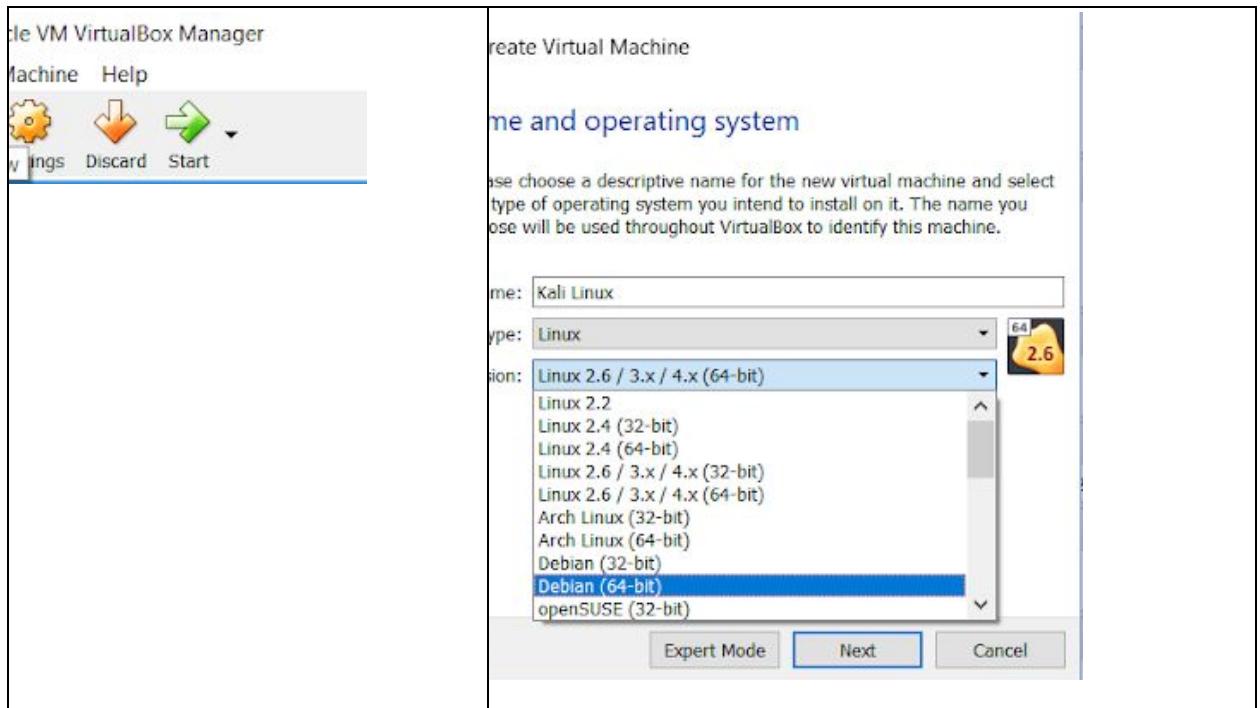


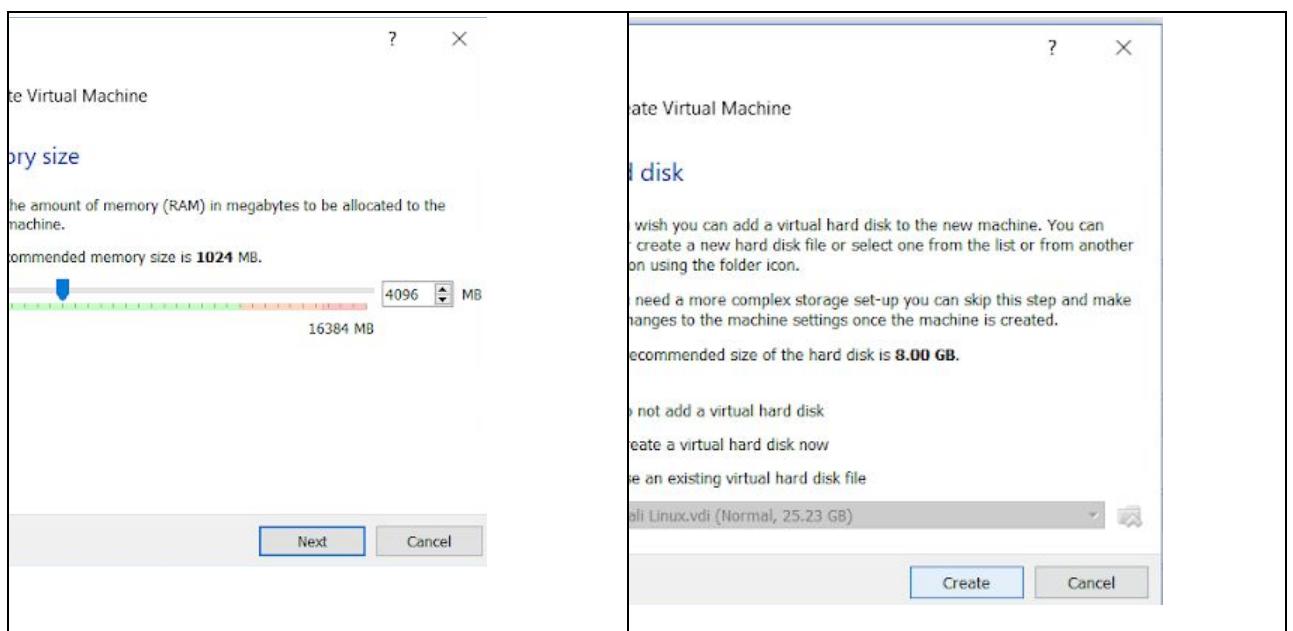
To exploit the vulnerabilities in the websites hosted in linux based Metasploitable OS, install Oracle VirtualBox VM and setup two linux environments:

Environment Setup:

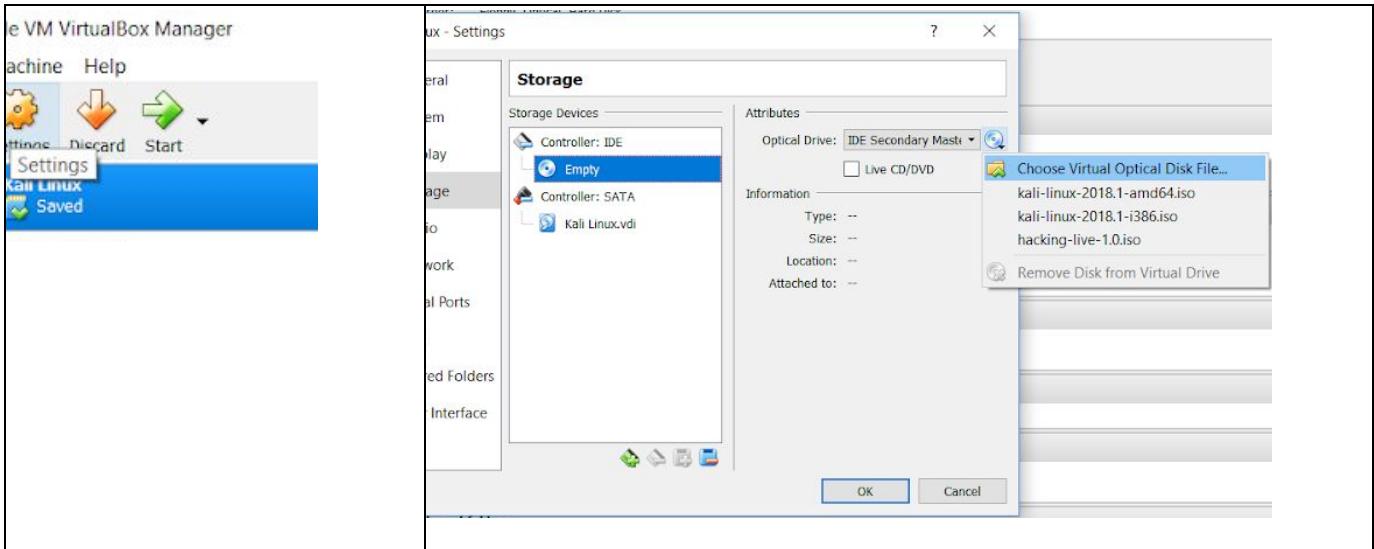
1. Kali Linux : install the ISO file from <https://www.kali.org/downloads/> website.
 - a. After downloading the ISO file follow below steps to create the VM:
 - b. Click the new button and on the ‘create virtual machine’ pop-up, give the name of the virtual machine, select the type of OS as ‘Linux’ and version as ‘Debian (64-bit)’ and click Next.



- c. Give the memory size as 4096 MB and click next, in the next pop-up, select ‘create virtual hard disk’ and then select create, to create the virtual machine.

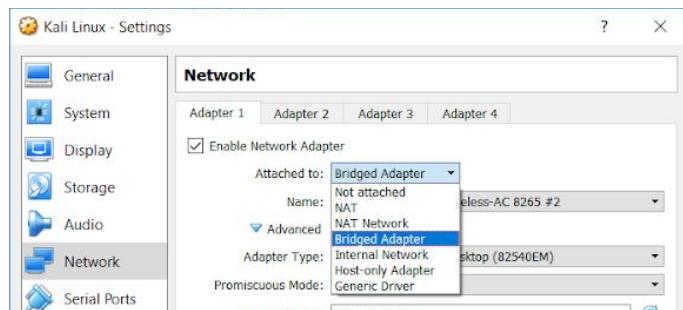


d. To add the ISO file to the Kali Linux VM, select the VM and click on settings as shown below:



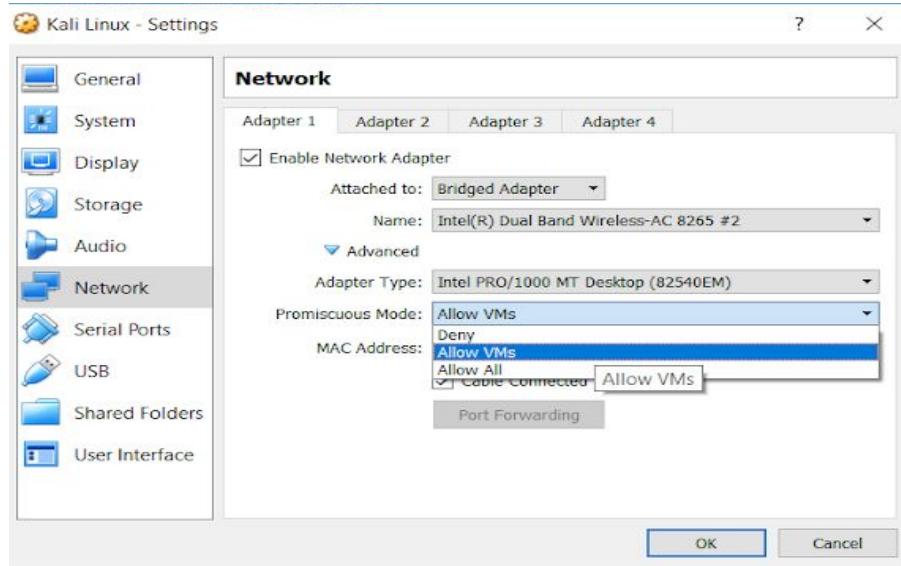
In the settings window, go to Storage, select click on the 'Empty' settings under the Controller, then click on the Disc image in the attributes section to choose the ISO file.

- e. If you were to setup the VM with 64-bit ISO file and the option to create a 64-bit Debian Linux VM is not showing up in the VirtualBox, change the BIOS setup of your machine as explained in this article here:
<https://superuser.com/questions/866962/why-does-virtualbox-only-have-32-bit-option-no-64-bit-option-on-windows-7>
- f. Once you have attached the ISO file to the VM, click on Network in the Settings window and change the "Attached to" attribute from NAT to Bridged Adapter.



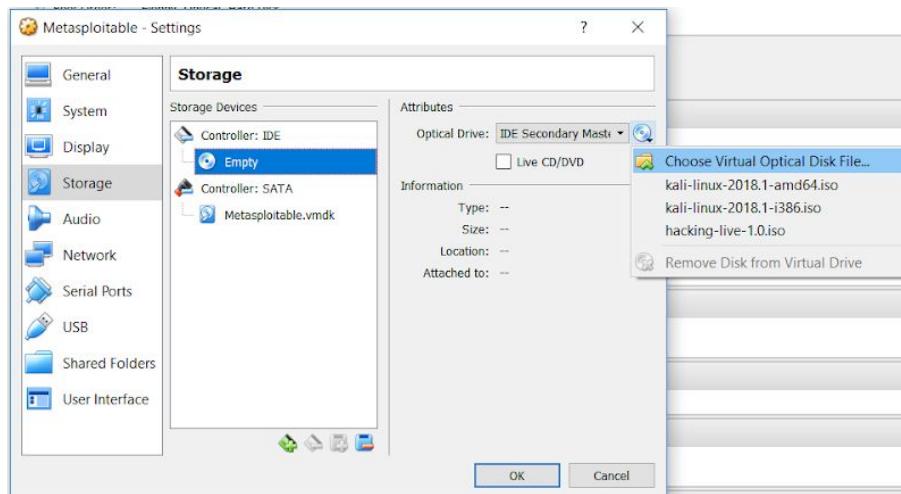
Then click on 'Advanced' attribute as shown below to change the Promiscuous mode from "Deny" to "Allow VMs".

These settings will let the Kali Linux to interact with other VMs.



- g. Click OK and start the VM to get started. Once you start the machine, choose appropriate settings to boot up the VM.
- Similar to the step 1 above, click on ‘New’ button on VirtualBox to create Metasploitable linux VM. Make sure to download the Metasploitable VM image from the below link:
<http://sourceforge.net/projects/metasploitable/files/Metasploitable2/>

Attach the .vmdk file to the VM in the storage section of the VM settings and start the machine:



- Once you start the metasploitable, use msfadmin as login username and password to login into the system. Type in “ifconfig” command as shown below to know the IP address of the VM, which will be used to exploit the websites hosted in the metasploitable VM from Kali Linux:

```
To access official Ubuntu documentation, please visit:  

http://help.ubuntu.com/  

No mail.  

msfadmin@metasploitable:~$ ifconfig  

eth0      Link encap:Ethernet HWaddr 08:00:27:08:e7:a8  

          inet addr:11.219.148.114 Bcast:11.219.255.255 Mask:255.255.128.0  

          inet6 addr: fe80::a00:27ff:fe08:e7a8/64 Scope:Link  

            UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  

            RX packets:40 errors:0 dropped:0 overruns:0 frame:0  

            TX packets:70 errors:0 dropped:0 overruns:0 carrier:0  

            collisions:0 txqueuelen:1000  

            RX bytes:12775 (12.4 KB) TX bytes:7864 (7.6 KB)  

            Base address:0xd010 Memory:f0000000-f0020000  

lo       Link encap:Local Loopback  

          inet addr:127.0.0.1 Mask:255.0.0.0  

          inet6 addr: ::1/128 Scope:Host  

            UP LOOPBACK RUNNING MTU:16436 Metric:1  

            RX packets:96 errors:0 dropped:0 overruns:0 frame:0  

            TX packets:96 errors:0 dropped:0 overruns:0 carrier:0  

            collisions:0 txqueuelen:0  

            RX bytes:21437 (20.9 KB) TX bytes:21437 (20.9 KB)  

msfadmin@metasploitable:~$  

msfadmin@metasploitable:~$
```

We would be using this highlighted IP address to exploit the websites.

4. Once you start the Kali Linux VM, hit this IP address in a browser to list the websites hosted on this VM:

Metasploitable2 - Linux - Mozilla Firefox

Metasploitable2 - Linux

11.219.148.114

Most Visited: Most Visited, Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter

This network may require you to login to use the internet. Show Login Page

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

We would be exploiting Mutillidae website. Below are the list of exploits and their possible fixes:

Exploits & Fixes:

1. Cross Site Request Forgery - CSRF

Go to add-to-your-blog.php and turn on the intercept of burp suite.

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

Developed by Adrian "Irongeek" Crenshaw and Jeremy Druin

Browser: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0
PHP Version: 5.2.4-2ubuntu5.10
The newest version of Mutillidae can downloaded from Irongeek's Site

Once the blog entry is submitted, we will see the POST request in burp suite.

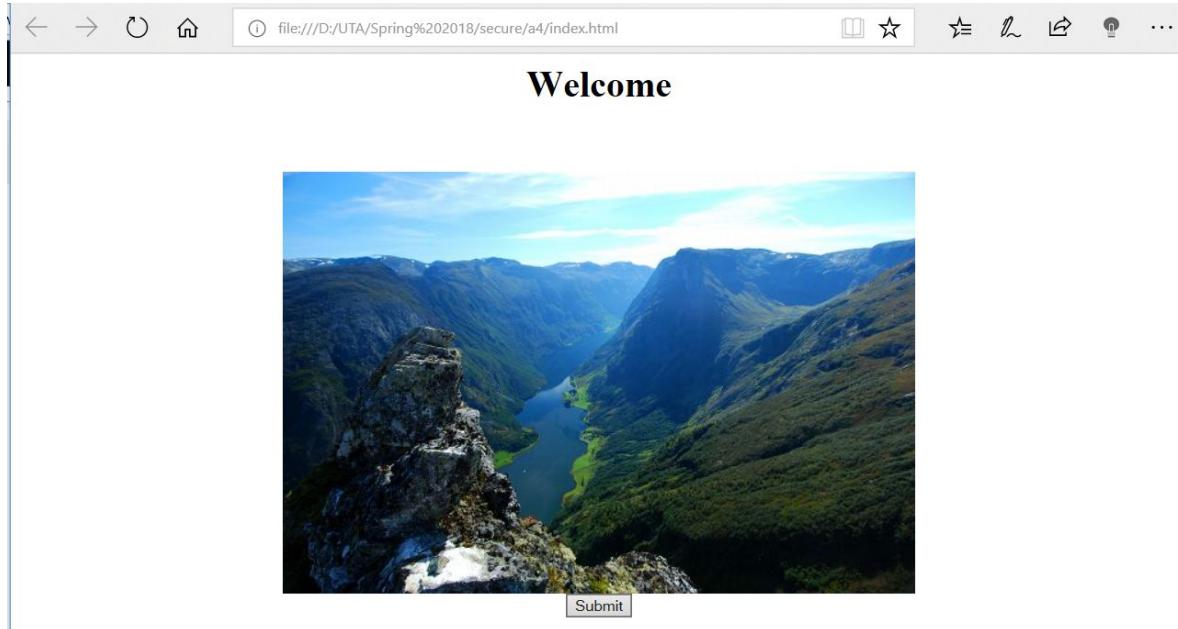
```
POST /mutillidae/index.php?page=add-to-your-blog.php HTTP/1.1
Host: 192.168.0.74
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.0.74/mutillidae/index.php?page=add-to-your-blog.php
Cookie: PHPSESSID=bdfc010843e166f55ef00c1eb92fd6e
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 101

csrf-token=SecurityIsDisabled&blog_entry=burpsuite&add-to-your-blog-php-submit-button=Save+Blog+Entry
```

In “Params” tab we will get the 3 body parameters name and value.

Type	Name	Value
URL	page	add-to-your-blog.php
Cookie	PHPSESSID	bdfc010843e166f55ef00c1eb92fd6e
Body	csrf-token	SecurityIsDisabled
Body	blog_entry	burpsuite
Body	add-to-your-blog-php-submit-button	Save Blog Entry

Create an html file with hidden input fields.



The hidden input tags contain the name of body parameters we obtained from burp suite.

```
1 <html>
2   <center>
3     <body><h1>Welcome</h1>
4       <br><br>
5       </body><br>
6
7
8     <!-- Form Post -->
9     <form name="csrf" action="http://192.168.0.74/mutillidae/index.php?page=add-to-your-blog.php"
10    method="POST">
11      <input type="hidden" name='csrf-token' value=''>
12      <input type="hidden" name='blog_entry' value='Got hacked'>
13      <input type="hidden" name='add-to-your-blog-php-submit-button' value='Save Blog Entry'>
14      <input type="submit" value="submit">
15    </form>
16    <!-- <script>document.csrf.submit();</script>-->
17    <!-- End Form Post -->
18 </html>
```

Upon click submit button the message will be posted in the add-to-your-blog.php



Back

Add New Blog Entry

[View Blogs](#)

Add blog for anonymous

Note: ,,<i>,</i>,<u> and </u> are now allowed in blog entries

[View Blogs](#)

3 Current Blog Entries

	Name	Date	Comment
1	anonymous	2018-04-20 11:56:26	Got hacked
2	anonymous	2018-04-20 11:53:39	burpsuite
3	anonymous	2009-03-01 22:27:11	An anonymous blog? Huh?

Solution:

This exploit results in making the user submit sensitive information to unauthorized users. It can be prevented by following one of the measures listed below:

- **Synchronizer Tokens** – It is unique per sessions and has large random values. Generated by a cryptographically secure random number generator.
- **Double Cookie Defense** – Send a random value in both a cookie and as a request parameter. Have the server verify if the cookie and the request value is the same.
- **Encrypted token pattern** – After authentication, have the server generate a unique. Token comprised of the user's ID, a timestamp values and a cryptographic nonce, using a unique key available only to the server.
- **Custom Request Header** – A non-complex defense suited for AJAX endpoints. It relies on same origin policy (SOP) restrictions that only JavaScript can be used to add a custom header and only within its origin. By default, browsers do not allow JavaScript to perform Cross-origin requests.

2. SQL Injection

Give username input as ' or 1=1 -- ' and password can be blank.

 **Mutillidae: Born to be Hacked**

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls ▾
OWASP Top 10 ▾
Others ▾
Documentation ▾
Resources ▾



Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and [these Mozilla Add-ons](#)


@webpwnized

View your details

 Back

Please enter username and password to view account details

Name
Password

Dont have an account? [Please register here](#)

The given input for name lists all the users from the table users and their passwords.

View your details

 Back

Please enter username and password to view account details

Name
Password

Dont have an account? [Please register here](#)

Results for . 16 records found.

```

Username=admin
Password=adminpass
Signature=Monkey!

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

```

Solution:

Keep the web application in sync by maintaining validation checks of all the inputs that are sent to a database.

3. Cross Site Scripting – XSS

To exploit this kind of flaw, go to DNS lookup page and using developer tools, increase the length of input textbook to add the script, as shown below:

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Logged In Admin: admin (Monkey!)

[Home](#) [Logout](#) [Toggle Hints](#) [Toggle Security](#) [Reset DB](#) [View Log](#) [View Captured Data](#)

DNS Lookup

 [Back](#)

Who would you like to do a DNS lookup on?

Enter IP or hostname

Hostname/IP

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Logged In Admin: admin (Monkey!)

Home Logout Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls ▾
OWASP Top 10 ▾
Others ▾
Documentation ▾
Resources ▾

 Back

DNS Lookup

Who would you like to do a DNS lookup on?

Enter IP or hostname

Hostname/IP

Lookup DNS

Site hacked...err...quality-tested with Samural WTF, Backtrack, Firefox, Burn Suite

Inspector Console Debugger Performance Memory Network Search HTML Rules Computed Animations Fonts

```
<tr><td><td><td><td>
```

DOM Tree:

- <tr><td><td><td><td>
- <tr><td><td><td><td>
- <td class="label">Hostname/IP</td>
- <td><input id="idTargetHostInput" name="target_host" size="100" type="text">
- </td>
- </tr>
- <tr><td><td><td><td>
- <tr><td><td><td><td>

Elements:

- Rules
- Computed
- Animations
- Fonts

Global Styles (global-styles.css):

```
table.main-table-frame { border-collapse: collapse; border-spacing: 0px; }
```

Inherited from HTML:

```
html { }
```

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Logged In Admin: admin (Monkey!)

[Home](#) [Logout](#) [Toggle Hints](#) [Toggle Security](#) [Reset DB](#) [View Log](#) [View Captured Data](#)

DNS Lookup

 [Back](#)

Who would you like to do a DNS lookup on?

Enter IP or hostname

Hostname/IP

With the sample script we added as shown above, the below alert is thrown to display the cookie information:

DNS Lookup

username=admin; uid=1; PHPSESSID=48402c4c1208ef09b576d433d7474507

OK

Hostname/IR

Lookup DNS

Results for

Solution:

Keep the web application in sync by maintaining validation checks of all the inputs that are sent to a database.

4. Broken authentication and session management

Login as admin

Please sign-in

Name admin

Password ••••••••

Login

Dont have an account? [Please register here](#)

get the session id by opening developer tools and select network tab. Select cookies under network tabs.

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Logged In Admin: admin (Monkey!)

Core Controls OWASP Top 10 Others Documentation Resources

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

Status	Method	File	Domain	Cause	Type	Tran	Headers	Cookies	Params	Response	Timings
200	GET	global-styles.css	192.168.0.79	stylesheet	css			PHPSESSID: '48402c4c1208ef09b576d433d7474507'			

Filter cookies Request cookies

username: "admin"

Open a new window

The screenshot shows a web browser interface with the following details:

- Header:** Version: 2.1.19, Security Level: 0 (Hosed), Hints: Disabled (0 - I try harder), Not Logged In.
- Navigation:** Home, Login/Register, Toggle Hints, Toggle Security, Reset DB, View Log, View Captured Data.
- Main Content:** A banner says "Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10". Below it, "Latest Version / Installation" links are listed.
- Toolbars:** Inspector, Console, Debugger, Style Edi..., Performance, Memory, Network.
- Network Tab:** Net (selected), CSS, JS, Security, Logging, Server.
- Logs:** An error message: "unreachable code after return statement [Learn More]". Below it, a session ID is shown: "document.cookie =\"PHPSESSID=48402c4c1208ef09b576d433d7474507\"".
- Bottom:** A note: "Samurai WTF and Backtrack contains all the tools needed or you may build your own collection".

refresh the page and we can see the user logged as admin

The screenshot shows a web browser interface with the following details:

- Header:** Version: 2.1.19, Security Level: 0 (Hosed), Hints: Disabled (0 - I try harder), Logged In Admin: admin (Monkey!).
- Navigation:** Home, Logout, Toggle Hints, Toggle Security, Reset DB, View Log, View Captured Data.
- Main Content:** A banner says "Mutillidae: Born to be Hacked". Below it, "Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10" is displayed.
- Toolbars:** Inspector, Console, Debugger, Style Edi..., Performance, Memory, Network.
- Network Tab:** Net (selected), CSS, JS, Security, Logging, Server.
- Logs:** No visible errors or logs.
- Bottom:** A note: "Samurai WTF and Backtrack contains all the tools needed or you may build your own collection".

With this kind of exploits, attacker can re-use the user's session id to hack into the user's session and take control of their account.

Solution:

Attacker uses leaks and flaws in the authentication or session management functions to impersonate the users.

- **Password Strength** – Have restrictions based on a minimum size and complexity.
- **Password Use** – Have restrictions based on number of login attempts per unit time and log number of failed attempts.
- **Password Change Controls** – A single password change mechanism should be used wherever users can change a password, regardless of the situation. Users should always be required to provide both their old and new password when changing their password.
- **Password storage** – All passwords have to be stored in hashed or encrypted forms.
- **Session ID protection** – Select the session via SSL.
- **Browser Caching** – Authentication and Browser data should never be submitted as a part of GET. POST method should be used.

5. Insecure direct object reference

Go to developer tools and then change the value of one of the select tag as “/etc/passwd” as shown.

The screenshot shows a web application interface titled "Mutillidae: Born to be Hacked". At the top, it displays "Version: 2.1.19", "Security Level: 0 (Hosed)", "Hints: Enabled (1 - 5cript K1ddle)", and "Logged In Admin: admin (Monkey!)". Below the header are navigation links: Home, Logout, Toggle Hints, Toggle Security, Reset DB, View Log, and View Captured Data. A sidebar on the left contains links for Core Controls, OWASP Top 10, Others, Documentation, and Resources, along with a logo for Site Hacked. The main content area is titled "Hacker Files of Old" and contains a message: "Take the time to read some of these great old school hacker text files. Just choose one from the list and submit." A "Text File Name" input field is populated with "Intrusion Detection in Computers by Victor H. Marshall (January 29, 1991)". A "View File" button is present. Below the input field, a message says "For other great old school hacking texts, check out <http://www.textfiles.com/>". On the right, developer tools are open, showing the DOM structure and CSS styles. The DOM highlights the selected option in the dropdown menu, which has the value "/etc/passwd".

On click of View File button, list of passwords is displayed as shown below:

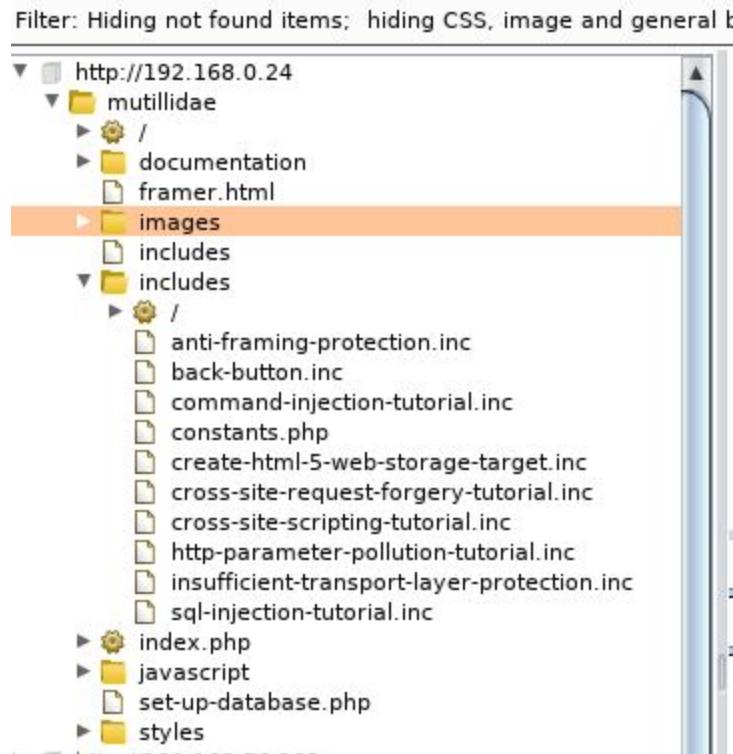
The screenshot shows the same application interface after clicking the "View File" button. The main content area now displays a large list of password entries under the heading "File: /etc/passwd". The list includes entries such as root:x:0:0:root:/root:/bin/bash, daemon:x:1:1:daemon:/usr/sbin:/bin/sh, bin:x:2:2:bin:/bin:/bin/sh, sys:x:3:3:sys:/dev:/bin/sh, sync:x:4:65534:sync:/bin:/bin/sync, games:x:5:60:games:/usr/games:/bin/sh, man:x:6:12:man:/var/cache/man:/bin/sh, lp:x:7:7:lp:/var/spool/lpd:/bin/sh, mail:x:8:8:mail:/var/mail:/bin/sh, news:x:9:9:news:/var/spool/news:/bin/sh, uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh, proxy:x:13:13:proxy:/bin:/bin/sh, www-data:x:33:33:www-data:/var/www:/bin/sh, backup:x:34:34:backup:/var/backups:/bin/sh, list:x:38:38:Mailing List Manager:/var/list:/bin/sh, irc:x:39:39:ircd:/var/run/ircd:/bin/sh, gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh, nobody:x:65534:65534:nobody:/nonexistent:/bin/sh, libuuid:x:100:101:/var/lib/libuuid:/bin/sh, dhcpc:x:101:102::/nonexistent:/bin/false, syslog:x:102:103::/home/syslog:/bin/false, klog:x:103:104::/home/klog:/bin/false, sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin, msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash, bind:x:105:113::/var/cache/bind:/bin/false, postfix:x:106:115::/var/spool/postfix:/bin/false, ftp:x:107:65534::/home/ftp:/bin/false, postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash.

Solution:

To fix this issue, refrain from providing the server details on the webpage directly. Instead maintain the path details in a separate file at the server and use JavaScript and access control to the access the server path.

6. Security Misconfiguration

Open burp suite and then open mutillidae website. Then we can see the directory listings of that website in burp suite at 'Target' tab as shown.



We can get the url of the files and will be able to access the files as shown in below.

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
anti-framing-protection.inc	11-Jan-2012 20:32	704	
back-button.inc	01-Apr-2012 14:54	1.4K	
command-injection-tutorial.inc	20-Jun-2011 20:35	2.5K	
constants.php	01-Apr-2012 15:11	3.5K	
create-html-5-web-storage-target.inc	15-Mar-2012 16:54	308	
cross-site-request-forgery-tutorial.inc	15-Mar-2012 16:28	11K	
cross-site-scripting-tutorial.inc	17-Dec-2011 13:36	5.2K	
http-parameter-pollution-tutorial.inc	09-Jul-2011 01:52	1.7K	
insufficient-transport-layer-protection.inc	16-Dec-2011 20:45	439	
sql-injection-tutorial.inc	20-Jan-2012 11:38	14K	

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.0.24 Port 80

Accessing the file

```
<?php
/*
 * @author: Jeremy Druin
 */
?>

<div>&nbsp;</div>
<table class="tutorial">
    <tr class="tutorial-title">
        <td colspan="10">Insufficient Transport Layer Protection Tutorial</td>
    </tr>
    <tr>
        <td>
            <br/>
            Use a packet sniffer such as TCPDump or Wireshark to observe traffic. Information
            such as the users password should be encrypted as it goes across the wire.
            <br/><br/>
        </td>
    </tr>
</table>
```

Solution:

Implement proper access control and visibility of the application folder structure to prevent attackers from accessing the application folders/ data. Ensure to implement user authorization rules in the application config files.

7. Insecure Cryptographic Storage

In order to exploit this vulnerability, go to html5-storage.php

The screenshot shows a web-based application titled "HTML 5 Storage". At the top, there is a "Back" button. Below it, a green bar says "HTML 5 Web Storage". Underneath is a table titled "Web Storage" with columns "Key", "Item", and "Storage Type". The data in the table is as follows:

Key	Item	Storage Type
CurrentBrowser	undefined	Session
LocalStorageTarget	This is set by the index.php page	Local
MessageOfDay	Go Cats!	Local

Below the table are two input fields, a radio button group ("Session" is selected), and a "Add New" button. At the bottom, there are three radio buttons: "Session Storage", "Local Storage", and "All Storage".

As shown below, this page is so poorly designed that from the console window of a developer tool, attacker can get the details of current session and locally stored details of the application.

The screenshot shows the browser's developer tools with the "Console" tab selected. The output area contains the following log entries:

```

Console was cleared
< undefined
> console.log(window.localStorage)
▼ Storage {LocalStorageTarget: "This is set by the index.php page", MessageOfDay: "Go Cats!", Secure.CurrentStateofHTML5Storage: "Completely Insecure", length: 3} ⓘ
  LocalStorageTarget: "This is set by the index.php page"
  MessageOfDay: "Go Cats!"
  Secure.CurrentStateofHTML5Storage: "Completely Insecure"
  length: 3
  ► __proto__: Storage
< undefined
> console.log(window.sessionStorage)
▼ Storage {CurrentBrowser: "undefined", Secure.AuthenticationToken: "DU837HHFYTEYUE9S1934", Secure.IsUserLoggedIn?: "No", SessionStorageTarget: "This is set by the index.php page", length: 4} ⓘ
  CurrentBrowser: "undefined"
  Secure.AuthenticationToken: "DU837HHFYTEYUE9S1934"
  Secure.IsUserLoggedIn?: "No"
  SessionStorageTarget: "This is set by the index.php page"
  length: 4
  ► __proto__: Storage

```

Solution:

- Make sure offsite backups are encrypted, but the keys are managed and backed up separately.
- Ensure appropriate strong standard algorithms and strong keys are used, and key management is in place.
- Ensure passwords are hashed with a strong standard algorithm and an appropriate salt is used.
- Ensure all keys and passwords are protected from unauthorized access.

8. Failure to Restrict URL Access

This vulnerability can be seen if the attacker knows the application structure beforehand. Below is the index page:

Applications ▾ Places ▾ Firefox ESR ▾ Sun 15:33 Mozilla Firefox

http://11.2.../index.php +
11.219.148.114/mutillidae/index.php 67% Search

Most Visited ▾ Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter

This network may require you to login to use the internet.

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

back|track

Samurai Web Testing Framework

BUILT ON

PHP MySQL Toad HACKERS FOR CHARITY

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons
@webpwnized

Now edit the url with a query string parameter 'page' and the webpage redirects to an unauthorized page as shown below, which has all the server configuration details:

Sun 15:32

Mozilla Firefox

http://11.2...hpinfo.php

11.219.148.114/mutillidae/index.php?page=phpinfo.php

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter

This network may require you to login to use the internet.

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Enabled (1 - Script Kiddie) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Secret PHP Server Configuration Page

Back

PHP Version 5.2.4-2ubuntu5.10

System	Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Build Date	Jan 6 2010 21:50:12
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
additional .ini files parsed	/etc/php5/cgi/conf.d/gd.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/pdo.ini, /etc/php5/cgi/conf.d/pdo_mysql.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	zip, php, file, data, http, ftp, compress.bzip2, compress.zlib, https, ftps

Site hacked...err...quality-tested with Samurai, WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

Developed by Adrian "IronGeek" Crenshaw and Jeremy Druin

Solution:

Implement proper access control and visibility of the application folder structure to prevent attackers from accessing the application folders/ data. Ensure to implement user authorization rules in the application config files. Make sure that all url redirections pass through authentication checks in the application.

9. Insufficient Transport Layer Protection

Open user-info.php page and also start wireshark.

View your details

Back

Please enter username and password to view account details

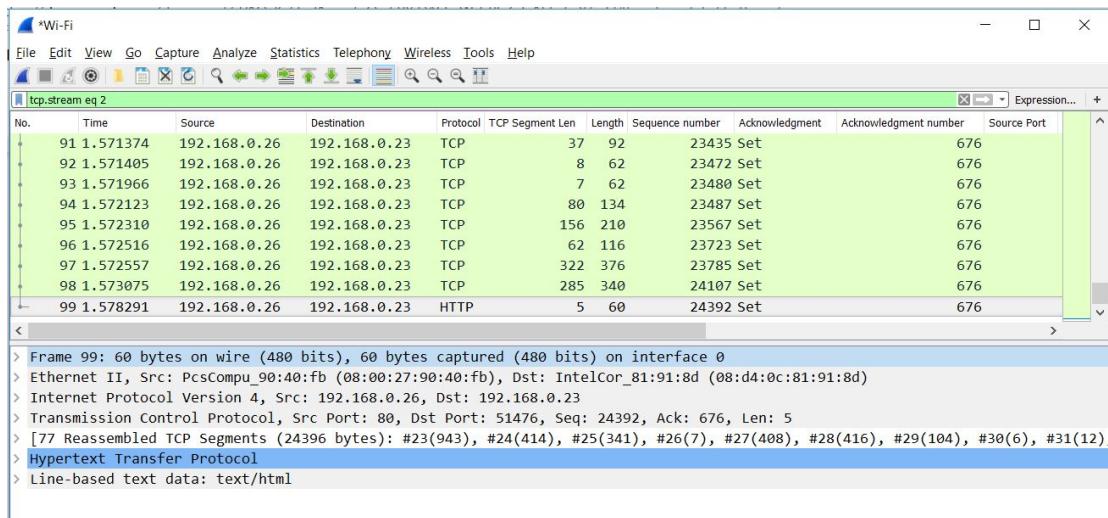
Name

Password

[View Account Details](#)

Dont have an account? [Please register here](#)

Results for . 1 records found.



Enter the credentials and click on ‘view account details’. Then stop the Wireshark. Then open the TCP stream of the Wireshark for that website. We will get the HTML page with username and password.

```

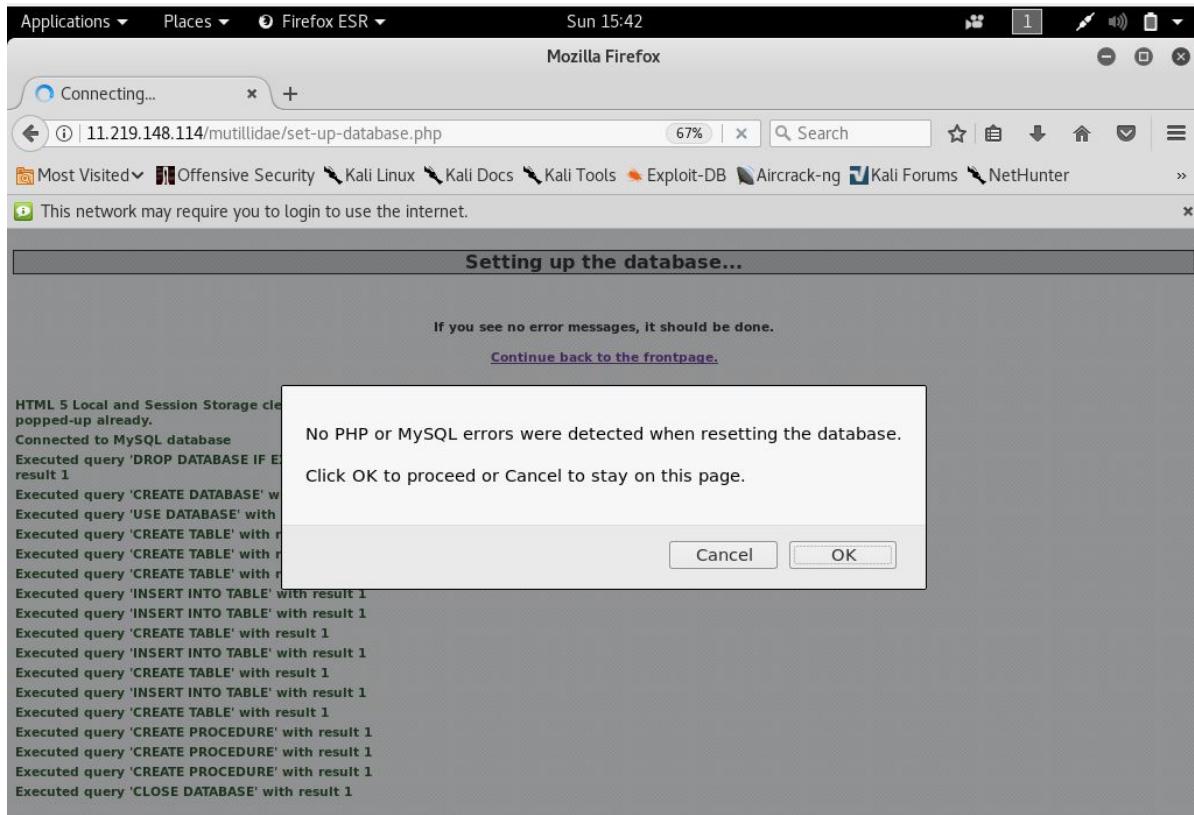
1 3a
<p class="report-header">Results for . 1 records found.<p>
19
<b>Username=</b>admin<br>
1d
<b>Password=</b>adminpass<br>
1f
<b>Signature=</b>Monkey!<br><p>
3
<p>
```

Solution:

- Implement HTTP Strict Transport Security in the browsers to enforce secure connections. Also implement Certificate and Public Key pinning in browsers where applicable.
- Redirect all HTTP requests to HTTPS
- Configure SSL properly on the server.
 - o Disable weak SSL/TLS protocols
 - o Disable weak ‘export’ algorithms
 - o Use SSL certificate with a minimum key size of 2048 bit
- Make sure session key size is 128 bits
- Disable algorithms offering no encryption

10. Unvalidated Redirects & Forwards

The following alert pop-up explains that anyone can reset the database, regardless of user being an admin or an attacker, which is an example of the Unvalidated Redirects & Forwards vulnerability.



Solution:

- Avoid using redirects and forwards.
- Do not allow url as user input, if it can't be avoided, make sure to implement the authorization rules for each user role that has access to the application.
- Implement appropriate input validation rules.
- Sanitize input by creating a list of trusted URLs.

11. Command Injection

To exploit this vulnerability, in the Hostname textbox, give input as ";ls". This particular command closes the earlier command and lists all the files that are present in the same directory as that of the dns-lookup.php page, as shown below:

Applications ▾ Places ▾ Firefox ESR ▾ Sun 15:54

Mozilla Firefox

http://11.2...lookup.php +

11.219.148.114/mutillidae/index.php?page=dns-lookup.php 67% | C Search

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter

This network may require you to login to use the internet.

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

DNS Lookup

Back

Who would you like to do a DNS lookup on?
Enter IP or hostname

Hostname/IP :ls

Lookup DNS

Results for ; ls

add-to-your-blog.php
arbitrary-file-inclusion.php
authorization-required.php
browser-info.php
capture-data.php
captured-data.php
captured-data.txt
change-log.htm
classes
closeddb.inc
config.inc
credits.php
dns-lookup.php
documentation
favicon.ico
footer.php
framer.html
framing.php
header.php
home.php
html5-storage.php

Site hacked...err..quality-tested with Samurai
WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

Developed by Adrian "Irongeek" Crenshaw and Jeremy Druin



12. Security Misconfiguration – GET for POST

In the user poll page, select any of the choices from the page and notice that the selected choice is passed through the 'choice' query string of the url when submit button was hit, as shown below:

Sun 11:40

Mozilla Firefox

http://11....ubmit+Vote +

mutillidae/index.php?page=user-poll.php&choice=nmap&initials=rsc&use 67% | Search | Star | Copy | Download | Home | Refresh | More

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter

This network may require you to login to use the internet.

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Enabled (1 - Script Kiddie) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls OWASP Top 10 Others Documentation Resources

User Poll

Back

User Poll

Choose Your Favorite Security Tool

Initial your choice to make your vote count

nmap
 wireshark
 tcpdump
 netcat
 metasploit
 kismet
 Cain
 Ettercap
 Paros
 Burp Suite
 Sysinternals
 inSIDder

Your Initials: [Text Input]

Submit Vote

Your choice was nmap

Now change the query string parameter from the user selected ‘nmap’ to wireshark to exploit the flaw, and on click of submit vote, it will be displayed that user selected wireshark.

The screenshot shows a Firefox browser window with the address bar containing `http://11....ubmit+Vote`. The page title is "Mozilla Firefox". The main content area displays a user poll titled "User Poll" with the sub-section "Choose Your Favorite Security Tool". The poll asks "Initial your choice to make your vote count" and lists various security tools as options:

- nmap
- wireshark
- tcpdump
- netcat
- metasploit
- kismet
- Cain
- Ettercap
- Paros
- Burp Suite
- Sysinternals
- inSIDder

Below the list is a text input field labeled "Your Initials:" with a placeholder box. A "Submit Vote" button is located below the input field. At the bottom of the poll section, a message states "Your choice was wireshark".

The left sidebar of the application includes a "Core Controls" menu with links to "OWASP Top 10", "Others", "Documentation", and "Resources". It also features sections for "Site" (with a note about being hacked), "@webpwnized", and "Mutilidae Channel". A footer at the bottom left credits the developer: "Developed by Adrian 'IronGeek' Crenshaw and Jeremy Druin".

Solution:

This flaw is due to the use of GET requests instead of POST requests that carry the parameters in the body of the requests instead of query strings like in GET requests.

13.Unvalidated Redirects & Forwards

To exploit this vulnerability, go to the credits.php and change the url redirection property - “forwardurl” of one of the links mentioned in this webpage.

Sun 16:07

Mozilla Firefox

http://11.2...credits.php

11.219.148.114/mutillidae/index.php?page=credits.php

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter

This network may require you to login to use the internet.

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls OWASP Top 10 Others Documentation Resources

Credits

Back

Created by Iongeek.com. Developed by Adrian "Iongeek" Crenshaw and Jeremy Druin

Adrian Crenshaw would like to thank the following people for helping him with the Mutillidae project:

OWASP for making the vulnerability list I based this on.
 ISSA Kentuckiana
 OWASP Louisville
 Brian Blankenship for his support of the idea.
 Mubix for confirming the name
 InfoSec Daily Podcast
 PaulDotCom Podcast
 All sorts of folks at PHPNet for code snippets:
 kaigilmann
 Professional Web Application Developer Quality Assurance Pack by Jeremy Druin

Site hacked...err...quality-tested with SamuraliWTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

Developed by Adrian "Iongeek" Crenshaw

Search HTML

<div></div>

OWASP

for making the vulnerability list I based this on.

Rules Computed Animations Fonts

Inherited from table

table.main-table-frame { border-collapse: collapse; border-spacing: 0px; }

Inherited from html

html { font-family: sans-serif, tahoma, verdana, serif; }

As shown below, OWASP link url is now changed to “nostarch.com”. Now, when the user clicks on this link, they will be redirected to nostarch.com website:

Sun 16:09

Mozilla Firefox

http://11.2...credits.php

11.219.148.114/mutillidae/index.php?page=credits.php

Most Visited: Offensive Security, Kali Linux, Kali Docs, Kali Tools, Exploit-DB, Aircrack-ng, Kali Forums, NetHunter

This network may require you to login to use the internet.

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Hints Toggle Security Reset DB View Log View Captured Data

Core Controls OWASP Top 10 Others Documentation Resources

Credits

Back

Created by Iongeek.com. Developed by Adrian "Iongeek" Crenshaw and Jeremy Druin

Adrian Crenshaw would like to thank the following people for helping him with the Mutillidae project:

OWASP for making the vulnerability list I based this on.
 ISSA Kentuckiana
 OWASP Louisville
 Brian Blankenship for his support of the idea.
 Mubix for confirming the name
 InfoSec Daily Podcast
 PaulDotCom Podcast
 All sorts of folks at PHPNet for code snippets:
 kaigilmann
 Professional Web Application Developer Quality Assurance Pack by Jeremy Druin

Site hacked...err...quality-tested with SamuraliWTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

Developed by Adrian "Iongeek" Crenshaw

Search HTML

<div></div>

OWASP

for making the vulnerability list I based this on.

Rules Computed Animations Fonts

Inherited from table

table.main-table-frame { border-collapse: collapse; border-spacing: 0px; }

Inherited from html

html { font-family: sans-serif, tahoma, verdana, serif; }

14.SQL Injection (Union exploit)

To exploit this vulnerability, go to DVWA site and select sql injection. Then pass the string as shown

' 'UNION select distinct(table_schema),null FROM information_schema.tables -- ' and click submit. Then we get the list of database name (mentioned in First name).

Vulnerability: SQL Injection

User ID:

```
ID: ' UNION select distinct(table_schema),null FROM information_schema.tables --
First name: information_schema
Surname:

ID: ' UNION select distinct(table_schema),null FROM information_schema.tables --
First name: dwva
Surname:

ID: ' UNION select distinct(table_schema),null FROM information_schema.tables --
First name: mysql
Surname:

ID: ' UNION select distinct(table_schema),null FROM information_schema.tables --
First name: owasp10
Surname:

ID: ' UNION select distinct(table_schema),null FROM information_schema.tables --
First name: tikiwiki
Surname:

ID: ' UNION select distinct(table_schema),null FROM information_schema.tables --
First name: tikiwiki195
Surname:
```

Put input as ' 'UNION select table_schema,table_name FROM information_Schema.tables where table_schema = "dvwa" -- ' to determine the table names as shown below. First name contains the database name and Surname contains the table name. Similarly we can obtain the column names of the table also.

Vulnerability: SQL Injection

User ID:

```
ID: ' UNION select table_schema,table_name FROM information_Schema.tables where table_schema = "dvwa" --
First name: dvwa
Surname: guestbook

ID: ' UNION select table_schema,table_name FROM information_Schema.tables where table_schema = "dvwa" --
First name: dvwa
Surname: users
```

15.Persistent Cross Site Scripting

To exploit this vulnerability, go to add-to-your-blog.php page and then add the message as shown in screenshot.

Welcome To The Blog

Back

Add New Blog Entry

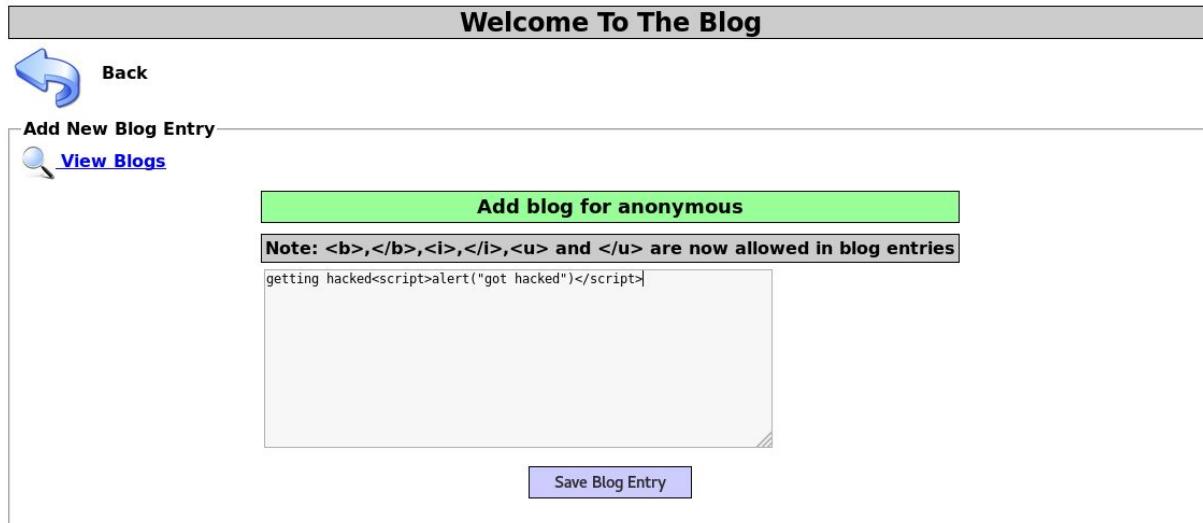
[View Blogs](#)

Add blog for anonymous

Note: ,,<i>,</i>,<u> and </u> are now allowed in blog entries

getting hacked<script>alert("got hacked")</script>

Save Blog Entry



[View Blogs](#)

1 Current Blog Entries		
	Name	Date
1	anonymous	2009-03-01 22:27:11

An anonymous blog? Huh?



Save the blog entry and then we will get an alert message since we have included a alert message as script and get excuted.

Welcome To The Blog

Back

Add New Blog Entry

[View Blogs](#)

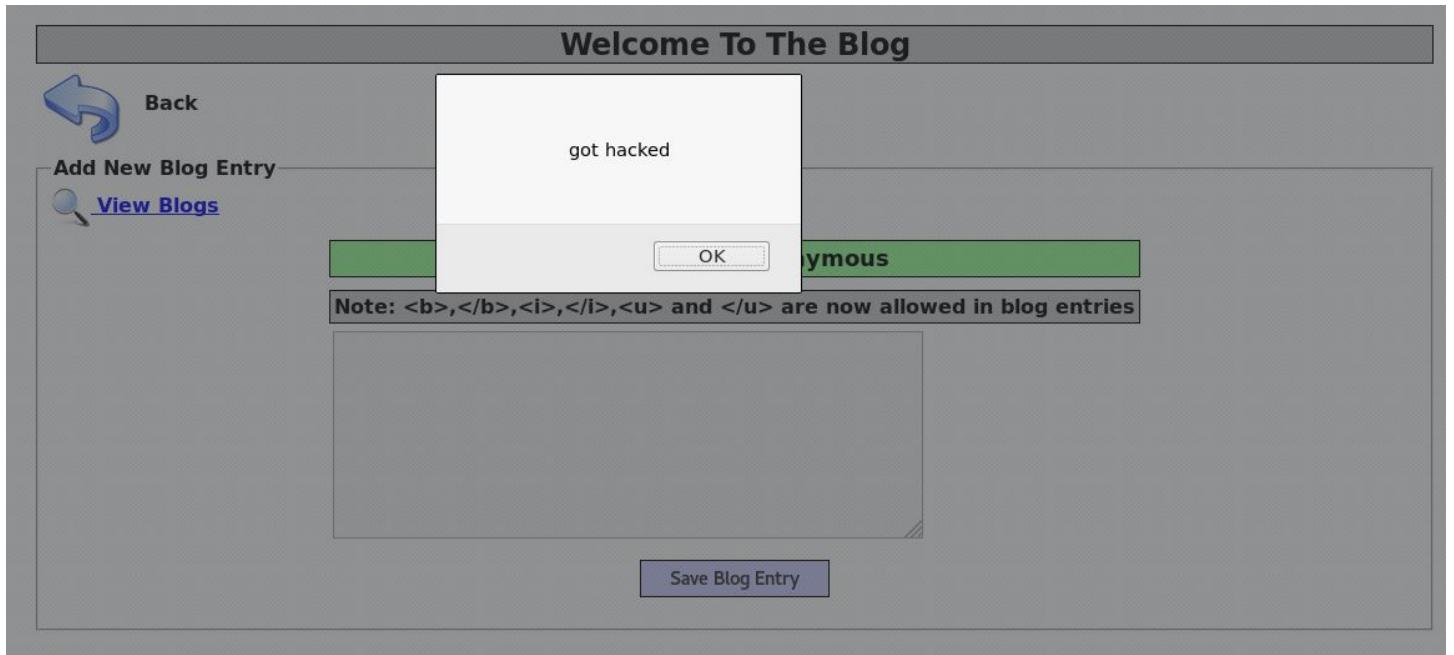
got hacked

OK

ymous

Note: ,,<i>,</i>,<u> and </u> are now allowed in blog entries

Save Blog Entry



If any user goes to view-someones-blog.php and click on blog entries

View Blogs

 [Back](#)

[View Blog Entries](#)  [Add To Your Blog](#)

Select Author and Click to View Blog

The script we included in message will be executed at that page as shown below. This is known as Persistent Cross Site Scripting.

View Blogs

 [Back](#)

[View Blog Entries](#)  [Add To Your Blog](#)

got hacked

to View Blog

13 Current Blog Entries			
	Name	Date	Comment
1	anonymous	2018-04-22 19:18:10	getting hacked

References

<http://cwe.mitre.org/top25/>

<https://www.sans.org/top25-software-errors>

[https://www.owasp.org/index.php/Cross-Site_Request_Forgery_\(CSRF\)_Prevention_Cheat_Sheet](https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)_Prevention_Cheat_Sheet)

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

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