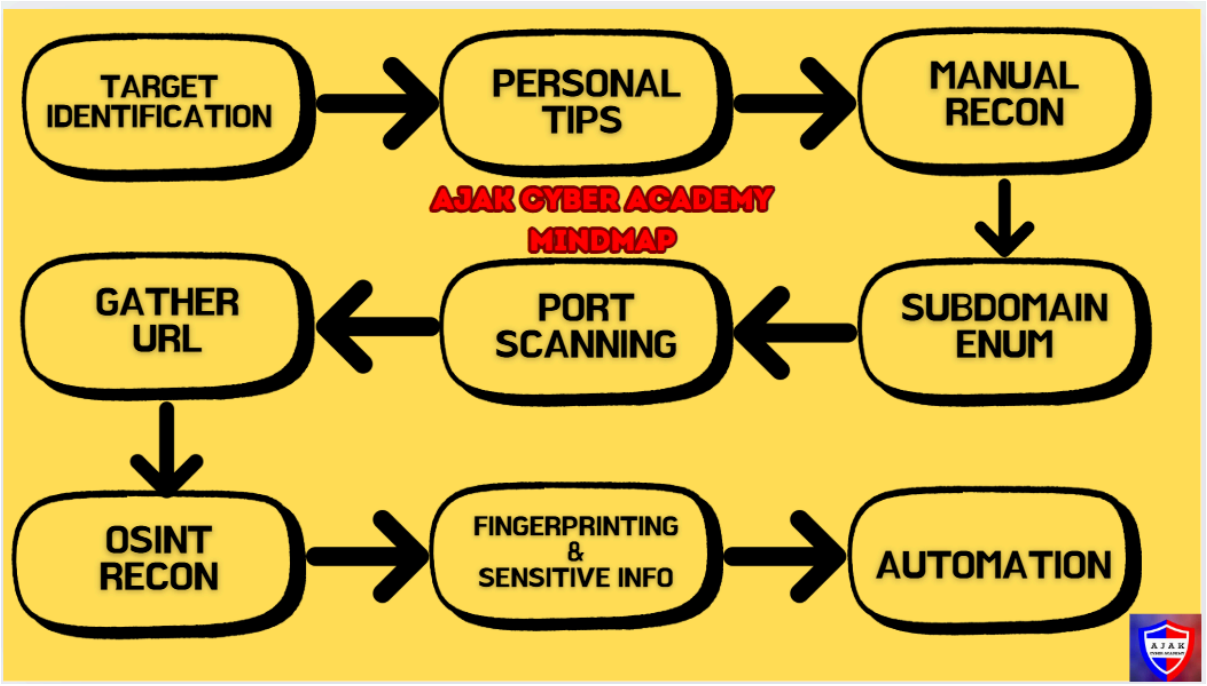
## 1) Mindmaps:

**AJAK Cyber Academy Roadmap**




**Other Roadmaps:**
https://gowthams.gitbook.io/bughunter-handbook/mindmaps

## 2) Note Taking:

http://github.com/ehrishirajsharma/SwiftnessX?tab=readme-ov-file -**SwiftnessX**
**https://getgreenshot.org/ - Greenshot or snipping Tool**

## 3) How Recon Help in Bounties

**https://infosecwriteups.com/how-fuzzing-helps-me-to-get-my-first-bounty-2c63eb864e08**

**https://hackerone.com/reports/17514**

**https://www.instagram.com/stories/highlights/17960390840676056/**

## 4) ASN:

### ***Tools:***

1) **Ping- (Kali linux inbuilt)**
2) **Whois- (kali linux inbuilt)**
3) **asnlookup.com**
4) **shodan.io**

#### ***Tools Commands:***

**Ping target.com**
**Whois target IP**
**CIDR notation/Target name**
**net:CIDR notation**

## 5) Certificate transparency

### ***Tools:***

5) **crt.sh**
6) **crt.sh automation**

### ***Installation of crt.sh:***

**git clone https://github.com/az7rb/crt.sh.git && cd crt.sh/**
**chmod +x crt.sh crt_v2.sh**

***Usage of tool:***

**./crt.sh -h**
**./crt.sh -d target.com**

## 5) DNS Enumration

### ***Tools:***

1) **Dig (inbuilt in kali linux)**
2) **MXToolbox.com**

***Usage of tool:***

**Dig target.com**

**Dig MX target.com**
**Dig NS target.com**
**Dig TXT target.com**

## 6) Download GO Language

**Kindly watch this blog and simply copy paste the commands if go is not installed in your kali linux**

**https://noureldinehab.medium.com/how-to-install-golang-latest-version-on-kali-linux-1afa2bd64ace**

## 7) Install subfinder

**Link: https://github.com/projectdiscovery/subfinder**

**Commands to Install:**

**go install -v github.com/projectdiscovery/subfinder/v2/cmd/subfinder@latest**

**Public Blogs:**
**https://ajakcybersecurity.medium.com/tips-tricks-to-get-hall-of-fame-in-nasa-53819d8221d3**

**https://medium.com/@halfcircassian/subfinder-automating-subdomain-enumeration-for-bug-bounty-in-2025-2fc17e385e4f**

**https://ajakcybersecurity.medium.com/find-this-easy-csrf-in-every-website-a-sweet-p4-372a3198bf47**

https://ajakcybersecurity.medium.com/i-found-an-idor-flaw-where-users-attached-pictures-and-documents-were-leaked-961d564ce72f

https://ajakcybersecurity.medium.com/idor-to-view-other-private-users-profile-pictures-in-un-org-358e464335e8

## 8) Amass

**Install AMASS:go install -v github.com/owasp-amass/amass/v4/...@master**
**Cd .config**
**Create a 2 file**
**https://github.com/owasp-amass/amass/blob/master/examples/config.yaml**
**https://github.com/owasp-amass/amass/blob/master/examples/datasources.yaml**

**Save it as .yaml file**
**Copy the APi keys from other websites and paste it**

## 9) Sublist3r

**git clone https://github.com/aboul3la/Sublist3r.git**
**cd Sublist3r**
**pip install -r requirements.txt**

**Commands:**

1) **python sublist3r.py -d nasa.gov -o sublist3r_subdomain.txt**

## 10) PureDNS

***Installation:***

**go install github.com/d3mondev/puredns/v2@latest**

**$HOME/go/bin**

**echo 'export PATH=$PATH:$HOME/go/bin' >> ~/.bashrc**

**source ~/.bashrc**

**cp /root/go/bin/puredns /usr/local/bin/**

https://github.com/d3mondev/puredns?tab=readme-ov-file
https://github.com/six2dez/OneListForAll
https://github.com/danielmiessler/SecLists/blob/master/Discovery/DNS/dns-Jhaddix.txt
https://wordlists.assetnote.io/
https://github.com/trickest/resolvers/blob/main/resolvers.txt

## 11) Chaos

**Installation:**

**GO111MODULE=on go install -v**
github.com/projectdiscovery/chaos-client/cmd/chaos@latest

**export CHAOS_KEY="your_api_key_here"** **(get your keys from here https://cloud.projectdiscovery.io/settings/api-key)**

**cd ~/go/bin**

**sudo cp chaos /usr/local/bin/**

**chaos -h**

**Links:**
https://github.com/projectdiscovery/chaos-client
https://cloud.projectdiscovery.io/settings/api-key
https://chaos.projectdiscovery.io/

## 12) Crunchbase

https://www.crunchbase.com/

## 13) google dorks for subenum

**For different Countries:**

**site:*.tesla.com.***

**For Fectching subdomain**

**site:nasa.gov**

**For Fetching Subdomain**

**site:*.nasa.gov**

**For Fetching Subdomain of subdomain**

**site:*.*.nasa.gov**

**For Fetching unique subdomain**

**site:*.nasa.gov inurl:dashboard**
**site:*.nasa.gov inurl:admin**
**site:*.nasa.gov inurl:community**

## 14) Virus Total

https://www.virustotal.com/\

## 15) LLM For Subdomain Enum

**Prompts:**

**I've uploaded a text file containing subdomain enumeration results. Summarize the list and highlight any subdomains that look sensitive, high-value, or internal**

**Please classify each subdomain in the file into categories: Public Content, Internal/Administrative, Authentication, Development/Staging, or Unknown.**

**From this subdomain list, extract keywords (like "vpn", "auth", "dev", "staging") and group subdomains accordingly. Present the results in a table showing Subdomain, Category, and Potential Value.**

## 16) shodan

org:"nasa"
hostname:"*.nasa.gov"
ssl.cert.subject.CN:"*nasa.gov"
ssl:*nasa.gov 403

pip install shodan
shodan init "YOUR_API_KEY"
shodan search 'ssl:"nasa.gov"'

## 17) cencys

https://platform.censys.io/home

## 18) Subdomain filtering

**Manual Method:**

```
cat Subdomainfinder.c99.nl.txt subfinder_nasa.gov sublister_subdomin.txt sublist.txt
cat Subdomainfinder.c99.nl.txt subfinder_nasa.gov sublister_subdomin.txt sublist.txt > merged.txt

 merged.txt | wc -l

sort -u finalsubdomains.txt
sort -u finalsubdomains.txt | wc -l
```

**One Liner:**

```
cat Subdomainfinder.c99.nl.txt subfinder_nasa.gov sublister_subdomin.txt sublist.txt virustotal_subenum.txt | sort -u | grep -v '^$' > final_unique_subdomains.txt
```

**Bash scripting:**

```
#!/bin/bash
cat Subdomainfinder.c99.nl.txt subfinder_nasa.gov sublister_subdomin.txt sublist.txt virustotal_subenum.txt | sort -u | grep -v '^$' > final_unique_subdomains.txt
echo "[+] Final list saved to: final_unique_subdomains.txt"
```

**LLM Prompts:**

Hi I am doing bug bounty, I have fetched subdomain from different forums and saved in multiple txt output file from here Give me a bash scripting script to combine all urls and remove duplicates and save them in an finaloutput file

Or

Upload multiple txt files and give "combine this multiple .txt file merge them in one and name it as merged.txt and remove duplicates from the merged.txt and give me as final_output.txt "

## 19) NMAP Port Scanning

### Commands:

nmap -h ---> help
nmap 192.168.0.1/24 -sL -----> list
nmap -sn ---> Ping
nmap -F -----> Fast common 100 ports
nmap target.com ---> =common 1000 ports
nmap target.com -p 23 ---> port specification
nmap target.com -p 23-1000---> Port range specification
nmap 172.217.31.206 -p-  ----> scan all ports
nmap iL sublist.txt -F
nmap -sV target.com ---> scans service version
nmap -O target.com ----> scans OS detection
nmap -v target.com ----> give verbose output
nmap -A target.com ----> agressionn scan
nmap -sT target.com ---> tcp scan
nmap -sS target.com ---> syn scan
nmap -sI <zombie-ip> target.com ---> zombie scan
nmap -f target.com ----> fragmentation
nmap -D 192.168.1.5,192.168.1.100,ME 192.168.1.10 target.com --> decoy
nmap --script vuln testphp.vulnweb.com
nmap -p22 --script ssh-brute --script-args userdb=user.txt,passdb=pass.txt
134.209.164.179
nmap -p21 --script ftp-anon.nse surfer.nmr.mgh.harvard.edu

‘

## 20) HTTPX

**Installation:**

go install -v github.com/projectdiscovery/httpx/cmd/httpx@latest
cd ~/go/bin/
ls
sudo cp httpx cp /usr/local/bin/

**Commands:**

httpx -l duplicate_filterted_subdomain.txt -sc -cl -ct -cdn -ip -mc  200,403 -t 60 > file.txt

Cat file.txt | awk '{print $1}'

## 21) Eyewitness

**https://github.com/RedSiege/EyeWitness**

**Chrome Install:**
sudo apt update
sudo apt install -y python3 python3-pip git chromium-driver xvfb

**Eyewitness Install:**
git clone https://github.com/FortyNorthSecurity/EyeWitness.git
cd EyeWitness
cd python
cd setup
./setup/setup.sh
cd ..

**Final Command:**
./EyeWitness.py -f /path/to/final_live_sub.txt --web

## 22) Nuclei Scanner

## Installation:

go install -v github.com/projectdiscovery/nuclei/v3/cmd/nuclei@latest
cd ~/go/bin
cp nuclei /usr/local/bin

**Nuclei Commands**

nuclei -u http://www.thevision.edu.pk/
nuclei -u testphp.vulnweb.com -as
nuclei -u testphp.vulnweb.com -tags admin,xss,api
nuclei -u testphp.vulnweb.com -t ~/nuclei-templates/
nuclei -list /root/Akashroxstarz/Recon/UK_defense/final_live_sub.txt severity low,info

**Nuclei AI:**

Cloud discovery~ Website

nuclei -auth
export PD_CLOUD_KEY="your api key"
nuclei -list /root/Akashroxstarz/Recon/subdomain_enum/unique.txt -ai "extract email address from webpages"

**Low Hanging Fruits:**

nuclei -list targets.txt -ai "Find exposed AI/ML model files (.pkl, .h5, .pt) that may leak proprietary algorithms or sensitive training data"
nuclei -list targets.txt -ai "Find exposed automation scripts (.sh, .ps1, .bat) revealing internal tooling or credentials"
nuclei -list targets.txt -ai "Identify misconfigured CSP headers allowing 'unsafe-inline' or wildcard sources"
nuclei -list targets.txt -ai "Detect pages leaking JWT tokens in URLs or cookies"
nuclei -list targets.txt -ai "Identify overly verbose error messages revealing framework or library details"
nuclei -list targets.txt -ai "Find application endpoints with verbose stack traces or source code exposure"
nuclei -list targets.txt -ai "Find sensitive information in HTML comments (debug notes, API keys, credentials)"
nuclei -list targets.txt -ai "Find exposed .env files leaking credentials, API keys, and database passwords"
nuclei -list targets.txt -ai "Find exposed configuration files such as config.json, config.yaml, config.php, application.properties containing API keys and database credentials."
nuclei -list targets.txt -ai "Find exposed configuration files containing sensitive information such as credentials, API keys, database passwords, and cloud service secrets."
nuclei -list targets.txt -ai "Find database configuration files such as database.yml, db_config.php, .pgpass, .my.cnf leaking credentials."
nuclei -list targets.txt -ai "Find exposed Docker and Kubernetes configuration files such as docker-compose.yml, kubeconfig, .dockercfg, .docker/config.json containing cloud credentials and secrets."
nuclei -list targets.txt -ai "Find exposed SSH keys and configuration files such as id_rsa, authorized_keys, and ssh_config."
nuclei -list targets.txt -ai "Find exposed WordPress configuration files (wp-config.php) containing database credentials and authentication secrets."
nuclei -list targets.txt -ai "Identify exposed .npmrc and .yarnrc files leaking NPM authentication tokens"

nuclei -list targets.txt -ai "Identify open directory listings exposing sensitive files"
nuclei -list targets.txt -ai "Find exposed .git directories allowing full repo download"
nuclei -list targets.txt -ai "Find exposed .svn and .hg repositories leaking source code"
nuclei -list targets.txt -ai "Identify open FTP servers allowing anonymous access"
nuclei -list targets.txt -ai "Find GraphQL endpoints with introspection enabled"
nuclei -list targets.txt -ai "Identify exposed .well-known directories revealing sensitive data"
nuclei -list targets.txt -ai "Find publicly accessible phpinfo() pages leaking environment details"
nuclei -list targets.txt -ai "Find exposed Swagger, Redocly, GraphiQL, and API Blueprint documentation"
nuclei -list targets.txt -ai "Identify exposed .vscode and .idea directories leaking developer configs"
nuclei -list targets.txt -ai "Detect internal IP addresses (10.x.x.x, 192.168.x.x, etc.) in HTTP responses"
nuclei -list targets.txt -ai "Find exposed WordPress debug.log files leaking credentials and error messages"
nuclei -list targets.txt -ai "Detect misconfigured CORS allowing wildcard origins ('*')"
nuclei -list targets.txt -ai "Find publicly accessible backup and log files (.log, .bak, .sql, .zip, .dump)"
nuclei -list targets.txt -ai "Find exposed admin panels with default credentials"
nuclei -list targets.txt -ai "Identify commonly used API endpoints that expose sensitive user data, returning HTTP status 200 OK."
nuclei -list targets.txt -ai "Detect web applications running in debug mode, potentially exposing sensitive system information."

## 23) Gau

go install github.com/lc/gau/v2/cmd/gau@latest
cd ~/go/bin
ls
cp gau /usr/local/bin

### Usage:

**gau target.com > gau.txt**

## 24) waybackurls

go install github.com/tomnomnom/waybackurls@latest
cd ~/go/bin
ls
cp waybackurls /usr/local/bin

### Usage:

**Waybackurls target.com > gau.txt**

## 25) Waybackup finder

git clone https://github.com/anmolksachan/WayBackupFinder.git
cd WayBackupFinder
python3 wayBackupFinder.py

### Usage:

**1**
**target.com**
**Load**

**To see output go to content folder**

## 26) back-me-up

git clone https://github.com/Dheerajmadhukar/back-me-up.git
cd back-me-up/
chmod +x back-me-up
bash backmeup.sh --check/-c
bash backmeup.sh --install/-i

### Usage:

**./backmeup.sh -d target.com**

## 27) waybackurls

go install github.com/projectdiscovery/katana/cmd/katana@latest
cd ~/go/bin
ls
cp katana /usr/local/bin

### Usage:

**Katna -u target.com**
**Katana -u target.com -jc**

## 28) JSFinder

go install -v github.com/kacakb/jsfinder@latest
cd ~/go/bin
ls
cp jsfinder /usr/local/bin

### Usage:

**Jsfinder -l list.txt**

## 29) Mantra

go install github.com/Brosck/mantra@latest
cd ~/go/bin
ls
cp mantra /usr/local/bin

### Usage:

**Cat js.txt | mantra**

## 30) Linkfinder

```
git clone https://github.com/GerbenJavado/LinkFinder.git
cd LinkFinder
python setup.py install
pip3 install -r requirements.txt
```

### Usage:

```
python3 linkfinder.py --input https://www.ajakcyberacademy.com
```

## 31) Gospider

```
go install github.com/jaeles-project/gospider@latest
cd ~/go/bin
ls
cp gospider /usr/local/bin
```

### Usage:

```
Gospider  -s https://target.com -a -w –js –robots
```

## 32) paramspider

```
go install github.com/jaeles-project/gospider@latest
cd ~/go/bin
ls
cp gospider /usr/local/bin
```

### Usage:

```
paramspider -d target.com -p '"<h1>test</h1>'
```

## 33) Arjun

pipx install arjun

### Usage:

Arjun -u target.com

## 34) URO

pipx install uro

### Usage:

All_URL.txt | uro

## 35) Grep automation

Download the grep.sh file locally on you PC, Now drag and Drop that file into your Linux  and type ‘chmod +x grep.sh’
(Or)
Copy then code from grep.sh, go to linux, open mousepad paste the script and save it as grep.sh file,  type ‘chmod +x grep.sh’

Now run the Script: **bash grep.sh**

## 36) dalfox

go install github.com/hahwul/dalfox/v2@latest
cd ~/go/bin
ls
cp dalfox /usr/local/bin

### Usage:

XSS_url.txt | dalfox

## 37) KXSS

go get github.com/Emoe/kxss
cd ~/go/bin
ls
cp kxss /usr/local/bin

### Usage:

XSS_url.txt | kxss

## 38) OpenRediWrecked

sudo apt-get install sed
sudo apt-get install grep
sudo apt-get install curl
pip install lolcat
sudo apt-get install -y figlet

git clone https://github.com/blackhatethicalhacking/OpenRediWrecked.git
cd OpenRediWrecked
chmod +x OpenRediWrecked.sh
./OpenRediWrecked.sh

### Usage:

Enter File name: open_redirect.txt

## 39) qsreplace

go install github.com/tomnomnom/qsreplace@latest
cd ~/go/bin
ls
cp qsreplace /usr/local/bin

## Usage:

Cat fuzz.txt | qsreplace '<h1>AJAK</h1>'

### 40) dirsearch

git clone https://github.com/maurosoria/dirsearch.git --depth 1
cd dirsearch
pip install -r requirements.txt

## Usage:

dirsearch -u http://testphp.vulnweb.com -w
/root/Akashroxstarz/Recon/wordlist/direserach.txt -x 500-509,301,302,404
--random-agent --delay=2 -o dirtest.txt

### 41) Gobuster

go install github.com/OJ/gobuster/v3@latest
cd /~go/usr/bin
cp gobuster /usr/local/bin

## Usage:

gobuster dir -u https://example.com -w /path/to/wordlist.txt

### 42) ffuf

go install github.com/ffuf/ffuf/v2@latest
cd /~go/usr/bin
cp ffuf /usr/local/bin

## Usage:

ffuf -u https://example.com/FUZZ -w /path/to/wordlist.txt -mc al

## 43) Extensions

Bulk url opener, screen recorder for POC, security header test
cookie editor, link grabber, shodan, urban vpn (IP rotate), wappalyzer
Hunter, foxy proxy, dot git, find something, D3coder.

Supporting Blogs:

https://ajakcybersecurity.medium.com/find-this-easy-csrf-in-every-website-a-sweet-p4-372a3198bf47

https://ajakcybersecurity.medium.com/how-i-bypassed-rate-limiting-to-account-takeover-1df722a527d5

## 44) WAFW00FF

git clone https://github.com/enablesecurity/wafw00f.git
cd wafw00f/
python3 -m pip install

## Usage:

Wafwoof target.com
Wafwoof -i list.txt

Subfinder -d target.com | httpx > alive.txt

Wafwoof -i alive.txt > nowwaf.txt

nmap selecthospital.in --script=http-waf-detect -p 80,443

## 45) Shodan Recon

[https://book.martiandefense.llc/notes/security-research/shodan-dork-cheatsheet](https://book.martiandefense.llc/notes/security-research/shodan-dork-cheatsheet) Cheatsheet

All other queries are in resources folder separately

## 46)Gitgrabber:

git clone https://github.com/hisxo/gitGraber.git
pip3 install -r requirements.txt
{Setup GitHub API token by navigating into config.py}

Usage:
python3 gitGraber.py -k wordlists/keywords.txt -q \"nasa.gov\"

## 47) Trugglehog:

git clone https://github.com/trufflesecurity/trufflehog.git
pip3 install -r requirements.txt
curl -sSfL
https://raw.githubusercontent.com/trufflesecurity/trufflehog/main/scripts/install.sh | sh -s -- -b /usr/local/bin

usage:

trufflehog git https://github.com/trufflesecurity/test_keys
--results=verified,unknown

trufflehog git https://file-to-repo (File to Repo)

trufflehog github --org=tesco  (Whole organization)

**XXX: All other queries are in resources folder separately**

## 48) GoogleDorks Recon

**All the manual dorks are available in resource Folder Individually**

**Automation:**

https://github.com/Viralmaniar/BigBountyRecon/blob/main/BigBountyRecon.exe

https://dorks.faisalahmed.me/#

## 49) Onliner Automation

All the one liner are available in resource Folder Individually

**Bolt Login Link:**

# https://bolt.new/?rid=sj9ypp

## 50) Wordpress Recon

The AJAK-WP recon tool is in resources, use git clone or copy the script and paste it in your mousepad and
**save it as .sh file and give**
**chmod +x {file name}**
**bash {filename}**

**WP-Recon**

**Tool usage:**

**wpscan --url https://target.com --enumerate vp,vt,tt,cb,dbe --random-user-agent**

**WP-Wordlist:**

**https://github.com/JavierOlmedo/UltimateCMSWordlists/tree/master/wordpress**

**51) Bonus Video:**

**All bug bounty tips and writesups link are there in resources PDF**

**https://www.proxynova.com/tools/comb**