Javascript

secretKEY - Endpoints - Subdomains

Mahmoud M. Awali

@0xAwali

attacker

Workflow Of Javascript Files

JS URLs Monitoring

Secret Keys

Subdomains

Endpoints

Host

JS Files

Note

When Parsing JS For Endpoints , Please Keep Your Eyes Open

- Tweet

```
/       Root Directory
 .      This Location
..      Up A Directory
./      Current Directory
 ../    Parent Of Current Directory
../../  Two Directories Backwards
```

attacker

My Methodology

Try To Extract All The Javascript Files From A Set Of Given Urls By Using Tools e.g. getJS

```
root@mine:~#./getJS --complete --header "Auth: token" --input domains.txt --output out.txt
```

" --complete " Complete The URLs e.g. https://www.company.com/api/file.js

" --header "Auth: token" " Custom Request Header

" --input domains.txt " Input File With URLs

" -output output.txt " The File Where To Save The Output

attacker

My Methodology

Try To Extract All The Javascript Files From A Set Of Given Urls By Integration Tools e.g. gau AND subJS

```
root@mine:~#cat domains.txt | gau -subs | subjs -ua "Value-User-Agent-Header" | tee -a out.txt
```

" jau -subs " Fetches URLs From AlienVault's And Wayback Machine Include Subdomains Of Target

" -ua "Value-User-Agent-Header" " Value Of User-Agent Header To Send In Requests

" | tee -a out.txt " Save The Output

attacker

My Methodology

Try To Extract All The Javascript Files From A Set Of Given Urls By Using One-Liner Command e.g. Integration Tools e.g. gospider , httpx , subJS AND anew

-  Tweet

Steps to produce :-
1 - Open Your Terminal
2 - Write This Command

```
xargs -I@ -a domains.txt -P10 sh -c 'gospider -a -d 4 -t 3 -c 50 -s @ | tr "[] " "\n" | grep -oE "(/\*([^*]|(\*+[^*/]))*\*+/)|(.*)|(//.*)" | httpx -silent -threads 200 | subjs -c 100 -t 5 | anew subjsUrls'
```

attacker

My Methodology

Try To Extract All The Endpoints From Javascript URL OR File
By Using Tools e.g. LinkFinder

```
root@mine:~#cat extractEndpoints.sh
#!/bin/bash
for jsFile in `cat out.txt`
do
        Python3 linkfinder.py -o cli -i $jsFile | tee -a newEndpoints.txt
done
root@mine:~#chmod +x extractEndpoints.sh && ./extractEndpoints.sh
```

" -o cli " Print Output to STDOUT " " -i urlJS OR out.txt " Input a Javascript URL OR File

attacker

My Methodology

Try To Extract All The Endpoints From Javascript URL OR File By Integration Tools e.g. subJS AND JSA

```
root@mine:~#cat domains.txt | subjs -ua "Value-User-Agent-Header" | python3 jsa.py -e -v
```

" -ua "Value-User-Agent-Header" " Value Of User-Agent To Send In Requests

" -e " Exclude 3rd Party JS Files

" -v " Verbose Mode

attacker

My Methodology

Try To Extract All The Endpoints From Any Domain By Using One-Liner Command e.g. One Of Those Commands

- Tweet
- Tweet

Steps to produce :-
1 - Open Your Terminal
2 - Write This Command

```
curl -L -k -s https://www.comapny.com | 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
```

attacker

My Methodology

Try To Extract All The Endpoints From Any Javascript File By Integration Tools e.g. js-beautify AND CyberChef

-  Tweet

Steps to produce :-

1 - Open Your Terminal
2 - Write This Command
    wget https://www.comapny.com/app.js
    js-beautify file.js > pretty.js
3 - Copy Content Of pretty.js To CyberChef

attacker

My Methodology

Try To Extract All Subdomains From Any Domain By Using Tools e.g. subscraper

```
root@mine:~#python3 subscraper.py -u URL -v -o output.txt
```

" -u URL " URL Of Target e.g. https://www.company.com

" -v " Enables Verbosity

" -o output.txt " File Where To Save Subdomains

attacker

My Methodology

Try To Extract All The Endpoints And Subdomains From Any Domain By Using Tools e.g. JSFinder

```
root@mine:~#python3 JSFinder.py -u URL -c "COOKIE-Value" -d -ou url.txt -os subdomains.txt
```

" -u URL " URL Of Target e.g. https://www.company.com

" -c "COOKIE-Value" " Value Of Cookie Header To Send In Requests

" -d " Rescan Every URL You Will Find It aka Deep Find

" -ou url.txt " File To Save Paths " -os subdomains.txt " File To Save Subdomains

attacker

My Methodology

Try To Scan Javascript URL File To Extract Tokens By Using Tools e.g. SecretFinder

```
root@mine:~#python3 SecretFinder.py -i out.txt -H 'Header: Value' -o secrets.txt
```

" -i out.txt " Input a Javascript URL From out.txt

" -H 'Header: Value' " Custom Request Header

" -o secrets.txt " Save Output To secrets.txt

attacker

My Methodology

Try To Scan Javascript URL File By Using Tools e.g. JScanner

```
root@mine:~#cat out.txt | JScanner --- -o scan.txt -t 10
```

" --- " Input a Javascript URL From STDIN

" -o scan.txt " Save Output To scan.txt

" -t 10 " Number Of Threads

attacker

My Methodology

Try To Analyze Javascript Files From Your Command Line e.g. js-beautify file.js > pretty.js && grep -Eo "(http|https)://[a-zA-Z0-9./?=_-]*" pretty.js | sort -u

-  **Tweet**

Steps to produce :-

1 - Open Your Terminal
2 - Write This Command

```
wget https://www.comapny.com/app.js
js-beautify file.js > pretty.js
grep -Eo "(http|https)://[a-zA-Z0-9./?=_-]*" pretty.js |
        sort -u
```

attacker

My Methodology

Try To Search For Javascript Manually To Discover New parameters , Endpoints And Subdomains OR References To More API Calls OR Get Dev Comments OR Tokens

- Video
- Writeup
- Writeup

Steps to produce :-

1 - Browse Your Target e.g. https://www.company.com
2 - Click Right , Choose View Page Source
3 - Press Ctrl Plus F To Display Search Box
4 - Search For Javascript Files e.g. .js
5 - Search For Certain Keywords e.g.
api , internal , url: , token , var = , // , https://
, company.com , parameter etc

attacker

Top 15 DOM Base Open URL Parameter

-  Tweet

```
location
location.host
location.hostname
location.href
location.pathname
location.search
location.protocol
location.assign()
location.replace()
open()
domElem.srcdoc
jQuery.ajax()
$.ajax()
XMLHttpRequest.open()
XMLHttpRequest.send()
```

attacker

My Methodology

Try To Execute Those Functions That Contains Sensitive Keywords e.g. key , API Key etc In Your Browser's Console To See What They Do

- Tweet

Steps to produce :-

1 - Browse Your Target e.g. https://www.company.com
2 - Click Right , Choose View Page Source
3 - Search For Functions That Contains Certain Keywords e.g. api , internal , url: , var = , // , https:// , company.com , parameter etc
4 - Click Right , Choose Inspect Element (Q)
5 - Click Console , Write Your Funcion e.g. secret() Then Press Enter

attacker

My Methodology

Try To Monitor Specific Javascript URL To Get Notification On Slack OR Telegram If There Are Changes By Using Tools e.g. JSMon

Steps to produce :-
1 - Open File .bashrc
2 - Write Those

```
export JSMON_NOTIFY_SLACK=True
export JSMON_SLACK_TOKEN=token
export JSMON_SLACK_CHANNEL_ID=channel
```

3 - Open Your Terminal
2 - Write This Command

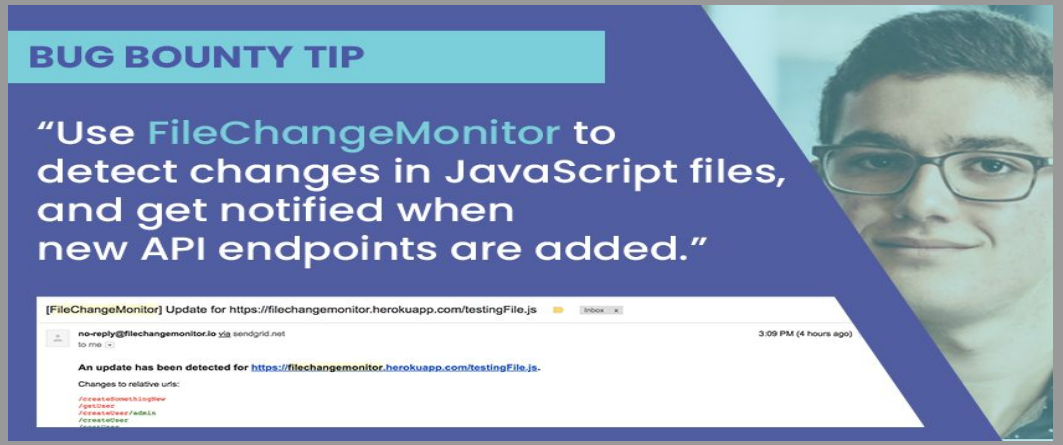
```
source .bashrc
cd path/to/jsmon/tool
crontab -e && @daily jsmon.sh
echo "https://www.company.com/file.js" >> targets/company.com
python3 jsmon.py
```

attacker

My Methodology

Try To Use FileChangeMonitor To Monitor JavaScript Files And Discover Endpoints When They're Added

- Tweet
- Tweet



Thank You

Mahmoud M. Awali

@0xAwali