

# Week 02

# Web Hacking I

Nathan and Kevin



`sigpwny{w3b_intr0}`



Client side validation



# Announcements

- CSAW (TOMORROW!!!)
- Fall Recruitment Event



# Table of Contents

- How the web works
  - Clients and Servers
- How websites work
  - The bones, skin, and brain of the internet
    - HTML
    - CSS
    - JavaScript
  - Cookies, local storage
- Chrome Devtools
- Challenge walkthrough



# How the web works

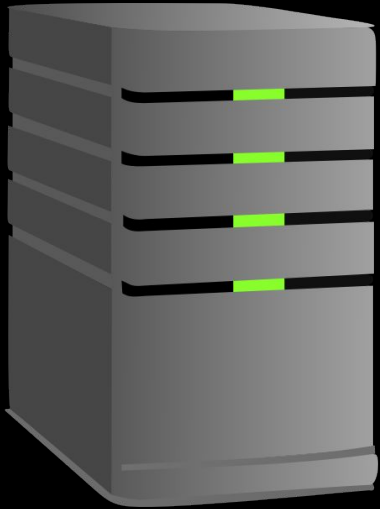
At a very high level!



# How the web works



Browser



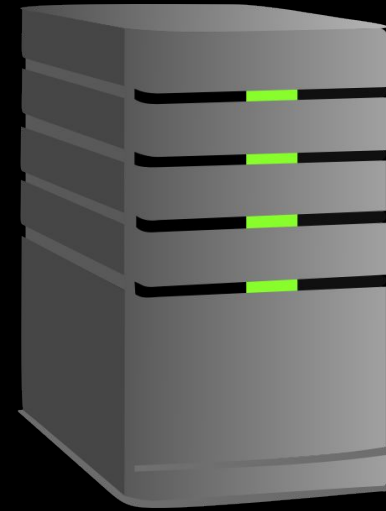
Server



# How the web works



Browser



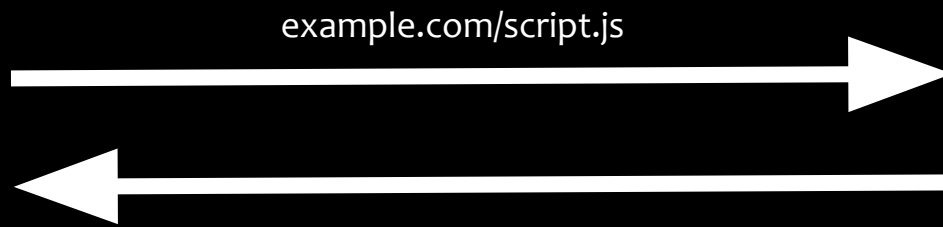
Server



# How the web works



Browser



Server





# How websites work

The Bones, Skin, and Brains of the Internet



# How websites work

- Websites displayed by browser according to
  - HTML
  - CSS
  - Javascript



# HTML - The Bones

- Defines the *layout* of websites
  - Where are the images, buttons, and textboxes?
- Defines where to load the javascript and CSS from

```
<html>  
  <p>Hello world!</p>  
    
  <script src="script.js"></script>  
</html>
```



# CSS - The Skin

- Defines what website elements should *look* like
- Can be written in the HTML or loaded from external file

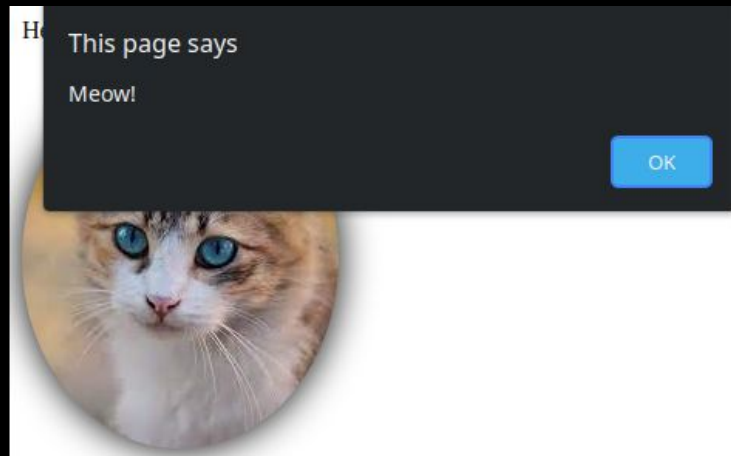
```
img {  
  border-radius: 50%;  
  filter: drop-shadow(0 0 0.75rem  
black);  
}
```



# JavaScript - The Brains

- Programming language to make website *do something*
  - Do something when button is pressed
  - Animate things on webpage
  - Make requests to other endpoints

```
document.getElementById("cat").onclick = () => alert("Meow!");
```



# How websites work

Cookies and Local Storage



# Cookies



- Small pieces of information stored across visits to same web page
- Maintained by browser, sent along with requests
- Main usages:
  - Maintain a “session” after you log in to a site
  - Track you for advertising purposes



# Local Storage

- Store key/value pairs like a cookie
- *Not* sent with requests to server
- Larger storage size limit (4KB vs 5MB)
- Can persist indefinitely





# Important Tools

Devtools



# Devtools - Inspect Element

1. Right click

2. Inspect

3.

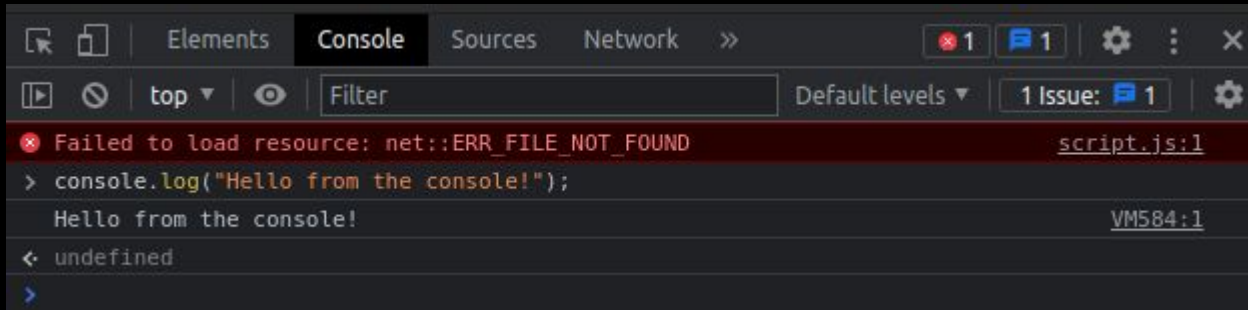
```
<html>
  <head>...</head>
  <body>
    <p>Hello world!</p>
    </script>
  </body>
</html>
```

- Inspect HTML of page
- Delete or add elements
- View event listeners and styles



# Devtools - Console

- View errors
- Execute your own javascript to interact with page and existing javascript

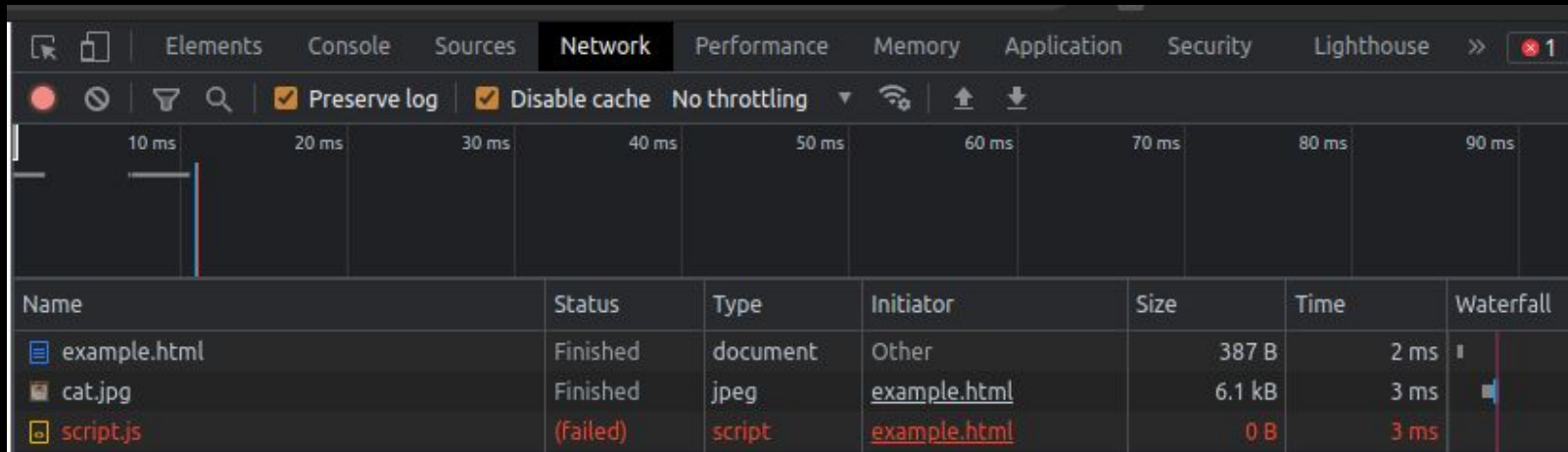


The screenshot shows the Chrome DevTools Console. At the top, there are tabs for Elements, Console, Sources, and Network. The Console tab is active, showing a list of messages. The first message is an error: "Failed to load resource: net::ERR\_FILE\_NOT\_FOUND" from "script.js:1". Below it is a log message: "Hello from the console!" from "VM584:1". The console also shows "undefined" and a prompt character ">".

```
Failed to load resource: net::ERR_FILE_NOT_FOUND script.js:1
> console.log("Hello from the console!");
Hello from the console! VM584:1
< undefined
>
```



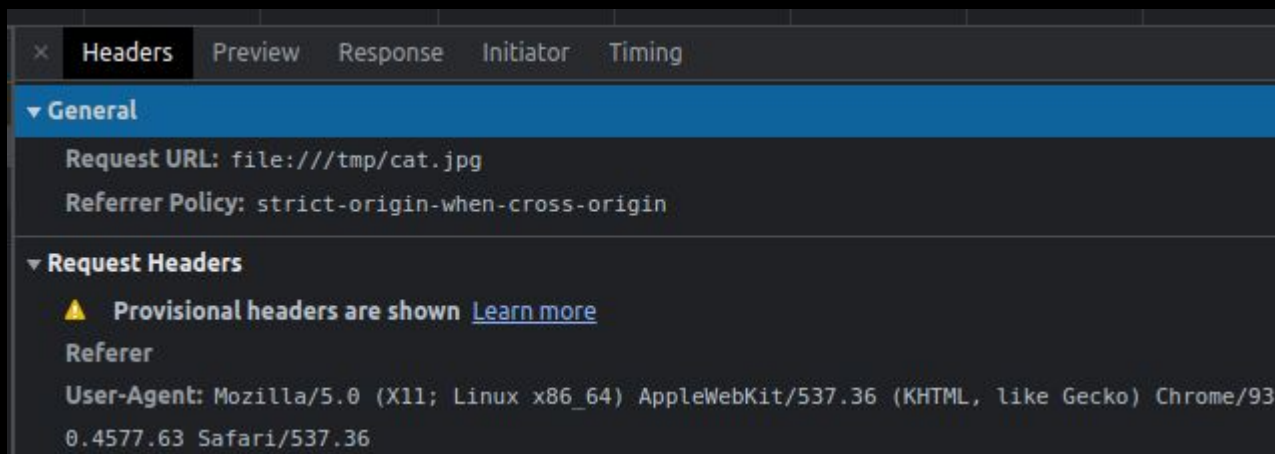
# Devtools - Network



The screenshot shows the Chrome DevTools Network tab. At the top, there are tabs for Elements, Console, Sources, Network (selected), Performance, Memory, Application, Security, and Lighthouse. Below the tabs, there are controls for Preserve log, Disable cache, and No throttling. A waterfall chart shows three requests: example.html (2 ms), cat.jpg (3 ms), and script.js (3 ms). Below the chart is a table with the following data:

Name	Status	Type	Initiator	Size	Time	Waterfall
example.html	Finished	document	Other	387 B	2 ms	
cat.jpg	Finished	jpeg	example.html	6.1 kB	3 ms	
script.js	(Failed)	script	example.html	0 B	3 ms	

- View requests sent from your browser
  - Resources requested from server
  - Login forms
  - File uploads



The screenshot shows the Chrome DevTools Headers tab for the request cat.jpg. The tabs are Headers (selected), Preview, Response, Initiator, and Timing. The General section shows the Request URL as file:///tmp/cat.jpg and the Referrer Policy as strict-origin-when-cross-origin. The Request Headers section shows a warning that provisional headers are shown and lists the Referer header with its value: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36.

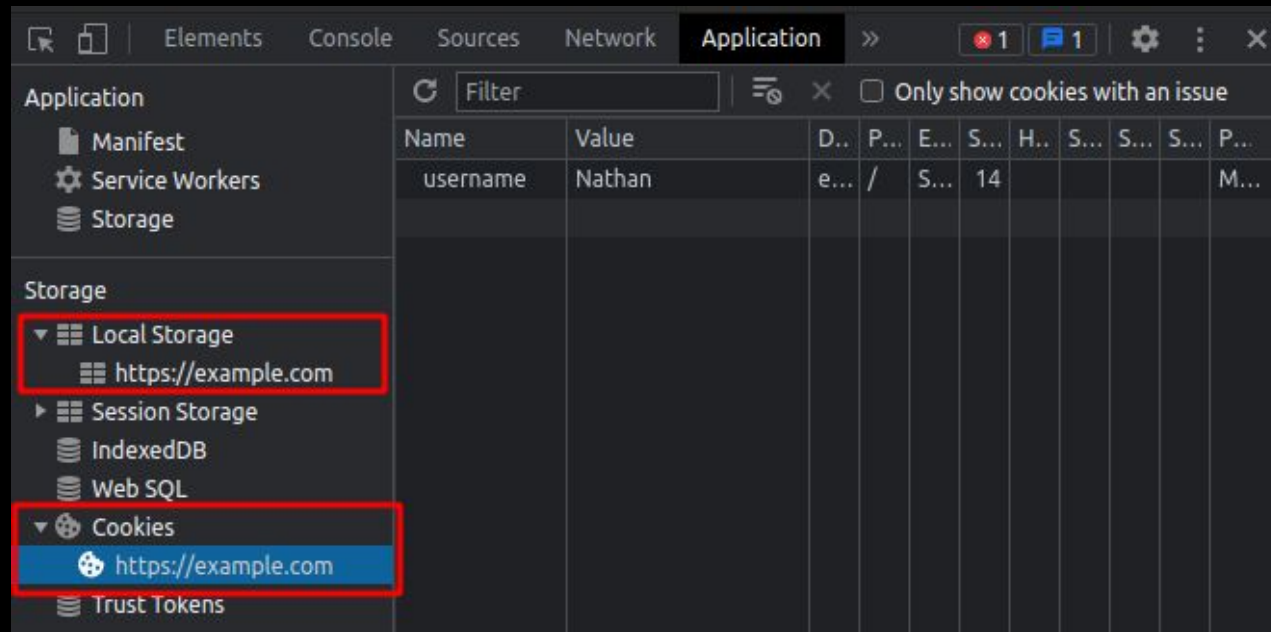
General
Request URL: file:///tmp/cat.jpg
Referrer Policy: strict-origin-when-cross-origin

Request Headers
Provisional headers are shown <a href="#">Learn more</a>
Referer
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36



# Devtools - Application



The screenshot shows the Chrome DevTools Application panel. The left sidebar is divided into 'Application' and 'Storage' sections. Under 'Storage', 'Local Storage' and 'Cookies' are expanded, with 'https://example.com' selected in both. The main area displays a table of cookies for the selected domain.

Name	Value	D..	P..	E..	S..	H..	S..	S..	S..	P..
username	Nathan	e...	/	S...	14					M...

- View cookies and local storage for a website
- Modify contents to mess with web service



# Challenge Walkthrough!



# Next Meetings

## **Weekend Seminar:** Web Hacking II!

- XSS, SQL injection
- There likely will be a web hacking 3!

## **Next Thursday:** Crypto I!

- Crypto fundamentals (keys, communication, encoding)
- Introduction to basic crypto schemes (symmetric and asymmetric)

