Bezpečnost informačních systémů

Projekt č.1 – jednoduchý rootkit
Radim Loskot

1 Spuštění rootkitu

Pro spuštění rootkitu byly vytvořeny následující cíle v Makefile:

make run	– vloží modul jádra pro skrytí rootkitu a spustí rootkit
make stop	– ukončí rootkit
make insert-module	– vloží modul jádra pro skrytí rootkitu
make run-rootkit	– spustí pouze rootkit

Pro spuštění rootkitu a jeho skrytí je určen cíl „run“. Došlo-li by ovšem k neočekávanému pádu systému z důvodu použitého modulu jádra, rootkit by měl být spouštěn samostatně a to cílem „run-rootkit“. I při spuštění pouze samotného rootkitu, bez nahrání modulu jádra, by měl být proces skryt druhou metodikou – viz další kapitola.

2 Metodiky skrývání procesu

Pro skrytí procesu z příkazu `ps` a `top`, ale i jiných využívajících virtuální souborový systém `/proc`, který je alokován v paměti, nikoliv na disku, byly vytvořeny 2 způsoby skrytí.

První způsob je založen na vloženém modulu jádra, který filtruje výpisy složek a souborů při dotazování na složku daného procesu. Samotný modul jádra je také skryt, takže není zjistitelný příkazem `lsmod`. Modul konkrétně ze souboru `/boot/System.map-$VERSION` zjistí adresu tabulky systémových volání, která nebývá obvykle implicitně exportována při kompilaci jádra. Dále při vypnutém chráněném režimu procesoru modifikuje callback funkci pro systémové volání `getdents()`, využívaném pro výpis obsahu složky funkcí `readdir()`. Modifikovaná funkce filtruje všechny soubory týkající se procesu rootkitu.

Druhý způsob využívá toho, že příkaz ps získává informace o procesu ze souboru `/proc/$PID/cmdline`. Jelikož se jedná o virtuální souborový systém, o čtení tohoto speciálního souboru se stará z pohledu jádra funkce `proc_pid_cmdline()` definovaná v `fs/proc/base.c`. Při prostudování funkce je obsah souboru získáván z adresy `mm->arg_start` až po `mm->arg_end`.

Abychom mohli modifikovat tento soubor, tak nám stačí přepsat paměť určenou těmito pozicemi. Používá-li cílový systém ELF binární formát, poté je tento rozsah paměti naplněn ve funkci `create_elf_tables()`. Z té lze vyčíst, že shodou okolností je rozsah umístěn v uživatelském prostoru procesu, konkrétně odpovídá poli `argv`, předávanému procesu funkcí `main()`. Tudíž pro modifikaci souboru `cmdline` nám stačí pouze změnit pole `argv`, pro skrytí procesu smazat řetězce argumentů.

3 Protokol

Pro komunikaci s rootkitem byl vytvořen jednoduchý protokol. Autentizace probíhá formou dialogu. Jako autentizační login a heslo bylo staticky zvoleno slovo „rootkit“:

```
[root@localhost bis]# telnet localhost 8000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
(Type in your login and password)
Login: rootkit
Password: rootkit
```

Při neúspěšné autentizaci je spojení uzavřeno. V opačném případě je možné zadávat příkazy rootkitu:

`start` – spustí sshd server
`stop` – zastaví sshd server
`info` – zobrazí stav sshd serveru a informaci o stavu skrytého modulu jádra
`exit` – uzavře spojení

Ukázka komunikace s rootkitem po úspěšné autentizaci:

```
info
Hidden kernel module: inserted
Status of sshd: is stopped
start
Status of sshd: is running
info
Hidden kernel module: inserted
Status of sshd: is running
stop
Status of sshd: is stopped
info
Hidden kernel module: inserted
Status of sshd: is stopped
```