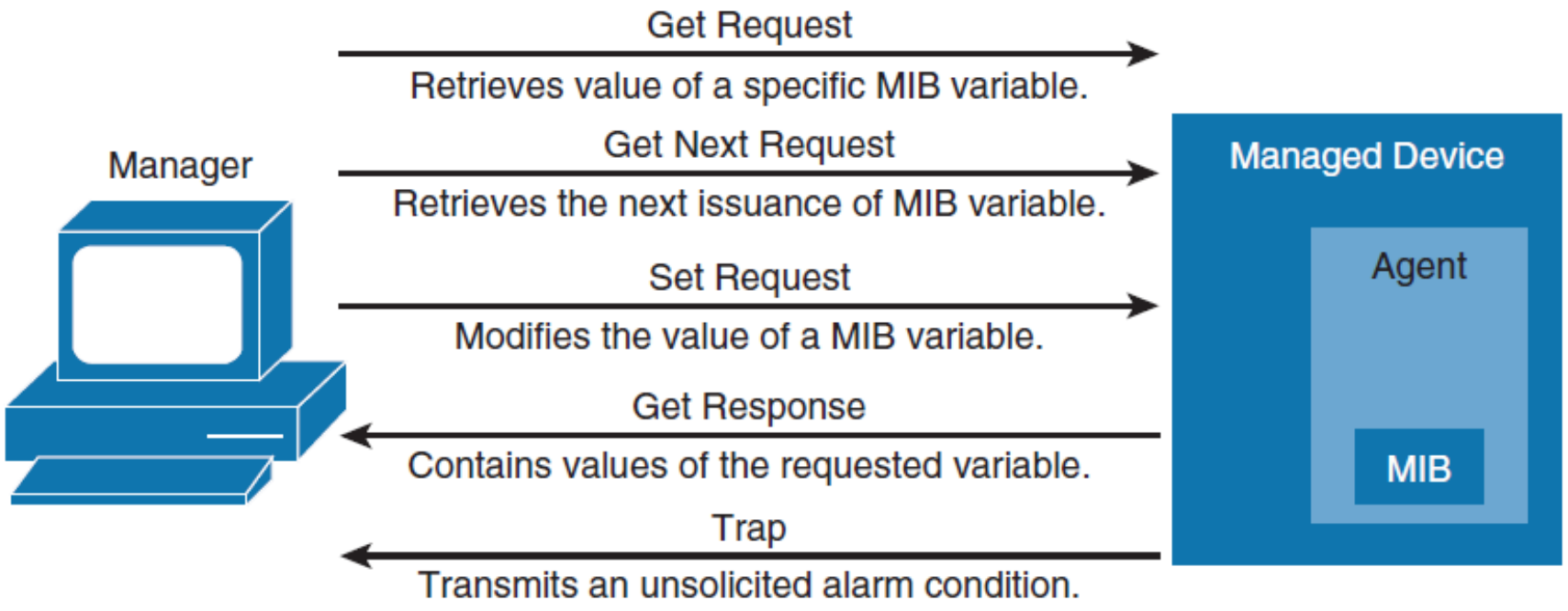
SNMP se sklásá ze dvou komponent:

- **Manager SNMP**, který periodicky dotazuje agenty SNMP na spravovaných zařízeních dotazem na zařízení pro data. Periodická výzva má nevýhodu: Mezi skutečným výskytem události a časem, kdy správce SNMP dotazuje data, dochází ke zpoždění.
- **Agenti SNMP** na spravovaných zařízeních shromažďují informace o zařízení a převádějí je do kompatibilního formátu SNMP podle MIB. MIB jsou kolekce definic spravovaných objektů. Agenti SNMP uchovávají databázi hodnot definic zapsaných v MIB.



- .

Procesy SNMP



SNMP Versions

- **Version 1**
  - Pět typů zpráv
    - Get Request,
    - Get Next Request
    - Set Request
    - Get Response
    - Trap.
  - Dnes řídce se vyskytující verze.
- **Version 2**
  - Zavádí nové zprávy
    - Get Bulk Request,
    - Inform Request, typ trapu očekávající potvrzení.
  - Version 2 přidala 64-bit čítač pro zvýšení rychlosti na rozhraních.
  - Přidal bezpečnostní model, který nebyl šířeji akceptovaný.
- ***Version 2c***
  - Community-based SNMP Version 2, je široce akceptovatelné
  - Community-based version of SNMP je nedostatečně bezpečné.
- **Version 3**
  - Doplnění bezpečnosti.

Bezpečnost SNMPv3

SNMPv3 podporuje následující tři úrovně bezpečnosti:

- **noAuthNoPriv**
  - Žádná autentizace a žádné šifrování není povoleno.
- **authNoPriv**
  - Autentizace je založena na Hashed Message Authentication Code (HMAC), MD5, nebo Secure Hash (SHA). Šifrování není poskytnuto.
- **authPriv**
  - Autentizace je doplněna šifrováním CBC-DES.

Chybná konfigurace z hlediska bezpečnosti

```
router ospf 100
 redistribute connected metric 100 metric-type 1 subnets
 redistribute static metric 100 metric-type 1 subnets
 passive-interface Ethernet0
 network 10.0.144.0 0.0.0.255 area 9
 area 9 nssa
!
no ip classless
logging buffered alerts
logging console informational
logging 10.192.17.3
access-list 1 permit 10.132.36.16
access-list 1 permit 10.132.37.11
access-list 1 permit 10.132.37.3
access-list 2 permit 10.0.0.0 0.255.255.255
snmp-server community public RW 1
snmp-server trap-source Loopback1
snmp-server packetsize 8192
snmp-server trap-authentication
snmp-server queue-length 50
snmp-server enable traps isdn
snmp-server enable traps config
snmp-server enable traps bgp
snmp-server enable traps frame-relay
snmp-server host 10.192.17.3 public tty snmp
```

Nejlepší praktiky SNMP

- Omezení přístupu na read-only.
- Použití práva write access s oddělenými kredity.
- Nastavení SNMP views na nezbytný přístup do MIB.
- Konfigurace ACLs pro omezení SNMP přístupu.
- Použití SNMPv3 autentizace, šifrování a kontroly integrity kdekoliv to lze.

Konfigurační kroky SNMPv3

- **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.

Konfigurace SNMPv3

```
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
```

Bezpečnostní konfigurace

- Skupina SNMPv3 je konfigurována s úrovní zabezpečení authPriv (**snmp-server group groupZ v3 priv**) a s uživatelem této skupiny (**snmp-serverZ userZ groupZ**) s hesly pro ověřování (**auth sha itsasecret**) a šifrování (**priv aes 256 anothersecret**).
- Příklad konfigurace pak povolí SNMP trapy pomocí příkazu **snmp-server enable traps**. Trapy jsou odesílány na NMS nebo ekvivalentní server; proto musí být konfigurován přijímající SNMP manažer. Z příkladu budou SNMPv3 trapy odeslány na adresu 10.1.1.50 (**snmp-server host 10.1.1.50 traps**) pomocí uživatele s úrovní zabezpečení **authPriv security level** (priv).
- Můžete také omezit události, pro které jsou odesílány trapy. V tomto příkladu jsou trapy omezeny na události související s CPU a zabezpečením portu (**cpu port security**).

Příkazy SNMP

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

Souhrn kapitoly 7

- Funkce AAA zahrnují autentizaci, autorizaci a účetnictví. Využití AAA je vyžadováno téměř ve všech sítích kampusu, protože zajišťuje a zajišťuje administrativní řízení a protokolování uživatelského přístupu k síťovým zařízením a samotné síti.
- Síť založená na identitě využívá protokoly 802.1X pro podporu mobility, zabezpečení, autentizace a autorizace uživatelů k síťovým prostředkům.
- Přesný čas je nezbytný pro služby zaznamenávání času v sítích kampusu, stejně jako mnoho bezpečnostních funkcí, jako je šifrování.
- Všechny přepínače Cisco Catalyst podporují NTP pro synchronizaci času.
- NTP obecně dosahuje přesnosti v milisekundách v sítích LAN.
- SNMP je light protokol, který nejen monitoruje a kontroluje zařízení, ale také podporuje upozornění na události.
- SNMPv3 je doporučení pro osvědčené postupy pro SNMP; nepoužívejte SNMPv2 (nebo v1), pokud je to možné (z důvodu nedostatku bezpečnostních funkcí).
- Bezpečnost v okolí SNMP musí být považována za součást implementačního plánu. Minimálně použijte ověřování a šifrování spolu s omezeným přístupem pro zápis a IP ACL, abyste omezili přístup k síti.

Chapter 7 Labs

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