YASH RUSTAGI BT22GCS180

Page :
Date :

VAPT LAB ASSIGNMENT
6/05/25

Company Name: Acme Retail Corp.

Infrastructure: 3 web-apps
Internal network with 25 systems
Cloud-hosted assets on AWS
2 mobile apps (android & IOS)
VPN access for remote employees

Business Priority: data security of customer transaction
protect PII
maintain 24/7 uptime

1. SCOPE

a) Assets covered in VA:
- all web-apps
- Internal Infrastructure systems
- cloud-hosted assets
- both mobile apps
- APIs

b) Assets covered in PT:
- Same as above

c) OUT OF SCOPE
- Testing VPN access which is there for external ~~exmf~~ employees
- personal devices
- social engineering

Page :
Date :

2. Tools

↳ for VA:

VA is mostly an automated process, done by using tools

① OWASP ZAP for web-apps

~~Cons~~ Pros

- easy to use
  - ↳ tester just have to enter the URL and it will automate the scans
- report-generation
  - ↳ onces the scan is done, it gives the choice to generate a complete report
- Accurate results
  - ↳ it provides proof and palyoads for all the alerts generated & very less false positive

② S3Scanner: for cloud assets on AWS

③ Nuclei

Pros:

- extremely fast scanning; uses parallel scan

- Templates
  - nuclei provides the feature to make custom templates for specific needs/vulnerabilities
- wide coverage
  - ~~detects~~ covers a wide range of cves, misconfigurations

(4) Drozer, frida - for mobile apps

- explores exposed components, privilege escalation
- frida is a dynamic purpose toolkit
  - provides real-time code hooking and analysis of runtime behavior

(5) Android Debug Package

- CLI tool
- install/uninstall apps, access logs, file systems
- extract sensitive files
- used for manual testing and reconnaissance

(6) for information gathering: ffuf, subfinder, httpx, subzy, nmap

Page :
Date :

for Penetration Testing

- It is mostly done manually, as the requests and responses needs to be understood

① BURPSUITE
- to capture requests
- alter requests

② ~~OWTF framework~~
OWASP WSTG for web app
- guide for testing security of web-app & services

3. TEST PLAN and TIMELINE

PLANNING

① Information Gathering
- search for information leakage using google hacking
- fingerprint web server
- enumerate web server
- Identify entry points
- Map application architecture

Page :
Date :

② Vulnerability Scanning

- using the tools mentioned
- document every step

③ Verification

- do a manual verification of the results from the previous step

④ Penetrating Testing

- Exploit assets that are in scope to demonstrate risk
- document the tool used, payload used, proof of concept for each issue
- Assign CVSS score and rank the findings

⑤ Reporting

- create an high-detailed overview and technical summary

⑥ Provide Remediation Support

- resolve the issues found
- Re-test to see if it is fixed
- final report

Page :
Date :

Different teams will be there for each asset:

1. 1 team for web-app testing + APIs
2. 1 team for android app testing
3. 1 team for ios app testing
4. 1 team for cloud pentesting
5. 5 teams for internal Infrastructe (1 team each for 5 systems)

Timeline

Day 1 to 7: Defining Scope, Objectives
Signing SLA
Rules of engagement

~~Day 3 to~~

→ Information gathering and enumeration

Week 2: Vulnerability Assement for web-app, android app, and internal systems parallely

Page :
Date :

TIMELINE

TOOLS

	Name	PROS	CONS
①	OWASP ZAP	• very few false +ves • Report generation • Automated scanning • easy to use	• can disrupt web services due to the huge amount of traffic generated
②	Nuclei	• fast scanning • custom templates	-
③	Drozer (mobile app)	• explores exposed components	
④	Frida	• provides real-time code hooking • analysis of runtime behavior	
⑤	Burpsuite	• capture, modify HTTP requests • best for web-app	• No remediation • false positives
⑥	Astra Pentest	• PCI-DSS, GDPR, HIPAA ISO, SOC2 compliance • 0 false positives • provides remediation	• costly

Page :
Date :

Timeline

Day 1-3: Define Scope
Objectives
Rules of engagement
Signing SLA

Day 4-6: Reconnaissance and Intelligence gathering

- each team works in parallel

Day 7-8: Scanning and Vulnerability Analysis

- run automated tools
- map out all the vulnerabilities
- document them
- rank them on basis of impact

Day-9-10: Threat modeling

- determine risk impact & liklihood
- propose mitigations
- review the vulnerabilities
- draw DFDs

Page :
Date :

Day 11 - [redacted] Exploitation

- manual exploitation takes time
- document the process, tools used, payloads used, proof of concept

Day 15 : Reporting and Advisory

- create a detailed reporting, consisting of all vulnerabilities, their impact, P.O.C, and mitigation

end