# Administratorens Hacker værktøjer. Del 3

## Her er så del 3, du kan læse denne allene, men bør læse de to andre først.

Skrevet den **08. Feb 2009** af **bufferzone** I kategorien **Sikkerhed / Generelt** | ★★☆☆☆

**Administratorens Hacker værktøjer. Del 3.**

Denne artikel er tredie del af serien om værktøjer der bruges af hackerne og som den seriøse administrator er nødt til at kende.
Del 1: http://www.eksperten.dk/artikler/387.
Del 2: http://www.eksperten.dk/artikler/400.
Del 3: http://www.eksperten.dk/artikler/426.
Del 4: http://www.eksperten.dk/artikler/445.

Reglerne fra de to foregående dele gælder stadig. Hack aldrig andre, hverken for sjov eller når de beder dig om det, og brug kun disse værktøjer i et sikkert testmiljø, der ikke har forbindelse med andre "skarpe" miljøer, hvor du kan gøre skade.

Her i del 3 vil jeg tage lidt flere værktøjer op. Jeg vil kikke på et par værktøjer, der er mere hacker værktøjer end de er administratorværktøjer. Hvor en administrator sagtens kan bruge f.eks. Nessus til at teste sit netværk med, vil jeg nu kikke på nogle værktøjer der decideret bruges til at omgå firewalls og andet. Når de er taget med, er det selvfølgelig for at give administretore viden om hvad de står overfor, hvilke værktøjer der bruges imod dem, så de kan teste og selv se resultaterne i logfiler og andet.

**FPipe.**
http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/proddesc/fpipe.htm

FPipe er en TCP/UDP source port forwarder/redirector. Den er en af de værktøjer som en administrator bør kende, men som han faktisk kun kan bruge til at lave proof-of-concept på forskellige hacks, og som han bevidst bør ungå at have på maskiner i sit net. Med andre ord, et rent hackerværktøj.

FPipe kan redirecte enhver TCP eller UDP stream til enhver source port du måtte ønske. Dette kan du bruge til at redirecte TCP trafik ud af en hvilken som helst port der måtte være til rådighed, herunder også de såkaldte ephemeral ports (port numre over 1023), der normalt vælges tilfældigt som destination ports og som derfor er åbne for udgående trafik på pakkefiltrerings routere der ikke har stateful inspection.
Det kunne også være, at firewallen var lidt løst konfigureret omkring nogle af de reserverede porte (port numre under 1023), og således tillod alt udgående trafik med destination port 53 (DNS trafik), 20/21 (FTP trafik), Port 23 (telnet trafik) port 25 (mail) eller port 80 (http trafik). Her kan du så anvende FPipe til at pipe alt muligt andet trafik ud gennem disse porte.
Hvis du f.eks. har lukket dit net af for uønsket tratik og det betyder at dine brugere ikke kan anvende Kazaa eller ICQ, da de nødvendige porte er lukket, kan de redirecte trafikken indefra ud gennem en anden port som du har åbnet for og på den made få den ulovlige trafik ud. Hvis de også vil have trafikken ind, er de dels nødt til at finde en port hvor indgående trafik tillades helt ind til klienten (det kunne f.eks. være port 80 trafik), og dels nødt til at anvende en redirect proxy, der modtagere deres trafik på de redirectede porte og via FPipe redirecter den tilbage til de normale porte. Denne proxy kunne de f.eks. sætte op på deres egen maskine. FPipe kan evt. kombineres med NetCat, se herunder.
Når du har installeret Fpipe skriver du "FPipe", "FPipe -h" eller "FPipe -?". Dette vil give dig nedenstående hjælpetekst.

```
FPipe v2.1 - TCP/UDP port redirector.
Copyright 2000 (c) by Foundstone, Inc.
http://www.foundstone.com
FPipe [-hvu?] [-lrs ] [-i IP] IP
-?/-h - shows this help text
-c - maximum allowed simultaneous TCP connections. Default is 32
-i - listening interface IP address
-l - listening port number
-r - remote port number
-s - outbound source port number
-u - UDP mode
-v - verbose mode
```

Herunder kan du se et eksempel på hvordan FPipe køres. Prøv at se om du kan gennemskue hvad der sker. Det er desværre rørende let at bruge.

fpipe -l 53 -s 53 -r 80 192.168.1.101

**Firewalk.**
Friewalk er et aktivt scanner værktøj, der bruges til at analysere firewalls og andre filtre for huller og løst konfigurerede porte. Dette værktøj bruges af hackere i de indledende undersøgelser, oftest skjult bag en proxy IP, da den slags værktøjer altid støjer og sætter tydelige spor i både firewall logs og til IDS. Værktøjet anvender samme teknik som Tracerout (trecert i Windows verdenen) kommandoen og andre tracerout værktøjer. Time To Live (ttl) værdien har egentlig intet med tid at gøre, der sker det at denne værdi bliver en mindre hver gang pakken krydser en router. Hvis ttl værdien bliver null, vil den router (eller anden enhed) der nedtæller den fra en, sende en ICMP_TIME_EXCEEDED besked tilbage til afsenderen. Det er denne egenskab tracerout værktøjer anvender, de starter med at sende en pakke med ttl værdien 1 mod den host de ønsker at traceroute til. Den første router pakken møder vil nedtælle værdien til null og sende en ICMP_TIME_EXCEEDED. Routerens IP adresse noteres, hvorefter der sendes en pakke med værdien to og det medføre at næste router IP kan noteres. Dette fortsættes indtil vi når målet (eller faktisk igennem målet når vi taler Firewalk).
Firewalker bruges aktivt til at kortlægge netværk, hvor firewallen enten er decideret dårligt konfigureret eller bare løst konfigureret med hensyn til ICMP pakker.

En firewalking session kan f.eks. se således ud

```
Root@bufferzone:#firewalk -n -P1-8 -pTCP 10.0.0.5 10.0.0.20
Firewalking through 10.0.0.5 (towards 10.0.0.20) with a maximum of 25 hops.
Ramping up hopcounts to binding host...

probe: 1 TTL: 1 port 33434: <response from> [10.0.0.1]
probe: 2 TTL: 2 port 33434: <response from> [10.0.0.2]
probe: 3 TTL: 3 port 33434: <response from> [10.0.0.3]
probe: 4 TTL: 4 port 33434: <response from> [10.0.0.4]
probe: 5 TTL: 5 port 33434: Bound scan: 5 hops <Gateway at 5 hops> [10.0.0.5]

port 1: open
port 2: open
port 3: open
port 4: open
port 5: open
port 6: open
```

```
port 7: *
port 8: open

13 packets sent, 12 replies received
```

**Hunt.**

Hunt er en avanceret pakke sniffer og connection intrusion tool der kun kører på Linux og udnytter velkendte svagheder i TCI/IP protokol suiten.
Hunt kikker på TCP connections, bryder ind i dem eller resætter dem.
Hunt kan mange ting, f.eks. selektive ARP relaying og connection synchronization efter angreb og hunt har aktive mekanismer til at sniffe switchede netværk.

At angreb med hunt kunne f.eks. se sådan ud:
1. Hackeren starter altid med at undersøge og teste det system han vil angribe.

2. Hackeren starter hunt fra sit system (logget på som root) og venter på at Hunt skal indikere at den har fanget/detekteret en session.

3. Hackeren starter ARP relay daemon1, forbereder RST daemon2 entry til senere brug og sætter option til "enable host name resolution".

4. Ofret logger f.eks. på via Telnet og kører pine mail programmet (Pine er en linus mail klient).

5. Hackeren ser nu alle nye TCP/IP forbindelser. Han kan liste active forbindelser for at se om der skulle være noget interessant. Hvis han ser noget han kan bruge, kan han vælge at sniffe, eller han kan hijacke sessionen.

6. Ofret vil nu se en ny prompt. Hans mail forbindelse er gået død, men alt andet virker tilsyneladende. Har vi ikke alle prøvet at noget går ned, uden at det er særligt farligt. Det er det i dette tilfælde, men vil ofte opdage det?

7. Hackeren opdager at dette er en simple bruger session (altso ikke root retigheder) og giver derfor ofret sessionen tilbage (han resynkronisere TCP/IP strømmen).

8. Ofret bliver promptet for "hit any key" og gør selvfølgelig dette, hvorefter hans mail session igen virker. Hvis hackeren nu er lidt heldig, vil ofret logge ind som root for at forsøge at finde ud af hvad der er galt.

9. Hackeren starter RST daemonen for at forebygge nye connections og venter derefter på at hijacke root sessionen.

10. Ofret køre ssu for at få en SecurID protected root shell. Så er han jo sikret ikke?

11. Hackeren hijacker root login sessionen.

12. Ofret ser igen en ny prompt og lige som før er hans ssu login død, hans web browser virker, man hverken telnet eller ftp virker.

13. Hackeren sætter en bagdør op, f.eks. med NetCat, disabler command history, resets session og standser RST daemonen.

14. Ofret får igen sin session tilbage og alt virker tilsyneladende normalt. Hvor mange gange har vi ikke prøvet den slags midlertidige problemer?

15. Hackeren venter på at aktiviteten på ofrets maskine standser (brugeren har forladt maskinen tændt). Herefter installeres rootkit, flere bagdøre og loggen renses.

Denne slags angreb kaldes "Man-in-the-middle" angreb eller session hijacking og den opmærksomme læser vil straks se at selvom hunt kan sniffe på switchede netværk, kan hunt ikke bare anvendes på via Internettet mod hvem som helst, men på et LAN eller hvis en hacker har fysisk adgang (f.eks. via trådløs connection) til et netværk, er det et farligt værktøj.

**Hping3.**
Hping kaldes ofte for ping på steroider. Hping er et kommando linie baseret TCP/IP pakke assembler og analyseværktøj. I modsætning til almindelig ping der er en del af ICMP protokollen, understøtter Hping også TCP, UDP RAW-IP protokoller og har tracerout mode samt understøttelse for covered channels og meget mere. Hping bruges bl.a. til følgende ting.

- Firewall testning
- Avanceret port scanning
- Network test med flere forskellige protokoller, TOS, og fragmentation
- Manual path MTU discovery
- Avanceret traceroute, under alle de understøttede protokoller
- Remote OS fingerprinting
- Remote uptime guessing
- TCP/IP stacks auditing

Hping kan en del af det som værktøjer NetWox også kan og må betegnes som et must hvis du vil have indgående kendskab til TCP/IP protokollen. Her er et par eksempler på den utrolige enkle syntaks som Hping anvender
hping -S 10.1.1.1 -p 21  som sender en syn pakke mod hosten på port 21
hping -R 10.1.1.1 som sender en rst pakke mod hosten
hping -F 10.1.1.1 som sender en fin pakke mod hosten
hping -1 10.1.1.1 icmp echo request
hping -2 10.1.1.1 en udp pakke afsendes

Her kommer lidt mere komplicerede eksempler

-P en push pakke afsendes
10.1.1.1 destination IP
-d giver dig mulighed for at +/- størrelsen af selve pakken. Herunder sættes størelsen til 80 bytes
-p destinations porten, herunder sættes den til port 80
-E angiver stein til den fil der skal indsættes som data payload. Herunder /home/don/test.sig meget brugbart hvis man f.eks. vil indsætte et prekompileret exploit f.eks. et buffer overflow.

hping -P 10.1.1.1 -d 80 -p 80 -E /home/don/test.sig

Som du kan se er mulighederne mange, det er faktisk kun fantasien der sætter grændser.

**NetCat.**
http://www.atstake.com/research/tools/network_utilities/
NetCat er en klassiker man ikke kommer uden om når man taler hacking. Lige som Nmap og Nessus er det et af de værktøjer man skal kende og kunne bruge hvis man vil tages alvorligt.
NetCat kaldes ofte for The Swish Army knife of hacking, en title den I fremtiden kommer til at kæmpe med NetWox om, en kamp den nok ikke vinder.
NetCat kan mange ting, den kan virke som en telnet klient, som en SMTP klient og en http klient. Den kan konfigureres som bagdør, eller som tunnel klient til en bagdør.

Her er et par eksempler på NetCat kommandoer

Hvis du vil undersøge hvilken web server du står over for, kan du bruge  netcat som en http klient

nc 209.237.227.195 80
tryk enter to gange og skriv
head / http/1.0
Du vil nu se bannerinformationen fra www.apache.org (dette er fuldt lovligt)

Du kan også opsætte netcat til at lytte på en port, klar til at modtage en tunnel fra en anden netcat

Nc -l -p 1500

Netcat vil nu lytte på port 1500 for en indkommende tunner. Selve tunlen laver du fra en anden maskine således

Nc 80.146.141.89 1500

Der vil nu være etableret en tunnel fra den fremmede maskine til din maskine (med IP adressen 80.146.141.89) på port 1500 prøv selv.

Du kan se de andre muigheder ved at skrive nc -help, det giver følgende resultat

```
-d        detach from console, stealth mode
-e prog    inbound program to exec [dangerous!!]
-g gateway    source-routing hop point[s], up to 8
-G num       source-routing pointer: 4, 8, 12, ...
-h        this cruft
-i secs     delay interval for lines sent, ports scanned
-l              listen mode, for inbound connects
-L        listen harder, re-listen on socket close
-n        numeric-only IP addresses, no DNS
-o file        hex dump of traffic
-p port    local port number
-r        randomize local and remote ports
-s addr    local source address
-t         answer TELNET negotiation
-u        UDP mode
-v        verbose [use twice to be more verbose]
```

Det var så Del 3. Del 4 kommer til at handle om trådløse hacking værktøjer. Om jeg skriver flere ved jeg ikke endnu. Skulle du være interesseret i at få oplysninger om nye værktøjer mmd så se på http://www.securiteam.com/ de har en mailingliste hvor der bl.a. udsendes anmeldelser at værktøjer og andet.

Skulle du have spørgsmål, kommentarer eller rettelsen (herunder især stavefejl) er du velkommen til at kontakte mig på kim@bufferzone.dk, ligesom jeg ofte er at finde på Eksperten. Jeg hjælper selvfølgelig også gerne med de forskellige værktøjer. Undlad venligst at stille spørgsmål i kommentarerne, dem kan jeg jo ikke svare på.

**Kommentar af cybermike d. 04. Dec 2004 | 1**

Kedelig artikel , du gennemgår programmerne på et mega lavt niveau og aligevel antager du mange

steder i teksten et kendskab til blandt andet tcp som unødvendigør den grundige gennemgang da det kan forventes at folk med indgående kendskab til tcp/ip også kan finde ud af at bruge tools som hping og netcat. Du skulle måske se til at rafinere din skrivetil lidt.

Og så harmer det mig væsentligt at du i dit forsøg på en artikel bruger ordet hacker om en person som bryder ind i IT Systemer og netværk man skulle forvente at en som sætter sig selv i en position hvor denne forsøger at afgive viden til andre har sat sig nok ind i sagen til at bruge de rigtige ord, eller som minimum ihvertfald ikke bruge konfliktende ord, så istedet for det omdiskuterede "hacker" bruger du "angriber" eller sågar "datatyv" noget som ikke kan mistolkes.

Mine 25 øre.

**Kommentar af jones d. 15. Sep 2004 | 2**

Udemærket... glæder mig til del 4

**Kommentar af kurtspurt d. 16. Sep 2004 | 3**

Lige i tråd med de to forgående c",)

**Kommentar af john_stigers (nedlagt brugerprofil) d. 28. Dec 2005 | 4**

Kan ikke slette en vurdering, så må give karakteren middel.

**Kommentar af barbarbo d. 20. Sep 2004 | 5**

Rigtig god serie her, der er lidt at arbejde med her og jeg glæder mig til del 4.

**Kommentar af mysund d. 02. Feb 2006 | 6**

**Kommentar af dustie d. 09. Aug 2005 | 7**

**Kommentar af jlykkegaard d. 20. Sep 2004 | 8**

**Kommentar af sorenbs d. 04. Oct 2004 | 9**

Han gør det igen... He He

**Kommentar af d4rkd3vil d. 28. Oct 2004 | 10**

Jeg læste lige. Synes specielt at den del med Hunt, er et meget godt eksempel :)

**Kommentar af cyber00 d. 11. Oct 2005 | 11**

sikke noget pis. at lære folk at hacke er ulovligt