**Class Notes Thursday**

**November 11th, 2021**

How Can We Fix JavaScript on The Web?

1.) Separate out content types

2.) Define policies for each content type

* Developers define the policy, the browser follows it

3.) JS Script Policy

* Run or Don’t Run
* Whitelist script sources to run, and Blacklist script sources that won’t run
* HTTPS (Only run scripts from secure connections)
* ORIGIN (Only run scripts form the origin website)
* NONE (Don’t run any scripts at all)

This exists today in browsers and is called **CSP: Content Security Policy**

* The CSP protects the bowser, not the web server running the website
* The CSP is set inside the HTTP headers

**Browser Responsibilities:**

* Browser is also called the User Agent
* Protect the user from malicious attackers

**CSP Examples:**

Content-Security-Policy: default-src ‘self’

* The default part is the directive for the content types
* Only allow content from ‘self’ which is the current website

Content-Security-Policy: default-src ‘self’ trusted.com

* Allow content also from trusted.com

Content-Security-Policy: default-src ‘self’ trusted.com sub-domain

* Allow conent from self, trusted, and all trusted.com sub domains

Content-Security-Policy: default-src https://secure.kittenware.com

* Only allow content from https and kittenwar.com

Content-Security-Policy: default-src ‘self’

img-src \*;

media-src media1.com media2.com

script-src scripts.example.com

* Only allow content from self, images from anywhere, media from media 1 and 2, and scripts form exapmle.com

To Allow inline scripts we can use nonce- or hash-