Hlo everyone, my name is Rhonny. All the stuffs you will find here is collected during my learning. Hope you will support me...

Part → 1

Understanding Recon

- Recon == **Increased Attack Surface ~=** More Vulnerabilities
- Recon == Finding Untouched Endpoints ~= **Less Dupies**

- Recon == Sharpening your Axe before Attack. BUT! Wait! We won't waste time into sharpening our bonds with EX. :p
- We will rather jump in to automate stuff as much as we can to reduce time consumption.

General Misunderstanding

- If I do Recon, I will get a lot of Vulnerabilities ?
  - Recon will help you increase attack surface, may allow you to get vulnerabilities but ultimate goal is to dig your target to deepest.

- Automated Recon is sufficient?
  - No, there are certain situations where you might need to look up manually like Github Recon, Google Dorking and others.

- Recon is a time consuming process so I avoid it, am I cool?
  - No, If you will try to play smart moves automating your Recon, you can do a lot of things!
- Recon is love bro!
  - Absolutely, Just like Chaai (Tea)

Before Recon V/S. After Recon

Before Recon

- Target's Name

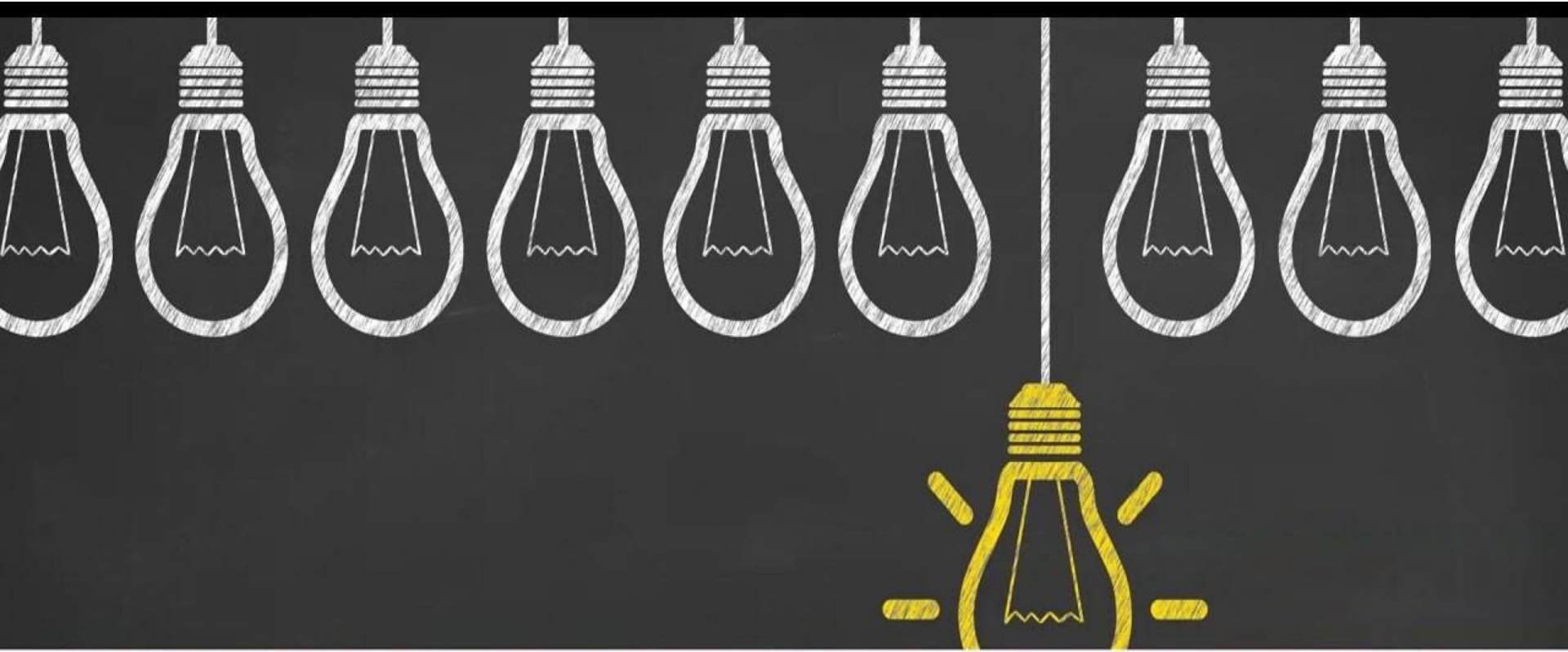
High-Level Overview of Application

- Credentials/Access to the Application

- And some other information based upon target, that's it on high level?

After Recon

- List of all live subdomains
- List of interesting open port
- Sensitive Data Exposed on Github
- Hidden Endpoints
- Juicy Directories with Sensitive Information
- Publicly exposed secrets over various platforms
- Hidden Parameters
- Low hanging bugs such as Simple RXSS, Open Redirect, SQLi ...



APPLICATION TESTING METHODOLOGY
&
SCOPE BASED RECON

SCOPES

Small Scope

- Specific set of Single URLs/Sandbox/QA/Staging Environment

Medium Scope

- Specific set of **"*.target.com"**

Large Scope

- Complete Internet presence including Acquisitions & Copyrights

Small Scope Recon

Scope – Single/Multiple Page Applications

- What to look for while Recon:
- Directory Enumeration
- Service Enumeration
- Broken Link Hijacking
- JS Files for Hardcoded APIs & Secrets
- GitHub Recon (acceptance chance ~ Depends upon Program)
- Parameter Discovery
- Wayback History & Waybackurls
- Google Dork (Looking for Juicy Info related to Scope Domains)
- Potential URL Extraction for Vulnerability Automation (GF Patterns + Automation Scripts)

Medium Scope Recon

- What to look for while Recon:
- Subdomain Enumeration
- Subdomain Takeovers
- Misconfigured Third-Party Services
- Misconfigured Storage Options (S3 Buckets)
- Broken Link Hijacking
- Directory Enumeration
- Service Enumeration
- JS Files for Domains, Sensitive Information such as Hardcoded APIs & Secrets
- GitHub Recon
- Parameter Discovery
- Wayback History & Waybackurls
- Google Dork for Increasing Attack Surface
- Internet Search Engine Discovery (Shodan, Censys, Fofa, BinaryEdge, Spyse Etc.)
- Potential URL Extraction for Vulnerability Automation (GF Patterns + Automation Scripts)

Large Scope Recon

- Tracking & Tracing every possible signatures of the Target Application (Often there might not be any history on Google related to a scope target, but you can still crawl it.)
- Subsidiary & Acquisition Enumeration (Depth – Max)
- DNS Enumeration
- SSL Enumeration
- ASN & IP Space Enumeration and Service Identification
- Subdomain Enumeration
- Subdomain Takeovers
- Misconfigured Third-Party Services
- Misconfigured Storage Options (S3 Buckets)
- Broken Link Hijacking
- Directory Enumeration
- Service Enumeration
- JS Files for Domains, Sensitive Information such as Hardcoded APIs & Secret
- GitHub Recon • Parameter Discovery
- Wayback History & Waybackurls
- Google Dork for Increasing Attack Surface
- Internet Search Engine Discovery (Shodan, Censys, Fofa, BinaryEdge, Spyse Etc.)
- Potential URL Extraction for Vulnerability Automation (GF Patterns + Automation Scripts)
- And any possible Recon Vector (Network/Web) can be applied.

This is the theory part, in next part we will see practical web pentesting recon cheaklist.

Share this checklist and support me....

Thank You