# Introduction to HTML Parsers and Mutation

## Road to mXSS

# whoami

- Online presence : GameOv3r
- I play Web & Crypto for TamilCTF
- Currently playing around with client side stuffs

# Quick history of HTML

**HTML** —which is short for HyperText Markup Language— is the official language of the World Wide Web and was first conceived in 1990. HTML is a product of **SGML** (Standard Generalized Markup Language)

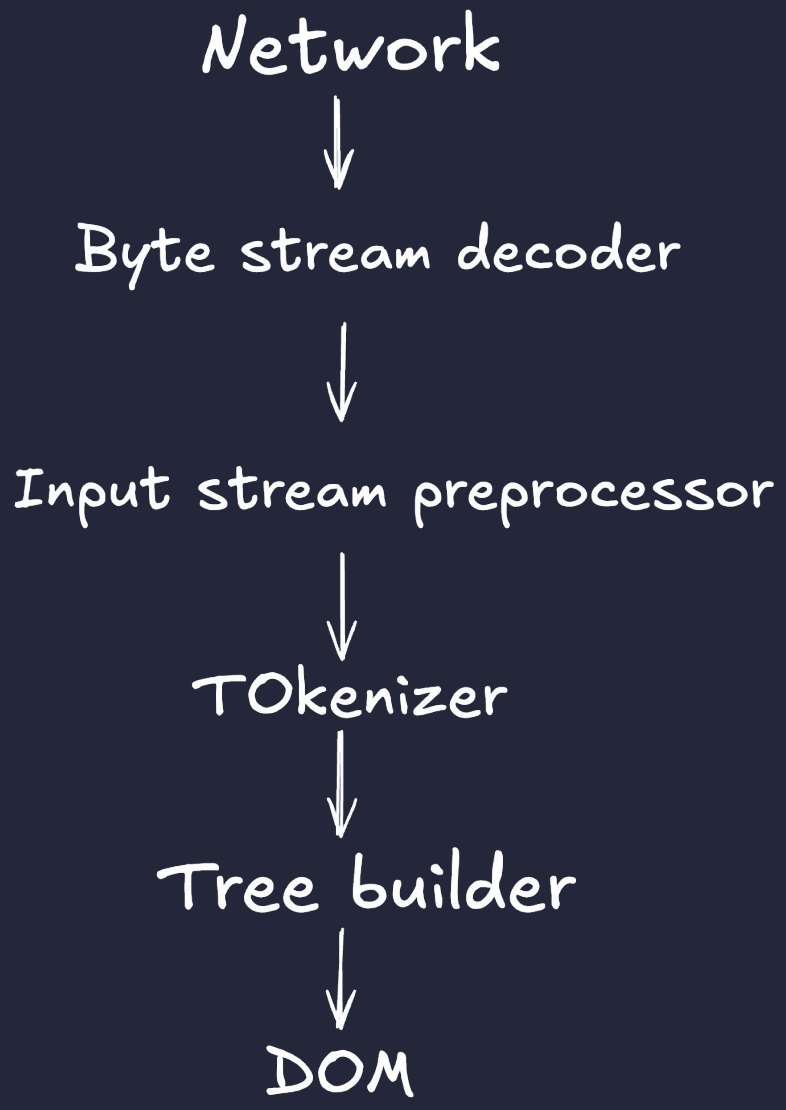
**Browser-Wars** : Soon, companies began creating browsers —the software required to view an HTML document, i.e., a web page— and as they gained popularity it gave rise to competition and other web browsers.

**Browser Makers** began inventing HTML elements which only works on their browsers,which let to webpage's working fine on one browser and failed on another.

This created a necessity for a standard and HTML standard was created and followed by all major web browsers. [**W3C** & **WHATWG**]

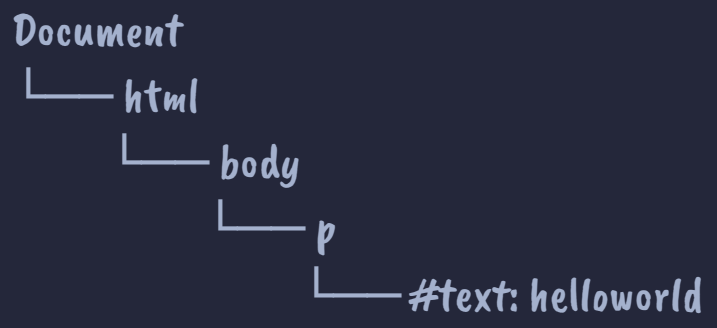
# What is a PARSER ?

# HTML PARSER




Ex : <!doctype html><p>helloworld

- Bytes go over the network
- Decoder will produce  a stream of code points
- The tokenizer walks through stream of code points character by character and emit  tokens : doctype token,a start tag token (p) and character token.
- The tree builder takes the tokens and build the dom.



# The birth of mutation?

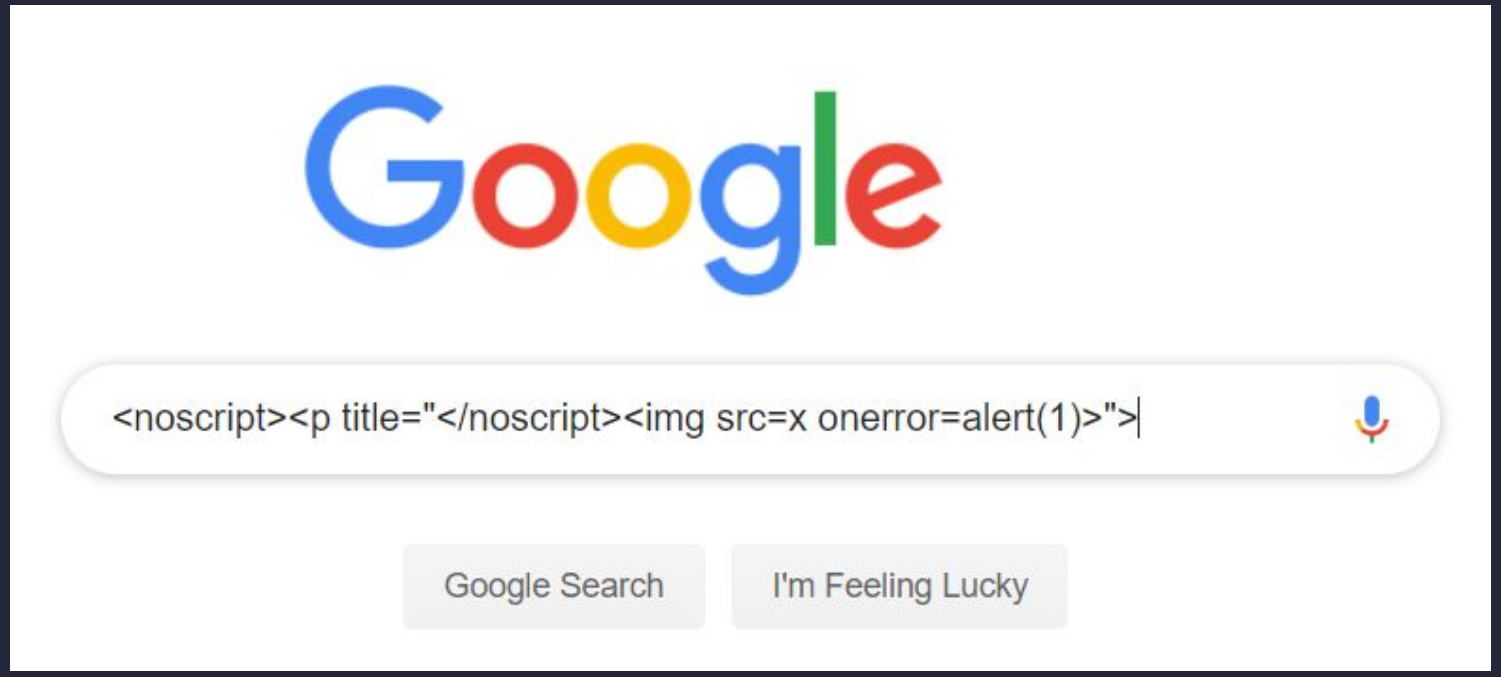
- The HTML parser spec specifies exactly what to do incase of an error.
- Technically it is allowed to abort processing upon an error,but no browser does that.
- Instead they follow the specification to recover from the error in some particular way, which is designed to compatible with web content.

# mutations.....

- When a parser fixes broken markup
- When a parser rearrange elements
- When a parser normalize attribute quotes

# mXSS

## The story of google search bar XSS



# noscript

## Funky element

## Two faced

Behave in one way if scripting is enabled


Behave in another way if scripting is disabled

# There's more to it !

Namespaces?
Element type?
Parsing rule?
Parsing mode?
Insertion points?

https://kabilan1290.github.io/sniper/

# <Table>

Forester parenting
<table> when encountering other HTML elements

# mXSS using table

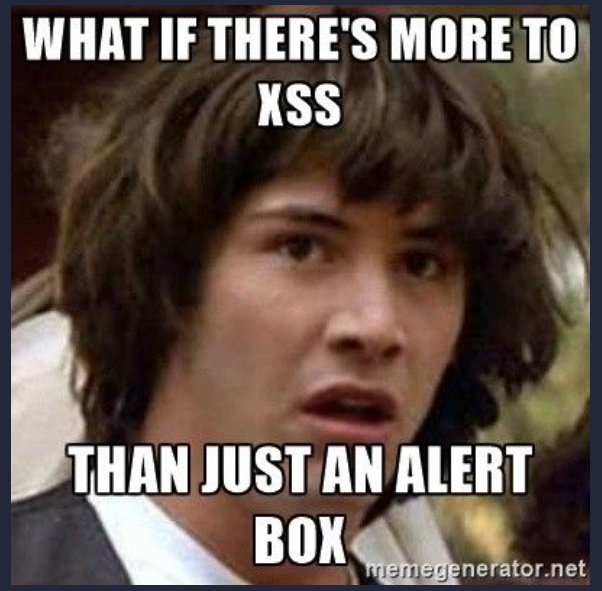
https://game0v3r.vercel.app/blog/wwctf-saas-challenge-writeup

# mXSS Cheatsheet

https://sonarsource.github.io/mxss-cheatsheet/

# Moving forward

- Don't spray and pray XSS payloads
- Understand the working ! read spec
- fancy payloads is not really fancy.
- Research provides quality output.




simple yet fascinating

# Questions?