JS & GF

Saturday, April 18, 2020 4:19 PM

Reference Used:
----

1. **BASH script from**
   Gwendal Le Coguic
   https://gist.github.com/gwen001/0b15714d964d99c740a7e8998bd483df

2. **GF from** https://github.com/tomnomnom/gf

3. **After setting up ParamSpider & GF . I tried to use above script with GF it was really nice.**

ParamSpider + GF (for massive pwnage)

Lets say you have already installed ParamSpider and now you want to filter out the juicy parameters from plethora of parameters.
No worries you can easily do it using GF(by tomnomnom) .

Note : Make sure you have go properly installed on your machine .

Follow along this :

```
$ go get -u github.com/tomnomnom/gf
$ cp -r $GOPATH/src/github.com/tomnomnom/gf/examples ~/.gf

Note : Replace '/User/levi/go/bin/gf' with the path where gf binary is located in your system.

$ alias gf='/User/levi/go/bin/gf'
$ cd ~/.gf/
Note : Paste JSON files(https://github.com/devanshbatham/ParamSpider/tree/master/gf_profiles)  in ~/.gf/ folder

Now run ParamSpider and navigate to the output directory

$ gf redirect domain.txt    //for potential open redirect/SSRF parameters
$ gf xss domain.txt         //for potential xss vulnerable parameters
$ gf potential domain.txt   //for xss + ssrf + open redirect parameters
$ gf wordpress domain.txt   //for wordpress urls

[More GF profiles to be added in future]
```

~~~~~~
~~~~~~

JS endpoint extractor combined with GF
---

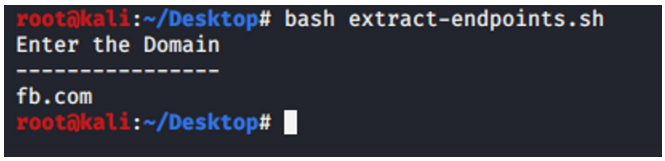
1. Save it as a bash file.

```
echo "Enter the Domain"
echo "----------------"
read DOMAIN
curl -L -k -s https://$DOMAIN | tac | sed "s#\\\/#\/#g" | egrep -o "src['\"]?
\s*[=:]\s*['\"]?[^\"]+.js[^\"> ]*" | awk -F '//' '{if(length($2))print "https://"$2}' | sort -fu | xargs -I
'%' sh -c "curl -k -s \"%\" | sed \"s/[;}\)>]/\n/g\" | grep -Po \"(['\\\"](https?:)?[/]{1,2}[^'\\\"> ]
{5,})|(\.(get|post|ajax|load)\s*\(\s*['\\\"](https?:)?[/]{1,2}[^'\\\"> ]{5,})\"" | awk -F "['\"]" '{print
$2}' | sort -fu  -o /root/go/bin/$DOMAIN

function ejs() {
  curl -L -k -s "$1" | tac | sed "s#\\\/#\/#g" | egrep -o "src['\"]?\s*[=:]\s*['\"]?[^\"]+.js[^\"> ]*" |
awk -F '//' '{if(length($2))print "https://"$2}' | sort -fu | xargs -I '%' sh -c "curl -k -s \"%\" | sed
\"s/[;}\)>]/\n/g\" | grep -Po \"(['\\\"](https?:)?[/]{1,2}[^'\\\"> ]{5,})|(\.(get|post|ajax|load)\s*
\(\s*['\\\"](https?:)?[/]{1,2}[^'\\\"> ]{5,})\"" | awk -F "['\"]" '{print $2}' | sort -fu
}

chmod 777 extract-endpoints.sh
```

2. You need to change the blue highlighted text to your gf path because here the output will be send to gf path.



3. Use the commands like we used in ParamSpider & gf

   ./gf redirect outputfile
   ./gf xss outputfile
   ./gf potential outputfile

```
root@kali:~/go/bin# ./gf potential twitter.com
twitter.com:18:https://twitter.com/search?q=%23
twitter.com:19:https://twitter.com/search?q=%24
twitter.com:38:/?logout=
twitter.com:42:/search?f=users&q=
twitter.com:43:/search?q=query&src=typd
root@kali:~/go/bin# ./gf potential fb.com
fb.com:57:/ajax/hovercard/user.php?id=
root@kali:~/go/bin# ./gf xss twitter.com
twitter.com:18:https://twitter.com/search?q=%23
twitter.com:19:https://twitter.com/search?q=%24
twitter.com:42:/search?f=users&q=
twitter.com:43:/search?q=query&src=typd
root@kali:~/go/bin# ./gf redirect twitter.com
twitter.com:38:/?logout=
root@kali:~/go/bin# ./gf potential hackerearth.com
hackerearth.com:9:/gtag/js?id=
hackerearth.com:55:https://www.hackerearth.com/zeus/facebook/login/?display=popup&next=%2Fsocial-login-complete%2F%3Forigin_protocol%3Dhttps%26origin_host%3Dwww.hackerearth.com%26redirect%3D/challenges/hiring/valuelabs-ml-hiring-2019/&update=facebook&source=challenge-hiring-869486&purpose=login
hackerearth.com:56:https://www.hackerearth.com/zeus/facebook/login/?display=popup&next=%2Fsocial-login-complete%2F%3Forigin_protocol%3Dhttps%26origin_host%3Dwww.hackerearth.com%26redirect%3D/challenges/&update=facebook&source=challenges&purpose=login
hackerearth.com:57:https://www.hackerearth.com/zeus/github/login/?next=%2Fsocial-login-complete%2F%3Forigin_protocol%3Dhttps%26origin_host%3Dwww.hackerearth.com%26redirect%3D/challenges/hiring/valuelabs-ml-hiring-2019/&update=github&source=challenge-hiring-869486&purpose=login
hackerearth.com:58:https://www.hackerearth.com//zeus/github/login/?next=%2Fsocial-login-complete%2F%3Forigin_protocol%3Dhttps%26origin_host%3Dwww.hackerearth.com%26redirect%3D/users/basic-details/&update=github&source=homepage&purpose=signup
hackerearth.com:59:https://www.hackerearth.com/zeus/google/login/?next=%2Fsocial-login-complete%2F%3Forigin_protocol%3Dhttps%26origin_host%3Dwww.hackerearth.com%26redirect%3D/challenges/hiring/valuelabs-ml-hiring-2019/&update=google&source=challenge-hiring-869486&purpose=login
hackerearth.com:60:https://www.hackerearth.com/zeus/google/login/?next=%2Fsocial-login-complete%2F%3Forigin_protocol%3Dhttps%26origin_host%3Dwww.hackerearth.com%26redirect%3D/challenges/&update=google&source=challenges&purpose=signup
hackerearth.com:90://www.googletagmanager.com/a?id=
root@kali:~/go/bin#
```