ISA - Laboratorní cvičení č.5

Správa a monitorování sítě

Vysoké učení technické v Brně

`https://github.com/nesfit/ISA/tree/master/management`

Cíl laboratorního cvičení

- seznámit se s nástroji pro správu sítě
- naučit se práci s nástrojem SSH a správu klíčů
- naučit se pracovat s protokolem Syslog a nástrojem rsyslog
- konfiguraci nástrojů pro práci s protokolem SNMP
- otestovat práci se syslog a SNMP
- seznámit se s protokolem NetFlow
- naučit se pracovat s nástroji `nfsen` a `nfdump`

Pokyny

- Do zadání nepište, slouží pro další skupiny. PDF verzi zadání i šablony konfiguračních souborů lze najít v IS u předmětu ISA.
- Na konci laboratorního cvičení nezapomeňte na poslední bod, tj. na **Ukončení práce v laboratoři**!

1 Vzdálený terminál – SSH

Pracujte pod uživatelem `user`. Zkontrolujte, že pracujete pod uživatelem `user` příkazem `whoami`, který vypisuje jméno aktivního uživatele.

Namísto řetězce `<login>` používejte své studentské přihlašovací jméno.

1. Ukončete démona spravujícího hesla k použitým klíčům SSH:
   `killall gnome-keyring-daemon`.

2. **Bezpečné připojení na vzdálený počítač bez autentizačních klíčů.**

   (a) Přihlaste se příkazem `ssh` *`host`* na vzdálený počítač. Jako *host* použijte jméno `hNN`, kde NN je číslo stanice souseda. Uživatele není nutné specifikovat.

   (b) Příkazem `exit` nebo stiskem `Ctrl-D` spojení ukončete.

3. **Vytvoření veřejného a privátního klíče.**

   (a) Příkazem `ssh-keygen` vygenerujte implicitní klíč. Neměňte jeho název a zvolte heslo o délce alespoň osmi znaků.

   (b) Příkazem `ssh-keygen -N "" -f ~/.ssh/nopass -C <login>@nopass` vygenerujte autentizační klíč bez hesla.

   (c) Příkazem `ssh-keygen -N <heslo> -f ~/.ssh/pass -C <login>@pass` vygeneruje klíč s heslem jiným než pro výchozí klíč.

   (d) Ověřte obsah a přístupová práva u nově vzniklých souborů (ls -l ~/.ssh). Jak se liší práva mezi souborem s privátním a veřejným klíčem?

4. **Distribuce klíčů**

   (a) Všechny tři veřejné klíče si zkopírujte na vzdálený počítač do souboru `.ssh/authorized_keys`. (např. `cat ~/.ssh/*.pub | ssh hXX "cat >> .ssh/authorized_keys"`).
   Jaké heslo bylo nutné zadat?

   (b) Zkuste se znovu přihlásit na stejný vzdálený počítač. Jaké heslo bylo nyní nutné zadat? Zkuste zadat špatné heslo a pozorujte, které další klíče se použily. Při experimentech můžete také využít tzv. verbose režim ssh (`ssh -v`). Pro experimenty s identitou využijte přepínač `-i`.

5. **Konfigurace použití klíčů**

   (a) Stejným způsobem nakopírujte své veřejné klíče ještě na další počítač v laboratoři.

   (b) Na svém počítači vytvořte soubor ~/.ssh/config a pomocí textového editoru přidejte následující konfiguraci (hostname definujte jen pro jeden ze dvou strojů, na který jste kopírovali veřejné klíče):

```
Host hNN hNN.netlab.fit.vutbr.cz
 IdentityFile ~/.ssh/pass

Host *
 IdentityFile ~/.ssh/id_rsa
 IdentityFile ~/.ssh/nopass
```

   (c) Přihlaste se postupně na oba počítače. Které klíče se použily? Co se stane, když pro některý klíč zadáte špatné heslo?

6. **Omezení použití klíčů**

   Nyní bude naším cílem omezit použití klíče, který není chráněn heslem tak, aby pomocí něj bylo možné na vzdáleném serveru vykonat pouze konkrétní příkaz.

   (a) Přihlaste se na počítač, kam jste nakopírovali své veřejné klíče a upravte soubor s autorizovanými veřejnými klíči tak, že na začátek řádku s klíčem nopass (řádek poznáte tak, že končí řetězcem `<login>@nopass`) napíšete `command="ntpq -p"` (od původního obsahu oddělený jednou mezerou).

   (b) Na vzdáleném počítači spusťte službu NTP příkazem `systemctl start ntp` (jak root),

   (c) Odhlaste se ze vzdáleného počítače a znovu se na něj přihlaste příkazem `ssh hXX -i ~/.ssh/nopass`. Aplikovalo se omezené využití klíče?

7. **Pohodlné opakované použití klíče zabezpečeného heslem.**

   (a) Spusťte program `gnome-keyring-daemon`.

   (b) Připojte se ke vzdálenému počítači za použití klíče chráněného heslem.

   (c) Odhlaste se a znovu přihlaste. Museli jste zadávat znovu heslo?

8. **HTTP proxy pomocí SSH tunelu**

   (a) Navštivte stránku `www.fit.vutbr.cz`. Všimněte si vaší IP adresy v levém dolním rohu stránky.

   (b) Přihalšte se na fakultní server merlin pomocí SSH příkazu:

   ```
   ssh -D 8080 -N xloginNN@merlin.fit.vutbr.cz
   ```

   Parametr `-N` způsobuje, že se na serveru neotevře příkazový řádek.

   (c) V prohlížeči jděte v nastavení vyberte Advanced → Network → Settings ... a nastavte adresu pro SOCKS Host na 127.0.0.1 a port 8080, vyberte SOCKS verze 5. Ostatní možnosti ponechte prázdné.

   (d) Znovu navštivte stránku `www.fit.vutbr.cz`. Jaká je nyní zobrazena IP adresa?

2 Syslog

- Úkol:
  - Seznámit se s protokolem Syslog, který slouží pro přenos logovacích zpráv ze spravovaných zařízení. Pojmem Syslog je často označováno také programové vybavení implementující samotný přenos, třídění a ukládání zpráv na disk.
  - Rozdělte se do dvojic. V každé dvojici zvolte jednu stanici jako klient a druhou jako server a nakonfigurujte přeposílání veškerých Syslog zpráv z klienta na server. Mějte na paměti možné zneužití Syslog protokolu útočníkem a omezte na straně serveru příjem pouze na zprávy od daného klienta a klientovi zamezte příjem jakýchkoliv zpráv ze sítě.
  - Pro práci využijte nástroj `rsyslogd`, který bude sloužit jako server i klient. K otestování využijte nástroj `logger`.
  - Na klientovi následně omezte přeposílání pouze na zprávy konkrétního typu.
- Příkazy:
  - `rsyslogd(8)` – démon pro Syslog.

- `rsyslog.conf(5)` – Popis konfigurace rsyslog démona.
- `logger(1)` – Nástroj pro generování Syslog zpráv.
- `tcpdump(1)`

- Postup:

1. Rozdělte se do dvojic a určete server a klient stanici.
2. **Na serveru** povolte naslouchání na síťovém soketu. Do souboru `/etc/rsyslog.conf` přidejte:

   ```
   module(load="imudp")
   input(type="imudp" port="514")
   ```

3. **Na klientovi** zakažte programu rsyslogd naslouchat na síťovém soketu, tj. odeberte/zakomentujte v souboru `/etc/rsyslogd.conf` řádky:

   ```
   module(load="imudp")
   input(type="imudp" port="514")

   module(load="imtcp")
   input(type="imtcp" port="514")
   ```

4. **Na klientovi** nakonfigurujte rsyslog démona tak, aby odesílal veškeré zprávy z klienta na serveri pomocí UDP. Jako oddělovač použijte výhradně tabulátor nikoliv mezery. Do souboru `/etc/rsyslog.conf` přidejte následující pravidlo:

   ```
   *.*<TAB>@<doménové_jméno_serveru>:<číslo_portu>
   ```

5. **Na serveru i klientovi** restartujte Syslog démona:

   ```
   systemctl restart rsyslog
   ```

6. **Z klienta** ověřte správnou konfiguraci vygenerováním testovací Syslog zprávy pomocí nástroje `logger`:

   ```
   logger -d <obsah_zprávy>
   ```

7. Zpráva byla přeposlána na server, kde ji lze najít na konci souboru `/var/log/messages`.

   ```
   tail -f /var/log/messages | grep <doménové_jméno_klienta>
   ```

8. Na klientovi pokračujte v generování Syslog zpráv a na serveru sledujte příchozí Syslog zprávy pomocí `tcpdump`. Na jakém portu a jakým protokolem jsou Syslog zprávy zasílány.
9. Otevřete si manuálovou stránku rsyslog.conf a zjistěte, jaké zařízení a priority zpráv Syslog poskytuje.

   ```
   man rsyslog.conf
   ```

10. Ze znalosti zařízení a priorit nakonfigurujte klienta, aby posílal na server pouze zprávy při selhání autentizace, tj. upravte již existující pravidlo pro přeposílání veškerých zpráv na server v souboru `/etc/rsyslog.conf`. Nezapomeňte restartovat Syslog démona.

    ```
    Syntax pravidel: <facility>.<priority><TAB><action>
    ```

11. Ze serveru se následně pokuste o neúspěšné ssh připojení na klienta a podívejte se do souboru pro autentizační záznamy, jakou zprávu zaslal klient serveru.

3 SNMP

- Úkol:
  - Seznámit se s protokolem SNMP, který slouží pro přenos informací o stavu spravovaných zařízení (např. hodnot čítačů). Během tohoto cvičení si vyzkoušíte práci se synchronními SNMP událostmi, tj. model komunikace dotaz-odpověď.
  - Každou stanici nakonfigurujte tak, aby lokálně umožňovala čtení i zápis SNMP proměnných a při vzdáleném přístupu k SNMP proměnným pouze čtení.
- Příkazy:
  - `snmpd(8)` – SNMP démon.
  - `snmpd.conf(5)` – Popis konfigurace SNMP démona.
  - `snmpwalk(1)` – Nástroj pro procházení a získávání podstromu hodnot SNMP proměnných.
  - `snmpget(1)` – Nástroj pro získání hodnoty SNMP proměnné.
  - `snmpset(1)` – Nástroj pro změnu hodnoty SNMP proměnné.
- Postup:
  1. Otevřete si soubor `/etc/snmp/snmpd.conf` pro editaci a postupně v něm proveďte základní konfiguraci:
     (a) Zvolte si vlastní název komunity - např. PUBLIC
     (b) Nastavte agenta, aby poslouchal na všech lokálních adresách:
     `agentAddress udp:0.0.0.0:161`
     (c) Namapujte sítě, ze kterých bude možné přistupovat k SNMP hodnotám. Povolte počítačům v učebně přístup pro čtení `rocommunity <COMMUNITY> 10.10.10.0/24` a lokálnímu počítači i zápis `rwcommunity <COMMUNITY> localhost`
     (d) V konfiguračním souboru ještě nastavte SNMP proměnnou `sysContact` na Vámi vymyšlenou hodnotu. Takto nastavená proměnná bude vždy pouze pro čtení bez ohledu na nastavení přístupových pravidel. Zakomentujte nastavení proměnné sysLocation.
     (e) Nastavte proměnnou sysLocation pomocí příkazu `snmpset`
     `snmpset -v1 -c <nazev_komunity> localhost 1.3.6.1.2.1.1.6.0 s <retezec>`
  2. Restartujte službu `snmpd`.
     ```
     systemctl restart snmpd
     ```
  3. Ověřte, že služba běží. Na kterých portech naslouchá?
     ```
     systemctl status snmpd
     ss -atun | grep snmpd
     ps -ax | grep snmpd | grep -v grep
     ```
  4. Příkazem `snmpwalk` přečtěte podstrom `system` a zjistěte jak váš soused nastavil proměnné `sysLocation` a `sysContact`. Např.:
     ```
     snmpwalk -v 1 -c <nazev_komunity> <domenove_jmeno>
     snmpwalk -v 1 -c <nazev_komunity> <domenove_jmeno> 1.3.6.1.2.1.1
     snmpwalk -v 1 -c <nazev_komunity> <domenove_jmeno> 1.3.6.1.2.1.1.6.0
     snmpwalk -v 1 -c <nazev_komunity> <domenove_jmeno> 1.3.6.1.2.1.1.4.0
     ```

5. Pokud znáte OID (Object Identifier) proměnné v databázi MIB (Management Information Base), pak můžete získat hodnotu této proměnné přímo pomocí `snmpget`, např. počet bytů odeslaných na rozhraní eth0:

```
snmpwalk -v 1 -c <nazev_komunity> \
         <domenove_jmeno_souseda> 1.3.6.1.2.1.2.2.1.16
snmpget -v 1 -c <nazev_komunity> \
         <domenove_jmeno_souseda> 1.3.6.1.2.1.2.2.1.16.2
```

6. Zkuste nastavit svému sousedovi proměnnou sysLocation pomocí nástroje `snmpset`:

```
snmpget -v 1 -c <nazev_komunity> <domenove_jmeno_souseda> 1.3.6.1.2.1.1.6.0

snmpset -v 1 -c <nazev_komunity> <domenove_jmeno_souseda> \
  1.3.6.1.2.1.1.6.0 s <nova_hodnota>
```

7. Proč předchozí pokus o nastavení proměnné sousedovi selhal? Zkuste nyní nastavit svoji proměnnou popisující název Vašeho počítače pomocí nástroje `snmpset`:

```
snmpget -v 1 -c <nazev_komunity> localhost 1.3.6.1.2.1.1.6.0

snmpset -v 1 -c <nazev_komunity> localhost \
  1.3.6.1.2.1.1.6.0 s <nove_jmeno_pocitace>

snmpget -v 1 -c <nazev_komunity> localhost 1.3.6.1.2.1.1.6.0
```

4 NetFlow

- Úkol:
  - Seznámit se možnostmi měření provozu pomocí NetFlow. NetFlow slouží pro přenos statistik o jednotlivých tocích dat vznikajících při komunikaci po síti. Záznamy NetFlow, s nimiž budete během cvičení pracovat, jsou pořízeny z napojení sítě VUT a anonymizovány. V druhé části úkolu budete pracovat s daty pořízenými sondou FlowMon.
  - Seznámit se s nástrojem `nfsen`, který graficky zobrazuje záznamy NetFlow ve webovém prohlížeči. Seznámit se s nástrojem `nfdump`, který slouží k dotazování na uložená data NetFlow.
- Příkazy:
  - `nfdump`
- Postup:
  1. Naučte se používat nástroj `nfdump`, který slouží k dotazovaní se nad záznamy NetFlow.
     (a) Na Vašem počítači se v adresáři `/root/isa5/netflow` nachází anonymizovaná kolekce NetFlow dat. Tento adresář bude vstupem programu `nfdump`, který využijte ke kladení dotazů nad NetFlow daty.
     (b) Prostudujte manuálovou stránku nástroje `nfdump`.
     (c) Dotažte se na následující statistiky. TOP 20 IP adres podle počtu přenesených bajtů.
        - V manuálové stránce si najděte, co dělají přepínače `-R, -s, -n`.
        - Nezapomeňte, že zpracovávanáý data jsou relativně objemná. Dosažení výsledku tedy chvili potrvá.

(d) Zjistěte, jak velké datové přenosy připadají na jednotlivé protokoly. (Statistika protokolů)
   - Všimněte si rozdílů v podílech podle toků a podle přenesených bajtů.

(e) Na základě získaných statistik se zamyslete nad velikostí sítě.

(f) Vyfiltrujte si toky se zdrojovou IP 162.35.0.190. Zaměřte se na čísla portů. Je aktivita zdroje něčím podezřelá?[1]

2. Přihlašte se na sondu FlowMon, běžící ve společnosti Flowmon Networks

(a) webová stránka: `https://demo.invea.cz`, login: `demo`, heslo: `demo`. Neměňte žádné nastavení.

(b) Seznamte se s jednotlivými stránkami, které vytváří program `nfsen`.
   - **Přehled** – Stručný přehled statistik pro provoz
   - **Zdroje** – Zdroje NetFlow dat
   - **Profily** – Nastavení pro specifický typ provozu
   - **Analýza** – Podrobné informace o nasbíraných datech za určité období
   - **Reporty** – Vytvoření reportu za určité období
   - **Alerty** – Hlášení pro specifické události

(c) Úkoly:
   - Zjistěte, jaké IM protokoly se v této síti používají.
   - Zjistěte, jaké P2P protokoly se v této síti používají.
   - Který protokol transportní vrstvy se používá nejčastěji? Jaký má přibližně podíl na celkovém provozu v síti? Jak se tento podíl mění v průběhu dne, týdne? Zamyslete se, proč tomu tak je.

- Pro zájemce:
   - Zájemci si mohou na stránce `https://demo.invea.cz` projít ostatní moduly vystavěné nad protokolem. Neměňte žádné nastavení.

Ukončení práce v laboratoři

- Pod uživatelem `root` spusťte dávku `/root/isa5/clean` pro zrušení vytvořených konfiguračních souborů a pro vypnutí počítače.

[1] Velké množství krátkých toků se stejným cílem a postupně stoupajícími čísly portů ukazuje na vertikální skenování.

Teorie

1 Secure Shell

Základní funkcí protokolu Secure Shell (SSH) [7] je umožnění bezpečného přístupu ke vzdálenému počítači přes nezabezpečenou síť. Díky tomu, že je protokol SSH navržen obecně, lze pomocí něj zabezpečovat i další služby, jako je např. X Window, přístup ke vzdálenému souborovému systému (SFTP, sshfs, scp), tunelování portů TCP apod. Protokol SSH zajišťuje šifrování dat, autentizaci, integritu dat a volitelně také kompresi přenášených dat.

V rámci předmětu ISA se zaměříme pouze na malou část možností, které protokol SSH přináší. V rámci cvičení si vyzkoušíme protokol SSH pro terminálový přístup ke vzdálenému stroji a ukážeme si využití přihlašování ke vzdálenému počítači pomocí klíčů. Dále si vyzkoušíme, jak využít SSH k vytvoření jednoduché HTTP proxy.

Jednou z nejčastěji používaných aplikací pro využití protokolu SSH je sada programů OpenSSH. Balíček programů obsahuje kromě klienta `ssh` i serverovou aplikaci `sshd`, program pro generování SSH klíčů `ssh-keygen`, agenta pro usnadnění práce s SSH klíči `ssh-agent` a další. My budeme předpokládat, že na počítačích, na které se budeme snažit připojit již běží SSH server `sshd`. Konfigurace tohoto programu je nad rámec tohoto manuálu. Zájemci mohou nalézt podrobnější informace v manuálové stránce `sshd_config(5)`, či v jiných návodech.

1.1 Připojení ke vzdálenému počítači

Pro připojení se k počítači pojmenovaném *h01* a otevření příkazového řádku na vzdáleném stroji je možné použít příkaz:

```
ssh h01
```

Příkaz ssh má celou řadu parametrů, které jsou detailně popsány v manuálové stránce `ssh(1)`. Z těch nejčastěji používaných zmíníme alespoň změnu uživatelského jména (`-l`), specifikování TCP portu vzdáleného serveru (`-p`), zvýšení výřečnosti programu (`-v`, tento parametr lze použít i vícekrát), zapnutí tunelování protokolu X Windows (`-X`, `-Y`) a přesměrování portů (`-L`). Často zadávané parametry se specifikují v konfiguračním souboru `~/.ssh/config`. Popis tohoto souboru obsahuje manuálová stránka `ssh_config(5)`.

Po připojení ke vzdálenému serveru jsou informace o použitém spojení dostupné např. v rámci proměnných prostředí. Proměnné prostředí související s protokolem SSH je možné zobrazit příkazem `env | grep SSH`. Následující výpis ukazuje příklad proměnných po připojí k serveru `merlin` protokolem IPv6:

```
local $ ssh merlin6.fit.vutbr.cz
merlin $ env | grep SSH
SSH_CLIENT=2001:67c:1220:80c:e138:4d11:c04c:c675 54514 22
SSH_TTY=/dev/pts/30
SSH_CONNECTION=2001:67c:1220:80c:e138:4d11:c04c:c675 \
  54514 2001:67c:1220:8b0::93e5:b013 22
```

Z výpisu vidíme, že připojení bylo realizováno na IPv6 adresu `2001:67c:1220:8b0::93e5:b013` z počítače s adresou `2001:67c:1220:80c:e138:4d11:c04c:c675`. Byl použit zdrojový port č. 54514, na straně serveru byl použit standardní protokol 22. Po připojení využíval vzdálený uživatel terminál č. 30. Odhlášení ze vzdáleného počítače probíhá standardními prostředky pro ukončení shellu, např. příkaz `exit`, nebo vložení konce souboru klávesovou zkratkou `Ctrl-D`.

1.2 Kopírování souborů mezi počítači

Pro kopírování souborů protokolem SSH je často využívaná utilita `scp`. Cesta na vzdáleném serveru je specifikována v následujícím formátu: `uživatel@jménoserveru:cesta`. Následující příkaz zkopíruje lokální soubor isa na vzdálený počítač h01 do složky fit umístěné v domovské složce uživatele student.

```
scp isa student@h01:fit/
```

1.3 SSH jako proxy a tunelování portů

Díky velmi obecné implementaci lze protokolem SSH tunelovat jakýkoli typ provozu. Je tedy možné např. zajistit šifrování provozu po určitou část komunikace, konkretně po stanici, ke které se připojujeme SSH protokolem, nebo přesměrovat provoz z některého portu na jiný port. Následujcí příkaz nám otevře SSH spojení, které pak můžeme použít jako proxy ve webovém prohlížeči.

```
ssh -D 12345 -N student@sshserver
```

Po autentizaci stačí pak v prohlížeči zadat jako nastavení proxy server localhost s portem 12345 a provoz bude směrován nejprve na localhost, poté půjde ssh spojením na sshserver, kde bude převeden na běžný HTTP/HTTPS provoz a poslán dále.

1.4 Klíče protokolu SSH

Klíče protokolu SSH mají několik výhod. Díky nim není nutné posílat heslo pro přístup ke vzdálenému stroji přes síť, byť v šifrované podobě. Délka používaných klíčů znesnadňuje potenciálnímu útočníkovi útok hrubou silou, protože síla klíče bývá typicky vyšší než síla běžně používaných hesel. Ve spojení s agentem pro správu klíče může uživatel přistupovat ke vzdálenému serveru bez nutnosti opakovaného zadávání hesla. Agent může být nastaven, aby pro použitý klíč nevyžadoval heslo po zbytek sezení, či po určitý počet minut.

SSH klíč se obvykle generují utilitou `ssh-keygen`. Po spuštění bez parametrů je vytvořen pár klíčů algoritmem RSA o délce 2048 B. Uživateli je nabídnuto umístění souboru, které může změnit. Dále je uživatel požádán o zadání passfráze, od které se očekává, že bude silnější než heslo. Parametr `-t` nastavuje jiný typ šifrovacího algoritmu, parametr `-b` mění délku klíče, parametrem `-f` specifikuje umístění souboru, parametr `-C` upravuje popis klíče, další parametry popisuje manuálová stránka `ssh-keygen(1)`. Následující příklad vygeneruje klíč o délce 4096 B, který nebude chráněn passfrázi:

```
ssh-keygen -t rsa -b 4096 -N "" -f ~/.ssh/nopass -C nopass
```

Po vytvoření klíčů vzniknou dva soubory. Ten který je bez přípony `.pub` je soukromý klíč, který by měl zůstat tajný a uživatel, který jej vytvořil by jej neměl dále distribuovat. Soubor s příponou `.pub` je určen pro další distribuci, protože data zašifrovaná tímto klíčem dešifruje pouze tajný soukromý klíč.

1.5 Konfigurace použití klíčů

Nejdříve je potřeba distribuovat soubor s veřejným klíčem na vzdálený počítač. K tomu slouží např. program `scp`. Každý z uživatelů si může specifikovat sadu klíčů pro přístup k danému stroji v souboru `~/.ssh/authorized_keys`. Nový klíč do tohoto souboru přidáme např. takto (všimněte si, že se klíč přidává na konec souboru pomocí `>>`):

```
cat id_rsa.pub >> ~/.ssh/authorized_keys
```

Nyní se již můžeme připojit ke vzdálenému počítači pomocí vytvořených klíčů. Pokud jsme klíč na lokálním počítači umístili do výchozího umístění, je klíč použit automaticky. Pokud jsme zvolili jiné umístění, je potřeba program `ssh` informovat o umístění klíče parametrem `-i`, nebo volbou `IdentityFile` v konfiguračním souboru. Informace o hledaných klíčích jsou zobrazeny po použití parametru `-v`.

2 Syslog

Protokol Syslog [3] popisuje způsoby a formát pro záznam systémových událostí. Je využíván především v unixových systémech pro záznam událostí jednak prvků součástí systém, jednak pro události běžících programů.

Součástí záznamů bývá zpravidla časová značka, zařízení (facility), závažnost záznamu (severity) a samotný záznam. V některých implementacích se může objevit i doménové jméno či IP adresa. Záznam může být uložen lokálně (např. soubor `/var/log/messages`) nebo lze záznamy přeposílat na vzdálený server. Takové záznamy se pak objeví v syslogu daného serveru. Součástí záznamů může být i doménové jméno resp. IP adresa původce záznamu.

V položce facility je uvedena součást systému (jádro, autentizační zprávy,apod.) nebo přímo název programu, případně jeho PID. Kompletní seznam zařízení lze najít ve starším RFC 3164 [4].

Závažnost záznamu je rozdělena na 8 úrovní od nejzávažnější (0) po ladící informace (7). Obvykle jsou v syslogu zaznamenávány událostí od úrovně varování (4). Úplný seznam úrovní pak vypadá takto:

Závažnost		Popis
0	EMERGENCY	Nejvyšší závažnost, systém je nestabilní
1	ALERT	Událost vyžadující okamžitou akci.
2	CRITICAL	Kritická chyba systému (např. HW chyby)
3	ERROR	Chyba programu.
4	WARNING	Varování programu.
5	NOTICE	Důležité informační záznamy (např. změna nastavení)
6	INFORMATIONAL	Informační zprávy nevyžadující speciální pozornost
7	DEBUG	Ladicí informace

3 Simple Network Monitoring Protocol

SNMP [1] je protokol určený k monitorování stavu zařízení. Protokol podporuje nejen čtení provozních paramterů připojených stanic (využití paměti, zatížení CPU apod.), ale i nastevní některých paramterů těchto stanic. Protokol je provozován nad UDP na portech 161 a 162 pro asynchronní zprávy (traps). V případě potřeby přenášení větších bloků informací lze SNMP provozovat i nad TCP [6].

Stanice zapojené v síti dělíme na "správce"(Network Management station, NMS0 a sledovaná zařízení. Správcovské stanice slouží ke shromažďování informací ze sledovaných zařízení a jejich případné prezentaci. Požadované informace mohou získat buď synchronní komunikací způsobem požadavek-odpověď, nebo mohou získávat informace od sledovaných stanic asynchronně pomocí tzv. traps. Tyto zprávy jsou odesílány přímo sledovanými zařízeními. Sbírané informace jsou pak uloženy v databázi MIB (Management Information Base). Každa položka je identifikována pomocí OID (Object Identifier), podobně jako v LDAP. Příkladem může být položka sysLocation uložená pod OID `1.3.6.1.2.1.1.6.0`. Seznam OID podle standrardu RFC 1213 [5] lze najít na stránce

`http://www.oidview/mibs/0/RFC1213-MIB.html.`

Sledované stanice provozují v síti SNMP programy pro získávání potřebných informací tzv. SNMP agenty. Ti sbírají data a zpracovávají požadavky z NMS.

Zařízení v síti SNMP jsou rozdělena do logických celků, komunit, identifikovaných textovým řetězcem. Komunita slouží k autentizaci zařízení v síti a definuje přístup k informacím MIB. Určuje tedy, které hodnoty a vlastnosti je povoleno v dané komunitě sledovat, případně nastavovat. Příklad takového nastavení může být následující:

```
rocommunity public 10.0.0.0/24
rocommunity public 192.168.0.0/24 1.3.6.1.2.1.1
rwcommunity public localhost
```

Konfigurace v příkladu umožňuje zařízením v síti 10.0.0.0/24 přístup k MIB komunity public pouze pro čtení. Při pokusu o zápis do jakékoli proměnné skončí akce neúspěchem, že proměnná neexistuje. Pro stanice v síti 192.168.0.0/24 je přístup pro čtení omezen pouze na základní szstémové informace (podstrom system). Při lokálním přístupu však budou záznamy přístupné i pro zápis.

4 Cisco NetFlow

Formát záznamu NetFlow umožňuje ukládání informací o IP provozu na síťovém zařízení. Formát byl vytvořen firmou Cisco jako funkcionalita jejich zařízení. Ve verzi 9 pak byl standartizován jako RFC 3954 [2].

NetFlow záznamy jsou využívány k monitorování a analýze provozu v síti. Data obsahují záznamy o IP adresách, portech, počty přenesených paketů či bajtů apod. Data jsou uchovávána pro tzv. toky neboli flow. Toky jsou identifikovány pěticí zdrojová a cílová IP adresa, zdrojový a cílový port a protokol transportní vrstvy. Tok je chápán jako jednosměrná komunikace.

Infrastruktura pro sběr a uchovánvání NetFlow dat sestává z kolektoru a sond. Kolektorem je míněna stanice, která sbírá data ze sítě a provádí jejich uchování. Sondou pak může být jakékoli zařízení ať už směrovač, přepínač nebo zařízení přímo vyhrazené pro tuto činnost. Sběr dat může probíhat pro všechny toky, nebo může být prováděno vzorkování či agregace.

Reference

[1] J.D. Case, M. Fedor, M.L. Schoffstall, and J. Davin. Simple Network Management Protocol (SNMP). RFC 1157 (Historic), May 1990.

[2] B. Claise. Cisco Systems NetFlow Services Export Version 9. RFC 3954 (Informational), October 2004.

[3] R. Gerhards. The Syslog Protocol. RFC 5424 (Proposed Standard), March 2009.

[4] C. Lonvick. The BSD Syslog Protocol. RFC 3164 (Informational), August 2001. Obsoleted by RFC 5424.

[5] K. McCloghrie and M. Rose. Management Information Base for Network Management of TCP/IP-based internets:MIB-II. RFC 1213 (Standard), March 1991. Updated by RFCs 2011, 2012, 2013.

[6] J. Schoenwaelder. Simple Network Management Protocol Over Transmission Control Protocol Transport Mapping. RFC 3430 (Experimental), December 2002.

[7] T. Ylonen and C. Lonvick. The Secure Shell (SSH) Transport Layer Protocol. RFC 4253 (Proposed Standard), January 2006.