CSIRT.SK
www.csirt.gov.sk

Linux Hardening v1

Príručka administrátora

Obsah

1. Všeobecné informácie 1
1.1. Zoznam definícií 1
1.2. Ako používať túto príručku 1
1.3. Hardening 2
1.4. Všeobecné princípy 2
2. Všeobecná konfigurácia operačného systému 3
2.1. Inštalácia systému 3
2.2. Nastavenie prístupových oprávnení 6
2.3. Konfigurácia siete 8
2.3.1. Hardening sieťovej prevádzky 8
3.3.2. Hardening koncových systémov 15
3.4. Pravidelná údržba 17
2.4.1. Auditné záznamy 17
2.4.2. Bežná údržba systému 20
3. Red Hat Enterprise Linux (RHEL) 5 25
3.1. Konfigurácia po inštalácií 25
3.2. Zabezpečenie používateľského prístupu 26
3.3. Ochrana pred riskantným správaním aplikácií 27
3.4. Konfigurácia siete na RHEL5 28
4. Debian 30
4.1. Drobná konfigurácia Debian-u 30
4.2. Pravidelná údržba 31
5. Konfigurácia vybraných aplikácií a služieb 32
5.1. Apache 2.2 32
5.2. MySQL 35
5.3. SSH 37
Referencie 39

1. Všeobecné informácie

Rozsah tejto príručky sa zameriava na operačné systémy založené na Linuxovom jadre. Pre systémy **Windows** odporúčame použiť nástroj **Security Compliance Manager**, dostupný na adrese tu: http://technet.microsoft.com/en-us/library/cc677002.aspx . Jedná sa o automatizovaný nástroj na aplikovanie bezpečnostnej politiky na systém.

1.1. Zoznam definícií

OS	Operačný Systém
APT	Advanced Persistent Threat – pokročilá vytrvalá hrozba. Útočník ktorý sa dlhodobo snaží preniknúť do inštitúcie.
cron	cron je Unixový systémový nástroj, ktorý vykonáva opakované činnosti
démon	softvér spustený ako služba, spustený na pozadí systému
kernel	jadro operačného systému
whitelist	spôsob kontroly prístupu, ktorý zakazuje všetky vstupy okrem vyslovene povolených
blacklist	spôsob kontroly prístupu, ktorý povoľuje všetky vstupy okrem vyslovene zakázaných
stateful	stateful firewall je taký, ktorý drží kontext správ a vie rozpoznať súvisiace správy.

1.2. Ako používať túto príručku

Táto príručka slúži ako zoznam odporúčaní, ktoré je vhodné aplikovať aby sa znížilo riziko úspešného útoku na daný linuxový systém. Je písaná pre všeobecné použitie a s bližšími detailmi pre distribúcie Debian a RHEL 5. Príručka sa používa tak, že sa aplikujú potrebné úpravy zo všeobecnej sekcie, následne sa aplikujú vybrané časti zo sekcie pre konkrétnu distribúciu.

Každé odporúčanie je označené na stupnici dôležitosti od 1 (najnižšia dôležitosť) až po 3 (najvyššia dôležitosť). Dôležitosť súvisí s požadovanou úrovňou zabezpečenia systému. Aplikovať zmeny označené úrovňou 3 sa odporúča na všetkých systémoch, zmeny označené 1 sa odporúča iba na systémoch s podstatne zvýšenou úrovňou bezpečnosti.

V jednotlivých odsekoch sa môžu vyskytnúť buď nastavenia, ktoré musí administrátor aplikovať v grafickom prostredí (hlavne počas inštalácie). Tieto sú väčšinou označené **tučným** písmom. Príkazy do príkazového riadku sú označené znakom **#**. Príkazy sa vkladajú bez tohto znaku. Pokiaľ je zaznamenaný skript, je označený prvým riadkom **#!/bin/bash**.

1.3. Hardening

Aplikácie ako webové servery, poštové servery alebo rôzne operačné systémy sú určené na mnohé činnosti a do mnohých prostredí. Často sa nevyužívajú všetky funkcie týchto aplikácií, avšak ich funkcionalita je stále dostupná. To rozširuje tzv. „attack surface“[1] aplikácie, čo priamo zvyšuje riziko napadnutia aplikácie (systému).

Cieľ hardeningu je zredukovať zraniteľnú „plochu“ aplikácie bezpečnou konfiguráciou, aby boli omnoho robustnejšie a odolnejšie voči útokom. Spočíva v odstránení množstva zraniteľných bodov z aplikácie blokovaním častí programov, ktoré nie sú potrebné, prípadne oklieštení funkčnosti určitých častí programu, dôsledkom čoho je systém (resp. aplikácia) omnoho odolnejšia voči útokom.

Odporúčané konfigurácie sa menia podľa aplikácií, operačných systémoch kde sú aplikácie spustené a aj verzií inštalovaných aplikácií.

1.4. Všeobecné princípy

Všeobecný princíp pri hardeningu je minimalizovať miesta, o ktoré by sa útočník mohol zachytiť pri počiatočných fázach útoku.

Takéto všeobecné princípy sú:

- Šifrovanie všetkej citlivej komunikácie
- Spúšťanie iba nevyhnutných aplikácií
- Rozloženie služieb na odlišné servery
- Prideľovanie minimálnych potrebných prístupových práv
- Časté aktualizácie
- Blokovanie nepotrebných častí (systému, programov)
- Použiť whitelist namiesto blacklistu
- Zabezpečenie prístupovej siete (IDS, IPS)
- Detekcia škodlivého softvéru (antivírus, kontrola na prítomnosť rootkitov, ...)
- Auditné záznamy (ukladanie a analýza)

Na základe týchto pravidiel boli odvodené odporúčané konfigurácie, ktoré mnohonásobne sťažujú neoprávnený prienik do systému, prípadne eskalovania prieniku na prevzatie kontroly nad systémom.

[1] Attack surface je kód, ktorý môžu vykonať neautorizovaní používatelia, tzn. všetky vstupné body do aplikácie.

2. Všeobecná konfigurácia operačného systému

Táto sekcia obsahuje všeobecné odporúčania a mechanizmy konfigurácie operačných systémov. Pre konkrétnejšie informácie pre systémy Debian a Red Hat prejdite aj nasledujúcu sekciu. V prípade iných operačných systémov je možné použiť tieto princípy ako odporúčania pre následné vyhľadanie korektnej konfigurácie daného OS.

Niektoré z postupov nie je možné zaznamenať ako skript a je nutné ich vykonať manuálne. Jedná sa hlavne o činnosti súvisiace s konfiguráciou BIOS, formátovania partícií a niektorých špecifík konkrétnych distribúcií.

2.1. Inštalácia systému

Tieto úkony sa týkajú zabezpečenia systému po fyzickej stránke a na najnižších úrovniach operačného systému.

Offline inštalácia *2*

Nie je odporúčané pripájať systém do internetu pred tým, ako je to potrebné. Tzn. pred nastavením firewallu a ochránením prítomných služieb, keďže existuje hrozba napadnutia systému automatizovaným útočným nástrojom alebo inou osobou, očakávajúcou takúto akciu (APT).

Je vhodné nakonfigurovať proxy server na balíčky, viac v sekcií „Používanie proxy servera na balíčky“.

Nastavenie hesla v BIOS *1*

Nastavenie tejto ochrany závisí od verzie BIOS / UEFI zavádzača. Je potrebné nastaviť heslo, aby bolo možné spustiť server iba pre fyzicky prítomné osoby.

Toto **nastavenie je obmedzujúce** v prípade že je potrebné vykonať reštart vzdialene.

Nastavenie hesla pre bootloader LILO/GRUB *1*

Bootloader je ďalší z procesov, ktoré sa vykonávajú pri inicializácií systému. Pridaním hesla do bootloaderu sa obmedzí možnosť reštartovať systém cez internet a je potrebné lokálne prevedenie za prítomnosti klávesnice a monitoru.

Implementácia pre LILO

Je potrebné upraviť súbor **/etc/lilo.conf**, do ktorého treba vložiť riadky „password“ a „restricted“ za názov boot image. Nahradiť heslo „**heslo**“ za požadované heslo.

Príklad:

```
image=/boot/2.2.14-vmlinuz
label=Linux
read-only
password=heslo
restricted
```

Implementácia pre Grub 2

V tejto príručke sa uvádza iba konfigurácia pre GRUB verzie 2.0, ktorá sa v moderných operačných systémoch nachádza cca. od roku 2010. Konfigurácia sa líši od verzie 1.0.

V adresári **/etc/grub.d/** sa nachádzajú súbory vykonávajúce nasledujúce funkcie:

00_header
: Ochrana spúšťania OS heslom.

05_debian_theme
: Nastavenie farieb a zobrazenia pozadia a textu.

10_linux
: Lokalizacia linuxového jadra na základe výsledkov z príkazu "lsb_release".

20_memtest86+
: Ak existuje súbor /boot/memtest86+.bin, je začlenený do menu.

30_os-prober
: Vyhľadanie OS na iných partíciách a začlenenie ich do menu.

40_custom
: Šablóna na vkladanie vlastných položiek menu ktoré budú vložené do **grub.cfg** po vykonaní príkazu "**update-grub**". Tento a každý ďalší vlastný súbor musí byť nakonfigurovaný ako spustiteľný, aby bolo umožnené ho vložiť do grub.cfg.

Nastavenie hesla pre GRUB2

Zadáme príkaz a zadáme heslo, ktoré chceme zahashovať.

```
# grub-mkpasswd-pbkdf2
```

Výsledok bude vyzerať nasledujúco:

```
Your PBDKF2 is
grub.pbdkf2.sha512.10000.2CCE056FBEC295D14A25F0B54707D134DBD8F7AB8EE2F290A0
E4108095651A5EA02DD75F902CE80E1B0D13FFA4313D638EC33DACF1716EA8736DF2EAD7D1E
2810B0961CF4348345B67A6FAC0BB9EC1A97D7501C2A92028226
```

Nasledujúci záznam vložíme na koniec súboru 00_header, kde ho priradíme používateľovi. (reťazec "user" nahradíme za meno používateľa a "GRUB_PASSWORD" nahradíme za reťazec z výstupu grub-mkpasswd-pbkdf2)

```
"
cat << EOF
set superusers="user"
password_pbkdf2 user GRUB_PASSWORD
EOF
"
```

Aktualizujeme nastavenia cez zadanie príkazu `update-grub`.

Aktivovať Execute Disable (XD) alebo No Execute (NX) 1

Tento parameter slúži na blokovanie vykonania kódu s granularitou podľa stránok v pamäti. Jedná sa o nastavenie BIOS-u, ktoré Intel procesory označujú tento parameter ako XD, AMD ako NX.

Parameter by mal byť nastaviteľný v sekcií bezpečnosti (Security), alebo Rozšírených nastavení BIOS (Advanced BIOS Features).

Rozdelenie partícií *3*

Pre obmedzenie rozšírenia útoku do iných častí systému je potrebné fyzicky rozdeliť súborový systém do viacerých partícií. Je potrebné vytvoriť samostatnú partíciu pre:

/

Koreňový adresár linuxovej súborovej štruktúry. Odporúčaná veľkosť pre používateľské systémy je 15-20 GB, pre servery postačuje 5 až 10 GB, podľa potreby použitia adresárov.

/boot

Adresár, ktorý uchováva súbory potrebné na zavedenie systému. Je rozumné tento adresár vložiť na začiatok disku, kvôli najrýchlejším prístupovým časom a vďaka tomu aj rýchlemu štartu systému-

/tmp

Adresár na uchovávanie dočasných súborov. Rozsah zložky je potrebné nastaviť podľa zamerania serveru. Pokiaľ server nepočíta s prácou s veľkými súbormi, zložka môže ostať malá. Pre veľké súbory je odporúčaná veľkosť cca 10 GB. Je rozumné tento adresár umiestniť na začiatok disku (za adresár /boot) kvôli častému čítaniu a zápisom.

/var

Táto zložka obsahuje súbory ako email, webstránky, zálohy a podobne. Podľa použitia serveru je potrebné prideliť rozsah. Pokiaľ neprevádzkujeme žiadne služby, postačuje 2-4 GB, inak podľa potreby služieb.

/var/log

Táto zložka obsahuje textové záznamy aktivity. Metrika na určenie veľkosti partície je *[priemerné množstvo dát za deň] * 180.*

/home

Ak sa používajú používateľské adresáre, tak je potrebné vytvoriť samostatnú partíciu.

Základná sieťová konfigurácia *3*

Pri inštalácií systému pri sieťových nastaveniach je potrebné nastaviť nasledujúce možnosti:

Vypnúť DHCP, pokiaľ táto služba nie je nevyhnutná pre chod systémov. Je odporúčané používať statické adresovanie v sieti.

Vypnúť IPv6, pokiaľ táto služba nie je nevyhnutná na chod systémov. (Zriedkavo je).

Root heslo *3*

Toto heslo býva vstupným bodom do administrátorského panelu v systéme. Je preto potrebné aby spĺňalo najprísnejšie parametre na tvorbu hesla, tj: obsahovalo **minimálne 12 znakov**, ktoré tvoria mix veľkých a malých písmen, čísel a špeciálnych znakov. Tiež by nemalo byť založené na slove zo slovníkov.

Pokiaľ administrátorské heslo nie je nastavené, je možné získať prístup do systému napríklad cez rozličné nástroje ako je napr. *Recovery* konzola bez použitia autentifikácie.

Premazať systémový banner 3

V súbore **/etc/issue** sa v základnom nastavení nachádza označenie verzie systému. Tento súbor je potrebné zmeniť podľa bezpečnostnej politiky.

2.2. Nastavenie prístupových oprávnení

Táto sekcia nasleduje po inštalácií systému, avšak v prípade potreby je možné ju vykonať kedykoľvek. Cieľom je upraviť oprávnenia používateľov a služieb tak, aby sa zredukovalo riziko rozšírenia útoku v systéme. To znamená, že v prípade úspešného prieniku útočníka do systému útočník nebude schopný prevziať plnú kontrolu nad systémom.

Blokovanie prihlásenia a prístupu na terminál pre systémové účty 3

Táto ochrana spočíva v tom, že pokiaľ sa útočníkovi podarí vykonať určité príkazy v službe spustenej ako systémový používateľ, nebude schopný sa na tento účet prihlásiť, prípadne vykonávať veci v termináli pod týmto menom.

Podstata je zamknúť prihlasovanie a nastaviť terminál **/sbin/nologin** pre každý systémový účet (ID medzi 1 až 500)

Konfigurácia sa vykoná nasledujúcim skriptom:

```
#!/bin/bash
for user in `awk –F: ‘{print $1 “:” $3 “:” $7}’ /etc/passwd`
do
int=`echo $user | awk –F: ‘{print $2}’`;
      if [[ “$int” –gt “0” && “$int” –lt “500” ]];
      then
            usermod –s /sbin/nologin `echo $user | awk –F: ‘{print $1}’`;
            usermod –L `echo $user | awk –F: ‘{print $1}’`;
      fi
done
```

Pridanie nodev na lokálne ne-koreňové partície 2

Upraviť súbor **/etc/fstab .** Dôležité časti sú stĺpec 2 (Mount point), stĺpec 3 (typ fs) a stĺpec 4 (možnosti).

Pre každý riadok, kde platí nasledujúce: Typ FS je **ext (2**,**3** a **4)** a mount point **nie je** /

Treba pridať záznam “**,nodev**” do zoznamu možností v stĺpci 4.

Pridanie nodev, nosuid, noexec na vymeniteľné média 2

Všetky partície ktoré obsahujú text „floppy“ alebo „cdrom“ v stĺpci 3 je potrebné doplniť textom

“,noexec,nodev,nosuid“

V prípade, že je potrebné spúšťať kód z vymeniteľných médií, príkaz „noexec“ spôsobí problémy.

Pridanie nodev, nosuid, noexec na partíciu /tmp 2

Pridanie parametra **noexec** sa zabráni vykonávanie akéhokoľvek súboru z adresára /tmp. Toto môže byť požadovaný efekt, keďže do tohto adresára sa najčastejšie zapisujú škodlivé súbory pri prieniku z internetu. Avšak, niektoré programy v tomto adresári ukladajú dočasné súbory, napríklad **apt**. Preto je potrebné dôkladne preskúmať konflikty oprávnení po aplikovaní nasledujúceho nastavenia.

Pridať text **“,noexec,nodev,nosuid“** do zoznamu možností pri partícií /tmp

Pre každú službu vytvoriť osobitného používateľa 3

Každá služba (web server, mail server, ...) potrebuje vlastného používateľa, nikdy by nemala byť spustená s administrátorskými oprávneniami. Toto je z dôvodu, že ak útočník vykoná škodlivý kód v mene služby, dokázal by ako administrátor okamžite prevziať kontrolu nad systémom.

Takto vytvorený používateľ býva nastaviteľný v spúšťacom skripte v **/etc/init.d/** , kde sa v konkrétnom súbore vyskytuje záznam „**RUNASUSER**“, prípadne podobná hodnota.

Ďalšie miesta kde sa môže nachádzať takýto parameter sú súbory v adresári **/etc/default**, prípadne v konkrétnych konfiguračných súboroch danej aplikácie.

Umiestniť vhodné služby do jail-u 2

Jail je spôsob, ako umiestniť aplikáciu do virtuálneho prostredia, kde sa nenachádza nič iné okrem súborov súvisiacich s touto aplikáciou. V prípade napadnutia procesu nebude útočník mať možnosť pristupovať na súbory mimo uzamknutého adresára. Proces sa vykonáva cez nástroj **chroot**, ktorý pre daný proces zmení koreňový adresár, teda napríklad služba FTP nebude schopná pristupovať k akýmkoľvek súborom, ktoré sú uložené vyššie v súborovej hierarchií ako nami nastavený adresár.

Niektoré služby podporujú používanie chroot jail, pokiaľ je to možné tak je odporúčané toto nastavenie použiť. Chroot jail avšak **nie je** triviálna záležitosť, preto je nevyhnutné pred týmto procesom vytvoriť zálohu nastavení a bezpodmienečne dodržať postup.

2.3. Konfigurácia siete

Sieť (lokálna alebo internet) je najčastejším zdrojom útokov. Je preto mimoriadne dôležité tieto parametre nastaviť čo najprecíznejšie. Konfigurácia sa prejavuje úpravou firewallu, prípadne úpravou **sysctl** cez konfiguračný súbor **/etc/sysctl.conf**. V súbore sysctl.conf je potrebné buď zmeniť konkrétne hodnoty, alebo ich pripísať na koniec súboru, keďže posledná načítaná hodnota bude platná.

Sieť sa logicky rozdeľuje na vnútornú alebo vonkajšiu. Toto delenie závisí od prvku siete, z ktorého topológiu vyhodnocujeme. Pokiaľ sa berie v úvahu smerovač, vnútorná sieť je rozhranie do ktorého je pripojená infraštruktúra spoločnosti, vonkajšia sieť je pripojenie do internetu. Pokiaľ sa jedná o okrajový počítač, vnútornú sieť predstavuje operačný systém a prípadné virtuálne stroje; vonkajšiu sieť predstavuje výstup do sieťovej infraštruktúry. Toto delenie je veľmi podstatné pri konfigurovaní filtrov (firewall) a prekladania adries (NAT).

2.3.1. Hardening sieťovej prevádzky

Sieťový prístup je možné chrániť buď na koncovom zariadení, alebo cez samostatné zariadenie v sieti. Táto časť sa zaoberá ochranou na koncových zariadeniach.

Firewall

Firewall je najdôležitejšou zložkou konfigurácie siete. Je to filtračný mechanizmus sieťovej prevádzky. V linuxových systémoch je firewall prirodzenou súčasťou jadra ako modul **netfilter.** Upravuje sa cez nástroj **iptables** Každý systém by mal mať aktívny firewall, keďže takéto riešenie umožňuje blokovať väčšinu škodlivého softvéru ovládaného cez sieť priamo na stroji.

Pravidlá budovania firewallu

Firewall sa buduje na základe whitelistu, aby bola zaistená bezpečnosť pred útokmi na neošetrených adresách. Tento whitelist sa buduje s prihliadnutím na nasledujúce pravidlá:

- Nové prichádzajúce spojenia sú povolené iba lokálnym službám pre povolené systémy.
- Odchádzajúce spojenia sú povolené iba pre známe služby (DNS, POP3, mail, web,...).
- Preposielanie dát je blokované (okrem situácie, kde systém slúži ako bridge/router).
- Všetky zvyšné spojenia sú blokované.

Konfigurácia firewallu musí podliehať určitým pravidlám, aby bol firewall účinný. ***Pravidlá je nutné vždy prispôsobiť systému, na ktorom je firewall spustený, inak hrozí, že systém prestane byť funkčný. Ako príloha tohto dokumentu je uvedený skript, ktorý inicializuje firewall.***

Implementácia firewallu

Klasicky sa firewall implementuje cez skript, ktorý sa aktivuje vždy po spustení systému z dôvodu, že príkazy zadávané cez iptables nie sú permanentné a vydržia iba do najbližšieho reštartu systému. Vzorový skript je pripojený k tomuto dokumentu, pričom je potrebné ho nastaviť podľa vašej siete a implementovať podľa postupu popísanom v sekcií

Firewall je treba osobitne nastaviť pre každú jednu službu v smere dnu aj von zo systému (siete). Z tohto dôvodu je preto rozumné používať skripty namiesto konfigurácie používateľom riadok po riadku. Toto je podmienené tým, že pokiaľ by sme na server pristupovali vzdialene, hrozí že by sme si odrezali prístup nesprávnym poradím príkazov. Skript je potrebné nastaviť tak, aby sa aktivoval vždy

pri inicializácií systému (konkrétny postup závisí od distribúcie systému), aby bola zabezpečená neustála ochrana. Súčasnú konfiguráciu firewallu môžeme vidieť cez príkaz **iptables –L**.

Pri všetkých **TCP** spojeniach smerom dnu do siete je potrebné dohliadať, či je prítomné nastavenie „-m state --state ESTABLISHED". Toto nastavenie blokuje všetky nové prichádzajúce spojenia, ponechá iba tie, ktoré vznikli ako odpoveď na poslanú požiadavku.

Ukážkový skript triviálnej implementácie povolenia všetkých spojení na službu SSH na rozhraní eth0:

```
#!/bin/bash
# premazanie sucasneho firewallu
iptables –F
iptables -A INPUT -i lo -j ACCEPT
iptables –A OUTPUT –i lo –j ACCEPT
# povolenie paketov pre uz existujuce spojenia
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT -m state --state ESTABLISHED, RELATED -j ACCEPT
# povolenie prístupu na port 22.
iptables -A INPUT-p tcp --dport 22 -j ACCEPT
# povolenie výstupu z portu 22.
iptables -A OUTPUT-p tcp --sport 22 -j ACCEPT
# zakazanie vsetkej zvysnej komunikacie
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP
```

Táto základná konfigurácia nám deaktivuje všetku súčasnú konfiguráciu a aktivuje nové pravidlá, ktoré blokujú všetku komunikáciu okrem spojení na port 22.

Príklady konkrétnych pravidiel

V tejto sekcií sú popísané základné pravidlá pre konkrétne bežne používané služby. Je možné ich použiť v rámci ukážkového skriptu. Všetky pravidlá sú konfigurované pre všetky sieťové rozhrania.

Použitie MultiPorts

Multiports je metóda, ako spojiť viacero filtrov na porty do jedného príkazu. Pre všetky vymenované porty budú platiť rovnaké pravidlá. Nasledujúce dva riadky kombinujú pravidlá povoľujúce spojenia z/do všetkých sietí pre porty 22,80 a 443.

```
iptables -A INPUT-p tcp -m multiport --dports 22,80,443 -j ACCEPT
iptables -A OUTPUT-p tcp -m multiport --sports 22,80,443 -j ACCEPT
```

Povolenie spojení na loopback

Operačný systém občas prenáša vlastné informácie cez loopbackové rozhranie. Je rozumné toto rozhranie odblokovať, keďže jeho filtrovanie môže narušiť chod systému.

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT
```

Povolenie prístupu zo všetkých vonkajších sietí na službu SSH

Aby bolo možné vzdialene administrovať systémy cez SSH z akejkoľvek lokality na internete je potrebné pridať výnimku do filtrov umožňujúcu takéto spojenia.

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

```
iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT
```

Zabezpečenie prístupu na SSH

Táto konfigurácia obmedzí prístup na SSH za predpokladu, že sa spojenie nejaví ako legitímne. (tj. vytvára viac ako 3 spojenia za minútu).

```
iptables -A INPUT -p tcp --dport 22 --syn -m limit --limit 1/m --limit-
burst 3 -j ACCEPT
iptables -A INPUT -p tcp --dport 22 --syn -j DROP
```

Povolenie prístupu z konkrétnej vonkajšej siete na SSH

Predchádzajúce riešenie nie je bezpečné z dôvodu, že ktokoľvek môže pristupovať na službu SSH. Je preto rozumnejšie vytvoriť zoznam správcovských IP adries, ktoré budú mať oprávnenie prístupu, zvyšok bude zamietnutý. V príklade má prístup akékoľvek zariadenie zo siete 192.168.100.0/24.

```
iptables -A INPUT -p tcp -s 192.168.100.0/24 --dport 22 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT
```

Povolenie odchádzajúcich SSH spojení na ľubovoľnú vonkajšiu sieť

Tento krok je podstatný, pokiaľ očakávame potrebu zamestnancov administrovať cudzie zariadenia.

```
iptables -A INPUT-p tcp --sport 22 -j ACCEPT
iptables -A OUTPUT -p tcp --dport 22 -j ACCEPT
```

Povolenie odchádzajúcich SSH spojení na konkrétnu vonkajšiu sieť

Komunikácia môže byť použitá na prenos škodlivej komunikácie alebo na nadviazanie spojenia k útočníkovi po spustení malwaru, preto je rozumné tieto spojenia plošne blokovať. Riešením je vylistovanie všetkých možných cieľov, aby sa podstatne obmedzila možnosť komunikovať s útočníkom.

```
iptables -A OUTPUT -p tcp -d 192.168.100.0/24 --dport 22 -j ACCEPT
iptables -A INPUT -p tcp --sport 22 -j ACCEPT
```

Povolenie vonkajších spojení na služby HTTP a HTTPS

Pokiaľ sa v sieti (systéme) nachádza web server, je potrebné povoliť tieto dva porty. Toto nastavenie je v prípade serveru pre čisto interné potreby rozumné obmedziť iba o zoznam adries patriacich do siete, keďže web server je populárny útočný vektor.

```
# sekcia HTTP
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 80 -j ACCEPT
# sekcia HTTPS
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 443 -j ACCEPT
```

Povolenie spojení HTTP a HTTPS na vonkajšie siete

Aby mohli používatelia pristupovať na internetové stránky, je potrebné povoliť túto komunikáciu. Pokiaľ sa však jedná o zariadenie, ktoré nemá dôvod používať tieto služby, je rozumné tieto spojenia ponechať zakázané.

```
iptables -A OUTPUT -p tcp -m multiport --dports 80,443 -j ACCEPT
iptables -A INPUT -p tcp -m multiport --sports 80,443 -j ACCEPT
iptables -A OUTPUT -p tcp -m multiport --dports 80,443 -j ACCEPT
iptables -A INPUT-p tcp -m multiport --sports 80,443 -j ACCEPT
```

Povolenie DNS

Aby bolo možné používať doménové názvy namiesto IP adries, je potrebné povoliť protokol DNS. Keďže DNS je protokol založený na UDP, nie je možné aktivovať stateful firewall (pravidlo ESTABLISHED). Je odporúčané, aby sa koncové stanice odvolávali iba na lokálny DNS server, tým pádom nebudú priamo napadnuteľné z internetu.

```
iptables -A OUTPUT -p udp --dport 53 -j ACCEPT
iptables -A INPUT -p udp --sport 53 –j ACCEPT
```

Povolenie NTP

NTP sa využíva na synchronizáciu času cez sieť. Podobne ako DNS, protokol NTP je tiež postavený nad UDP a teda platia podobné bezpečnostné riziká.

```
iptables -A OUTPUT -p udp --dport 123 -j ACCEPT
iptables -A INPUT -p udp --sport 123 –j ACCEPT
```

Povoliť zasielanie a prijímanie e-mailov z vonkajšej siete

Táto séria otvorených portov povolí prístup z vonkajšej siete na mail server. Mail server používa väčšie množstvo portov, preto je potrebné zvážiť, ktoré z nich je potrebné otvoriť. Je potrebné aplikovať pravidlo minimálnych nutných oprávnení, tzn. sa neodporúča otvárať porty, ktoré nebudú priamo využité.

```
iptables -A INPUT -p tcp -m multiport --dports 25,110,143,993,995 -j ACCEPT
iptables -A OUTPUT -p tcp -m multiport --sports 25,110,143,993,995 -j
ACCEPT
```

Popis portov:

25	SMTP	–	zasielanie mailov
110	POP3	–	prijímanie pošty metódou POP3
993	POP3S	–	Bezpečná verzia POP3 (POP3 + SSL)
143	IMAP	–	prijímanie pošty metódou IMAP
995	IMAPS	–	Bezpečná verzia IMAP (POP3 + SSL)

Povolenie prístupu na MySQL

Prístup na MySQL databázu by nemala byť štandardná prax, avšak v určitých prípadoch, kde spoločnosť má vývojárov ktorí potrebujú prístup k službe je potrebné povoliť tieto spojenia. Je odporúčané povoliť iba určité adresy.

```
iptables -A INPUT -p tcp -s 192.168.100.0/24 --dport 3306 -j ACCEPT
iptables -A OUTPUT -p tcp --sport 3306 -j ACCEPT
```

Adaptívny firewall

Adaptívny firewall je taký, ktorý sa dynamicky prispôsobuje podmienkam na sieti. Príklad môže byť aktivovanie filtra v sieti na adresy, z ktorých bol zaznamenaný sieťový scan. Nástroj ktorý sa na takúto činnosť používa sa volá **psad**.

1. Nainštalujeme psad

Debian	RHEL
`# apt-get install psad`	`# yum install psad`

Psad funguje na princípe monitorovania komunikácie (napr z iptables) a hľadaním podozrivých spojení. Psad načítava tieto záznamy a vyhodnocuje ich na prítomnosť útokov.

2. Konfigurácia psad

Konfigurácia je uložená v súbore **/etc/psad/psad.conf**

Vstupné dáta sú načítavané štandardne zo súboru **/var/log/psad/fwdata**. Pokiaľ túto hodnotu chcete zmeniť, treba upraviť premennú „**IPT_SYSLOG_FILE**“.

Treba aktivovať funkciu adaptívneho firewallu úpravou nasledujúcich hodnôt:

ENABLE_AUTO_IDS Y; a **IPTABLES_BLOCK_METHOD Y;**

Tieto nastavenia zapnú IDS a automatické pridávanie do iptables.

3. Konfigurácia iptables

Po nastavení psad je potrebné nastaviť firewall, aby zaznamenával správy.

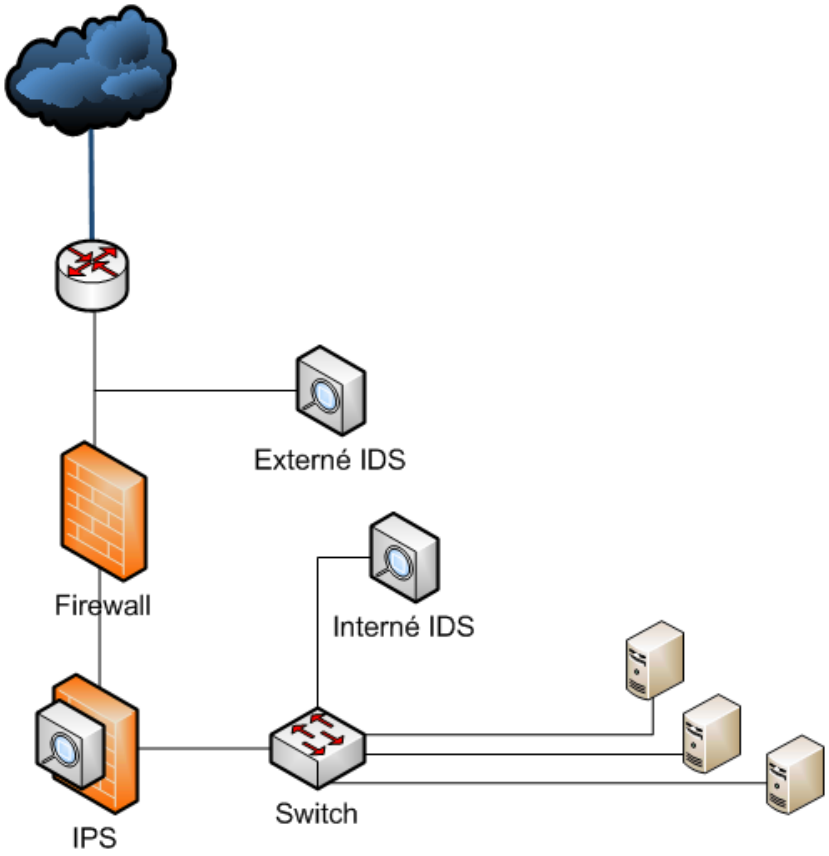
```
# iptables –A INPUT –j LOG
# iptables –A FORWARD –j LOG
```

Je potrebné nastaviť ukladanie správ do súboru, z ktorého načítavame dáta do psad, pokiaľ to nie je štandardný súbor **/var/log/syslog**

IDS/IPS

Intrusion Detection System alebo Intrusion Prevention System sú mechanizmy detekcie alebo prevencie pred sieťovými útokmi.

IDS aj IPS môžu byť tvorené rovnakou technológiou. Rozdiel je v spôsobe, akým je technológia použitá. IDS je senzor pripojený na „záznam“ sieťovej komunikácie (zvyčajne riešené cez span-port[2] na prepínači) a IPS je postavené ako most na sieti, kde prepája segmenty siete aby mohlo blokovať nežiaducu prevádzku.



Ochranu je možné inštalovať aj na koncový systém (HIPS – host based IPS), kde IPS monitoruje lokálny zásobník TCP/IP na vstupe do systému a nepustí dnu pakety, ktoré spĺňajú filtračné kritéria.

[2] Span-port je port, z ktorého vystupuje kópia všetkých paketov ktoré vystupujú z iných portov prepínača.

Snort

Snort je populárny opensource IDS/IPS systém. Pracuje na základe porovnávania dát so signatúrami škodlivej prevádzky. Signatúry sú vzory a pravidlá, podľa ktorých je prevádzka označovaná ako škodlivá. Je preto potrebné, aby používateľ mal tieto signatúry **vždy aktuálne**.

1. Nainštalujeme Snort

Debian	RHEL
`# apt-get install snort`	`# yum install snort`

Snort pri inštalácií požaduje informáciu o domácej sieti vo forme CIDR bloku. Je možné nakonfigurovať iba jednu adresu ako domácu sieť, formou masky 32. (Napr. 192.168.0.1/32)

2. Konfigurácia Snort

Snort sa konfiguruje pre IDS a IPS rovnako, pričom sa jedná o rozsiahlu činnosť. Po základnej konfigurácií domácej siete (HOME NET) a spustení procesu je systém funkčný. Rozpoznaná škodlivá komunikácia je zaznamenávaná do súboru **/var/log/snort/alert**.

Pokiaľ je Snort spustený s parametrom –Q, bude spustený v režime IPS a škodlivú prevádzku bude zahadzovať.

Spustenie Snort s parametrom -Q vyžaduje špeciálne moduly, ktoré je možno treba inštalovať zo zdrojových súborov. Inštalačná príručka pre tento prípad je dostupná v referenciách (1)

```
# snort –Q
```

Úprava parametrov detekcie zmenou spôsobov spracúvania paketov a pravidiel je mimoriadne rozsiahla činnosť, táto príručka nevyčerpáva túto tému.

Pravidlá je potrebné pravidelne aktualizovať (podobne ako vzorky vírusov pre antivírus). Takéto aktualizácie je možné získať po registrácií z web stránky autora programu Snort: https://www.snort.org/start/rules

Automatické aktualizácie sú možné použitím skriptov z tretích strán, ako je napríklad PulledPork. (http://code.google.com/p/pulledpork/)

Technická príručka k službe Snort je k dispozícií na oficiálnej stránke autora: http://manual.snort.org/

3. Monitorovanie výstupu

Prítomnosť IDS systému je zbytočná, pokiaľ sa záznamy ďalej neanalyzujú, keďže samotné IDS nemá žiadnu ochrannú funkciu. Je možné použiť monitorovací systém ako Splunk.

3.3.2. Hardening koncových systémov

Blokovanie zmien súboru /etc/resolv.conf *2*

Súbor /etc/resolv.conf obsahuje konfiguráciu DNS serverov systému. Je veľmi dôležité, aby tieto servery boli nastavené staticky a súbor bol blokovaný na prepisovanie. Takáto úprava je odstrániteľná kedykoľvek, avšak zvyšuje obtiažnosť útoku, hlavne pre menej skúsených útočníkov.

```
# sudo chattr +i /etc/resolv.conf
```

Vypnutie preposielania komunikácie koncových staníc *2*

Pokiaľ systém neslúži ako sieťové médium na preposielanie komunikácie, napríklad ako vstupná brána alebo smerovač, do súboru **/etc/sysctl.conf** je potrebné upraviť alebo pridať nasledujúce riadky:

```
net.ipv4.ip_forward = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
```

Zablokovanie rizikovej komunikácie pre koncové stanice a smerovače *2*

Toto nastavenie je určené pre systémy, ktoré sú umiestnené v menších, menej komplexných sieťach. Požiadavka na menej komplexné siete vyplýva z parametra **rp_filter**, ktorý blokuje overovanie zdrojov na základe RFC (IP Spoofing), čo nemusí byť splnené použitím určitých sieťových nastavení. Tento parameter je možné vynechať.

V súbore **/etc/sysctl.conf** je potrebné upraviť alebo pridať nasledujúce riadky:

```
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.accept_source_route = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.default.secure_redirects = 0
net.ipv4.icmp_echo_ignore_broadcasts = 1
net.ipv4.icmp_ignore_bogus_error_messages = 1
net.ipv4.tcp_syncookies = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1
```

Všeobecné blokovanie bezdrôtových spojení *1*

Bezdrôtové systémy tvoria riziko pre používateľa, keďže napadnúť bezdrôtový prístup je omnoho jednoduchšie ako napadnúť káblovú infraštruktúru. Riešenia môžu spočívať v odinštalovaní radičov pre bezdrôtové zariadenia, fyzické odstránenie antén alebo zablokovaním zariadení.

Zablokovanie hardvéru v BIOS/UEFI

V niektorých verziách BIOS/UEFI je možné zablokovať bezdrôtové zariadenia. Konkrétny postup závisí od výrobcu softvéru.

Zablokovanie IPv6 *2*

Protokol IPv6 je zriedkavo používaný, avšak je podporovaný na zariadeniach a teda je ho možné využiť na útoky, z dôvodu nenakonfigurovania sieťových filtrov vzhľadom na tento protokol. Pokiaľ vaša sieť explicitne nevyžaduje protokol IPv6, je ho potrebné zablokovať na všetkých bodoch infraštruktúry.

V súbore **/etc/sysctl.conf** treba upraviť alebo pridať nasledujúci riadok:

```
net.ipv6.conf.all.disable_ipv6 = 1
```

Ochrana pred detekciou časového razítka *1*

Pokiaľ server reaguje na ICMP požiadavku o časovú pečiatku, ICMP timestamp odpoveď obsahuje dátum a čas servera. Tieto informácie môžu byť teoreticky použité proti niektorým systémom, pokiaľ sa využijú slabiny v časových generátoroch.

Riešením je vypnúť ICMP timestamp odpovede. Realizácia je možná dvomi spôsobmi:

Filtrovanie na firewalle

Nasledujúce riadky treba pridať do inicializačného skriptu na firewall, inak ich budete musieť zadať opakovane po každom reštarte systému.

```
iptables -A INPUT -p ICMP –icmp-type timestamp-request -j DROP
Iptables -A INPUT -p ICMP –icmp-type timestamp-reply -j DROP
```

Vypnutie v sysctl

Pridať záznam `net.ipv4.tcp_timestamps = 0` do súboru **/etc/sysctl.conf**

Šifrovanie diskov *2*

Šifrovanie diskov je podstatné zabezpečenie pred neoprávneným prístupom k uloženým dátam. Dáta sú uchovávané šifrované a odšifrované sú len počas behu operačného systému, pokiaľ je zadaný správny šifrovací kľúč.

Výber softvéru na šifrovanie disku nie je jednoznačný, pričom populárny výber je **TrueCrypt** a **cryptsetup**.

3.4. Pravidelná údržba

Bezpečnosť nie je jednorazová činnosť ale proces. Je kriticky dôležité, aby systém bol pravidelne aktualizovaný a čistený od škodlivého kódu, inak sa veľmi ľahko stane terčom útoku.

Menej kritickú časť údržby tvorí sledovanie a zaznamenávanie činnosti systému čo sa týka plynulosti prevádzky, prítomnosti nepotrebných a rizikových služieb a aplikácií. Nasledujúce inštrukcie je potrebné vykonávať pomerne pravidelne, ideálne automatizovane, aby sme dostali základný prehľad o dianí v systéme.

2.4.1. Auditné záznamy

Auditné záznamy (po anglicky log) sú prvé miesto, kde bude administrátor hľadať informácie o chybách systému. Sú dôležité pre ladenie systému a majú nesmiernu hodnotu v prípade incidentu. Na systémoch Linux sú tieto záznamy uchovávané v adresári **/var/log**, kde sú uchovávané logy pre skoro všetky procesy.

Významné súbory

/var/log/syslog

Hlavný systémový log, je tu uchovávaná väčšina záznamov procesov.

/var/log/auth.log

Záznam z každého prihlásenia na systém. Tento log je mimoriadne dôležitý pre účely forenznej analýzy, preto je rozumné ho nastaviť iba na zápis.

```
# sudo chattr +a /var/log/auth.log
```

/var/log/messages

Log pre niektoré iné systémové procesy. Často sa tu uchovávajú chybové hlášky procesov.

/var/log/mail.log , /var/log/mail.err

Logy súvisiace s mailovou komunikáciou.

Implementácia centrálneho úložiska

Lokálne súbory dostupné (podľa štandardnej konfigurácie) v adresároch **/var/log/*** môžeme posielať z dôvodov centrálneho monitorovania na spoločný uzol, kde ich je možné ďalej spracúvať. Tieto súbory sa zaznamenávajú službou rsyslog alebo syslog-ng. Implementáciu centrálneho uzla dosiahneme jednoduchým spôsobom:

V pravidlách pre firewall povolíme UDP port 514 v smere dnu.

Debian

Debian a Ubuntu využívajú program **rsyslog** namiesto obyčajného syslogu. Tento program umožňuje veľmi jednoduché narábanie so syslog serverom. Pokiaľ nie je dostupný, je ho treba nainštalovať cez

```
# apt-get install rsyslog
```

V súbore **/etc/rsyslog.conf** odstránime komentár (znak #) z nasledujúcich riadkov:

```
# $ModLoad imudp
# $UDPServerRun 514
```

Následne reštartujeme službu rsyslog:

```
# service rsyslogd restart
```

RHEL

Red Hat používa klasický syslog. V súbore **/etc/sysconfig/syslog** treba pridať parameter **–r** do **SYSLOGD_OPTIONS**. (SYSLOGD_OPTIONS="-m 0 -r"). Následne reštartujeme službu:

```
# /usr/bin/sudo /sbin/service syslog restart
```

Implementácia klientov

Konfigurácia klientov spočíva v nastavení záznamov (varovania, informačné správy, chyby, ...), ktoré chceme zaznamenávať a adresu úložiska, kam sa majú posielať. Táto konfigurácia je rovnaká naprieč systémami.

V pravidlách pre firewall povolíme UDP port 514 v smere von.

V súbore **/etc/rsyslog.d/50-default.conf** (Debian, Ubuntu) alebo **/etc/sysconfig/syslogd** (RHEL, CentOS) pridáme nasledujúci záznam:

```
*.* @IP_ADRESA_SERVERA
```

Následne reštartujeme syslog server.

Spracúvanie auditných záznamov

Najčastejším problémom auditných záznamov je, že je ich príliš veľké množstvo na ľudské spracovanie a strácajú informačnú hodnotu, keďže ich nikto nebude čítať. Tento problém riešia automatické nástroje, ktoré spracúvajú záznamy do človekom prijateľnejšej formy.

Logwatch

Logwatch je nástroj, ktorý agreguje logy a spracúva ich do podoby ľahko čitateľnej človekom. Dokáže spracované logy posielať na e-mailovú adresu, alebo manuálne na požiadavku vygenerovať výpis.

1. Inštalácia Logwatch

Debian	RHEL
`# apt-get install logwatch`	`# yum install logwatch`

2. Konfigurácia a použitie

Konfiguračné súbory sa nachádzajú v zložke **/etc/logwatch/conf/logwatch.conf** alebo **/usr/share/logwatch/default.conf/logwatch.conf** (Závisí od distribúcie systému)

Dôležité parametre sú:

Detail Nastavuje ktoré správy sa majú posielať.

Mailer Nastavuje poštový klient pre odchádzajúce správy

MailTo Nastavuje poštovú schránku, na ktorú majú byť posielané reporty

Output Output umožňuje zadať miesto, kam budú reporty posielané. Štandardné je ich vytlačiť na obrazovku, ale je možné aj zapísať do súboru (zadaného pri spúšťaní programu) alebo poslať na mail.

Format Formát určí, či sa budú súbory vypisovať v čistom texte alebo ako HTML web stránka.

Spustiť výpis môžeme cez príkaz `# sudo logwatch`

Pre periodické výpisy je potrebné použiť cron skript s periodicitou štandardne 1 deň.

Synchronizovanie času

Je veľmi dôležité mať medzi systémami synchronizovaný čas. Táto požiadavka je o to silnejšia, keď sa jedná o **virtualizované systémy**, kde sú posuny hodín bežná situácia. Požadovanú presnosť nám zaručí proces **ntp**, ktorý synchronizuje čas sám v štatisticky vhodných intervaloch.

Prítomnosť NTP démona zvýši použiteľnosť logov zo systému.

1. Inštalácia NTP

NTP deamon nie je štandardne inštalovaný na mnohých systémoch. Pokiaľ ntpd nie je nainštalované, je potrebné vykonať:

Debian	RHEL
`# apt-get install ntp`	`# yum install ntp`

2. Konfigurácia

Konfiguračný súbor je **/etc/ntp.conf** . Do tohto súboru sú zapisované servery, voči ktorým sa bude synchronizovať čas. Po úprave je potrebné proces reštartovať. Ďalšiu konfiguráciu proces nevyžaduje, sám si určuje čas kedy sa bude synchronizovať. Drží aj štatistické informácie o odchýľkach, a na základe nich synchronizuje v čase, keď je to najvhodnejšie.

Pokiaľ vo vašej sieti nie je dostupný referenčný NTP server, je vhodné použiť čo najpresnejší verejne dostupný, napr. **time.windows.com**.

3. Overenie

Overiť funkčnosť ntp je možné cez nástroj ntpq. Parameter –p vypíše zoznam použitých ntp serverov aj s kontrolnými parametrami ako je stratum (vzdialenosť od autoritatívneho zdroja) a rôzne parametre týkajúce sa odchýlok.

```
# ntpq –p
```

Pokiaľ je časový rozdiel veľmi veľký, je potrebné ntp daemon vypnúť, a manuálne posunúť čas na aktuálnu úroveň nástrojom **ntpdate**. V príklade bol použitý server time.nist.gov, pričom je možné použiť náhradný.

```
# sudo service ntp stop && sudo ntpdate time.nist.gov && sudo service ntp start
```

2.4.2. Bežná údržba systému

Procesy bežnej údržby je potrebné vykonávať pravidelne, aby bola dosiahnutá čo najvyššia úroveň bezpečnosti.

Pravidelné aktualizácie systému *3*

Zanedbané aktualizácie systému sú najčastejší útočný vektor. Objavené zraniteľnosti sú publikované na internete a aplikovať verejne známy exploit na takto zraniteľný server je triviálne. **Pravidelné bezpečnostné aktualizácie sú nutné, pokiaľ máme systém v živej prevádzke.**

Súčasné operačné systémy majú bezpečnostné aktualizácie štandardne zapnuté. Pokiaľ nie, je potrebné ich aktivovať. Takúto procedúru je možné urobiť buď cez cron script, alebo využiť prostriedky programov dostupných na systéme.

V tejto sekcií sú uvádzané všeobecné postupy, v sekciách venovaných konkrétnym distribúciám sú dodatočné riešenia tejto problematiky.

Debian

Štandardná Debian distribúcia obsahuje balík `unattended-upgrades`. V prípade jeho neprítomnosti je potrebné balík doinštalovať.

Kontrola: # sudo dpkg -s unattended-upgrades

Prípadnú inštaláciu vykonáme príkazmi:

```
# sudo apt-get update ; sudo apt-get install unattended-upgrades
```

V súbore **/etc/apt/apt.conf.d/50unattended-upgrades** odstránime komentár (//) z riadku `"${distro_id}:{distro_codename}-security";`

RHEL

Vložiť nasledujúci skript do **/etc/cron.daily**:

```
#!/bin/bash
YUM=/usr/bin/yum
$YUM -y -R 120 -d 0 -e 0 update yum
$YUM -y -R -e 0 -d 0 update
```

Skript nazvite napr. **updater.sh** a aplikujte naňho **chmod +x** .

Antivírus

Škodlivé súbory (malware) nie sú unikátne iba pre Windows. Všetky systémy môžu byť napadnuté škodlivým kódom. Antivírus je dôležitou časťou systému, hlavne ak sa na ňom nachádzajú programy, s ktorými používateľ prichádza do styku. (FTP server, poštový server, web server,...).

ClamAV

ClamAV je populárny antivírus na Linux. Je voľne dostupný z balíčkov.

1. inštalácia

Debian	RHEL
`# apt-get install clamav`	`# yum install clamav`

S balíčkom clamav sa nainštaluje aj obslužný softvér **freshclam**, ktorý slúži na aktualizovanie vírusovej databázy programu. Aktualizácie vieme robiť manuálne (crontab), avšak odporúčaná metóda je spustiť démona ktorý bude za nás vykonávať túto činnosť.

2. Konfigurácia

Číslo za parametrom c určuje počet aktualizácií za deň <1 až 50>

```
# sudo freshclam –d –c 2
```

Okrem aktualizácií je treba nastaviť aj samotné zložky, ktoré chceme kontrolovať a nastaviť patričný výpis. Takáto konfigurácia sa môže premietnuť do skriptu, ktorý sa bude spúšťať v pravidelných intervaloch. Parameter **/cesta/k/adresáru** je potrebné nahradiť za adresár, ktorý chceme skenovať. (príkladom /var/www pre web, / pre celý systém.) Adresár **/var/log/clamav** bude použitý na odkladanie výpisov. V našom prípade je skript uložený v **/home/csirt/clamav.sh**

```
#!/bin/bash
LOG=clam-`date +"%d-%b-%y"`.log
cd /cesta/k/adresaru
sudo clamscan -r -l /var/log/clamav/$LOG
exit 0
```

Skript pridáme do **/etc/cron.daily**, (podľa potreby môžeme použiť aj hodinový interval) prípadne zadať do crontabu.

```
# sudo chmod +x /home/csirt/clamav.sh
# sudo crontab –e
```

Vložiť:

```
0 */1 * * * /home/csirt/clamav.sh
```

Pre pridanie rozpoznania výpisov do logwatch je potrebné upraviť hodnoty v súbore **/usr/share/logwatch/default.conf/logfiles/clamav.conf** tak, aby boli kompatibilné s ukladanými dátami. V našom prípade je log ukadaný do adresára /var/log/clamav/ vo formáte „clam-02-Jul-13.log“ , teda pridáme záznam:

```
LogFile = clamav/*.log
```

Úprava konfigurácie ClamAV

Upraviť súbor **/etc/clamav/freshclam.conf**

```
HTTPProxyServer IP_Adresa
HTTPProxyPort PORT
```

Do nasledujúcich súborov **/etc/environment ; /etc/profile/ ; /home/user_name/.profile ; /root/.profile** treba zapísať:

```
export http_proxy=http://IP_adresa:číslo_portu
export https_proxy=http://IP_adresa:číslo_portu
```

Ďalej je potrebné nastaviť UTF-8 pre celý systém kvôli diakritike.

```
# apt-get install locales
# dpkg-reconfigure locales
```

Do súborov **/home/user_name/.profile** a **/root/.profile**

Treba zapísať:

```
export LC_ALL=en_US.UTF-8
export LANG=en_US.UTF-8
export LANGUAGE=en_US.UTF-8
```

Ochrana pred rootkitmi

Rootkit je (škodlivý) softvér, ktorý dokáže maskovať prítomnosť procesov v systéme. Obvykle je využívaný na udržanie perzistentného spojenia na napadnuté systémy, maskovanie škodlivej aktivity a tak podobne. Pokiaľ je napadnutý systém, ktorý nemá žiadnu prevenciu, je veľmi zložité tento rootkit odhaliť a odstrániť.

Odporúčaný softvér je **chkrootkit** a **rkhunter**.

1. Inštalácia

Debian	RHEL
`# apt-get install chkrootkit`	`# yum install chkrootkit`
`# apt-get install rkhunter`	`# yum install rkhunter`
`# apt-get install chkconfig`	`# yum install chkconfig`

2. Použitie

Chkrootkit je proces, ktorý treba spustiť zakaždým keď chceme spustiť kontrolu. Použitie chkrootkitu je možné automatizovať cez cron.

```
# sudo chkrootkit
```

RKHunter je komplexnejší nástroj, ktorý aktualizuje svoje definície a spúšťa manuálne.

```
# sudo rkhunter --update
# sudo rkhunter --propupd
# sudo rkhunter --check
```

Overiť, či žiadne nebezpečné zložky nie sú prítomné v $PATH 2

Je potrebné občas kontrolovať túto premennú na prítomnosť relatívnych (začínajúcich znakom . alebo ..) a prázdnych ciest (:/bin , /bin: , /bin::/sbin). Všetky cesty by mali začínať znakom /.

```
# echo $PATH
```

Deaktivovať všetky nepotrebné sieťové služby 3

Táto úloha vyžaduje činnosť administrátora na identifikovanie potrebných služieb v systéme. Nasledujúcim príkazom dokážeme identifikovať všetky služby, ktoré majú aktívne sieťové spojenia.

```
# netstat –plntu
```

Iná metóda detekcie otvorených spojení je cez nasledujúci príkaz:

```
# lsof -i -n | egrep 'COMMAND|LISTEN|UDP
```

Následne identifikujeme spustené služby a deaktivujeme nepotrebné.

Deaktivovať všetky nepotrebné služby 2

Bežné systémové služby sú menej kritické ako sieťové, keďže nekomunikujú cez sieť von. Vždy však môžu predstavovať riziko, ako napríklad malware ktorý nepotrebuje sieťové spojenia na výkon svojej činnosti. Identifikovať momentálne zapnuté služby vieme napríklad nástrojom **chkconfig.**

```
# chkconfig --list | grep 3:on
```

Po analýze, či danú službu potrebujeme alebo nie, ju vieme vypnúť nasledujúcim príkazom:

```
# chkconfig --del ‘service-name’
```

Overiť, či v systéme existuje neadministrátorský účet s UID 0 3

Nasledujúci príkaz by mal mať jediný výstup a to účet „root“. Pokiaľ existuje ďalší účet, je nutné overiť či je tento účet autorizovaný a že či existuje postačujúci dôvod na jeho prítomnosť.

```
# awk -F: '($3 == "0") {print}' /etc/passwd
```

Skontrolovať služby spúšťané pri štarte systému 3

Nižšie uvedeným príkazom vypíšeme služby, ktoré sú inicializované pri štarte systému. Spôsob inicializácie systému sa môže líšiť medzi distribúciami.

V súčasných systémoch (napr. posledné verzie systému Debian) sa používa tzv. „**dependancy booting**“ a v starších systémov sa používajú adresáre **/etc/rcX.d/**.

Tieto adresáre sú mapované na úrovne behu systému (runlevel) systému, napr. zložka rc2.d obsahuje služby spúšťané na úrovni 2, rc5.d na úrovni 5. Inicializácia systému sa líši v rozličných distribúciách, preto je potrebné prispôsobiť nasledujúci postup vašej konkrétnej distribúcií. Skripty označené „S“ sú spúšťané pri štarte systému, skripty označené „K“ sú vykonávané pri vypínaní systému.

Dependancy booting využíva súbory uložené v **/etc/init.d/.depend.{boot,start,stop}**. Tieto súbory sú podobné Makefile súborom. Tvorba init skriptov presahuje túto príručku.

Inštalácia nových skriptov pre štart systému sa vykonáva cez príkaz **update-rc.d**.

Debian

Debian spúšťa pri štarte súbory s úrovňou S, následne sa preklopí do úrovne 2-5, čo je plný beh systému. Je preto potrebné sledovať adresáre **`/etc/rcS.d`** a **`/etc/rc2.d až /etc/rc5.d`**

```
# ls -l /etc/rc[2-5S].d/S* ; ls –l /etc/init/ ; ls –l /etc/init.d/.depend.*
```

RHEL

Red Hat Enterprise Linux rozlišuje úrovne omnoho striktnejšie. Úroveň 3 je určená pre beh systému v konzolovom režime (viacero používateľov, prístup na sieť), úroveň 5 v grafickom režime. V prípade že používame konzolový režim je potrebné sledovať adresár **`/etc/rc3.d`**, ak používame grafické rozhranie tak adresár **`/etc/rc5.d`**

```
# ls -l /etc/rc3.d/S* alebo  # ls -l /etc/rc5.d/S*
```

Výpis obsahuje symbolické linky do adresára **/etc/init.d**, kde sú uložené spúšťacie skripty služieb.

Overiť prítomnosť procesov odpočúvajúcich sieť *3*

Každý proces, ktorý odpočúva všetko dianie na sieti do ktorej je pripojený zanecháva stopu v konkrétnom súbore. Pokiaľ je nasledujúci výpis prázdny (žiadne hodnoty označené číslom), systém neeviduje žiadne odpočúvacie procesy.

```
# cat /proc/net/packet
```

Scan otvorených portov *3*

Z iného systému sa môžeme pozrieť perspektívou útočníka, ktoré porty sú dostupné. Nahradíme IP adresu za IP adresu lokálneho systému.

Pozor: nasledujúci príkaz bude pravdepodobne zablokovaný ak sú prítomné sieťové ochranné prvky.

```
# nmap –A 127.0.0.1
```

Po vykonaní scanu je potrebné spustiť príkaz `netstat –plntu` a vyhodnotiť, ktoré procesy využívajú daný port a či sú v súlade s očakávanými hodnotami.

3. Red Hat Enterprise Linux (RHEL) 5

Red Hat Enterprise Linux je populárna distribúcia v komerčnej sfére, spolu s distribúciou CentOS, ktorá je nekomerčný klon. Konfigurácia RHEL sa dá priamo aplikovať na CentOS.

3.1. Konfigurácia po inštalácií

Overenie prítomnosti GPG kľúča Red Hat 2

Tento kľúč overuje dôveryhodnosť a integritu inštalovaných Red Hat balíčkov

```
# rpm -q --queryformat "%{SUMMARY}\n" gpg-pubkey
```

Zablokovanie RHSND démona 2

Tento démon kontroluje sieť Red Hat-u pre naplánované aktivity. Pokiaľ táto služba nie je potrebná, je odporúčané ju vypnúť.

```
# chkconfig rhnsd off
```

Používanie softvéru yum na balíčky 3

Je potrebné deaktivovať klasické aktualizácie a vytvoriť cron skript

```
# chkconfig yum-updatesd off
```

Nasledujúci skript vložiť do **/etc/cron.daily**:

```
#!/bin/sh
/usr/bin/yum -R 120 -e 0 -d 0 -y update yum
/usr/bin/yum -R 10 -e 0 -d 0 -y update
```

Skontrolovať overovanie podpisu na balíčkoch 3

V **/etc/yum.conf** v sekcií [Main] overiť prítomnosť hodnoty **gpgcheck=1**

Používanie proxy servera pre balíčky 2

V súbore **/etc/yum.conf** je potrebné pridať alebo upraviť nasledujúce riadky

```
# Proxy server - proxy server:port number
proxy=http://mycache.mydomain.com:3128
# Nastavenie prihlasovacich udajov
proxy_username=yum-user
proxy_password=qwerty
```

3.2. Zabezpečenie používateľského prístupu

Obmedzenie administrátorských príkazov iba na členov skupiny wheel

V prvom kroku sa overí prítomnosť skupiny wheel.

```
# grep ^wheel /etc/group
```

Následne v súbore **/etc/pam.d/su** odkomentujeme, prípadne upravíme nasledujúci riadok:

```
auth required pam_wheel.so use_uid
```

Následne je potrebné overiť, či všetci administrátori a používatelia s administrátorskými oprávneniami sú členmi skupiny **wheel**.

Konfigurácia bezpečnostnej politiky hesiel *3*

V súbore **/etc/login.defs** sa nastavia hodnoty expirácie hesiel. Je potrebné pridať alebo upraviť nasledujúce riadky podľa bezpečnostnej politiky:

```
PASS_MAX_DAYS 60
PASS_MIN_DAYS 7
PASS_MIN_LEN 14
PASS_WARN_AGE 7
```

Tieto zmeny sa týkajú iba nových používateľov. Pre každého existujúceho je potrebné spustiť nasledujúci príkaz: (USER označuje používateľské meno používateľa)

```
chage -M 60 -m 7 -W 7 USER
```

3.3. Ochrana pred riskantným správaním aplikácií

Táto ochrana poskytuje všeobecnú ochranu pred takým správaním, kde by aplikácia mohla zobraziť nežiaduce informácie o systéme, prípadne iné podobné správanie. Táto ochrana je aplikovaná na úrovni kernelu a chráni pred zle nakonfigurovanými aplikáciami a niektorým škodlivým kódom.

Nastavenie parametra umask pre démony *2*

Toto nastavenie sa aplikuje pri inicializácií systému na všetky démon procesy. Nastavenie s hodnotou 027 umožní čítať súbory (vrátane dočasných súborov a logov) iba autorizovaným procesom.

V prípade konfliktu oprávnení je možné nastaviť menej reštriktívnu hodnotu 022, ktorá iba zakazuje vytvárať súbory, do ktorých môžu všetci zapisovať.

V súbore **/etc/sysconfig/init** pridať alebo upraviť riadok na hodnotu **`umask 027`**

Zablokovať výpisy jadra systému *2*

Core dump je proces, kde sa vypíše kompletný blok pamäti súvisiaci s daným procesom kvôli chybnému správaniu. Tieto výpisy často obsahujú citlivé informácie, preto je rozumné ich zablokovať z dôvodu ochrany systému pred útočníkmi.

Pre zablokovanie výpisov pre všetkých používateľov, v súbore **/etc/security/limits.conf** treba pridať alebo upraviť nasledujúci riadok:

```
hard core 0
```

Ako dodatok k tejto ochrane slúži blokovanie výpisov pre všetky procesy s parametrom setuid.

Štandardne je táto hodnota **0**, avšak počas života systému sa môže meniť. Overenie prebieha cez zadanie príkazu

```
# sysctl fs.suid_dumpable
```

Pokiaľ výstup nie je 0, v súbore **/etc/sysctl.conf** treba pridať alebo upraviť nasledujúci riadok:

```
fs.suid_dumpable = 0
```

Aktivácia ExecShield

ExecShield je sada funkcií systému RHEL, ktorá bráni pretečeniu vyrovnávacích pamätí. Medzi tieto funkcie patrí náhodné umiestňovanie dát do pamäte a zásobníkov, blokovanie vykonávania pamäťových blokov ktoré by mali držať pasívne dáta a pridáva špeciálne zaobchádzanie s vyrovnávacími pamäťami pre reťazce textu.

V súbore **/etc/sysctl.conf** treba pridať alebo upraviť nasledujúce riadky:

```
kernel.exec-shield = 1
kernel.randomize_va_space = 1
```

Overenie, či ExecShield je spustený je možné cez zadanie nasledujúcich príkazov:

```
# sysctl kernel.exec-shield
# sysctl kernel.randomize_va_space
```

3.4. Konfigurácia siete na RHEL5

Zablokovanie automatického načítania IPv6 modulu jadra

Automatické načítanie tohto modulu sa dosiahne cez pridanie nasledujúceho riadku do súboru **/etc/modprobe.conf**:

```
install ipv6 /bin/true
```

Keď jadro systému vyžiada modul IPv6, bude miesto toho presmerované na program **/bin/true**

Zablokovať používanie IPv6 na rozhraniach

Na zablokovanie konfigurácie IPv6 pre všetky rozhrania, je potrebné pridať alebo upraviť riadky v súbore **/etc/sysconfig/** v sekcií **network**:

```
NETWORKING_IPV6=no
IPV6INIT=no
```

Pre každé rozhranie IFACE je potrebné pridať alebo upraviť nasledujúce riadky v súbore **/etc/sysconfig/network-scripts/ifcfg-IFACE** ako dodatočný mechanizmus prevencie:

```
IPV6INIT=no
```

Ak v budúcnosti nastane situácia, kde je potrebné povoliť IPv6, je dôležité aby bolo povolené iba na rozhraniach, kde je to potrebné.

Zablokovanie automatickej konfigurácie IPv6 rozhraní

V protokole IPv6 je možné prijať od smerovača informáciu, ktorá automaticky nakonfiguruje lokálne rozhranie. Tento mechanizmus predstavuje riziko pre systémy, preto je rozumné ho deaktivovať.

V súbore **/etc/sysconfig/network** je potrebné pridať alebo upraviť nasledujúci riadok:

```
IPV6_AUTOCONF=no
```

Toto nastavenie nezablokuje zasielanie oslovovacích správ na smerovače (router solicitation), preto je ich potrebné deaktivovať osobitne.

V súbore /**etc/sysctl.conf** je potrebné pridať alebo upraviť nasledujúce riadky:

```
net.ipv6.conf.default.accept_ra=0
net.ipv6.conf.default.accept_redirect=0
```

Nastaviť statické IPv6 adresy

V prípade, že je nutné zachovať IPv6 konektivitu a nie je možné ju zablokovať, je potrebné ju zabezpečiť. Statická IP adresa je bezpečnejšia ako dynamicky pridelená. IPv6 adresy sa konfigurujú v súbore **/etc/sysconfig/network-scripts/ifcfg-IFACE** , kde IFACE je meno sieťového rozhrania.

Nasledujúci riadok obsahuje globálnu IPv6 adresu. Je potrebné pridať alebo zmeniť nasledujúci riadok (s tým, že bude vložená korektná adresa):

```
IPV6ADDR=2001:0DB8::ABCD/64
```

Nasledujúci riadok obsahuje predvolenú bránu smerovača. Je potrebné pridať alebo zmeniť nasledujúci riadok (s tým, že bude vložená korektná adresa):

```
IPV6_DEFAULTGW=2001:0DB8::0001
```

Obmedziť dynamickú konfiguráciu IPv6 adries

Tento blok konfigurácie súvisí so statickým nastavením IPv6 adries, a ako bezpečnostný mechanizmus touto konfiguráciou vypneme dynamické adresovanie zariadení.

V súbore **/etc/sysctl.conf** je potrebné zmeniť alebo pridať nasledujúce riadky:

```
net.ipv6.conf.default.router_solicitations = 0
net.ipv6.conf.default.accept_ra_rtr_pref = 0
net.ipv6.conf.default.accept_ra_pinfo = 0
net.ipv6.conf.default.accept_ra_defrtr = 0
net.ipv6.conf.default.autoconf = 0
net.ipv6.conf.default.dad_transmits = 0
net.ipv6.conf.default.max_addresses = 1
```

4. Debian

Debian a jeho odnože sú veľmi populárne distribúcie. Medzi hlavné patria Ubuntu a Mint.

4.1. Drobná konfigurácia Debian-u

Používanie proxy servera na balíčky

Aby sa koncové stanice nepripájali priamo na vonkajšie servery (a teda mali otvorenú ďalší komunikačný kanál cez firewall), je rozumné mať jedno centrálne úložisko na prijímanie aktualizácií z vonkajšej siete a aktualizovať koncové stanice lokálne z tohto úložiska.

Konfigurácia koncových staníc je nasledujúca:

V súbore **/etc/apt/apt.conf** vložiť nasledujúce hodnoty, kde URL je adresa nakonfigurovaného systému (napr. apt-cacher):

```
Acquire::http::Proxy "http://proxy.example.com:8080";
```

Kde sa adresa proxy.example.com:8080 nahradí za príslušnú (IP / URL) adresu a port.

Nastavenie minimálnej dĺžky hesla

V súbore **/etc/pam.d/common-password** treba upraviť nasledujúci riadok:

```
password required pam_unix.so md5 nullok obscure min=6 max=16
```

Zablokovanie reštartu cez Ctrl-Alt-Del

Aby sme zabránili neoprávneným používateľom reštartovať stroj cez stlačenie kláves Ctrl-Alt-Del, je možné vykonať nasledujúce opatrenia:

- Zablokovať reštart pre všetkých používateľov
- Umožniť reštart iba vybraným používateľom

Plošné zablokovanie

V súbore **/etc/inittab** sa nachádza nasledujúci riadok:
`ca:12345:ctrlaltdel:/sbin/shutdown -t1 -r now`

Ak sa pred neho vloží znak **#** funkcia reštartovania cez klávesy Ctrl-Alt-Del bude zablokovaná.

Vybraní používatelia

V súbore **/etc/inittab** sa nachádza nasledujúci riadok:
`ca:12345:ctrlaltdel:/sbin/shutdown -t1 -r now`

Vložíme doňho parameter **–a** tak, že výsledok bude vyzerať podobne ako príklad:
`ca:12345:ctrlaltdel:/sbin/shutdown -t1 –a -r now`

Parameter **–a** označuje prítomnosť whitelistu **/etc/shutdown.allow**, ktorý označuje vybraných používateľov.

Vytvoríme tento súbor, do ktorého vložíme zoznam používateľov (jedno používateľské meno na riadok), ktorým umožníme reštartovať systém klávesnicou.

Zablokovať klávesové príkazy SysRq

Klávesové skratky na základe System Request je možné použiť na zastavenie procesov, reštart systému alebo vyvolania kernel panic.

Zablokujú sa cez pridanie nasledujúceho riadku do súboru **/etc/sysctl.conf**

```
kernel.sysrq = 0
```

4.2. Pravidelná údržba

Aktualizácie kernelu

Pri inštalovaní Debian-u do verzie 3.0 neboli zavedené automatické aktualizácie jadra systému, preto je nutné ich aktualizovať manuálne.

Je možné overiť integráciu jadra systému do systému balíčkov nasledujúcim príkazom:

```
# dpkg -S ‘readlink -f /vmlinuz‘
```

Predpokladaný výstup tohto príkazu je text podobný „`linux-image-2.6.18-4-686: /boot/vmlinuz-2.6.18-4-686“`. Pokiaľ je výstup chybová hláška o neprítomnosti priradeného súboru, je potrebné inštalovať nový balíček kernelu manuálne. Výber závisí od architektúry systému a osobných preferencií na verziu. Po inštalácií je balíček asociovaný na správu balíčkov a podlieha automatickým aktualizáciám.

Aktualizácie kernelu sú vždy iba v rámci rovnakej verzie, tzn. verzia 2.4. nikdy nebude automaticky aktualizovaná na verziu 2.6.

Aktualizácia na novú verziu

```
# kernfile=‘readlink -f /vmlinuz‘
# kernel=‘dpkg -S $kernfile | awk -F : ’{print $1}’‘
# apt-cache policy $kernel
```

Očakávaný výstup:

```
linux-image-2.6.18-4-686:
Installed: 2.6.18.dfsg.1-12
Candidate: 2.6.18.dfsg.1-12
Version table:
*** 2.6.18.dfsg.1-12 0
100 /var/lib/dpkg/status
```

5. Konfigurácia vybraných aplikácií a služieb

Aplikácie a služby tvoria najzraniteľnejšie body systému. Čím je aplikácia populárnejšia a rozšírenejšia, tým je väčšia motivácia hľadať zraniteľnosti a vyvíjať exploity. O to dôležitejšie je korektne nakonfigurovať tieto aplikácie, keďže sú útočníkom dôverne známe.

Všeobecný postup spočíva v odstraňovaní schopnosti aplikácií zasahovať do inej časti systému, ako je potrebné na chod. V prípade, že cez aplikáciu bude vykonaný prienik do systému, útočník nadobudne všetky oprávnenia danej aplikácie. Je preto mimoriadne dôležité, aby aplikácia nemala absolútne práva na systém. Bezpečné konfigurácie aplikácií sa menia podľa verzií.

5.1. Apache 2.2

Apache je najpopulárnejšia distribúcia web servera. Štandardná inštalácia sa nachádza v adresári **/etc/apache2**. Pokiaľ je na vašom systéme Apache nainštalovaný do iného adresára, je treba tento návod prispôsobiť vašej inštalácií.

Apache je potrebné inštalovať s podporou SSL a SO.

1. Pridáme používateľa „apache“ a skupinu „apache“.

```
# groupadd apache
# useradd -g apache -d /dev/null -s /bin/false apache
```

2. Odstránime predvolený obsah.

```
# rm -rf /etc/apache2/cgi-bin
# rm -rf /etc/apache2/htdocs
# rm -rf /etc/apache2/icons
# rm -rf /etc/apache2/man
# rm -rf /etc/apache2/manual
# rm -rf /etc/apache2/conf/extra
# rm -rf /etc/apache2/conf/original
```

3. Aktualizovať vlastnícke práva na súboroch

```
# chown root:root /usr/sbin/apachectl
# chmod 770 /usr/sbin/apachectl
# chown -R root:root /etc/apache2
# chmod -R go-r /etc/apache2
# chown -R root:root /etc/apache2/logs
# chmod -R 700 /etc/apache2/logs
```

4. Vyčleniť zložku pre webový obsah a prideliť jej oprávnenia

```
# mkdir -p /var/www
# chown -R root /var/www
# chmod -R 775 /var/www
```

V prípade špeciálnych aplikácií s potrebou vyhradenie kvóty odporúčame vyhradiť samostatnú partíciu.

1. V súbore **/etc/httpd/conf/httpd.conf (RHEL)** alebo v súbore konkrétnej stránky (**Debian**) , resp. v základnej stránke **/etc/apache2/sites-enabled/000-default** vykonať nasledujúce zmeny:

V tabuľke sú uvádzané zmeny v prehľadnom formáte, kde je treba zmeniť konfiguráciu na novú:

Pôvodná konfigurácia	Nová konfigurácia
`DocumentRoot "/var/www/html"`	`DocumentRoot "/www"`
`User www`	`User apache`
`Group www`	`Group apache`
`Listen 80`	`Listen Server_FQDN:80`
`ServerAdmin root@localhost`	`ServerAdmin webmaster@spoločnosť.sk`
`#ServerName www.example.com:80`	`ServerName Server_FQDN`
`LogLevel warn`	`LogLevel notice`
`ScriptAlias /cgi-bin/ "/etc/apache2/cgi-bin/"`	`# ScriptAlias /cgi-bin/ "/etc/apache2/cgi-bin/"`

`<Directory />`
`  Options FollowSymLinks`
`  AllowOverride None`
`  Order deny,allow`
`  Deny from all`
`</Directory>`
	`<Directory />`
	`  Options None`
	`  AllowOverride None`
	`  Order deny,allow`
	`  deny from all`
	`</Directory>`

`<Directory "/etc/apache2/htdocs">`	`<Directory "/www">`
	`<LimitExcept GET POST>`
	`deny from all`
	`</limitexcept>`

`Options Indexes FollowSymLinks`	`Options -FollowSymLinks -Includes -Indexes -MultiViews`

Na koniec súboru pridať nasledujúce riadky:

```
ServerSignature Off
ServerTokens Prod
Timeout 60
# Maximum size of the request body.
LimitRequestBody 10000
# Maximum number of request headers in a request.
LimitRequestFields 40
# Maximum size of request header lines.
LimitRequestFieldSize 4094
# Maximum size of the request line.
LimitRequestLine 500
```

Vymazať zo súboru nasledujúce riadky:

```
<Directory "/etc/apache2/cgi-bin">
```

2. V súbore **/etc/apache2/include/ap_release.h** zmeniť nasledujúce linky:

V tabuľke sú uvádzané zmeny v prehľadnom formáte, kde je treba zmeniť konfiguráciu na novú:

Pôvodná konfigurácia	Nová konfigurácia
`#define AP_SERVER_BASEVENDOR "Apache Software Foundation"`	`#define AP_SERVER_BASEVENDOR "Restricted server"`
`#define AP_SERVER_BASEPRODUCT "Apache"`	`#define AP_SERVER_BASEPRODUCT "Secure Web Server"`

RHEL

Automatické spúšťanie Apache servera:

Do súboru **/etc/rc.local** treba pridať nasledujúci záznam:

```
/etc/apache2/bin/apachectl start
```

Debian

Pridávanie nových služieb medzi automaticky spúšťané služby sa vykonáva príkazom

```
# update-rc.d apache2 defaults
```

5.2. MySQL

MySQL je najbežnejšia implementácia SQL serveru.

Uistite sa, že na administračné rozhranie v iptables môžu pristupovať iba dôveryhodné osoby.

Pokiaľ nie je potrebné otvárať žiadne sieťové spojenia a všetka administrácia prebieha lokálne, je možné zablokovať sieťové spojenia v súbore **/etc/my.cnf** alebo **/etc/mysql/my.cnf**.

Do sekcie **[mysqld]** je potrebné pridať záznam **skip-networking.**

Iným riešením môže byť nastavenie parametra na lokálnu IP adresu (**bind-address=127.0.0.1**)

Zablokovať procedúru INFILE

Procedúra INFILE môže spôsobiť exfiltráciu dát z nepovolených súborov, ako je /etc/passwd.

V sekcií **[mysqld]** treba upraviť alebo vložiť záznam **set-variable=local-infile=0**

Overiť, či MySQL proces nie je spustený ako root

Je potrebné overiť, či útočník po nadobudnutí prístupu do MySQL bude môcť vykonať viac, než len obmedzenú sadu príkazov. Nasledujúcim príkazom zistíme meno vlastníka procesu, v štandardnom nastavení je vlastník „mysql“.

```
# ps aux | grep mysql
```

Pokiaľ meno vlastníka nie je obmedzený používateľ, túto hodnotu je treba zmeniť v súbore **/etc/mysql/my.cnf (resp. /etc/my.cnf)**.

Taktiež je potrebné zabezpečiť, aby iba vlastník mysql procesu a root mali prístup k súborom **/var/lib/mysql** a nikto iný.

Znížiť privilégiá používateľov databázy

Okrem administrátorského účtu sa v databáze nachádza mnoho ďalších používateľov, ktorí potrebujú rozdielne úrovne privilégií.

Iba jeden účet (administrátor) potrebuje privilégiá SUPER / PROCESS / FILE a prístup do datatabázy mysql. Treba preveriť zvyšných používateľov, či ich privilégiá nepresahujú ich účel.

Overenie sa vykoná pomocou nasledujúcich krokov:

```
mysql> use mysql;
mysql> select * from users;
```

Pre každého používateľa vykonáme nasledujúci krok (so zámenou používateľa „root“ za mená, ktoré sme dostali z predošlého výpisu):

```
mysql> show grants for 'root'@'localhost';
```

Je rozumné odobrať administrátorské oprávnenia všetkým používateľom, ktorí ich nepotrebujú.

Ďalší krok je odstrániť oprávnenie prezerať databázy pre používateľov, ktorí toto oprávnenie nepotrebujú.

Vykonanie opatrenia:

Pridať parameter **skip-show-database** do konfiguračného súboru **/etc/mysql/my.cnf** v sekcií **[mysqld]**

Následne povoliť oprávnenie SHOW DATABASES iba vybraným používateľom.

Premenovať administrátorské účty a zmeniť základné heslá

```
mysql> RENAME USER root TO novy_pouzivatel;
```

príkaz RENAME je dostupný až od verzie MySQL 5.0.2, pokiaľ používate predošlú verziu je potrebné vykonať nasledujúce príkazy:

```
mysql> use mysql;
mysql> update user set user="novy_pouzivatel" where user="root";
mysql> flush privileges;
```

Zmena hesla:

```
mysql> SET PASSWORD FOR 'pouzivatel'@'%hostname' = PASSWORD('noveheslo');
```

odstránenie predvolenej databázy „test“

táto databáza môže byť použitá ako dočasné úložisko škodlivého kódu. Pokiaľ sa nepoužíva, je rozumné ju odstrániť.

```
mysql> drop database test;
```

Overiť, či sú v databáze anonymní používatelia

Nasledujúci príkaz by nemal vrátiť žiadne výsledky. Pokiaľ sa takíto používatelia objavia, je potrebné ich odstrániť.

```
mysql> select * from mysql.user where user="";
```

Povoliť vytváranie logov

V súbore **/etc/mysql/my.cnf (resp. /etc/my.cnf)** je potrebné do sekcie **[mysqld]** vložiť nasledujúci záznam:

log =/var/log/mysql-logfile

5.3. SSH

Mnohé systémy používajúce SSH na vzdialený prístup sú zraniteľné. V súbore **/etc/ssh/sshd_config** je uložená konfigurácia.

Nasledujúce hodnoty je potrebné pridať do konfiguračného súboru, prípadne nastaviť na nové:

```
Port <ENTER YOUR PORT>
```

Táto hodnota zmení súčasnú hodnotu portu, na ktorom počúva služba SSH.

```
PermitRootLogin no
```

Zákaz priameho prihlasovania sa na administrátorský účet.

```
UsePrivilegeSeparation yes
```

Táto hodnota môže spôsobovať problémy na starších systémoch. Núti SSH daemona aby vykonával iba časť kódu ako root a nepotrebné časti v oddelenom prostredí.

```
Protocol 2
AllowTcpForwarding no
X11Forwarding no
```

Táto sekcia súvisí s parametrami sieťovej bezpečnosti. Vynucuje použitie protokolu verzie 2 a blokuje preposielanie dát.

```
IgnoreRhosts yes
HostbasedAuthentication no
RhostsRSAAuthentication no
PubkeyAuthentication yes #default
AuthorizedKeysFile /home/csirt/.ssh/authorized_keys #default
RSAAuthentication no
PasswordAuthentication no
UsePAM no
KerberosAuthentication no
GSSAPIAuthentication no
```

Táto sekcia blokuje prihlasovanie sa s lokálnymi účtami. Taktiež, nastavuje prihlasovanie sa iba cez PKI, nie cez meno/heslo.[3] Po vygenerovaní kľúča je treba verejnú časť kľúča pridať medzi rozpoznané na serveri do súboru **~/.ssh/authorized_keys**. Táto sekcia potrebuje, aby adresár **~/.ssh/known_hosts** bol zapisovateľný (mod 755, vlastník používateľ).

```
#Subsystem      sftp    /usr/lib/misc/sftp-server
```

Pokiaľ nepoužívame SFTP, je potrebné tento modul vypnúť.

```
DebianBanner no
```

[3] Tento mechanizmus mnohonásobne zvýši bezpečnosť. PKI kľúče treba generovať cez príkaz # ssh-keygen –t rsa –b 4096, alebo použiť puttygen

Zablokujeme posielanie informácií o lokálnom systéme.

```
AllowGroups sshLogin
```

Tieto nastavenia umožnia prihlásenie sa **iba** používateľom, ktorí sú súčasťou skupiny **sshLogin**.

Generovanie SSH kľúčov

Windows

Na systéme Windows sa kľúče generujú cez program PuTTYGen (Dostupný z http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html)

Cez PuTTYGen vygenerujeme pár kľúčov (SSH-2 RSA). Následne exportujeme verejný a privátny kľúč.

Linux

Program `ssh-keygen` vygeneruje pár kľúčov, kde verejný je označený ako .**pub**.

Implementácia kľúča

Pri štandardnej konfigurácií nakopírujeme na server do súboru **/home/meno_pouzivatela/.ssh/authorized_keys** .

Do tohto súboru zapíšeme kľúč bez medzier a úvodných identifikátorov ako napr. `------BEGIN RSA PUBLIC KEY------.` Kľúč musí byť uložený v súbore v nasledujúcom tvare:

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC3zgYfU6wA0QijCdEmDljstmJobtbwmZAe0op+xxK2RZ+iXkO
HT0+Vs51kScS6k+fDuGRdoylMai+MzjFKxKfOFJYR47YMxVU6jmj40PU88Om4dwgHlWBkn/IPURSxTgFe
1bMw6MvdEU8P3tO1OmzHjhuedl71WytgzEqlJFWdIVYN4D+T4g1DPV6JhO0j6xrgMey/4TFfFfrRVVU9
D5/m5RMV2IGN7P8uGqmAo52nEEX09DiiXqesqjvK/7anRkNzraU8tBoofPK1iJCfG7JOWdEP7xHg638Ud
sI0X/WGXGmrQTl6QadiiUQPLMkSycGypqFbyx8q5R/VLr3U+9PR comment

Server začne rozpoznávať používateľský kľúč po reštarte SSH. (`service ssh restart`)

Použitie SSH cez kľúč

Windows (PuTTY)

Na strane klienta je potrebné sa prihlasovať s privátnou časťou kľúča, ktorý je uložený na serveri. V kliente putty tento kľúč priložíme privátny kľúč v sekcií Connection > SSH > Auth

Linux

Štandardne je ssh klient nastavený na čítanie súborov **/home/meno_pouzivatela/.ssh/id_rsa**, kde je uložený privátny kľúč používateľa. Do tejto zložky je štandardne aj uložený kľúč vygenerovaný cez ssh-keygen.

Prihlásenie sa pri prítomnom kľúči prebieha jednoducho príkazom `ssh meno@server`

Referencie

1. Using Snort as service IPS. [Online] http://techminded.net/blog/using-snort-as-service-ips.html.