Deakin Wargames

Problem Solving Task

NOTE – This was produced as part of a Uni assignment. Some elements contained were included as they were required as part of the assignment brief and/or marking rubric. Additionally, the assignment had a strict word count, some elements had to be sacrificed.

This is my work, produced for the Bachelor of Cyber Security. Universities take plagiarism very seriously and automated tools are very effective at identifying the source of information. I am happy for this to be used as a source for learning. Keep in mind, I am learning also, some info may not be correct, you should always confirm with reputable sources. This information is likely out of date as it was produced some time between 2018 – 2021.

	Information Leakage	Level No(s)	War Game Level 0

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/index.html

Description
Login credentials for level 1 were stored in plain text as a comment in the page's html file. This is viewable by any user.

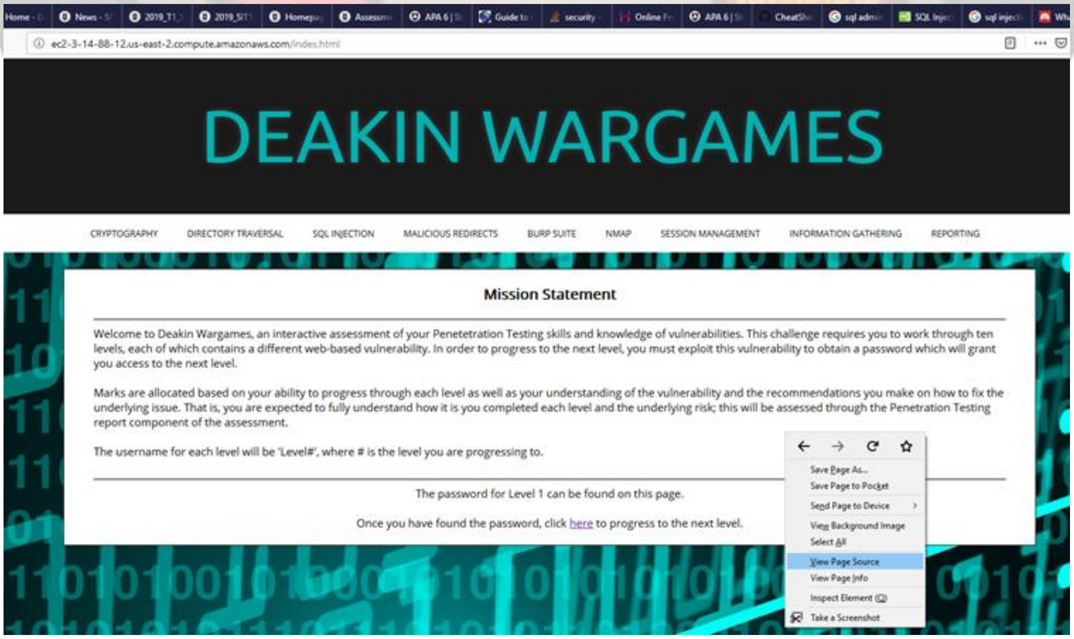
Observation
Level 0 provided a header 'Deakin Wargames' and links to various information pages, this is the same for all levels.
The body of the page detailed the mission statement and described how the challenge worked. It stated the password for level 1 could be found on the page; it was not visible.
A link was provided for the next level, this link is the same for all levels.

Steps to complete;

1. Right click and select **View Page Source** – this opens a new tab in HTML. Figure 1.
2. Visually inspect all elements, searching for access credentials.
3. Credentials were found in a comment. Figure 2.
4. Test access to level 1.

Method. After reading the page the next step was to view the page source and look for clues. This is identified some leaked information, being the credentials for level 1 in plain text.

Screenshot



Mission Statement

Welcome to Deakin Wargames, an interactive assessment of your Penetetration Testing skills and knowledge of vulnerabilities. This challenge requires you to work through ten levels, each of which contains a different web-based vulnerability. In order to progress to the next level, you must exploit this vulnerability to obtain a password which will grant you access to the next level.

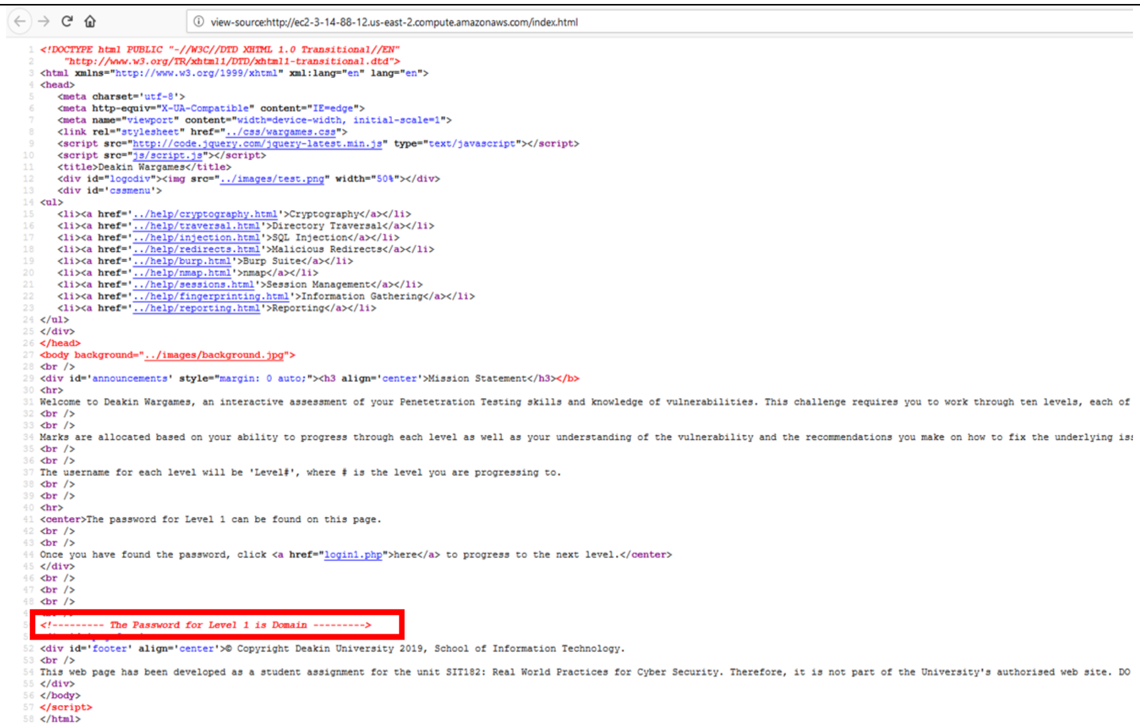
Marks are allocated based on your ability to progress through each level as well as your understanding of the vulnerability and the recommendations you make on how to fix the underlying issue. That is, you are expected to fully understand how it is you completed each level and the underlying risk; this will be assessed through the Penetration Testing report component of the assessment.

The username for each level will be 'Level#', where # is the level you are progressing to.

The password for Level 1 can be found on this page.

Once you have found the password, click here to progress to the next level.

Fig 1: Selecting view page source.


view-source:http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/index.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
   <meta charset='utf-8'>
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <link rel="stylesheet" href="../css/wargames.css">
   <script src="http://code.jquery.com/jquery-latest.min.js" type="text/javascript"></script>
   <script src="js/script.js"></script>
   <title>Deakin Wargames</title>
   <div id="logodiv"><img src="../images/test.png" width="50%"></div>
   <div id='cssmenu'>
<ul>
   <li><a href='../help/cryptography.html'>Cryptography</a></li>
   <li><a href='../help/traversal.html'>Directory Traversal</a></li>
   <li><a href='../help/injection.html'>SQL Injection</a></li>
   <li><a href='../help/redirects.html'>Malicious Redirects</a></li>
   <li><a href='../help/burp.html'>Burp Suite</a></li>
   <li><a href='../help/nmap.html'>nmap</a></li>
   <li><a href='../help/sessions.html'>Session Management</a></li>
   <li><a href='../help/fingerprinting.html'>Information Gathering</a></li>
   <li><a href='../help/reporting.html'>Reporting</a></li>
</ul>
</div>
</head>
<body background="../images/background.jpg">
<br />
<div id='announcements' style="margin: 0 auto;"><h3 align='center'>Mission Statement</h3></b>
<hr>
Welcome to Deakin Wargames, an interactive assessment of your Penetetration Testing skills and knowledge of vulnerabilities. This challenge requires you to work through ten levels, each of
<br />
<br />
Marks are allocated based on your ability to progress through each level as well as your understanding of the vulnerability and the recommendations you make on how to fix the underlying is:
<br />
<br />
The username for each level will be 'Level#', where # is the level you are progressing to.
<br />
<br />
<hr>
<center>The password for Level 1 can be found on this page.
<br />
<br />
Once you have found the password, click <a href="login1.php">here</a> to progress to the next level.</center>
</div>
<br />
<br />
<br />
<!--------- The Password for Level 1 is Domain --------->
<div id='footer' align='center'>© Copyright Deakin University 2019, School of Information Technology.
<br />
This web page has been developed as a student assignment for the unit SIT182: Real World Practices for Cyber Security. Therefore, it is not part of the University's authorised web site. DO
</div>
</body>
</script>
</html>
```

Fig 2: HTML with access credentials.

<table>
<tr><td>Level Credentials –
Level 0 Password: Domain</td><td><em>Impact Analysis</em>
A user would not need any specific training to view the page source and read and understand the credentials. Any user can view this data and gain access to level 1 without difficulty.</td></tr>
<tr><td colspan="2">Recommendation
Remove the comment.
Disable right click access on the page to prevent easy access to the pages source.</td></tr>
</table>

	Information Leakage	Level No(s)	War Game Level 1

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level1/level1.php

Description
Login credentials for level 1 were stored in plain text as a comment in the page's html file. This is viewable by any user who knows how to use a command to open the page source (ctrl u) or inspect elements using F12.

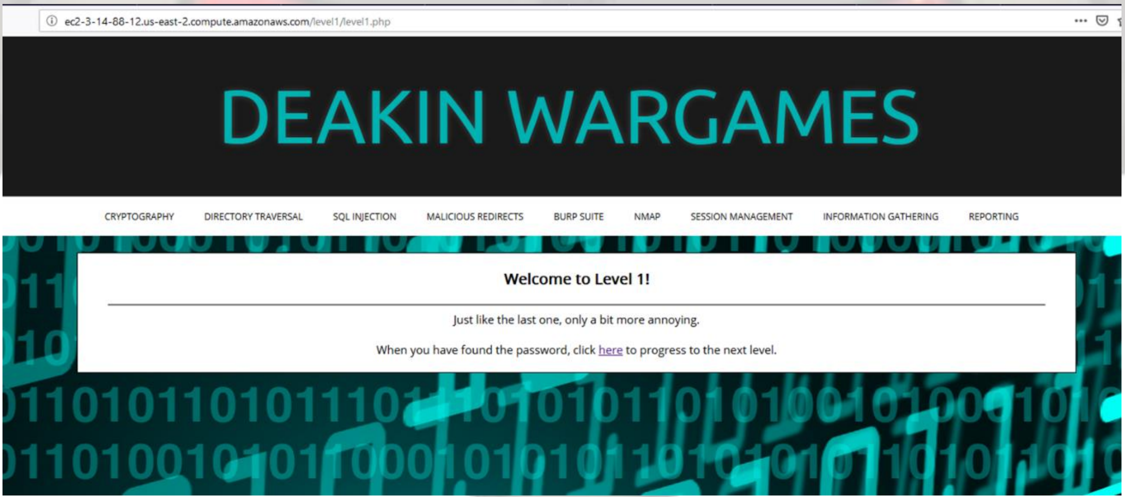
Observation
As seen in figure 3; the body of the page detailed that this vulnerability was 'just like the last one'.

Steps to complete;

1. (Windows10 enter ctrl u to view the page source – this opens a new tab in HTML.
2. Visually inspect all elements, searching for access credentials – found in a comment. Figure 4.
3. Test access to level 2.

Method. The first step was to attempt to right click to view the page source, this was disabled. Second use Ctrl u command to view source. This identified some leaked information; the credentials for level 2 in plain text.

Screenshot

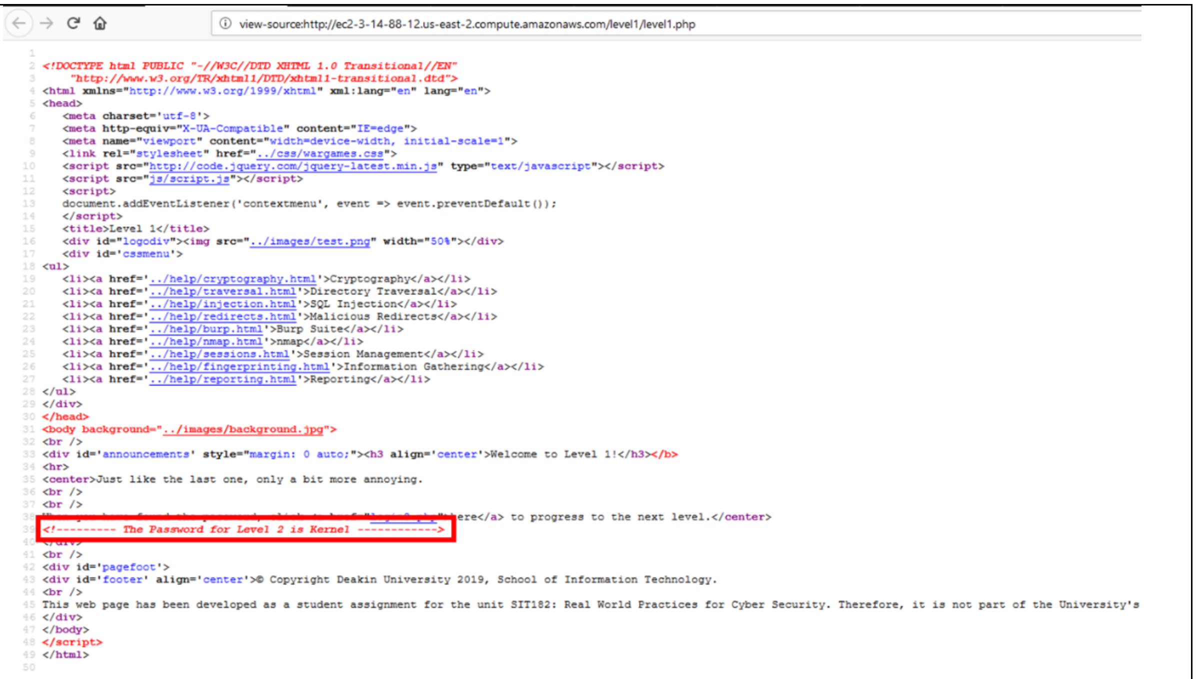


Welcome to Level 1!

Just like the last one, only a bit more annoying.

When you have found the password, click here to progress to the next level.

Fig 3: Level 1 challenge.



view-source:http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level1/level1.php

```html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
   <meta charset='utf-8'>
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <link rel="stylesheet" href="../css/wargames.css">
   <script src="http://code.jquery.com/jquery-latest.min.js" type="text/javascript"></script>
   <script src="js/script.js"></script>
   <script>
   document.addEventListener('contextmenu', event => event.preventDefault());
   </script>
   <title>Level 1</title>
   <div id="logodiv"><img src="../images/test.png" width="50%"></div>
   <div id='cssmenu'>
<ul>
   <li><a href='../help/cryptography.html'>Cryptography</a></li>
   <li><a href='../help/traversal.html'>Directory Traversal</a></li>
   <li><a href='../help/injection.html'>SQL Injection</a></li>
   <li><a href='../help/redirects.html'>Malicious Redirects</a></li>
   <li><a href='../help/burp.html'>Burp Suite</a></li>
   <li><a href='../help/nmap.html'>nmap</a></li>
   <li><a href='../help/sessions.html'>Session Management</a></li>
   <li><a href='../help/fingerprinting.html'>Information Gathering</a></li>
   <li><a href='../help/reporting.html'>Reporting</a></li>
</ul>
</div>
</head>
<body background="../images/background.jpg">
<br />
<div id='announcements' style="margin: 0 auto;"><h3 align='center'>Welcome to Level 1!</h3></b>
<hr>
<center>Just like the last one, only a bit more annoying.
<br />
<br />
When you have found the password, click <a href="login2.php">here</a> to progress to the next level.</center>
<!--------- The Password for Level 2 is Kernel ------------>
</div>
<br />
<div id='pagefoot'>
<div id='footer' align='center'>© Copyright Deakin University 2019, School of Information Technology.
<br />
This web page has been developed as a student assignment for the unit SIT182: Real World Practices for Cyber Security. Therefore, it is not part of the University's
</div>
</body>
</script>
</html>

```

Fig 4: Level 1 HTML with access credentials.

Level Credentials – Level 1 Password: Kernel	*Impact Analysis* A user would not need any specific training to read and understand the credentials. Any user can view this data and gain access to level 2 without difficulty.
Recommendation Remove the comment from the html file. Login credentials should not be written anywhere.	

	Directory Traversal	Level No(s)	War Game Level 2

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level2/level2.php

Description
As described by OWASP (2015) and Acunetix (n.d.) login credentials were able to be obtained due to a vulnerability in the web site allowing the user to navigate to files outside the root directory. Inadequate security settings allowed the user to browse the files looking for sensitive information.

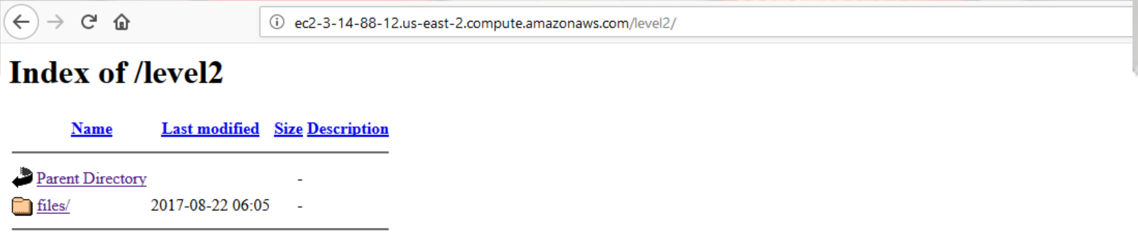
Observation
The body of the page indicates that the password cannot be found on this page.

Steps to complete;

1. Modify the URL to http://ec2-3-14-88-12.us-east2.compute.amazonaws.com/level2/level2.php**/../** and execute it – This tell the site to go up a directory and opens the index of files for level 2. Figure 5.
2. Open the folder labelled 'files/' – this presents all the stored files to the user. Figure 6.
3. Open the file labelled 'members.txt' – This provides the access credentials for level 3. Figure 7.
4. Use credentials to test access to level 3.

Method. Initially '/../' was added to the end of the URL to see if there was anything at /level2/. This returned access to the level 2 files. Then simply browsed thee files.

Screenshot

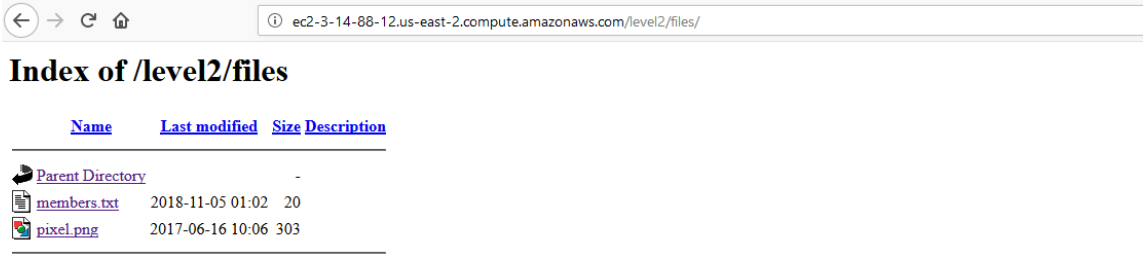


ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level2/

Index of /level2

Name	Last modified	Size	Description
Parent Directory		-	
files/	2017-08-22 06:05	-	

Fig 5: Root directory for level 2.

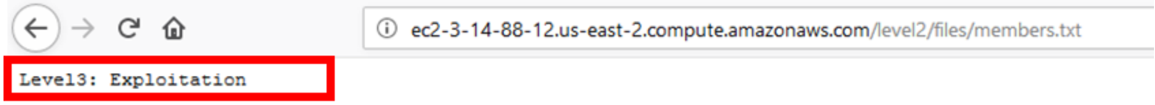


ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level2/files/

Index of /level2/files

Name	Last modified	Size	Description
Parent Directory		-	
members.txt	2018-11-05 01:02	20	
pixel.png	2017-06-16 10:06	303	

Fig 6: Files contained within level2/files.



ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level2/files/members.txt

```
Level3: Exploitation
```

Fig 7: Level 3 access credentials.

Level Credentials – Level 2 Password: Exploitation	*Impact Analysis* An attacker gaining access to information outside of the root folder could be used to escalate their privileges or obtain sensitive information.

Recommendation

Acunetix (n.d.) suggests ensure the latest version of web server software is installed and all patches have been applied.

OWASP (2015) and Acunetix (n.d.) suggest user input should be filtered and only known ‘good data’ should be accepted.

	Directory Traversal	Level No(s)	War Game Level 3

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level3/level3.php

Description
OWASP (2015) and Acunetix (n.d.) describe that login credentials were able to be obtained due to a vulnerability in the web site allowing the user to navigate to files outside the root directory. This attack required the use of another page on the site that was know to be vulnerable (Level 2).

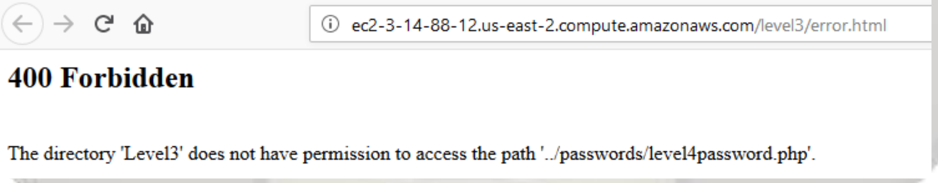
Observation
The body of the page provides a link that leads to the next password. Once clicked it returns an error with the password file path, '../passwords/level4password.php'.

Steps to complete;

1. Click the link provided to get the error due to inadequate permissions. Returned the password path, '../passwords/level4password.php'. Figure 8.
2. Modify the URL for level 2 to gain access to the level 4 password file from level 2. http://ec2-18-191-11-183.us-east-2.compute.amazonaws.com/level2/passwords/level4password.php.
3. Executing this URL will return the access credentials for level 4. Figure 9.
4. Test access to level 4.

Method. The link for the password was followed and this provided the file location. Trial and error of entering /level3/ and ../../../ etc. Finally, level 2 was tried as it had been previously used to obtain a password file.

Screenshot

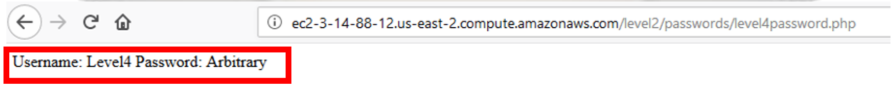


ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level3/error.html

400 Forbidden

The directory 'Level3' does not have permission to access the path '../passwords/level4password.php'.

Fig 8: Error due to insufficient permissions.



ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level2/passwords/level4password.php

Username: Level4 Password: Arbitrary

Fig 9: Level 4 access credentials.

Level Credentials – Level 3 Password: Arbitrary	*Impact Analysis* An attacker gaining access to information outside of the root folder could be used to escalate privileges or obtain sensitive information.

Recommendation
OWASP (2015) and Acunetix (n.d.) suggest user input should be filtered and only known 'good data' should be accepted.
Debug information should not be returned as an error.
Check all pages for the vulnerability.

	Weak Encryption	Level No(s)	War Game Level 4

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level4/level4.php

Description
The password was protected by a weak encoding method known as Caesar Cipher which provides no security against decryption as it a substitution cypher. Each letter is substitute with a letter X number away. This can be easily solved without a computer (Shift +3 = Cat => Fdw).

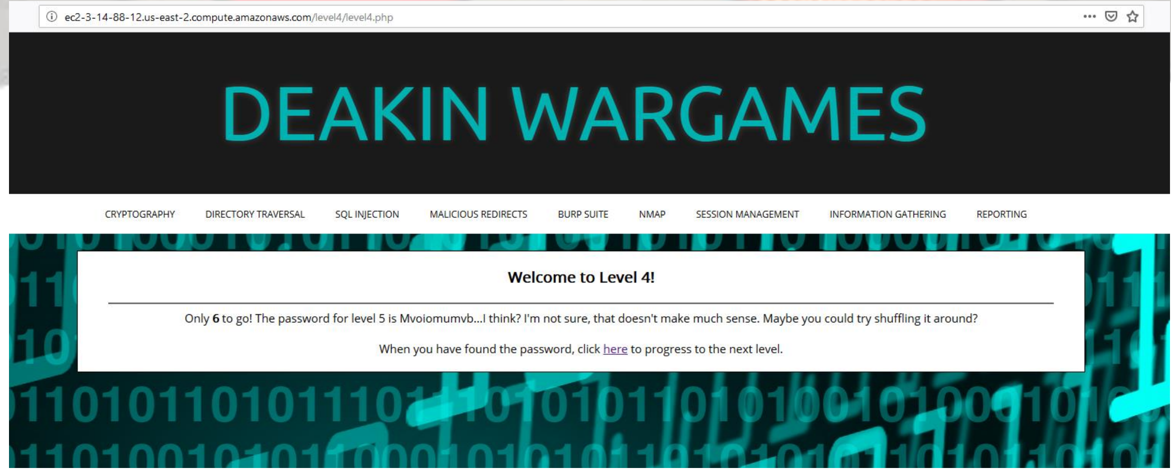
Observation
Figure 10 shows the body of the page which provides the password for level 5 as ‘Mvoiomumvb’. The clue is that it needs to be ‘shuffled’ around.

Steps to complete;

1. Enter the encrypted password, ‘Mvoiomumvb’, into a decoding tool. An online tool such as https://cryptii.com/pipes/caesar-cipher will work. Figure 11.
2. Set to decode and set the shift to + 8 – this will work backwards as we are decoding.
3. Ensure the decode creates a word – Engagement.
4. Test access to level 5.

Method. The encrypted word was run through online scrabble generators. This produced no viable words. The code was then entered into the Caesar Cypher decoding tool and the shift was set to + 1 then incremented by 1 until a word was formed. Plus 8 produced the word Engagement.

Screenshot

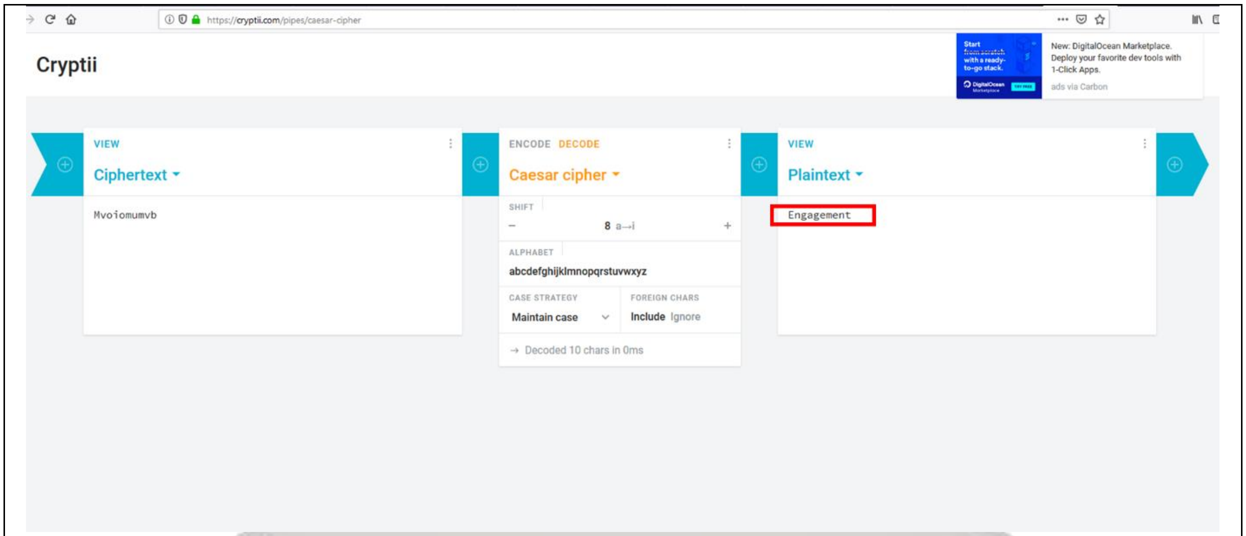


Welcome to Level 4!

Only **6** to go! The password for level 5 is Mvoiomumvb...I think? I'm not sure, that doesn't make much sense. Maybe you could try shuffling it around?

When you have found the password, click here to progress to the next level.

Fig 10: level 4 encrypted password.


https://cryptii.com/pipes/caesar-cipher
Cryptii
Start from scratch with a ready-to-go stack.
DigitalOcean Marketplace
New: DigitalOcean Marketplace. Deploy your favorite dev tools with 1-Click Apps.
ads via Carbon
VIEW
ENCODE DECODE
VIEW

Ciphertext

```
Mvoiomumvb
```

Caesar cipher

SHIFT: 8 a→i

ALPHABET: abcdefghijklmnopqrstuvwxyz

CASE STRATEGY: Maintain case

FOREIGN CHARS: Include Ignore

→ Decoded 10 chars in 0ms

Plaintext

```
Engagement
```

Fig 11: Decryption using online Caesar Cypher decoding tool.

Level Credentials –	*Impact Analysis*
Level 4 Password: Engagement	Easy to solve this with minimal resources. Once decoded the user has full access to level 5 with these credentials.

Recommendation

Hash the password so it cannot be easily identified, then encrypt the Hash.

	Weak Encryption	Level No(s)	War Game Level 5

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level5/level5.php

Description
The password is protected by an unsalted MD5 hash. This simply masks the plain text password. A single MD5 hash can be decoded through a dictionary attack as the same input will produce the same hashed output.

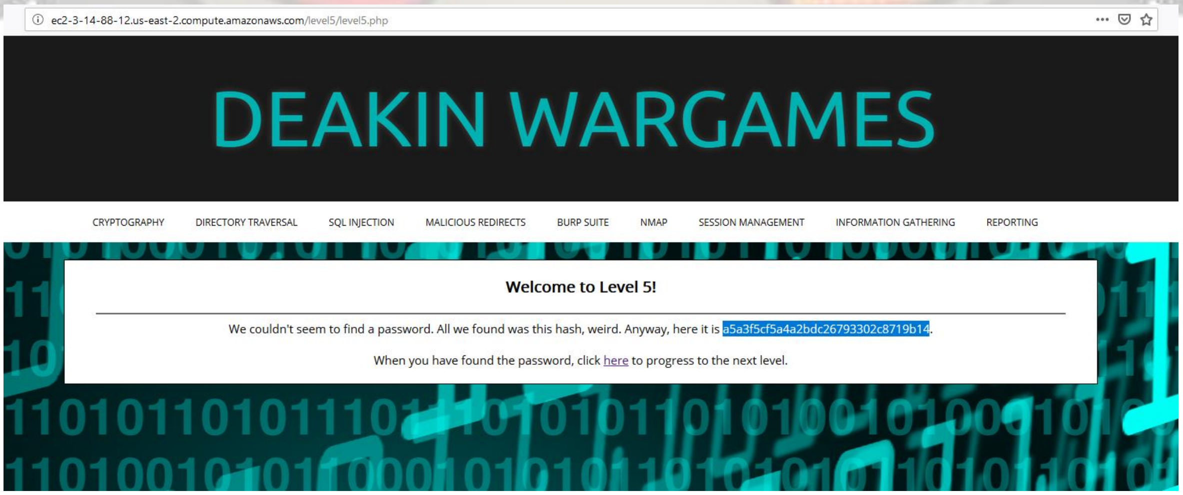
Observation
Level 5 provided a header 'Deakin Wargames' and links to various information pages. Figure 12 shows the body of the page which provides a string of 32 characters (a5a3f5cf5a4a2bdc26793302c8719b14) and identifies it as a Hash.

Steps to complete;

1. Enter the hash, 'a5a3f5cf5a4a2bdc26793302c8719b14', into a hash cracking tool. An online tool such as https://crackstation.net/ and https://www.md5online.org/md5-decrypt.html. Figure 13.
2. These conduct a dictionary attack to identify a word that will produce the same hash – this returned Escalation. Figure 13.
3. Test access to level 6.

Method. The Hash contained 32 characters this indicated it was likely MD5. To be sure the hash through an online identifier. This returned a list with the most likely being MD5. The hash was then run through a generic tool that would detect the hash, https://crackstation.net/.

Screenshot

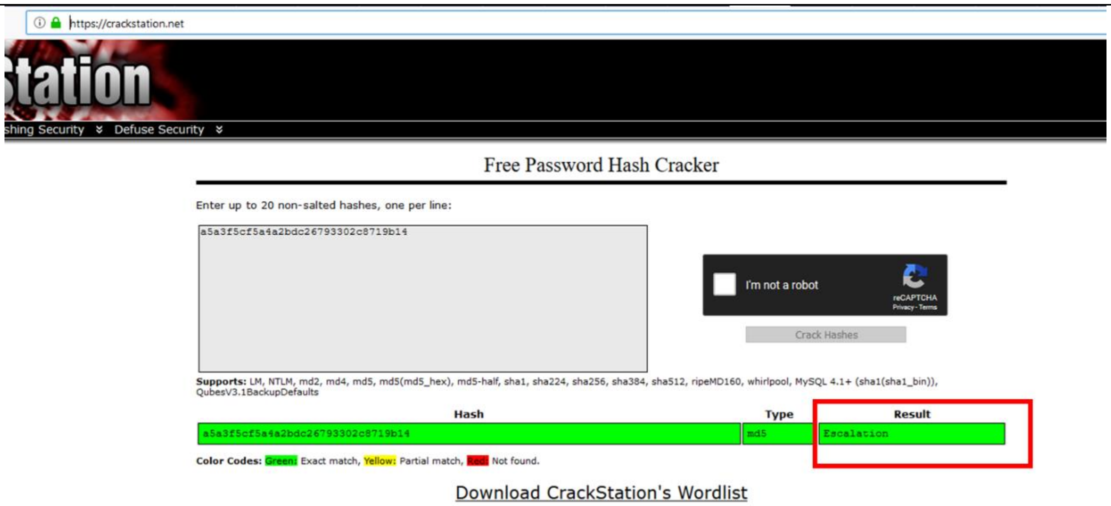


Welcome to Level 5!

We couldn't seem to find a password. All we found was this hash, weird. Anyway, here it is a5a3f5cf5a4a2bdc26793302c8719b14.

When you have found the password, click here to progress to the next level.

Fig 12: Level 5 password hash.



https://crackstation.net

Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
a5a3f5cf5a4a2bdc26793302c8719b14
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
a5a3f5cf5a4a2bdc26793302c8719b14	md5	Escalation

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Download CrackStation's Wordlist

Fig 13: Hash cracking tool – credentials identified.

Level Credentials –	*Impact Analysis*
Level 5 Password: Escalation	Online Hashing tools can detect the type of hash and decode it. Once decoded the user has full access to level 6. The user will also be able to decrypt other passwords quickly.

Recommendation

Salt the passwords before Hashing. As per Arias (2018) This is a random string of characters appended to the password that will increase its length and complexity.

	Weak Encryption	Level No(s)	War Game Level 6

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level6/level6.php

Description
The key is protected by an unsalted MD5 hash. Additional encryption layers were added, but these were reversable, exposing the unsalted hash. This exposes the password to being decrypted.

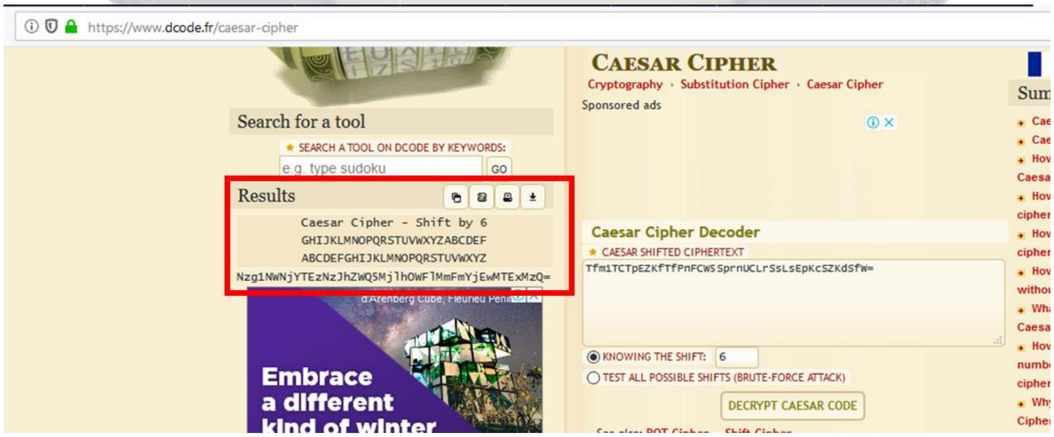
Observation
The body of the page provides an encrypted password, 'a3 fb cc 42 29 e9 43 d0 db f5 c9 9e a5 bd 44 1d' and that it is encrypted using AES, DES or 3DES.
Four decryption keys were provided and their method of encryption. MD5, base64 encoding and Caesar Cipher shift 6.

Steps to complete;

1. Keys are decrypted by working backwards through the provided encryption method.
2. Enter key two 'Tfm1TCTpEZKfTfPnFCW5SprnUCLrSsLsEpKcSZKdSfW=' into https://www.dcode.fr/caesar-cipher. Shift by 6 – Outputs 'Nzg1NWNjYTEzNzJhZWQ5MjlhOWFlMmFmYjEwMTExMzQ=**'.** Figure 14.
3. Enter 'Nzg1NWNjYTEzNzJhZWQ5MjlhOWFlMmFmYjEwMTExMzQ=' into https://www.base64decode.org/ - Outputs '7855cca1372aed929a9ae2afb1011134'. Figure 15.
4. Enter '7855cca1372aed929a9ae2afb1011134' into https://www.md5online.org/md5-decrypt.html - Outputs 'Pineapple'. Figure 16.
5. Enter 'Pineapple' as the key to the password into http://aes.online-domain-tools.com/ – this outputs 'Spectre'. Figure 17.
6. Test access to level 7.

Method. The tools and procedure to decode came from the SIT182 lecture notes for week 9, Hutchinson (2019). It was decoded by working backwards through the provided encoding method.

Screenshot

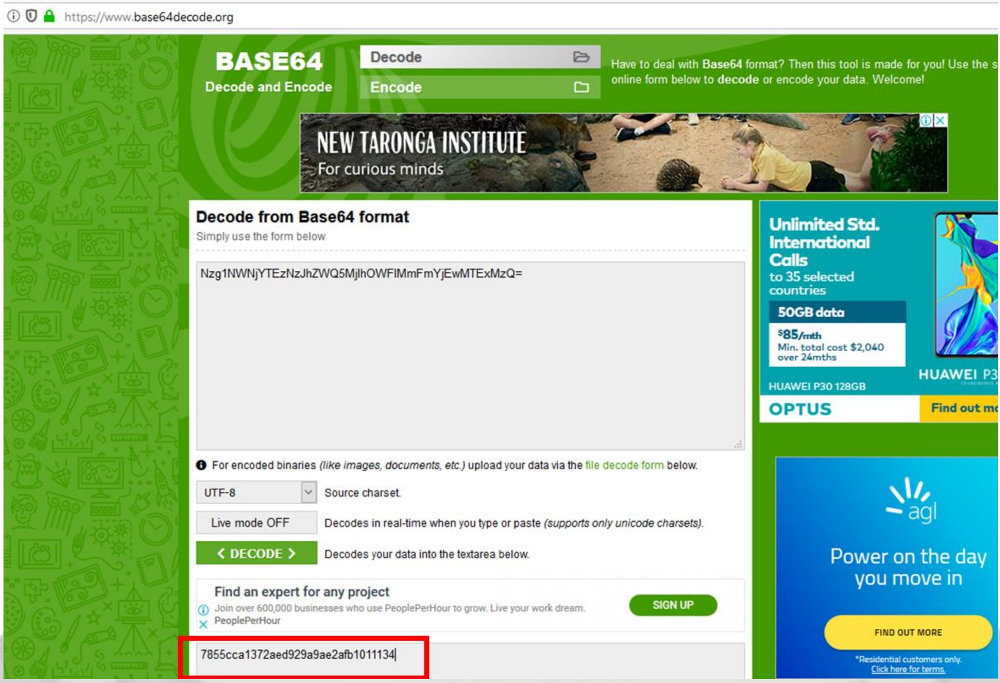

https://www.dcode.fr/caesar-cipher
CAESAR CIPHER
Cryptography › Substitution Cipher › Caesar Cipher
Sponsored ads
Search for a tool
SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type sudoku
GO
Results
Caesar Cipher Decoder
CAESAR SHIFTED CIPHERTEXT
DECRYPT CAESAR CODE
Embrace a different kind of winter

```
Caesar Cipher - Shift by 6
GHIJKLMNOPQRSTUVWXYZABCDEF
ABCDEFGHIJKLMNOPQRSTUVWXYZ
Nzg1NWNjYTEzNzJhZWQ5MjlhOWFlMmFmYjEwMTExMzQ=
```

CAESAR SHIFTED CIPHERTEXT: Tfm1TCTpEZKfTfPnFCW5SprnUCLrSsLsEpKcSZKdSfW=

● KNOWING THE SHIFT: 6
○ TEST ALL POSSIBLE SHIFTS (BRUTE-FORCE ATTACK)

Fig 14: Caesar Cypher shift and output.



https://www.base64decode.org

BASE64 Decode and Encode

Decode
Encode

Have to deal with **Base64** format? Then this tool is made for you! Use the s
online form below to **decode** or encode your data. Welcome!

Decode from Base64 format

Simply use the form below

```
Nzg1NWNjYTEzNzJhZWQ5MjlhOWFlMmFmYjEwMTExMzQ=
```

For encoded binaries *(like images, documents, etc.)* upload your data via the file decode form below.

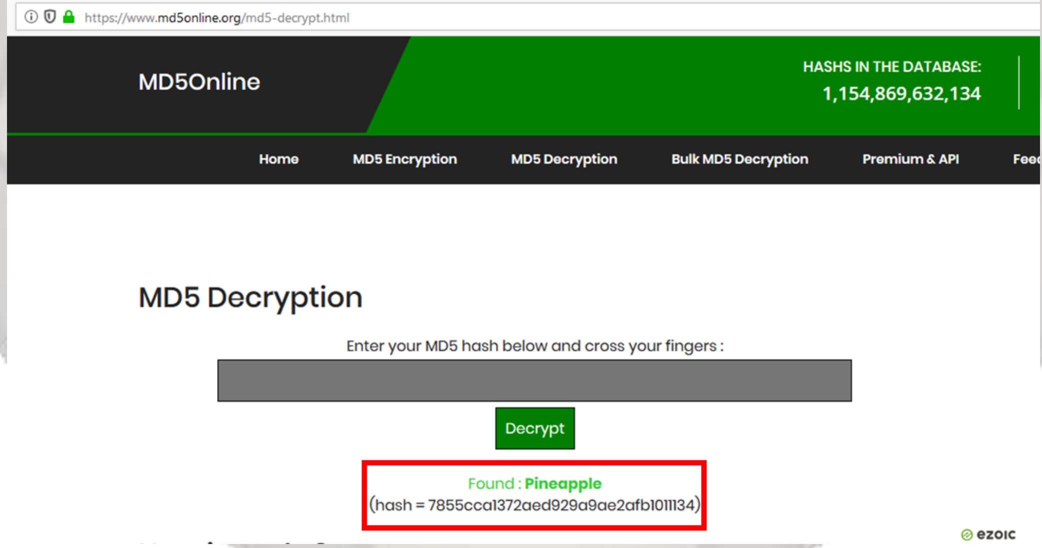
UTF-8 — Source charset.

Live mode OFF — Decodes in real-time when you type or paste *(supports only unicode charsets)*.

< DECODE > — Decodes your data into the textarea below.

```
7855cca1372aed929a9ae2afb1011134
```

Fig 15: Decoding Base64 and output.



https://www.md5online.org/md5-decrypt.html

MD5Online

HASHS IN THE DATABASE:
1,154,869,632,134

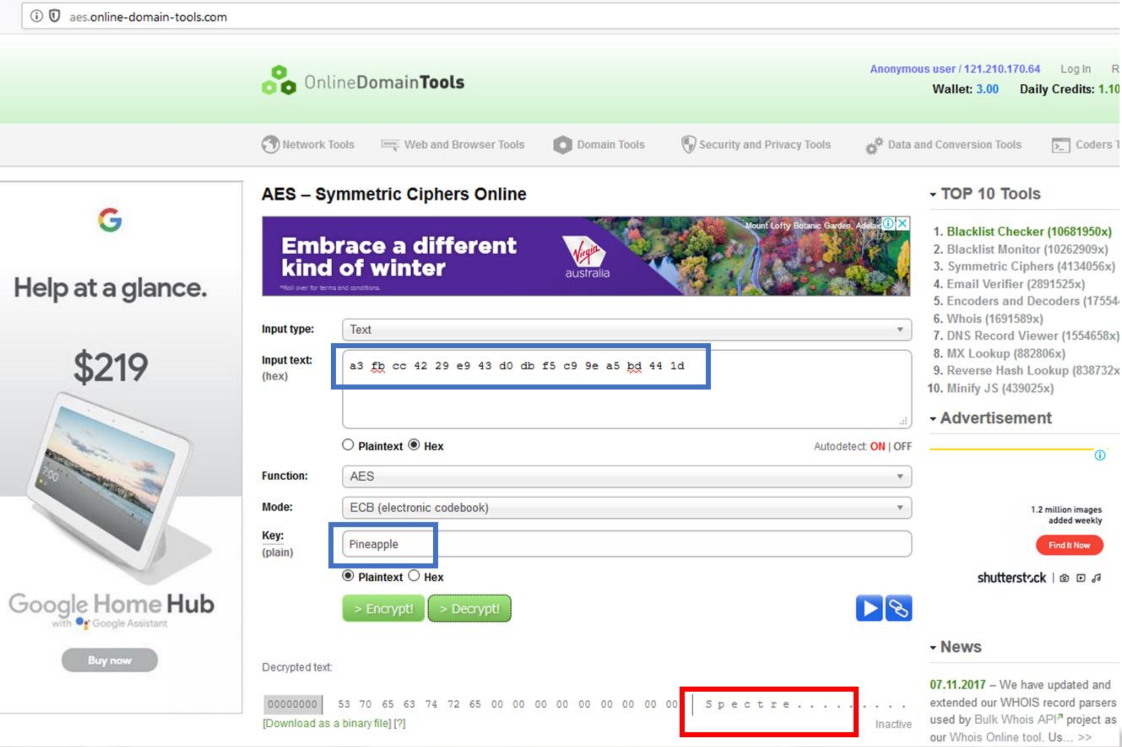
MD5 Decryption

Enter your MD5 hash below and cross your fingers :

Decrypt

Found : **Pineapple**
(hash = 7855cca1372aed929a9ae2afb1011134)

Fig 16: MD5 dictionary decryption.


aes.online-domain-tools.com
OnlineDomainTools
Anonymous user / 121.210.170.64 Log In
Wallet: 3.00 Daily Credits: 1.10
Network Tools
Web and Browser Tools
Domain Tools
Security and Privacy Tools
Data and Conversion Tools
Coders T
Help at a glance.
$219
Google Home Hub
Buy now
Embrace a different kind of winter
australia
TOP 10 Tools
1. Blacklist Checker (10681950x)
2. Blacklist Monitor (10262909x)
3. Symmetric Ciphers (4134056x)
4. Email Verifier (2891525x)
5. Encoders and Decoders (17554
6. Whois (1691589x)
7. DNS Record Viewer (1554658x)
8. MX Lookup (882806x)
9. Reverse Hash Lookup (838732x
10. Minify JS (439025x)
Advertisement
1.2 million images added weekly
Find It Now
shutterstock
News
07.11.2017 – We have updated and extended our WHOIS record parsers used by Bulk Whois API project as our Whois Online tool. Us... >>

AES – Symmetric Ciphers Online

Input type: Text

Input text: (hex)

```
a3 fb cc 42 29 e9 43 d0 db f5 c9 9e a5 bd 44 1d
```

○ Plaintext ● Hex Autodetect: ON | OFF

Function: AES

Mode: ECB (electronic codebook)

Key: (plain) Pineapple

● Plaintext ○ Hex

> Encrypt! > Decrypt!

Decrypted text:

```
00000000  53 70 65 63 74 72 65 00 00 00 00 00 00 00 00 00  | S p e c t r e . . . . . . . . .
```

[Download as a binary file] [?] Inactive

Fig 17: Decrypting password.

Level Credentials – Level 6 Password: Spectre	*Impact Analysis* With this information a user can gain access to level7. Additionally, all other passwords on the system are compromised.
Recommendation Salt the passwords before Hashing. As per Arias (2018) this is a random string of characters appended to the password that will increase its length and complexity.	

	Cookie Manipulation	Level No(s)	War Game Level 7

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level7/level7.php

Description
Cookie Manipulation – Due to inadequate server security settings the user is able to use the URL to adjust the incoming authenticated cookie to allow access to the login credentials.

Observation
The body of the page provides a notice that the session is not authorised. It asks you navigate to the page using an authorised session cookie.

Steps to complete;

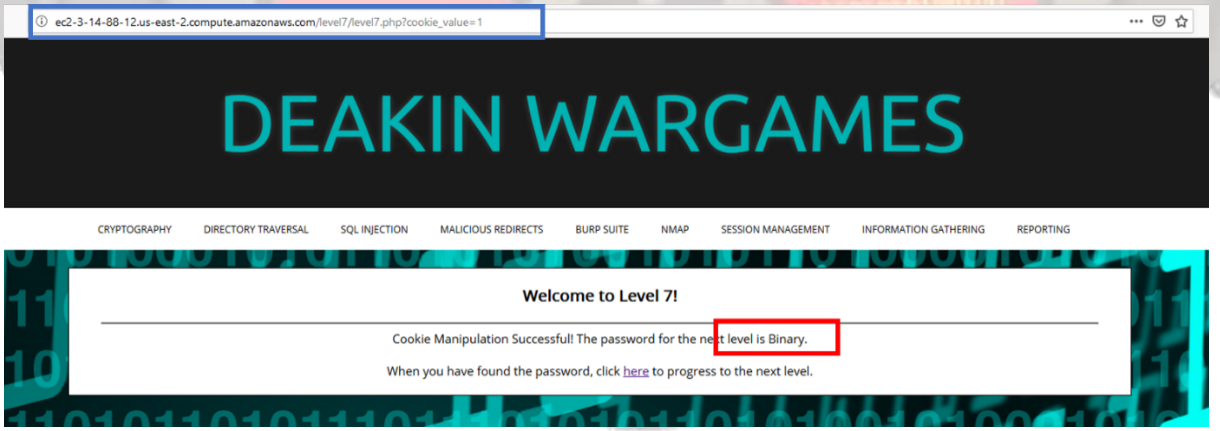
1. Identify the cookie and authentication value.

Modify the URL to change the value to '1' and execute. http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level7/level7.php**?cookie_value=1**.

2. Confirm manipulation success as seen in figure 18.
3. Use the credentials to test access to level 8.

Method. Initially the cookies were checked. The note, 'navigate to the page' suggested the cookie could be modified when the page was requested. Using the information provide by Deakin University (2019) the URL was modified and re-submitted, http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level7/level7.php?cookie_value=1.

Screenshot



ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level7/level7.php?cookie_value=1

DEAKIN WARGAMES

Welcome to Level 7!

Cookie Manipulation Successful! The password for the next level is Binary.

When you have found the password, click here to progress to the next level.

Fig 18: Successful cookie manipulation.

Level Credentials – Level 7 Password: Binary	*Impact Analysis* This method allows the user to authenticate their session and gain access to level 8. A user that understands session management will be able to circumvent restricted pages.

Recommendation
Ensure adequate security settings.
Use user input sanitisation

	SQL Injection	Level No(s)	War Game Level 8

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level8/level8.php

Description
An attacker users specific SQL inputs to attempt to bypass security measures of a database or retrieve information. Poorly configures SQL databases and websites are vulnerable to this attack.

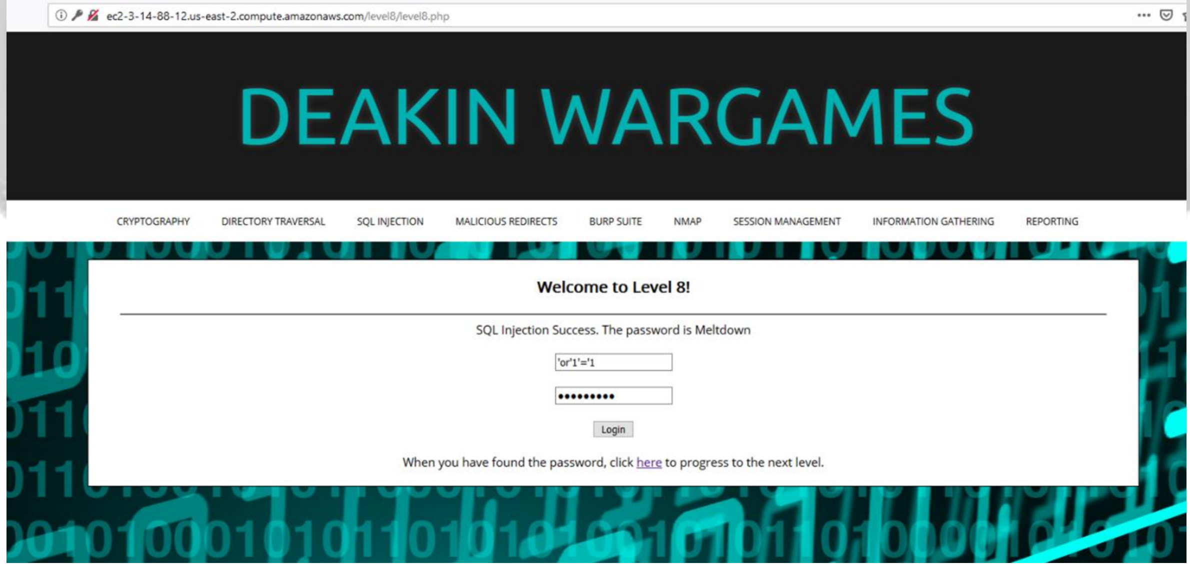
Observation
The body of the page provides two input fields, username and password with a login button.

Steps to complete;

1. Enter **'OR'1'='1** into the username and password field – The idea is to send a query that returns true to get data. 1 = 1 is always true. Figure 19.
2. Press login to execute this command – this returns the access credentials. Figure 20.
3. Use the credentials to test access to level 9.

Method. Acunetix (n.d.) provided examples of statements to inject to query a potentially vulnerable database to get a statement to return true, thus providing data. **'OR'1'='1** will always return true.

Screenshot



ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level8/level8.php

DEAKIN WARGAMES

Welcome to Level 8!

SQL Injection Success. The password is Meltdown

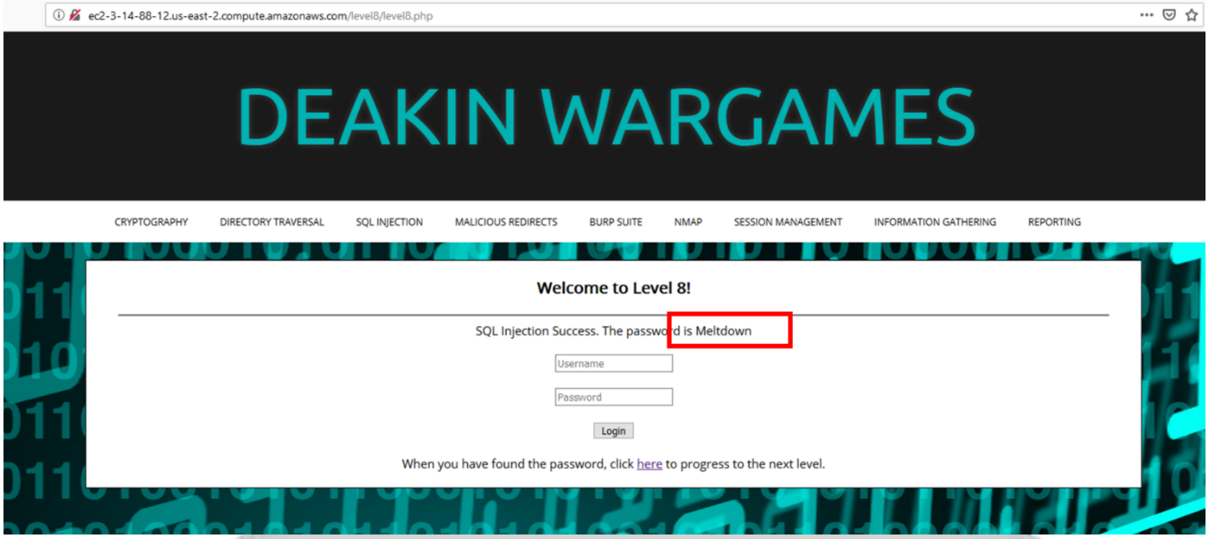
'or'1'='1

•••••••••

Login

When you have found the password, click here to progress to the next level.

Fig 19: SQL injection statement.

<table>
<tr><td colspan="2">
ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level8/level8.php
DEAKIN WARGAMES

Welcome to Level 8!
SQL Injection Success. The password is Meltdown

When you have found the password, click here to progress to the next level.
Fig 20: Successful execution of SQL injection.</td></tr>
<tr><td>Level Credentials –
Level 8 Password: Meltdown</td><td><i>Impact Analysis</i>
A user with knowledge of SQL will be able to obtain sensitive information from the database. An attacker could impersonate any other user.</td></tr>
<tr><td colspan="2">Recommendation
Sanitise user input to know good requests.</td></tr>
</table>

	SQL Injection	Level No(s)	War Game Level 9

Affected resources:
http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level9/level9.php

Description
An attacker uses SQL inputs to attempt to retrieve sensitive information from poorly configured SQL databases. Websites that return SQL syntax errors are vulnerable to this attack.

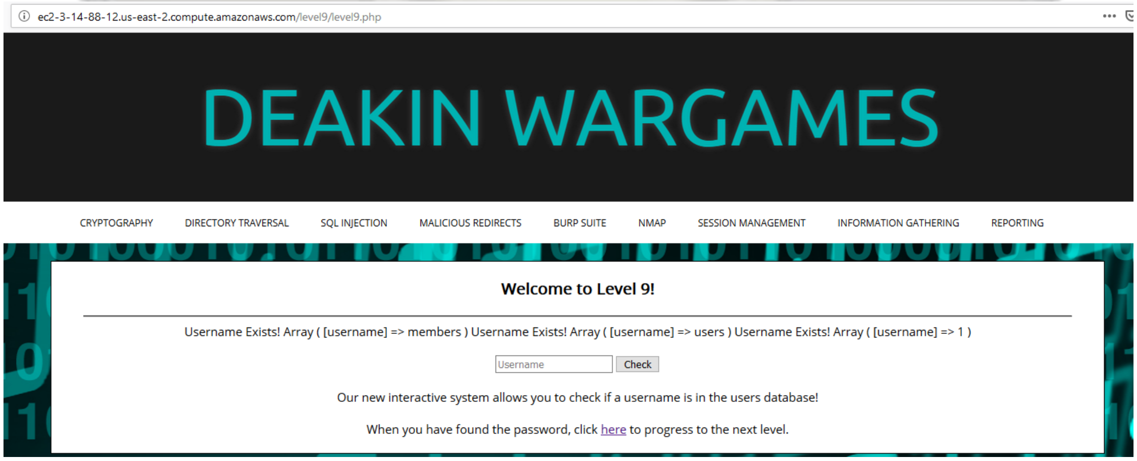
Observation
The body of the page provides the ability to check a username is in the database. A single input box and check button.

Steps to complete;

1. Identify the database names using ' UNION SELECT table_schema FROM information_schema.tables UNION SELECT '1 – Returns list of databases.
2. Identify the tables inside ‘Wargames’ database using ' UNION SELECT table_name FROM information_schema.tables WHERE table_schema = 'Wargames' UNION SELECT '1 – Returns list of tables names. Figure 21.
3. Identify the column names in ‘users’ table using ' UNION SELECT column_name FROM information_schema.columns WHERE table_name = 'users' UNION SELECT '1 – Returns list of columns. Figure 22.
4. Identify ‘username’ and ‘password’ form ‘users’ table using ' UNION SELECT CONCAT(username,”:”,password) FROM users UNION SELECT ‘1 – Returns concatenated username and password; Level10:Mimikatz. Figure 23.
5. Test access to level 10.

Method. Solved through trial and error of SQL statements based of the examples provided by Aaron(2018). The injection statements from SQLi were modified to get a successful response.

Screenshot



ec2-3-14-88-12.us-east-2.compute.amazonaws.com/level9/level9.php

DEAKIN WARGAMES

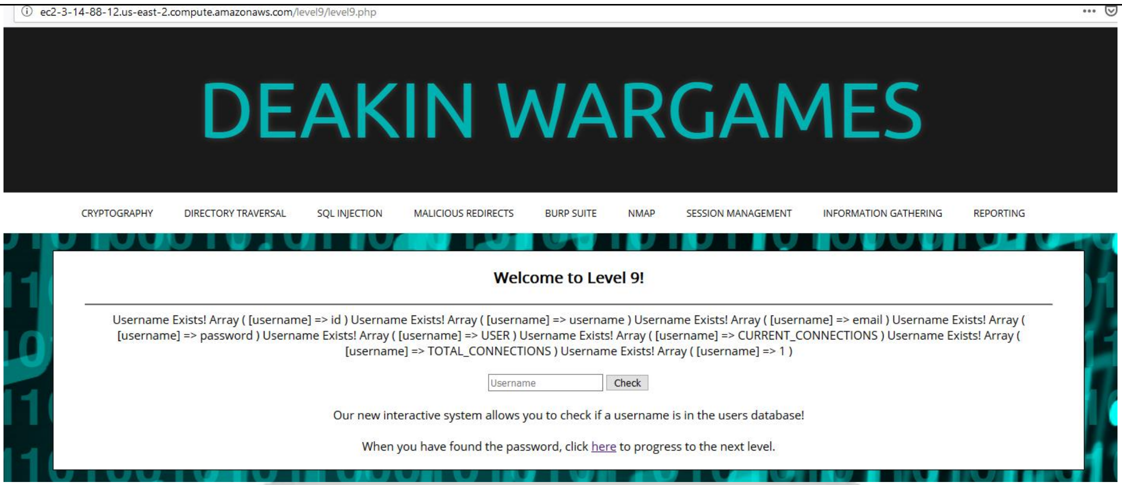
Welcome to Level 9!

Username Exists! Array ([username] => members) Username Exists! Array ([username] => users) Username Exists! Array ([username] => 1)

Our new interactive system allows you to check if a username is in the users database!

When you have found the password, click here to progress to the next level.

Fig 21: Table names in ‘Wargames’ database.



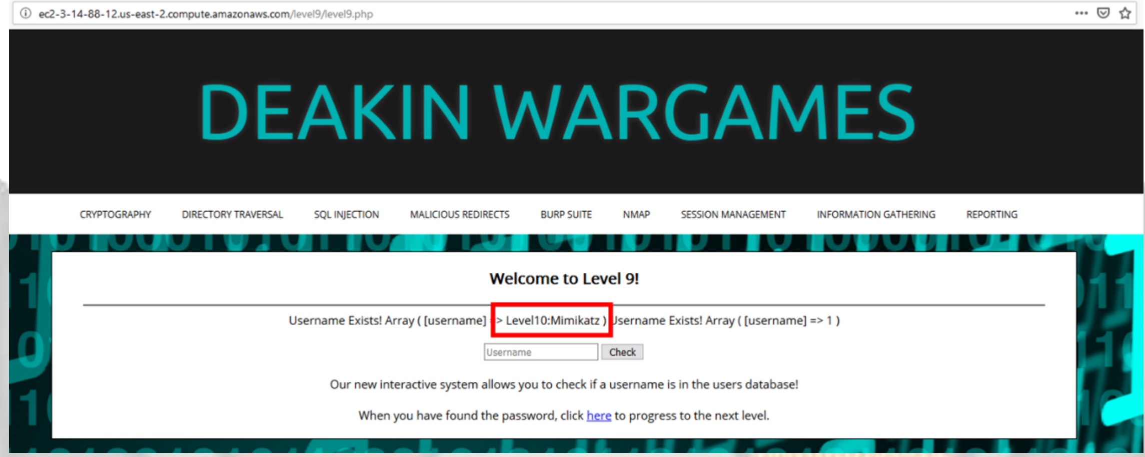
Welcome to Level 9!

Username Exists! Array ([username] => id) Username Exists! Array ([username] => username) Username Exists! Array ([username] => email) Username Exists! Array ([username] => password) Username Exists! Array ([username] => USER) Username Exists! Array ([username] => CURRENT_CONNECTIONS) Username Exists! Array ([username] => TOTAL_CONNECTIONS) Username Exists! Array ([username] => 1)

Our new interactive system allows you to check if a username is in the users database!

When you have found the password, click here to progress to the next level.

Fig 22: Column names in 'users' table.



Welcome to Level 9!

Username Exists! Array ([username] => Level10:Mimikatz) Username Exists! Array ([username] => 1)

Our new interactive system allows you to check if a username is in the users database!

When you have found the password, click here to progress to the next level.

Fig 23: Successful SQL injection.

Level Credentials – Level 9 Password: Mimikatz	*Impact Analysis* A user with knowledge of SQL will be able obtain sensitive information from the database. An attacker could impersonate any other user.
Recommendation Sanitise user input to know good requests. Do not return SQL syntax back to the user as an error.	

References

1. OWASP, (2015). Path Traversal. Retrieved from https://www.owasp.org/index.php/Path_Traversal

2. Acunetix, (n.d.). Directory Traversal Attacks. Retrieved from https://www.acunetix.com/websitesecurity/directory-traversal/

3. Arias, D. (2018 May 03). Adding Salt to Hashing: A Better Way to Store Passwords [Blog post]. Retrieved from https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/

4. OWASP, (2018). Guide to Cryptography. Retrieved from https://www.owasp.org/index.php/Guide_to_Cryptography

5. Hutchinson, D., SIT182 and Real World Practices For Cyber Security, Deakin University, PowerPoint slides, 02 May 2019

6. Deakin University, (2019). Session Management. Retrieved from http://ec2-3-14-88-12.us-east-2.compute.amazonaws.com/help/sessions.html

7. Acunetix, (n.d.). What is SQL Injection (SQLi) and How to Prevent It. Retrieved from https://www.acunetix.com/websitesecurity/sql-injection/

8. Aaron, P. [hackHappy]. (2018, Jun 22). SQL Injection Attack Tutorial (2019) [Video file]. Retrieved from https://www.youtube.com/watch?v=WFFQw01EYHM