**Ethical Hacking Day6**

**Positions**

**- Web Penetration Testing**

**+++++++**

- Bug Hunting+ CTFs

- iOS Penetration Testing

- Secure Code Review

securecodewarrior.com

- Android Penetration Testing

- Internal Penetration Testing

Active Directory → CRTP / CRTE

Web Penetration Testing

- OWASP Top 10

→ 2013

→ 2017

Payload

Logic

AND

1 & 1 → 1

1 & 0 → 0

0 & 1 → 0

0 & 0 → 0

OR

1 | 1 → 1

1 | 0 → 1

0 | 1 → 1

0 | 0 → 0

$username = $\_POST[‘username’];

$password = $\_POST[‘password’];

SELECT \* FROM `users` WHERE username = $username AND password = $password;

SELECT \* FROM `users` WHERE username = “bla” AND password = “test”;

$username = bla” or 1 = 1;--

SELECT \* FROM `users` WHERE username = “bla” OR 1 =1;-- AND password = “test”;

- Vulnerabilities

- SQL injection

- Directory Listing

- Server Fingerprint

- Sensitive Data Exposure/Disclosure in Robots.txt

- OS Injection / Command Injection

system(‘ls’)

exec(‘whoami’)

eval()

- SSTI

- Improper Error Handling

2 types of attacks

- Client-Side Attack →

Cross-Site Scripting (XSS)

- Server-Side Attack

**Penetration Testing Report**

- Vulnerability Name

- Vulnerability Risk (Critical – High – Medium – Low – Info)

- Vulnerability Description

- Vulnerability Impact

- Recommendations

- Proof of Concept (PoC) / Steps-to-Reproduce

Screenshots + description

**Spoofing Vs Hijacking**

**Spoofing**: Attacker pretends to be another user or machine to gain access

Attacker does not take over an existing active session.

Instead, he initiates a new session using the victim’s stolen credentials

**Hijacking**: Session hijacking is the process of taking over an existing active session

Attacker relies on the legitimate user to make a connection and authenticate

**Password Policy Misconfiguration**

- use weak Passwords (Password must be Complex)

1 capital character

1 small character

1 special character

1 digit

At least 8 characters

a

aa

ab

ac

Dictionary-Based Attack

[P@s5w0rd](mailto:P@s5w0rd)

<https://password.kaspersky.com/>

- Password revealed in user token

- Password stored as a plain text

- Password encrypted with weak algorithm

**Compromising session IDs using Sniffing**

- Attacker uses a sniffer to capture a valid session token or session ID

- Attacker then uses the valid token session to gain unauthorized access to the web server

**Compromising Session IDs by Predicting Session Token**

- Attackers can predict session IDs generated by weak algorithms and impersonate a website user

- Attackers perform analysis of variable sections of session IDs to determine a pattern

- The analysis is performed manually or by using various crypt-analytic tools

- Attackers collect a high number of simultaneous session IDs in order to gather samples in the same time window and keep the variable constant

**Protecting against Session Hijacking**

- Use Secure Shell (SSH) to create a secure communication channel

- Implement the log-out functionality for user to end the session

- Generate the session ID after successful login and accept session IDs generated by server only

- Ensure data in transit is encrypted and implement defense-in-depth mechanism

- Use string or long random number as a session key

- Use different username and passwords for different accounts

- Implement timeout() to destroy the session when expired

- Do not transport session ID in query string

- Ensure client-side and server-side protection software are in active state and up to date

- Use Strong authentication (like kerberos) or peer-to-peer VPNs

- Configure the appropriate internal and external spoof rules on gateways

- Use IDS products or ARPwatch for monitoring ARP cache poisoning

- Use HTTP Public Key Pinning (HPKP) to allow users authenticate web servers

- Enable browsers to verify website authenticity using network notary servers

Penetration Testing References

- eLearn Security - eWAPT course

- eLearn Security - eWAPTx course

- Ibrahem Hegazy Course – Youtube

<https://www.youtube.com/playlist?list=PLv7cogHXoVhXvHPzIl1dWtBiYUAL8baHj>

- eLearnSecurity – eJPT (PTS) course [Entry Level]

- Offensive Security - OSCP course

- eLearn Security – PTP course

- Ippsec – Youtube

[https://docs.google.com/spreadsheets/d/1dwSMIAPIam0PuRBkCiDI88pU3yzrqqHkDtBngUHNCw8/edit#gid=1839402159](https://docs.google.com/spreadsheets/d/1dwSMIAPIam0PuRBkCiDI88pU3yzrqqHkDtBngUHNCw8/edit" \l "gid=1839402159)

Active Directory Penetration Testing

- Pentester Academy - CRTP course

- Pentester Academy - CRTE course

VulnHub: <https://www.vulnhub.com/>

HackTheBox: <https://www.hackthebox.eu/>

- LiveOverFlow – Youtube

- Intro to Bug Bounty Hunting and Web Application Hacking:

<https://www.udemy.com/course/intro-to-bug-bounty-by-nahamsec/>

OWASP Top 10

[https://owasp.org/www-pdf-archive/OWASP\_Top\_10-2017\_%28en%29.pdf.pdf](https://owasp.org/www-pdf-archive/OWASP_Top_10-2017_(en).pdf.pdf)

Web Application Hackers Handbook

[https://github.com/briskinfosec/Books/blob/master/Web%20App%20Pentest/the-web-application-hackers-handbook.pdf](https://github.com/briskinfosec/Books/blob/master/Web App Pentest/the-web-application-hackers-handbook.pdf)

Training

Attack & Defense – Pentester Academy

<https://attackdefense.com/>

Pentester Lab

<https://www.pentesterlab.com/exercises?only=free>

XVWA : [https://mega.nz/#!4bJ2XRLT!zOa\_IZaBz-doqVZz77Rs1tbhXuR8EVBLOHktBGp11Q8](https://mega.nz/" \l "!4bJ2XRLT!zOa_IZaBz-doqVZz77Rs1tbhXuR8EVBLOHktBGp11Q8)

DVWA: <https://dvwa.co.uk/>

bWAPP: <https://www.vulnhub.com/entry/bwapp-bee-box-v16,53/>

<http://www.itsecgames.com/>

WebGoat: [https://owasp.org/www-project-webgoat/#:~:text=WebGoat%20is%20a%20deliberately%20insecure,and%20popular%20open%20source%20components](https://owasp.org/www-project-webgoat/" \l ":~:text=WebGoat is a deliberately insecure,and popular open source components).

OverTheWire: <https://overthewire.org/wargames/>