Forord

Da jeg startede Hackademiet, slog jeg fast fra start, at jeg ikke ville hjælpe script-kiddies med at hacke og at ødelægge andres maskiner. Mit mål var at hjælpe alle med at forstå, hvordan hackingverdenen kører, hvilke værktøjer der er, hvad de kan gøre, og hvordan det bliver gjort, samt hvordan man kan gardere sig imod det (hvis det er muligt). Det synes jeg lykkedes rimelig godt.

Der er mange ting, jeg kan lide ved Hackademiet ud over dette princip.

Jeg kan godt lide at greje informationerne, forstå dem og gengive dem på godt dansk, så de bliver mindre skræmmende og mere forståelige. Jeg kan lide at være med til at gøre det tilgængeligt for alle.

Men det, som glæder mig mest, er at jeg jævnligt modtager mails fra abonnenter til nyhederne eller købere, og de er som regel personlige, søde og spændende. Nogle giver mig praj om noget, andre bare en idé, de har fået. Det er dejligt.

Det er det, der har betydet, at Hackademiet er blevet til mere end bare et sted, man kan "lære at hacke". Der er et eller andet fælles. Alt foregår i glæde og godt humør.

Også denne bogskrivning er foregået i glæde. Det er med til at give inspiration, rent faktisk.

Så med denne bog, som jeg håber, du vil kunne lide, vil jeg gerne takke jer alle fra Hackademiet for at være kilde til så meget inspiration og så mange smil.

Jeg vil naturligvis også takke min kone, Britt Malka, som har været nødt til at gennemlæse manuskriptet op til flere gange for at rette mine sdavøfajl og syntagsfajl.

Men, da var igge så mejet som hun sige da var.

God læsning,

Chill

1. Introduktion

Jeg vil starte med at fortælle dig en lille historie. Måske forstår du ikke alting i første omgang. Du vil højst sandsynligt ikke forstå det hele i detaljer, da det er jo det, jeg skal fortælle om i denne bog. Men hvis du har læst Hackerguiden, vil du måske være med på nogle af noderne i hvert fald.

Lad os nu tage en fyr, som vi vil kalde "Johnny Hacker".

Johnny Hacker er ekspert i at finde hullede systemer, og han nyder at smadre dem. Han synes, det er sjovt.

Han benytter sig af Tor for at skjule sin identitet, og han har i øvrigt downloadet Tors bundle, som indbefatter en del værktøjer. Disse er egentligt beregnet som en hjælp for "good guys", men, som alle værktøjer, kan disse bruges til at opbygge eller til at smadre med.

Efter at han har installeret Tor og konfigureret det, benytter han Google til at finde servere, som er dårligt konfigurerede. Dem er der nemlig mange af rundt omkring på nettet. Så i Googles søgefelt, taster han:

Intitle:Test.page.for.Apache "It worked!" "this Web site!"

Google giver et hav af forskellige muligheder, og han klikker på en af dem og finder standardsiden: "It Worked on [navn på serveren] ! The Apache Web Server is Installed on this Web Site !"

CTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">

It Worked on Completel ! The Apache Web Server is Installed on this Web Site !

If you can see this page, then the people who own this domain have just installed the Apache Web server software successfully. They now have to add content to this directory and replace this placeholder page, or else point the server at their real content.

If you are seeing this page instead of the site you expected, please **contact the administrator of the site involved.** (Try sending mail to <Webmaster@*domain*>.) Although this site is running the Apache software it almost certainly has no other connection to the Apache Group, so please do not send mail about this site or its contents to the Apache authors. If you do, your message will be **ignored**.

The Webmaster of this site is free to use the image below on an Apache-powered Web server. Thanks for using Apache!

Powered by APACHE

Nu hvor han har webserveren og domænenavnet, vil han gerne have den IP-adresse, der er knyttet dertil. Han ved at en host-kommando vil opgive hans IP, og det ønsker han ikke. Derfor vil han benytte sig af Tors "tor-resolve" (som kom med i pakken).

```
Tor-resolve www.target.com
192.10.10.10
```

(Target= mål på engelsk, er bare et navn jeg bruger for at fortælle at der er tale om et mål. På samme måde, så er 192.10.10.10 et eksempel IP. Alt sammen til illustration).

Johnny Hacker starter Nmap, som han kører igennem Tor for at anonymisere det. Han starter Proxychain, som er et Linux program, der tvinger ethvert andet program (Nmap i det tilfælde) til at benytte sig af Tor eller af en liste af proxy-server.

Han vil bruge Nmap med specifikke muligheder: -ST for at få en fuld forbindelse i stedet for et SYN-scan. -PN for ikke at spilde tid på at finde ud af, om værten er til stede (eftersom han er forbundet til den, så ved han det, jo!). -N for at sikre sig at ingen DNS-request sker uden for Tor-netværket. -SV for at finde ud af hvilke tjenester og versioner, der er på de åbne porte, og -p option skal pege på de porte, der skal tjekkes.

Da Tor er meget langsom, kan en fuld scan simpelthen tage to eller tre evigheder, og det gider Johnny Hacker ikke vente på. Han foretrækker at plukke enkelte interessante porte ud.

```
# Proxychains nmap -sT -PN -n -sV -p 21, 22, 53, 80, 110, 139, 143, 443
192.10.10.10
```

```
bt ~ # proxychains nmap -sT -PN -n -sV -p 21,22,53,80,110,139,143,443
10.10.10.100
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 4.60 ( http://nmap.org ) at 2008-07-12 17:08 GMT
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:21-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:22-<--denied
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:53-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:80-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:443-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:110-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:143-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:139-<--timeout
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:21-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:53-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:80-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:110-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:143-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-10.10.10.100:443-<><>-OK
```

Johnny Hacker har nu nogle yderst vigtige oplysninger til rådighed. Han leder efter huller, som kan udnyttes på afstand.

Det aner ham, at systemet ikke er up to date, siden der er her tale om en standard version af Apache … Men hvad er det for en version, lige præcis? Forskellige versioner har forskellige huller.

Johnny bestemmer sig for at undersøge det via et værktøj der hedder socat. Han kunne godt bruge Torify, som er et værktøj fra Tor-bundlen, men socat er smartere, da man kan have en konstant forbindelse til målets server og køre en hele masse probes igennem.

Johnny Hacker forbinder sig til Apache serveren, og han taster: HEAD / HTTP/1.0 og trykker <ENTER> to gange:

Nc 127.0.0.1 8080
HEAD / HTTP/1.0

```
bt ~ # nc 127.0.0.1 8080
HEAD / HTTP/1.0

HTTP/1.1 200 OK

Date: Wed, 14 Dec 2011 18:36:23 GMT
Server: Apache/2.2.2 (Debian)
X-Powered-By: PHP/5.2.17-0.dotdeb.0
X-FIRSTBaseRedirector: LIVE
Vary: Accept-Encoding
Connection: close
Content-Type: text/html; charset=UTF-8
```

Bingo!

Apache version 2.2.2 er en oldgammel version af serveren, og Johnny Hacker har et hav af værktøjer, der kan benyttes til formålet …

Så er det bare om at gå i gang. Om et øjeblik er den server på knæ.

Jeg kan forsikre dig, at det tager Johnny Hacker mindre tid at nå hertil, end det har taget mig at skrive det. Det hele sker i løbet af ganske få minutter.

Det er vigtigt for alle at forstå disse værktøjer. At forstå, hvad de kan gøre. At forstå, hvordan de kan skade. Det er yderst vigtigt, at du kender de huller, du kan have på din hjemmeside eller på din server, og at du lapper disse. Enten tager du dig af disse huller, eller også vil en hacker gøre det for dig snart eller siden. Men det er ikke sikkert, at du vil bryde dig om det.

Nu vil vi gå i dybden med de forskellige værktøjer, og det, der kan ske, og især hvad du kan gøre ved det, så du ikke bliver det næste offer.

Lad os gå i gang.