Søren Krogh - 20105661
Emil Madsen - 20105376
K. Rohde Fischer - 20052356

November 17, 2014

Introduction

Embedded JavaScript

```
// The test object containing tests
var testObject = {
  test1: function() { /* ... */ },
  test2: function() { /* ... */ }
  // ...
}

Object.defineProperty(testObject, "init", { dontenum: true })
```

Object oriented hierarchy

Fun Sorting Game

Asynchronous loading

XMLHttpRequest

XMLHttpRequrest(xhr) provides some security features e.g: Same origin policy (sikkerhedspolitik). This tries to prevent cross domain attack and does not blindly parse code(You can still explicitly parse it). Thus it's still possible to implement the security errors. Furthermore using xhr also gives acess to http statuses which can be used to provide better feedback. A downside to xhr is that it can be rather tedious to code, but most libraries provide nice wrappers and if not using a library, it's fairly easy to code yourself.

iframe & Script Tags

Iframe and Script tags both have the advantage that they are rather light on code. On the other hand, neither iframe nor Script tags prevents cross origin attacks or execution of code. Usually this method is definitely not recomended but is in some cases used when browser permossions or similar prevents xhr. A common format in that case is JSONP.

Boss snooping

JSLint

The concept of validating JavaScript is justified because the interpretors operate by the ”keep on trucking” philosophy. This in turn means that validation is limited at best. Thus JavaScript code is quite error prone and alot of pages have errors due to this fact.

The validation by JSLint is probably quite usefull, but seems to overly pedantic. For instance it require a whitespace in places where it has, and can never have, any semantic difference.