Ethical Hacking Day #4

Bug Bounty – Bug Hunting

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

Triage Team

Proof of Concept (PoC)
Steps to reproduce

Web Programming Languages
HTML + CSS → Not a programming languages

JS → Interactive website
    Frontend: AngularJS – ReactJS – VueJS
    Backend: NodeJS
    Cross-Platform: Flutter – React Native – Ionic

PHP
    Laravel – Yii – Codegnitor – Symfony
    CMS (Content Management System) : Wordpress – Joomla – Drupal
    Wordpress Add-ons: WooCommerce

Python:
    Django – Flask

Ruby:
    Ruby on Rails

Java EE:
    Spring – Hibernate – Struts

C#:
    ASP.Net

Go Lang

Android
    Java – Kotlin
    JS → Cross-Platform

iOS
    Swift – Objective-C
    JS → Cross-Platform

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

Databases:
MySQL : 3306
Postgres: 5432
SQLite
SQL Server + ASP.NET
Oracle + JAVA
NoSQL + JS as a backend

Operating Systems

- Windows → Admin
- Linux → root
    - distributions
        - Debian - Ubuntu – Kali – Redhat – CentOS – Fedora
- MacOS

Recon Steps:
- Scan the open ports
    - using nmap
- Find out the programming Languages
    - using wapplayzer
    - using builtwith.com
- Find out the used DB
- Search for the CVEs
- Open the robots.txt file
- Search for the subdomains
- Login on the system

Web Hacking

Protocols:
HTTP → 80
HTTPs → 443

scan for the opening ports
Tool: nmap

Known Ports

- 21 → FTP
- 22 → SSH
- 23 → telnet
- 25 → SMTP
-

- 8080
- 8081
- 9001
- 9080

Programming Languages:
- JavaScript
    Frontend & Backend
- PHP
- C#
- Python
- Java
- Ruby
- Go

Frontend:
- Web Application
    HTML + CSS
    JavaScript Frontend Frameworks
    - ReactJS
    - Angular
    - VueJS

- Mobile Application
    - Android → Java / Kotlin
    - iOS → Objective-C / Swift
    - Cross-Platform Frameworks
        - Flutter
        - React Native
        - Ionic

→ Have to talk to an API [Backend]

Backend:

- JavaScript → NodeJS Framework → **.js**
- PHP → **.php, .php7, .php5**
    - Laravel
    - Symfony
    - Yii
- C# → ASP.NET → **.asp** , **.aspx**
- Java → Java EE → **.jsp**
    - Hibernate
    - Spring
- Python → .py
    - Django
    - Flask
- Ruby → .rb
    - Ruby on Rails
- Go → .go

Content Management System – CMS
- Wordpress
- Joomla
- Drupal

Core →
Extensions →

Vulnerability: Remote Code Execution (RCE)
from the web application, you can get full access on the web server

upload web shell

Search Engines:
Google
Bing
Yahoo

Spiders → indexing

robots.txt

SEO

Testing Types:
- Black-Box →
- White-Box →

domain
example.com

subdomain
api.example.com
bla.example.com

Discover the subdomains using sublist3r

sudo apt install sublist3r

---

netdiscover -r 10.0.0.1/24 -i eth0

Internal/External Penetration Testing
- netdiscover → to find out the IPs

ifconfig → linux machine
ipconfig → windows machine

Network Adapter Option

- Bridged → get an IP from the router directly
- NAT → Share the host IP
- Host-Only →located in a private network on the host

192.168.0.5

Host-only 10.0.0.3

MAC Address → physical address

Walkthrough = Write-up = Solution

List of Tools
- netdiscover → to find out the machine's IP
- nmap → to find out the open ports
- nikto → to scan the web application
- wpscan →
    - to scan the wordpress
    - to enumerate the users

netcat [nc]→

www-data → apache user

Vulnerability: Privilege Escalation →
upgrade ur privileges from www-data to c0ldd

The most powerful user on
Linux → root
Windows → admin

sudo vim -c ':!/bin/bash'

Asset

An asset is what we're trying to protect.

Threat

A threat is what we're trying to protect against.

Threat – Anything that can exploit a vulnerability, intentionally or accidentally, and obtain, damage, or destroy an asset.

Vulnerability

A vulnerability is a weakness or gap in our protection efforts.

Vulnerability – Weaknesses or gaps in a security program that can be exploited by threats to gain unauthorized access to an asset.

Risk

Risk is the intersection of assets, threats, and vulnerabilities.

Risk – The potential for loss, damage or destruction of an asset as a result of a threat exploiting a vulnerability.

A + T + V = R

That is, Asset + Threat + Vulnerability = Risk.

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