

Web Application Enumeration, Revisited

Installing GO

He goes into installing [PimpMyKali](#). I use this script here:

```
sudo apt install -y golang-go
echo 'export GOPATH="$HOME/.go"' >> ~/.zshrc
echo 'export PATH="$PATH:${GOPATH}://bin:}/bin"' >> ~/.zshrc
source ~/.zshrc
mkdir -p ~/.go/{bin,pkg,src}
```

This installs GO unto the users home directory under a hidden folder `.go` to prevent clutter in my home directory, puts the GOPATH and the path for GO into my `.zshrc` file, then makes the bin, pkg, and src directories in the hidden `.go` folder.

Finding Subdomains in Assetfinder

[AssetFinder Github](#)

Installing Assetfinder

```
go get -u github.com/tomnomnom/assetfinder
```

Usage

```
assetfinder domain
```

```
(root@kali)-[~]  
# assetfinder tcm-sec.com >> tcmsec.txt
```

```
(root@kali)-[~]  
# cat tcmsec.txt | wc -l
```

22

```
(root@kali)-[~]  
# cat tcmsec.txt  
staging.tcm-sec.com  
webdisk.tcm-sec.com  
cpanel.tcm-sec.com  
webmail.tcm-sec.com  
certifications.tcm-sec.com  
cpcalendars.tcm-sec.com  
cpcontacts.tcm-sec.com  
dev.tcm-sec.com  
www.dev.tcm-sec.com  
www.staging.tcm-sec.com  
autodiscover.tcm-sec.com  
www.certifications.tcm-sec.com  
tcm-sec.com  
academy.tcm-sec.com  
www.tcm.rocks  
lnkd.in  
ow.ly  
email.m.teachable.com  
definitelynotavirus.zip  
davidbombal.wiki  
t.co  
themayor.tech
```

```
(root@kali)-[~]  
# _
```

```
(root@kali)-[~]  
# assetfinder --subs-only tcm-sec.com  
staging.tcm-sec.com  
webdisk.tcm-sec.com  
cpanel.tcm-sec.com  
webmail.tcm-sec.com  
certifications.tcm-sec.com  
cpcalendars.tcm-sec.com  
cpcontacts.tcm-sec.com  
dev.tcm-sec.com  
www.dev.tcm-sec.com  
www.staging.tcm-sec.com  
autodiscover.tcm-sec.com  
www.certifications.tcm-sec.com  
tcm-sec.com  
academy.tcm-sec.com  
staging.tcm-sec.com  
www.staging.tcm-sec.com  
autodiscover.tcm-sec.com  
cpanel.tcm-sec.com  
cpcalendars.tcm-sec.com  
cpcontacts.tcm-sec.com  
mail.tcm-sec.com  
tcm-sec.com  
webdisk.tcm-sec.com  
webmail.tcm-sec.com  
www.tcm-sec.com  
cpanel.server.tcm-sec.com  
cpcalendars.server.tcm-sec.com  
cpcontacts.server.tcm-sec.com  
mail.server.tcm-sec.com  
server.tcm-sec.com  
webmail.server.tcm-sec.com  
whm.server.tcm-sec.com  
www.server.tcm-sec.com  
cert-dev.tcm-sec.com  
www.cert-dev.tcm-sec.com  
merch.tcm-sec.com  
certifications.tcm-sec.com  
www.certifications.tcm-sec.com  
dev.tcm-sec.com  
www.dev.tcm-sec.com  
cpanel.tcm-sec.com  
cpcalendars.tcm-sec.com  
cpcontacts.tcm-sec.com  
tcm-sec.com  
webdisk.tcm-sec.com  
webmail.tcm-sec.com  
www.tcm-sec.com  
tcm-sec.com  
tcm-sec.com  
academy.tcm-sec.com  
www.academy.tcm-sec.com  
tcm-sec.com  
www.tcm-sec.com  
  
(root@kali)-[~]  
#
```

Automating

```
#!/bin/bash

url=$1 # Uses the first argument for the url

# Creates the URL Folder if it does not exist
if [ ! -d "$url" ];then
    mkdir $url
fi

# Creates the recon folder inside the URL if it does not exist
if [ ! -d "$url/recon" ];then
    mkdir $url/recon
fi

echo "[+] Finding subdomains with Assetfinder..."

# Saves the output into a recon folder inside the URL folder it creates
assetfinder $url >> $url/recon/assets.txt

# Cats for only the domain given and outputs this into a file, final.txt
cat $url/recon/assets.txt | grep $1 >> $url/recon/final.txt
```

Finding Subdomains with Amass

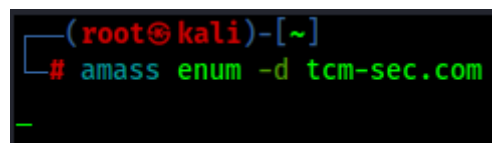
[Amass Github](#)

Install

```
go install -v github.com/owasp-amass/amass/v4/...@master
```

Usage

```
amass enum -d domain
```



```
(root@kali)-[~]
# amass enum -d tcm-sec.com
_
```

Automation

To the previous scripts, we add:

```
echo "[+] Finding subdomains with Amass..."
```

```
# Runs amass and saves the output into a recon folder inside the URL folder  
it creates
```

```
amass enum -d $url >> $url/recon/amass.txt
```

```
sort -u $url/recon/amass.txt >> final.txt
```

Finding Alive Domains with Httpprobe

[HTTPProbe Github](#)

Install

```
go install github.com/tomnomnom/httpprobe@latest
```

Usage

```
cat ListOfDomains.txt | httpprobe | sed 's/https\?:\/\/\\/'
```

Automation

To the previous scripts, we add:

```
echo "[+] Finding alive domains with HTTPProbe..."
```

```
# Runs httpprobe and saves the output into a recon folder inside the URL  
folder it creates
```

```
cat $url/recon/final.txt | sort -u | httpprobe | sed 's/https\?:\/\/\\/' >>  
Alive.txt
```

Screenshotting Websites with GoWitness

[GoWitness Github](#)

Install

```
go install github.com/sensepost/gowitness@latest
```

Usage

```
gowitness single https://domain
```

Automating the Enumeration Process

[sumrecon](#)

[TCM's modified script](#)

```
#!/bin/bash
url=$1
if [ ! -d "$url" ];then
    mkdir $url
fi
if [ ! -d "$url/recon" ];then
    mkdir $url/recon
fi
# if [ ! -d '$url/recon/eyewitness' ];then
#     mkdir $url/recon/eyewitness
# fi
if [ ! -d "$url/recon/scans" ];then
    mkdir $url/recon/scans
fi
if [ ! -d "$url/recon/httpprobe" ];then
    mkdir $url/recon/httpprobe
fi
if [ ! -d "$url/recon/potential_takeovers" ];then
    mkdir $url/recon/potential_takeovers
fi
if [ ! -d "$url/recon/wayback" ];then
    mkdir $url/recon/wayback
fi
if [ ! -d "$url/recon/wayback/params" ];then
    mkdir $url/recon/wayback/params
fi
if [ ! -d "$url/recon/wayback/extensions" ];then
    mkdir $url/recon/wayback/extensions
fi
if [ ! -f "$url/recon/httpprobe/alive.txt" ];then
    touch $url/recon/httpprobe/alive.txt
fi
if [ ! -f "$url/recon/final.txt" ];then
    touch $url/recon/final.txt
fi

echo "[+] Harvesting subdomains with assetfinder..."
assetfinder $url >> $url/recon/assets.txt
cat $url/recon/assets.txt | grep $1 >> $url/recon/final.txt
rm $url/recon/assets.txt
```

```
#echo "[+] Double checking for subdomains with amass..."
#amass enum -d $url >> $url/recon/f.txt
#sort -u $url/recon/f.txt >> $url/recon/final.txt
#rm $url/recon/f.txt

echo "[+] Probing for alive domains..."
cat $url/recon/final.txt | sort -u | httpprobe -s -p https:443 | sed
's/https?:\\/.\\/.\\/' | tr -d ':443' >> $url/recon/httpprobe/a.txt
sort -u $url/recon/httpprobe/a.txt > $url/recon/httpprobe/alive.txt
rm $url/recon/httpprobe/a.txt

echo "[+] Checking for possible subdomain takeover..."

if [ ! -f "$url/recon/potential_takeovers/potential_takeovers.txt" ];then
    touch $url/recon/potential_takeovers/potential_takeovers.txt
fi

subjack -w $url/recon/final.txt -t 100 -timeout 30 -ssl -c
~/go/src/github.com/hacker/subjack/fingerprints.json -v 3 -o
$url/recon/potential_takeovers/potential_takeovers.txt

echo "[+] Scanning for open ports..."
nmap -iL $url/recon/httpprobe/alive.txt -T4 -oA $url/recon/scans/scanned.txt

echo "[+] Scraping wayback data..."
cat $url/recon/final.txt | waybackurls >>
$url/recon/wayback/wayback_output.txt
sort -u $url/recon/wayback/wayback_output.txt

echo "[+] Pulling and compiling all possible params found in wayback
data..."
cat $url/recon/wayback/wayback_output.txt | grep '?*=' | cut -d '=' -f 1 |
sort -u >> $url/recon/wayback/params/wayback_params.txt
for line in $(cat $url/recon/wayback/params/wayback_params.txt);do echo
$line='';done

echo "[+] Pulling and compiling js/php/aspx/jsp/json files from wayback
output..."
for line in $(cat $url/recon/wayback/wayback_output.txt);do
    ext="${line##*.}"
    if [[ "$ext" == "js" ]]; then
        echo $line >> $url/recon/wayback/extensions/js1.txt
```

```

        sort -u $url/recon/wayback/extensions/js1.txt >>
$url/recon/wayback/extensions/js.txt
    fi
    if [[ "$ext" == "html" ]];then
        echo $line >> $url/recon/wayback/extensions/jsp1.txt
        sort -u $url/recon/wayback/extensions/jsp1.txt >>
$url/recon/wayback/extensions/jsp.txt
    fi
    if [[ "$ext" == "json" ]];then
        echo $line >> $url/recon/wayback/extensions/json1.txt
        sort -u $url/recon/wayback/extensions/json1.txt >>
$url/recon/wayback/extensions/json.txt
    fi
    if [[ "$ext" == "php" ]];then
        echo $line >> $url/recon/wayback/extensions/php1.txt
        sort -u $url/recon/wayback/extensions/php1.txt >>
$url/recon/wayback/extensions/php.txt
    fi
    if [[ "$ext" == "aspx" ]];then
        echo $line >> $url/recon/wayback/extensions/asp1.txt
        sort -u $url/recon/wayback/extensions/asp1.txt >>
$url/recon/wayback/extensions/asp.txt
    fi
done

rm $url/recon/wayback/extensions/js1.txt
rm $url/recon/wayback/extensions/jsp1.txt
rm $url/recon/wayback/extensions/json1.txt
rm $url/recon/wayback/extensions/php1.txt
rm $url/recon/wayback/extensions/asp1.txt
#echo "[+] Running eyewitness against all compiled domains..."
#python3 EyeWitness/EyeWitness.py --web -f $url/recon/httpprobe/alive.txt -d
$url/recon/eyewitness --resolve

```

Additional Resources

[The Bug Hunter's Methodology Youtube](#)

[Nahamsec Recon Playlist](#)