Ethical Hacking Day #3

Bug Bounty – Bug Hunting

Bug Bounty Program →
- Hall of Fame (HoF)
- Bounty

Triage Team

Proof of Concept (PoC)
Steps to reproduce

Arab Security Conference
Cairo Security Camp
CyberTalents
Besides

CTFs
Capture The Flag

- Jeopardy
- Attack & Defense

Flag{You_r_The_Best}

HackerOne{Bla_Bla_Bla}

Write-ups for the challenges

FLAG……FLAG

Categories:

- Information Gathering
- Web Security
- Forensics
- Reverse Engineering
- Cryptography
- Machines

WarGame

Cairo Security Camp
Arab Security Conference

CTF challenge write-ups

Encoding

- Base64
- Base32
- URL Encoding
- HTML Encoding

Hashing
Encryption

-

Penetration Testing Assignment

- Walkthrough [Scoping]
  - how many roles access the application
  - Which programming language

→ Estimate man-days: 20

- Penetration Testing
  - Login
  - Forget Password
  - Registration
  - Profile

  - Recon
  - testing
  - Reporting
    - Vulnerability Name
    - Risk [Critical – High – Medium – Low – Info]
    - Description
    - Impact
    - Recommendation
    - Proof of Concept (PoC) / Steps to Reproduce
      1. step 1
      Screenshot
      2. step 2
      screenshot
  - Retesting

  Black-box Vs White-box

Brute-Forcing
- Username: admin
- Password: Bl@Te$tC0mplex
dictionary-based attack

Vulnerabilities:

- User enumeration

    Login Page
    username wrong → Your username is wrong
    username correct & password wrong → your password is wrong
    → Generic message : You entered invalid Username/Password

    Forget Password Page
    email
    → correct → An email has been sent to your email
    → wrong → Invalid email

    Generic message : If you entered a correct email, an email will be sent.

- Missing Rate-Limit

    → if user entered invalid credentials 3-5 times → block the user

    Generic → Block the user for 10-15 minutes

- Weak Password Policy

Search for File uploads → RCE (Remote Code Execution)

https://ctftime.org/
https://backdoor.sdslabs.co/
picoctf.org

Penetration Testing Categories:
- Web App
- Mobile
- Network
- Internal/External
- Wireless
- IoT

Asset
Vulnerabilities

- Penetration Testing Phases
- **-** Scoping
  - Walkthrough
  - → Functions of the application
  - → How many mandays?

  - Prerequisites
  - - 2 Users from each role
  - - File sample for each file upload with dummy data

- Reconnaissance & Enumeration || Scanning
  - - Users : Admin, seller, user
  - - List each user's functions
  - - Brute-Force on the directories/files
  - Tools :
  - dirb, dirbuster, dirsearch

  - **Brute-Forcing**
  - - Dictionary-based [wordlist]
  - - Simple Brute-Force

  - abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890!@#$%^&*()

  - 8 characters

  - aaaaaaaa
  - aaaaaaab

aaaaaaac
aaaaaaad
bbbbbbbb
bbbbbba

6 digits

111111
111112
111113
……
999999

Vulnerability: Missing Rate-limit

Recommendations:
- Block the user after 3-5 invalid trails
- Block the user after 3-5 invalid trails for 15-30 minutes

Username & Password wrong for 3-5 times ?
User got blocked ?
User can login?
User got blocked for 15 mins ??
User got blocked and should by activated by admin

Denial of Service (DoS)

Availability Issues:
DoS → Denial of Service

List of users
Try to login with each user for 3 times until he is blocked

Brute-Force
wordlists

Vulnerability: User Enumeration

Username	Password	
Correct	Correct	→ Credentials Correct
Correct	Wrong	→ “Password Is wrong” XXXXX
Wrong	Wrong/Correct	→ “Username is not exist” XX

Forget Password
email → Wrong email
This email is not in our Dbs

Login
Registration
Forget password

Recommendation:
Use Generic Message in Login page
“Username/Password is wrong”
“Credentials are wrong”

Use Generic Message in Forget Password
“An email has been sent to your email, Please check!”

example.com/index
example.com/images
example.com/2006

Penetration Tester

Inputs, file upload, get request with parameter

example.com → production
staging.example.com → Staging [clone/copy from the production]
uat-qc.example.com → UAT Server [Quality Control – QC]
web developer → Local machine

Payloads

400 → -400

x – y
3000 - (-400) = 3400

http://api.example.com/login

domain → example.com
subdomain → api

SSL/TLS Certificate

HTTPs

If the application is using HTTP not HTTPs
Vulnerability → No SSL certificate installed

If the application is using HTTPs, but the certificate has some issues
Vulnerability → Using vulnerable SSL certificate

- Vulnerability Analysis
  - SQL injection
  - Cross-Site Scripting (XSS)
  - Unrestricted File Upload

- Exploitation
  - Extract for the DB
  - Extract for the user's session/cookies
  - Get Full access on the server [Shell file]
- Reporting
  - Detailed Observation
    - Vulnerability Name
    - Vulnerability Risk
    - Description
    - Impact
    - Recommendation
    - Steps to Reproduce – PoC
      - Screenshots with description

Operating Systems

- Windows → Admin
- Linux → root

    distributions

    - Debian - Ubuntu – Kali – Redhat – CentOS – Fedora

- MacOS

Vulnerability: Server Fingerprint

Server Default Page revealed

Web Applications Programming Languages

- PHP
    .php
- Python
    .py
- JS
    .js
- C# - .Net
    .asp .aspx
- Java
    .jsp
- Ruby on Rails
    .rb
- GO
    .go

Cyber Security Certificates:

- Offensive Security: OSCP, OSWE

- eLearnSecurity: PTS – PTP – WAPT – WAPTx – MAPT

- Penetester Academy: CRTP – CRTE

HackTheBox

Vulnhub

- IppSec

https://www.youtube.com/channel/UCa6eh7gCkpPo5XXUDfygQQA

OSCP-Like Machines
https://docs.google.com/spreadsheets/d/1dwSMIAPIam0PuRBkCiDI88pU3yzrqqHkDtBngUHNCw8/edit#gid=0

https://academy.hackthebox.eu/modules

Links:

https://www.youtube.com/c/GeneralEG/videos

The Web Application Hacker's Handbook: Discovering and Exploiting Security Flaws

https://raw.githubusercontent.com/briskinfosec/Books/master/Web%20App%20Pentest/the-web-application-hackers-handbook.pdf

- Install & use Kali Linux - a penetration testing OS.
https://images.kali.org/virtual-images/kali-linux-2021.2-vmware-amd64.7z