Javascript for hackers:

(xss stuff backticks `` can replace () in javascript so if a filter is picking up () use `` instead. Double URL encoding can be useful!)

Javascript is a CLIENT SIDE, object oriented scripting language

--its important to note the client side aspect as this is how we will use it for exploitation

Makes static webpages dynamically

--eg adding animations, or making chnages to a page without a reloading it

Javascript often has some secrets hidding in it, such as:

-new endpoints

-api keys

-business logic

-HTML/javascript sinks (look for xss here)

-secrets/ passwords

-potentially dnagerous areas of code (eval, dangerouslySetInnerHTML ...)

------------------------------------------------------------------------------------------------------------------------------------------

Attack Strategy:

First step is to find the exposed javascript files

-we can do this simply by exploring the webpage with burp running to build the site map

--then we can filter for only javascript files as show below.

Graphical user interface, text, application, email

Description automatically generated

--can use some tools here too Linkfinder & SecretFinder are usually regarded as the best

-- BUT .. Manually exploring and reading the javascript itself is NECESSARY!. Get used to exploring, reading and understanding javascript. This is extremely important as our automated tools will never give us insights on for example, business logic which can only be found using manual exploration and analysis and can be extraordinarily helpful for us when attacking an app.

---we can also run bits of javascript within our own environment to dynamically analyze the code to get more meaning as to its function

--we can also use the waybackurls tool and grep for .js files as shown below:

waybackurls google.com | grep “\.js” | uniq | sort

-----------------------------------------------------------

once we have this list of javascript we can then manually look through them and analyze the code as well as passing them into the linkfinder and secretfinder

python3 secretfibder.py -i <https://google.com/> -e -o cli -> -e will check all .js files for that domain and output will be on the command line

python3 secretfinder.py -i <https://google.come/example.js> -o cli -> checks that 1 .js file and outputs to cli

cat jsfiles.txt | xargs -I %% bash -c ‘python3 secretfiner.py -i %% -o cli’ -> form a list of javascript files and pass them all in (do this with the output from waybackurls or even just a regular list you compile with burp sitemap)

often javascript files we find will be compressed and look super ugly..

-we can fix this using a uglifyjs

uglify copypastedjsfile.js -b -> “beautifies” the javascript making it much more readable.

-------------------------------------------------

Javascript can also be obfuscated which we will have to research and then fix manually which is much more challenging

Javascript can also be “chunked” basically just a big javascript file that is split up across a few or many separate URLs. For this we also just have to copy and paste it together and beautify if necessary.