. . . už nechodí bosa

aneb poučení z krizového vývoje

Michal Švamberg

XXX. konference EurOpen.CZ, Jablonné v Podještědí

23. května 2007

Čas je neúprosný

Nebudeme mluvit o:

- zabezpečení dat,
- síťových nástrojích (snort, netflow, labrea, honeypot, . . .),
- řešení pro každého, aneb zachraňme svět.

Vyděržaj pijaněr

Příprava, nastavení a správa bezpečnosti je nikdy nekončící práce. Existuje však mnoho nástrojů jenž s tím mohou pomoci. Tyto nástroje znají velmi dobře také útočníci.

Pokud je 99% systému zajištěno a jen jediné procento způsobí kompromitaci, pak je kompromitován celý systém.

Systém nebude nikdy zajištěn na 100%.

Množství rutinní práce narůstá s počtem technických opatření.

Základní pojmy

proaktivní nástroje předcházejí útoku, monitorují a hledají potencionální pokusy o útok nebo více zabezpečují systém.

reaktivní nástroje se snaží odhalit útok a provést protiopatření.

forenzní analýza provádí zjištění rozsahu a způsob útoku.

Legislativa je více než technika

Bezpečnost by se měla primárně řešit "papírovou" cestou. Pokud se pravidla dodržují, jsou účinnější než technika. Bezpečnostní politika se snaží předejít průniku a v tom je její největší síla.

Technické prostředky jsou až na druhém místě. Lze si jimi také vynucovat bezpečnostní politiku.

Nastavuje mantinely uživatelům a správcům tak, aby omezovaly vznik problémů. Příkladem může být zákaz provozování služeb na desktopech a správcovských segmentech.

být informovaný Přihlásit se do konferencí, kde jsou hlášeny opravy bezpečnostních chyb. Použít nějaký nástroj (`debsecan`) na informování o stavu bezpečnostních záplat pro daný systém.

být podezíravý Důsledkem napadení systému může být jeho změna chování. Je třeba vždy hledat příčinu, pokud není zřejmá.

To lze snadno detekovat na desktopech, hůře na serverech, kde například vyšší load nebo traffic může být i očekáván. Je potřeba serverové systémy mít více pod drobnohledem.

být paranoidní Všichni jsou zlí a nelze od nich očekávat žádné slitování.

Napadená služba $\Rightarrow$ napadený stroj.

Proaktivní nástroje

Hledají nebo omezují slabá místa v systému, na které upozorňují. Tyto nástroje (zvláště síťové) jsou zneužívány také útočníky.

- firewally
- tiger
- nessus
- rkhunter
- aide, tripwire
- ...

Dále lze doporučit netstat, nmap, ps, ...

Reaktivní nástroje

Snaží se reagovat na detekovaný útok

- portsentry
- snort
- firewall
- ...

Existují i HW zařízení, jejich výhodou je vysoká propustnost.

Forenzní analýza

Poslouží nám k tomu, aby se:

- dohledal způsob útoku
- nalezly další napadené stroje,
- poučili se.

Zde nemá smysl mluvit o napadené službě, vždy je napaden stroj a pro úplné vyčištění je nutná REINSTALACE. Pozor na obnovy ze záloh, mohou být také napadené, nebo na nich útok rozpracován.

Služby

Pokud možno, vyčlenit, metod je několik:

- na samostatné stroje
- virtualizace
- chroot/jail

Cílem je zamezit křížení služeb, šíření útoku mezi nimi a snížit možnosti útočníka.
Provozovat službu a mít "blízko" ní uživatele je riziko.

Systémová nastavení

Základem je

Vyhnout se problémovým službám, jako je

- inetd
- portmap (rpc daemon)

Zcela určitě mít nastaveno

- vzdálený syslog
- firewall

Provozovat jen ty služby, které jsou nezbytné!

Firewall

Firewally dnes v zabezpečení charakterizuje:

- nezbytné minimum
- není to "svatý grál"
- nebrání útoku "zevnitř" s oblíbenými pravidly RELATED, ESTABLISHED

Vzdálené logování

Nasazení vzdáleného logování a kontrola logů je základem bezpečného systému!
Logy bez včasné analýzy lze považovat za plýtvání místem. Použití analyzátoru vede k:

- – častým a obsáhlým reportům,
- – neustálým úpravám vzorů,
- + přehledu a včasnému zjištění problémů (nejen bezpečnostních).

Často se používá `logcheck`, `swatch`, `logwatch` již s existující základní databází vzorů.

Chroot

Dobrý způsob, jak provozovat více služeb na jednom stroji, ale za předpokladu dalšího zabezpečení chrootu (grsecurity).
Pro vytváření chrootu lze použít nástroj `makejail`:

```
chroot="/var/chroot/mysql"
doNotCopy=["/etc/init.d","/etc/cron.daily","/etc/fstab","/proc"]
cleanJailFirst=1
forceCopy=["/etc/mysql/my.cnf","/etc/hosts"]
preserve=["/var/lib/mysql","/var/log/mysql","/proc","/dev/null",
          "/etc/passwd","/etc/group","/etc/mysql/debian.cnf"]
packages=["mysql-server","coreutils","grep","sed","bsdutils"]
useDepends=1
maxRemove=3000
users=["mysql"]
groups=["mysql"]
userFiles=["/etc/password","/etc/shadow"]
groupFiles=["/etc/group","/etc/gshadow"]
```

Wrappery

Rozhodně používat a nesnažit se obcházet. Pro logování (ale i ladění služeb) lze nastavit pro `hosts.deny`:

```
ALL: ALL: SPAWN ( \
          echo -e "\n\
          TCP Wrappers\: Connection refused\n\
          By\: $(uname -n)\n\
          Process\: %d (pid %p)\n\
          User\: %u\n\
          Host\: %c\n\
          Date\: $(date)\n\
        " | /usr/bin/mail -s "Connection to %d blocked" root) &
```

Obdobně lze také sledovat, například použití služby SSH.

Zabezpeční na úrovni kernelu

Mezi nejpropracovanější a dnes nejpoužívanější patří:

- grsecurity – `http://www.grsecurity.net/`
- SELinux – `http://www.nsa.gov/selinux/`

Modifikace jádra:

- + způsob zabezpečení, který nelze nahradit jiným řešením
- − náročnost (kompilace, konfigurace, snížení výkonu)
- − možné problémy s během aplikací

Nejčastěji se jedná o nespustitelný zásobník a haldu, ošetření `chroot()`, dočasných souborů, zavedení ACL a rozšířené logování.

grsecurity

Rozšiřuje vlastnosti jádra tak, aby zastavil nejběžnější metody útoku na systém. Součástí je robustní ACL systém.

- + bezpečnostních rozšíření (PaX, RBAC)
- + pro RBAC k dispozici learning režim
- + systémová opatření (`fork()` rate, memory page protection, různá síťová a systémová omezení, . . .)
- − možné problémy při nasazování
- − není součástí jádra
- − patch existuje jen vůči některým jádrům

Významně omezuje možnosti útočníka a mnoha exploitů. Zvláště vhodné je nasazení ve spojení s chroot.

Monitoring systému

Monitorování systému může pomoct při forenzní analýze.

- `atop` – monitoring systémové aktivity a procesů
- `lastcomm` – kdo a co pustil (process accountingu)
- `last` – kdo se přihlásil (součást systému)
- `sar` – zatížení systému

Drobnosti v systému

- mount `/tmp` a `/var` s parametry `noexec`, `nosuid`
- volby pro `sysctl`, například:

```
net/ipv4/icmp_echo_ignore_broadcasts=1
net/ipv4/icmp_ignore_bogus_error_responses = 1
net/ipv4/tcp_syncookies = 1

net/ipv4/conf/default/accept_redirects = 0
net/ipv4/conf/default/secure_redirects = 1
net/ipv4/conf/default/send_redirects = 0
net/ipv4/conf/default/forwarding = 0
net/ipv4/conf/default/log_martians = 1
net/ipv4/conf/default/rp_filter = 1
net/ipv4/conf/default/accept_source_route = 0
```

Nástroje pro zajištění integrity systému

AIDE pro souborové systémy

AIDE upozorňuje na změny v lokálním souborovém systému.

- + srovnatelný s Tripwire
- + jednoduchá a přehledná konfigurace na základě regexp
- + snadná údržba
- + používá více hashovacích algoritmů
- + distributoři přidávají do balíků konfiguraci pro AIDE
- – je třeba se pravidelně o něj starat
- – časově náročné odladění konfigurace
- – problém se zajištěním databáze

AIDE: konfigurace definic

```
Checksums     = md5+sha1+rmd160+haval+gost+crc32+tiger+whirlpool
OwnerMode     = p+u+g                    # p      - permissions
Size          = s+b                      # u,g    - user, group
InodeData     = OwnerMode+n+i+Size       # m,a,c  - (m|a|c)time
RamdiskData   = InodeData-i              # s,S    - size, growing size
StaticFile    = m+c+Checksums            # l      - link name
                                         # E      - empty group
Full          = InodeData+StaticFile     # b      - block count
VarFile       = OwnerMode+n              # i      - inode
VarDir        = OwnerMode+n+i            # n      - number of links
RotatedLogs   = Full+I
Logs          = OwnerMode+n+S            # I      - ignore changed filenam
LowLogs       = Logs-S                   # ANF    - allow new files
LinkedLogs    = Logs-n                   # ARF    - allow removed files
```

AIDE: apache2

```
/var/log/apache2/(access|error)\.log\.1$ LowLogs
/var/log/apache2/(access|error)\.log\.2\.gz$ RotatedLogs+ANF
/var/log/apache2/(access|error)\.log\.[0-9]+\.gz$ RotatedLogs
/var/log/apache2/(access|error)\.log$ Logs
/var/run/apache2\.pid$ VarFile
/var/run/apache2/ssl_scache$ VarFile
/var/(log|run)/apache2$ VarDir
@@ifdef APACHE2_SUEXEC
/var/log/apache2/suexec\.log\.1$ LowLogs
/var/log/apache2/suexec\.log\.2\.gz$ RotatedLogs+ANF
/var/log/apache2/suexec\.log\.[0-9]+\.gz$ RotatedLogs
/var/log/apache2/suexec\.log$ Logs
@@endif
```

AIDE: výstup obsahuje

1. Hlavičku (stroj, datum spuštění, počet řádek, . . .)
2. Celková statistika změn (sumář):

```
Summary:
 Total number of files:          41470
 Added files:                    7
 Removed files:                  13
 Changed files:                  102
```

3. Seznamy souborů podle změn (přidané, odebrané, změněné).
4. Detailní výpis ke změnám u souborů:

```
File: /etc/motd
 Mtime     : 2007-02-14 14:04:22          , 2007-05-15 14:10:34
 Ctime     : 2007-02-14 14:04:22          , 2007-05-15 14:10:34
 Inode     : 149374                       , 148085
```

5. Charakteristické znaky porovnávané a nové databáze (velikost, datum, součty, . . .).

`rkhunter` nejen na root-kity

Hledá známe i *neznámé* rootkity, backdoory, sniffery a exploity.

- + dostatečná konfigurace k testům
- + možnost automatických updatů vyhledávacích vzorů
- + navíc kontrola účtů, ssh, syslogu, md5 součtů, práv, ...

Jako alternativu lze použít `chkrootkit`.

`tiger` aneb najdi cestu k 'root' účtu

Systém pro hlášení zranitelnosti vedoucí ke kompromitaci roota:

- + jednoduché
- + prakticky bez konfigurace
- + hlásí pouze změny (zbytečně neotravuje)
- + široký záběr testů (listen, součty, účty, soubory mimo balíčky, práva, logy, ...)
- + lze navázat na aide/tripwire
- + lze použít i na jiných Unixových systémech (SunOS, IRIX, AIX, MacOS, ...)

Nasazení doporučuji, pohlídá základní systém a nestojí to příliš času.

Nessus – najdi mi vstupní vrátka pro útočníka

Provádí bezpečnostní audit vzdálených systémů.

- + simuluje a následně analyzuje úspěšnost útoku
- + obsahuje velkou sadu útoků
- – nelze příliš automatizovat
- – změna licenční politiky

Často jej používají i útočníci, proto je vhodné se s ním seznámit.

Pavel Vachek provozuje pro síť CESNET Nessus server.

Správa bezpečnosti

Je to běh na dlouhou trať, který zabírá spoustu času, přitom nikdy nekončí.
Základní zajištění je v dobře zvládnutém logování (syslog) a analýze logů (logcheck). Dalším krokem je provádění auditu (nessus, tiger) stroje. V kritických místech (logovací server) pak použít detekci průniku (aide).

Je to užitečné?

Po zkušenostech s nepříjemným útokem (popsáno v předchozím příspěvku) jsme lépe zajistili nejkritičtější stroje (terminálové servery). Nedlouho na to, jsme zaznamenali podobný útok, tentokrát jsme byli připraveni.

- `logcheck` nachází pády programu (hlášky grsec) a chybové hlášky cronu o špatném nastavení konf. souboru
- v systémovém logu toto chybí, ale také chybí pravidelné záznamy
- na vzdáleném logovacím serveru jsou logy kompletní
- `lastcomm` ukazuje na uživatele jehož program končí core dumpem

Podrobněji ve sborníku.

Otázky

Nyní otázky, potom oběd :-)