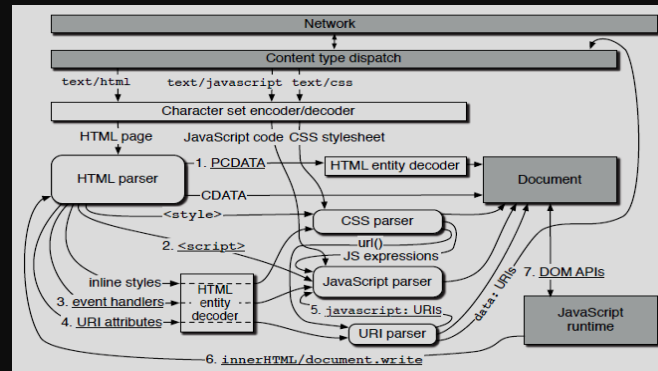
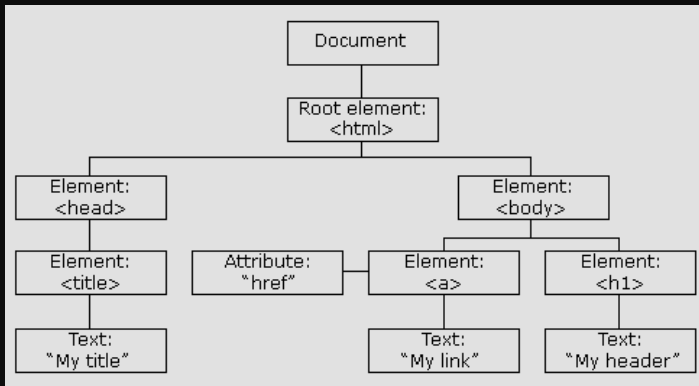


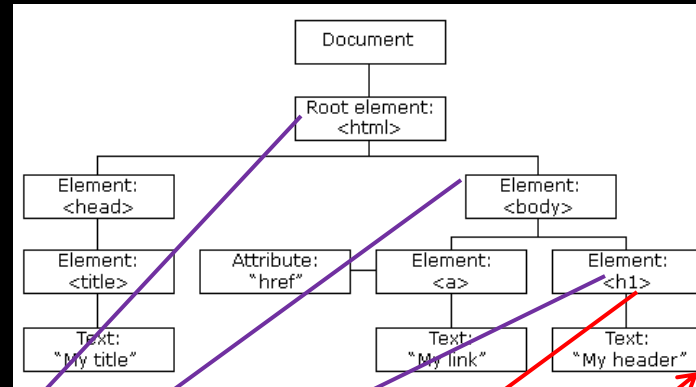


DOM XSS: ENCOUNTERS OF THE 3RD KIND

OBJECTIVES



UNDERSTANDING DOM



Tryit Editor v1.5 - Windows Internet Explorer

http://www.w3schools.com/html/dom/tryit.asp?fi

Yahoo!

Favorites Tryit Editor v1.5

Page Safety Tools

```
<html>
<body>

<h1 id="intro">Hello World!</h1>

<script type="text/javascript">
txt=document.getElementById
("intro").innerHTML;
document.write("<p>The text from the
intro paragraph: " + txt + "</p>");
</script>

</body>
</html>
```

Hello World!

The text from the intro paragraph: Hello World!

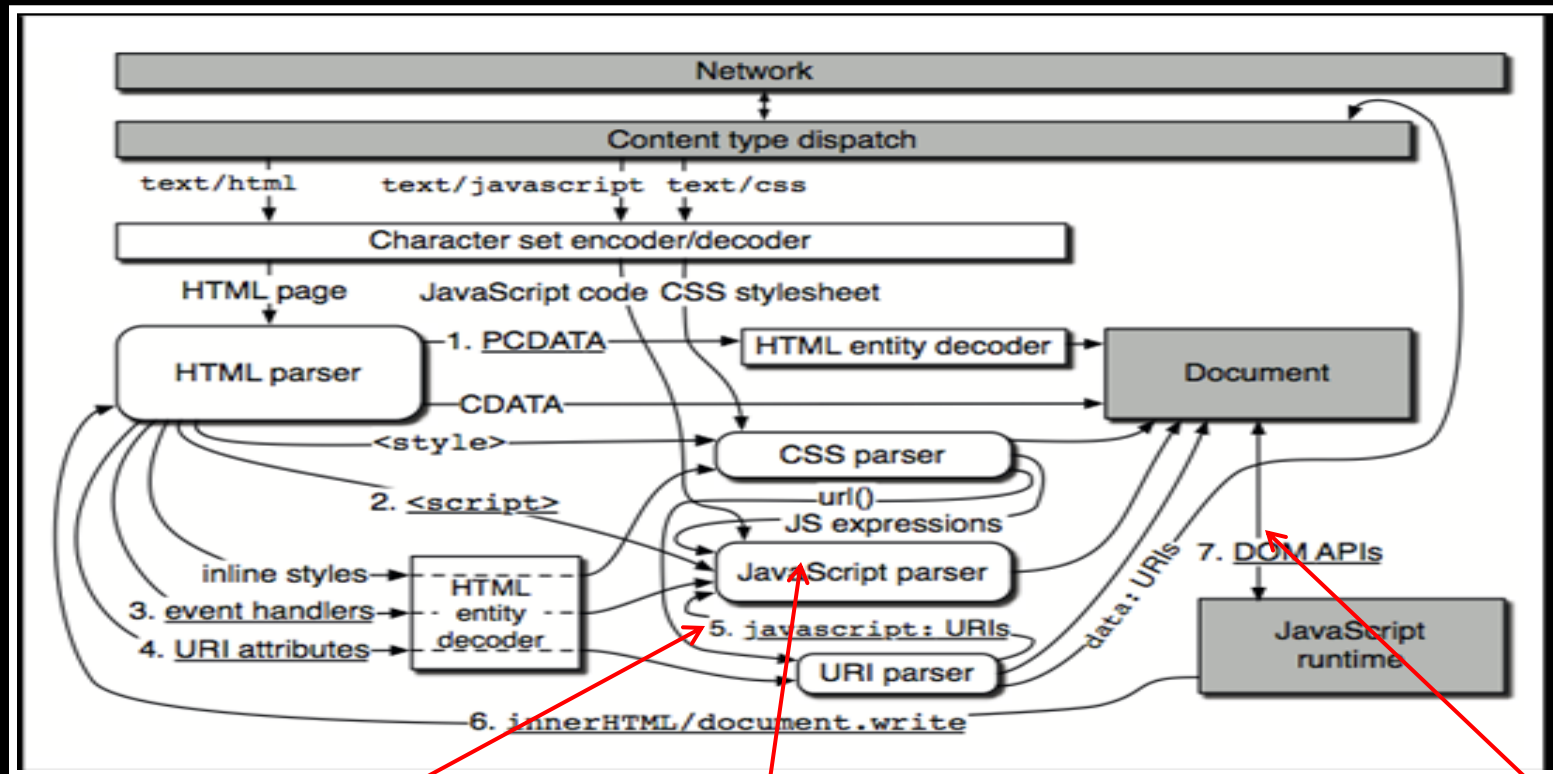
Edit the code above and click to see the result

W3Schools.com - Try it yourself

Done Internet | Protected Mode: On 100%

WARNING: DOM TEMPLATING IS EVIL

COMPLEX BROWSER CONTEXTS



JavaScript URI XSS

HTML->DOM->HTML Auto Decoding
(to be covered in Demo#7)

JavaScript Auto Decoding
(not covered. Similar to Demo#7)

WHY WORRY?



Who is safe? Those who write quality code – **DOM Construction** and **Input Sanitization**

But, could they (YUI/jQuery/Browsers) do better?

Yes, MY WISHLIST

- make it easier to do the right thing
- Warn on unsafe & abuse-able APIs
- Provide in-function sanitization capability

(Aah, context-sensitive auto-sanitization would be great, but let's not be too optimistic ATM)

Predicted to be one of the top 5 security issues for 2011

<http://jeremiahgrossman.blogspot.com/2011/02/top-ten-web-hacking-techniques-of-2011.html>

Native APIs & Frameworks do no protect. Context, performance & security after thought.

IBM found 2370 vulnerabilities on 92 sites out of 850 Fortune 500

<http://public.dhe.ibm.com/common/ssi/ecm/en/raw14252usen/RAW14252USEN.PDF>

(They released a commercial add-on to AppScan called JSA. Not available for eval yet)

Minded Security found 56 out of Alexa top 100 sites vulnerable

<http://blog.mindedsecurity.com/2011/05/dominator-project.html>

(They also released a free tool - DOMinator, we will eval that)

SAMPLE #1: DOM XSS (WITH DOMINATOR)

The screenshot illustrates a DOM XSS attack using the DOMinator extension. The browser's address bar shows a URL with a payload: `http://[redacted]/demos/unsafeDOM.html?url=<img src=0 onerror=alert(0)%3E`. The page content shows a link with text "undefined" and a "Change link" button. The DOMinator extension is active, showing a console log with the alert message and a stack trace. The Alerts panel shows the alert was triggered by the payload.

Q#1: New? No, first discovered by Amit Klein in 2005 www.webappsec.org/projects/articles/071105.shtml

Q#2: Then why now? Because code shifted client side - RIA, AJAX, Web2.0

Q#3: What are the tools?

- Do you think they solve the problem?
- Clever people solve, wise avoid. **Code Defensively**
- Anyways DOMinator and AppScan appear to do a bit but not enough

SAMPLE #1: WHAT WENT WRONG? WHAT WOULD HAVE SAVED THE DAY?

```
<html>
<head>

<script src="http://yui.yahooapis.com/3.4.0/build/yui/yui-min.js"></script>

<script type="text/javascript">
function changeLink()
{
  if (document.URL.indexOf('?urls=') != -1)
  var url=unescape(document.URL.substring(document.URL.indexOf('?urls=')+6,document.URL.length));
  else if (document.URL.indexOf('&name=') != -1)
  var url = unescape(document.URL.substring(document.URL.indexOf('&urls=')+6,document.URL.length));

  document.getElementById('myAnchor').innerHTML=url;

  if (document.URL.indexOf('?href=') != -1)
  var src= unescape(document.URL.substring(document.URL.indexOf('?href=')+6,document.URL.length));
  else if (document.URL.indexOf('&href=') != -1)
  var src = unescape(document.URL.substring(document.URL.indexOf('&href=')+6,document.URL.length));
  //document.getElementById('myAnchor').setAttribute("href", src);

  //node.set('href',src);
  YUI().use('node', function (Y) {
    var node = Y.one('#myanchor');

    node.set('text',src);

  });
  //document.getElementById('myAnchor').href="http://finance.yahoo.com";
  document.getElementById('myAnchor').target="_blank";
}
</script>
</head>
<body>

<a id="myAnchor" href="http://yahoo.com">Yahoo</a>
<input type="button" onclick="changeLink()" value="Change link">

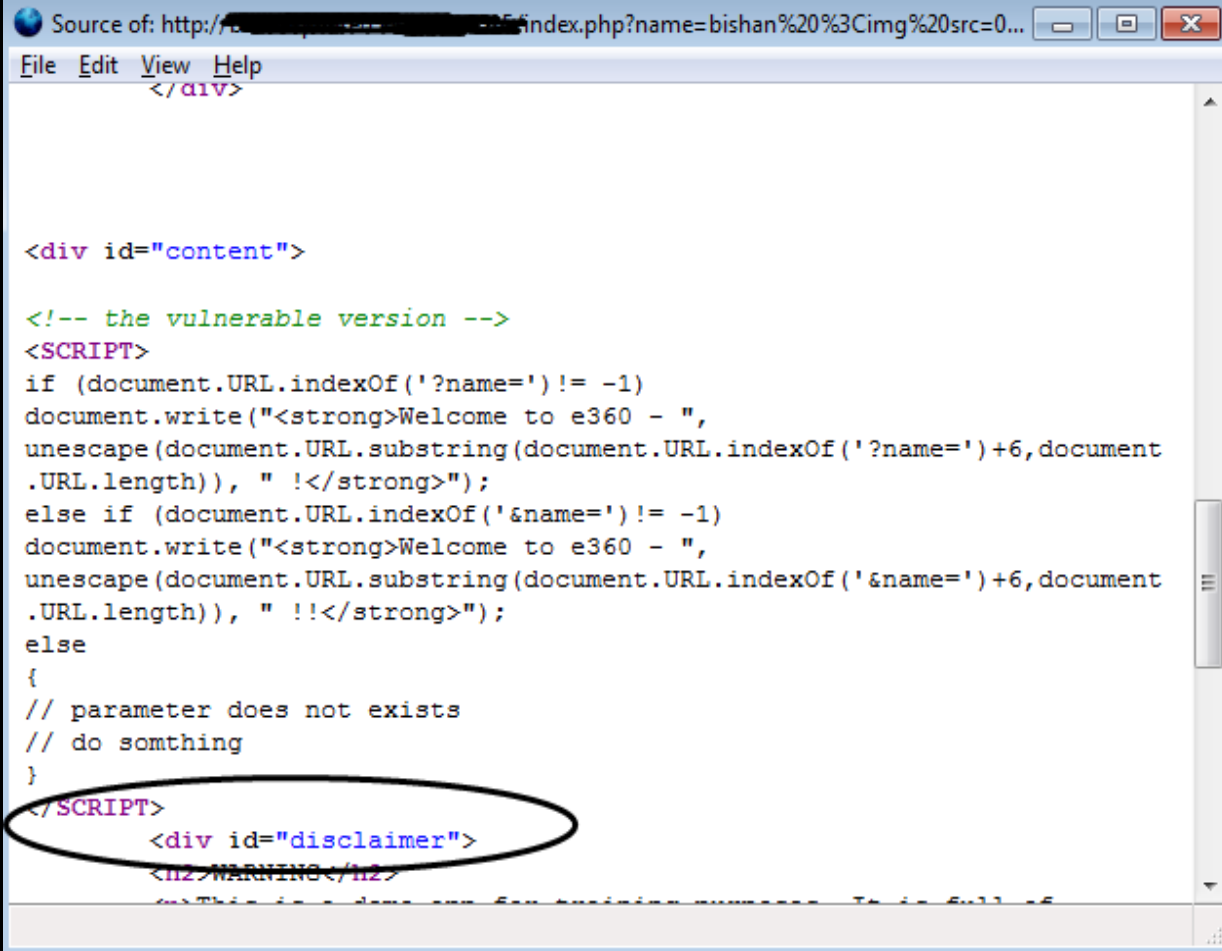
</body>
</html>
```

Taint Sources
(Direct or Indirect)

Taint Sinks
(eval, location.replace)

Defensive Coding

SAMPLE #2: NOT IN VIEW SOURCE



The screenshot shows a web browser window with the address bar displaying a URL. The source code is visible, showing HTML and JavaScript. A JavaScript snippet is highlighted with a black oval. The snippet is a comment followed by a script tag containing a conditional statement that checks for the presence of a 'name' parameter in the URL and writes a welcome message to the document. The script is enclosed in a <div id='disclaimer'> tag.

```
Source of: http://[redacted]index.php?name=bishan%20%3Cimg%20src=0...
File Edit View Help
</div>

<div id="content">

<!-- the vulnerable version -->
<SCRIPT>
if (document.URL.indexOf('?name=') != -1)
document.write("<strong>Welcome to e360 - ",
unescape(document.URL.substring(document.URL.indexOf('?name=')+6,document
.URL.length)), " !</strong>");
else if (document.URL.indexOf('&name=') != -1)
document.write("<strong>Welcome to e360 - ",
unescape(document.URL.substring(document.URL.indexOf('&name=')+6,document
.URL.length)), " !</strong>");
else
{
// parameter does not exists
// do something
}
</SCRIPT>
<div id="disclaimer">
<h2>WARNING</h2>
<div>This is a demo app for training purposes. It is full of
```

Myth#1 : we have default framework auto-sanitization at the server

- Server-side auto-sanitization like PHP Filter will not protect
- They has no way of intercepting DOM

SAMPLE #2: GENERATED SOURCE DOES



SAMPLE #2: DOMINATOR FALSE NEGATIVE

The screenshot shows a web browser window with the address bar displaying `http://[redacted]:8315/index.php?name=bishan <img src=0`. The page content includes "Welcome to [redacted] bishan !". A DOMinator alert box is visible, titled "The page at [redacted] says:", containing a yellow warning icon and the number "9". The alert box has an "OK" button. Below the alert, the browser's console is open, showing two log entries. The first entry is at 13:53:40.204 and the second is at 13:53:40.206. Both entries show the URL `http://[redacted]:8315/index.php?name=bishan%20%3Cimg%20src=0%20onerror=alert(9)%3E` and the target `[indexOf(?name=) CallCount: 1]`. The console also shows "Data:" and "Stack Trace" for each entry. On the right side of the browser, there is a red box highlighting the "Alerts (0)" and "Warning (0)" sections of the DOMinator interface.

Welcome to [redacted] bishan !

The page at [redacted] says:

9

OK

Log Enabled Clear Log Cookies Disabled Stack

1 13:53:40.204 :[http://[redacted]:8315/index.php?name=bishan%20%3Cimg%20src=0%20onerror=alert(9)%3E]
Target: [indexOf(?name=) CallCount: 1]
Data:
+ http://[redacted]:8315/index.php?name=bishan%20%3Cimg%20src=0%20onerror=alert(9)%3E
+ Stack Trace

2 13:53:40.206 :[http://[redacted]:8315/index.php?name=bishan%20%3Cimg%20src=0%20onerror=alert(9)%3E]
Target: [indexOf(?name=) CallCount: 1]
Data:
+ http://[redacted]:8315/index.php?name=bishan%20%3Cimg%20src=0%20onerror=alert(9)%3E
+ Stack Trace

Alerts (0) Warning (0)

SAMPLE #3: YUI / JQUERY ISN'T BAD. DOM TEMPLATING IS!

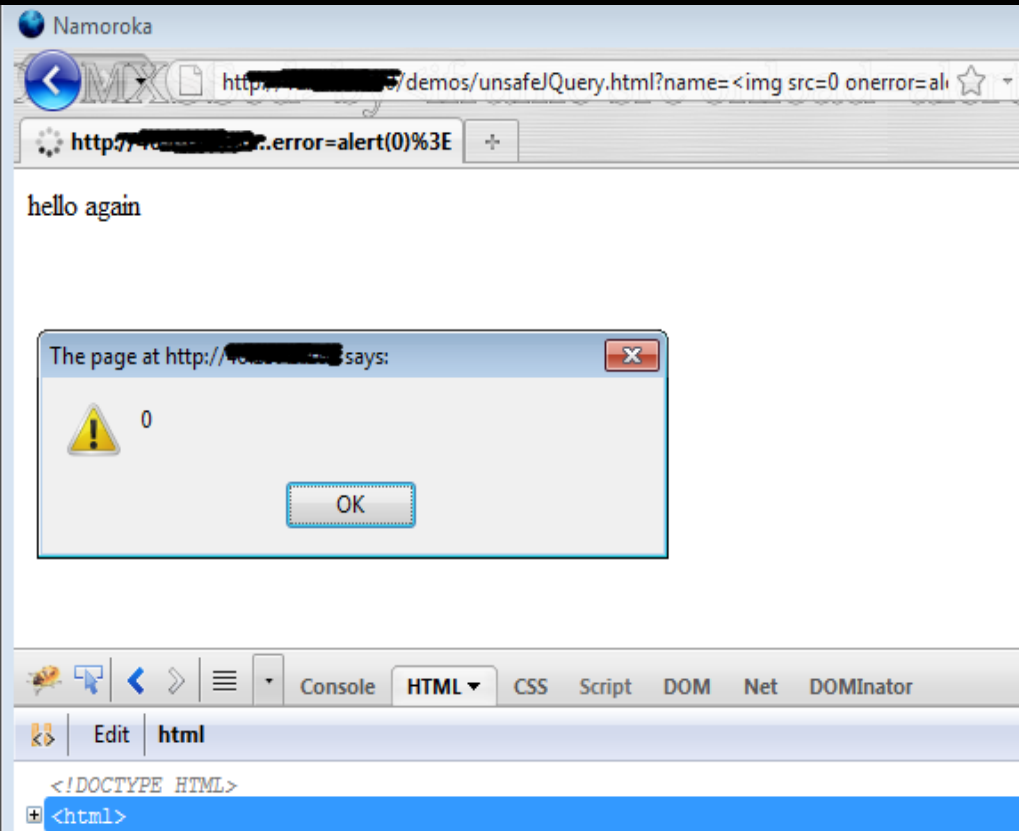
```
Source of: http://[redacted]/demos/unsafejQuery.htm...
File Edit View Help

<!DOCTYPE html>
<html>
<head>
  <style>

    .red { color:red; }
  </style>
  <script src="http://code.jquery.com/jquery-
latest.js"></script>
</head>
<body>

  <div></div>

<script>
if (document.URL.indexOf('?name=') != -1) {
  var name =
unescape(document.URL.substring(document.URL.inde
xOf('?name=')+6,document.URL.length));
  var str ="again";
  $("div").html("hello "+str+" "+name);
}
```



SAMPLE #3: YUI / JQUERY ISN'T BAD. DOM TEMPLATING IS!

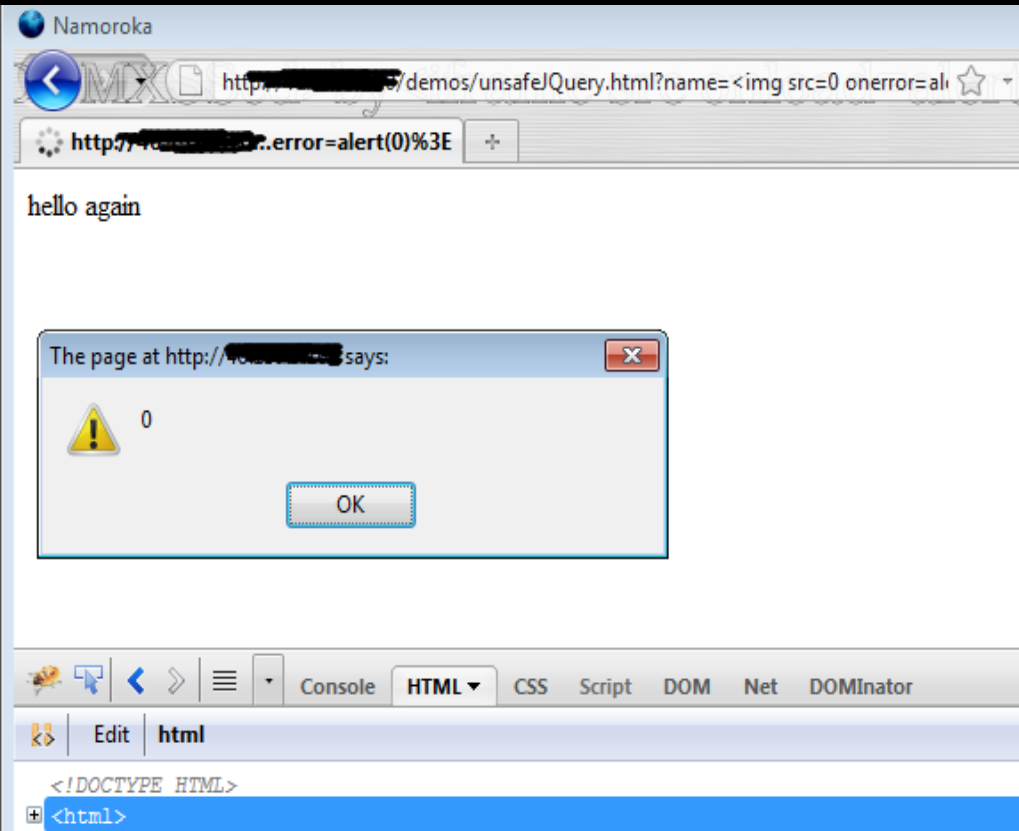
```
Source of: http://[redacted]/demos/unsafejQuery.htm...
File Edit View Help

<!DOCTYPE html>
<html>
<head>
  <style>

    .red { color:red; }
  </style>
  <script src="http://code.jquery.com/jquery-
latest.js"></script>
</head>
<body>

  <div></div>

<script>
if (document.URL.indexOf('?name=') != -1) {
  var name =
unescape(document.URL.substring(document.URL.inde
xOf('?name=')+6,document.URL.length));
  var str ="again";
  $("div").html("hello "+str+" "+name);
}
```

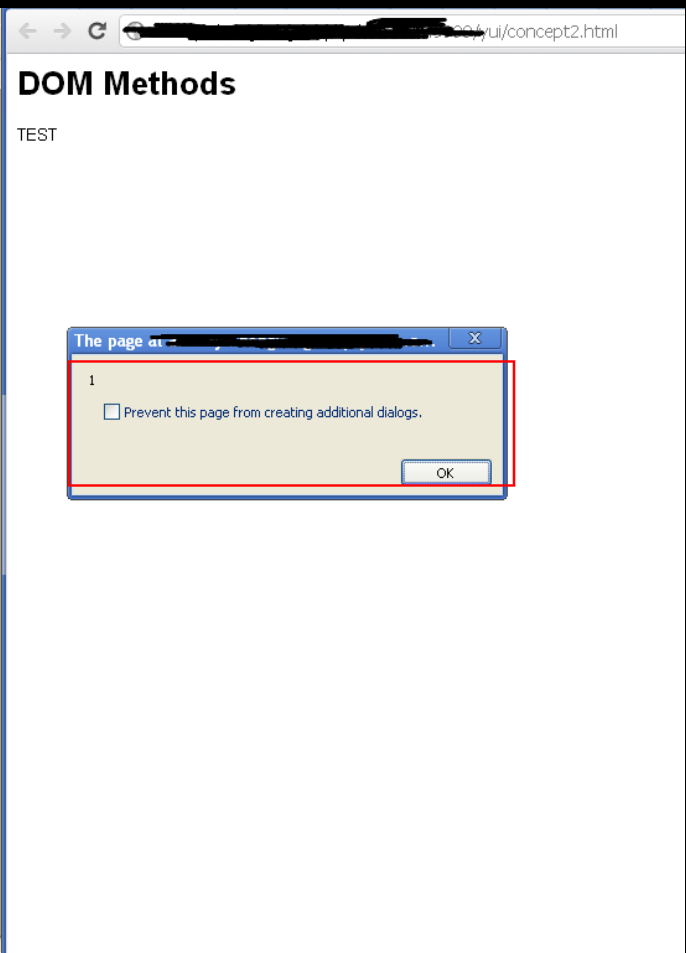


SAMPLE #4: YUI / JQUERY ISN'T BAD.

DOM TEMPLATING IS!

(DOMINATOR DIDN'T CATCH THIS ONE TOO)

```
11 <body class="yui3-skin-sam yui-skin-sam">
12
13 <h1>DOM Methods</h1>
14
15 <div id="demo">
16 TEST
17 </div>
18
19 <script type="text/javascript">
20 YUI({ filter: 'raw' }).use("node", function(Y) {
21 // YUI 3.3
22 var node = Y.Node.create('<img src=x onerror="alert(1)">');
23
24 var n3 = document.createElement('img');
25 n3.async = false;
26 n3.setAttribute("src", "x");
27 n3.setAttribute("onerror", "alert(3)");
28 // var n2 = Y.one('#demo');
29 // n2.setContent ( '<img src=x onerror="javascript:alert(2)">' );
30
31
32 // var node = Y.one('#demo').appendChild(document.createTextNode(''));
33 // var payload = '<b><script> alert(1); </script></b>';
34 // node.set('text', payload );
35
36 // YUI 3.5
37 // http://yuilibrary.com/projects/yui3/ticket/2529955
38 // completely eliminated document.*
39 // var node = Y.one('#demo').appendChild(Y.Node.create('')) ;
40
41 // Vulnerable example
42 // Y.one('#demo').set('innerHTML', '<script>alert(1);</script>' );
43 // node.set('innerHTML', '<div>test this </div>');
44 // Y.log(Y.one('#demo').getContent() );
45
46 });
```



SAMPLE #5: YOU DON'T NECESSARILY NEED FILTERING. YUI / NATIVE JS API (INNERTEXT) / OTHERS LET YOU PLAY SAFE. THIS IS CALLED DOM CONSTRUCTION

The screenshot displays a web browser window and the DOMinator extension interface. The browser's address bar shows the URL `http://46.137.9.100/demos/unsafeDOM.html?href=<img src=0 onerror=alert(0)%3E`. The DOMinator tool's console shows two log entries:

```
1 15:15:17.703 :[ http://46.137.9.100/demos/unsafeDOM.html?href=%3C/a%3E%3Cimg%20src=0%20onerror=alert(0)%3E ]
Target: [ indexOf(?urls=) CallCount: 1 ]
Data:
+ http://46.137.9.100/demos/unsafeDOM.html?href=%3C/a%3E%3Cimg%20src=0%20onerror=alert(0)%3E
+ Stack Trace

2 15:15:17.711 :[ http://46.137.9.100/demos/unsafeDOM.html?href=%3C/a%3E%3Cimg%20src=0%20onerror=alert(0)%3E ]
Target: [ indexOf(&name=) CallCount: 1 ]
Done
```

The DOMinator tool's DOM view shows the following HTML structure:

```
<a><img src=0 onerror=alert(0)> Change link
```

The source code of the page is visible in the left pane, showing the following JavaScript code:

```
Source of: http://46.137.9.100/demos/unsafeDOM.html?...
File Edit View Help
xOf('&href=')+6,document.URL.length));
//document.getElementById('myAnchor').setAttribute("href",src);

//node.set('href',src);
YUI().use('node', function (Y) {
var node = Y.one('#myanchor');

node.set('text',src);

});
//document.getElementById('myAnchor').href="http://finance.yahoo.com";
document.getElementById('myAnchor').target="_blank";
}
</script>
</head>
<body>

<a id="myAnchor"
href="http://yahoo.com">Yahoo</a>
<input type="button" onclick="changeLink()"
```

SAMPLE #5: BEWARE OF CONTEXTS.

AGAIN YUI / NATIVE JS API / OTHERS ARE NOT BAD.

NO FILTERING / CONTEXT INSENSITIVE FILTERING IS!

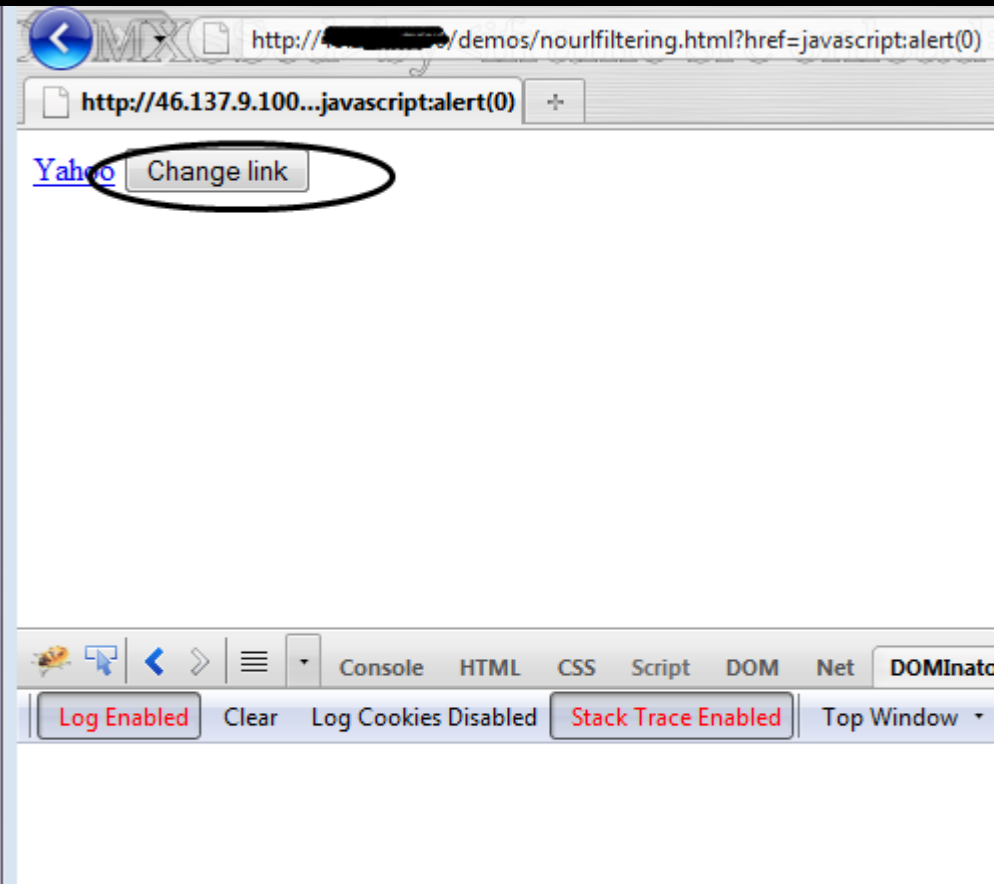
```
Source of: http://[redacted]/demos/nourlfiltering.html...
File Edit View Help
var src =
unescape(document.URL.substring(document.URL.indexOf('&href=')+6,document.URL.length));
//document.getElementById('myAnchor').setAttribute("href", src);

//node.set('href',src);
YUI().use('node', function (Y) {
var node = Y.one('#myanchor');

node.set('href',src);

});
//document.getElementById('myAnchor').href="http://finance.yahoo.com";
document.getElementById('myAnchor').target="_blank";
}
</script>
</head>
<body>

<a id="myAnchor"
```



SAMPLE #6: BEWARE OF CONTEXTS.

AGAIN YUI / NATIVE JS API / OTHERS ARE NOT BAD.

NO FILTERING / CONTEXT INSENSITIVE FILTERING IS!

The screenshot shows a web browser window with the address bar displaying `http://[redacted]/demos/nourlfiltering.html?href=javascript:alert(0)`. The browser's address bar also shows `http://[redacted].javascript:alert(0)`. The browser's search bar contains the text "Yahoo" and a "Change link" button.

The browser's developer tools are open, showing the "DOMinator" tab. The "Log Enabled" button is highlighted. The "Stack Trace Enabled" button is also highlighted. The "Top Window" dropdown is set to "Top Window".

The log shows two entries:

- 5 14:32:29.642 :[http://[redacted]/demos/nourlfiltering.html?href=javascript:alert(0)]
Target: [indexOf(https) CallCount: 1]
Data:
+ http://[redacted]/demos/nourlfiltering.html?href=javascript:alert(0)
+ Stack Trace
- 6 14:32:29.785 :[http://[redacted]/demos/nourlfiltering.html?href=javascript:alert(0)]
Target: [A.href CallCount: 1]
Data:
+ javascript:alert(0)
+ Stack Trace

The "Alerts (1)" tab is active, showing the alert message: `javascript:alert(0)`.

SAMPLE #7: BEWARE OF AUTO-DECODING. AGAIN YUI / NATIVE JS API / OTHERS ARE NOT BAD. INSECURE CODING / INSUFFICIENT FILTERING IS! (ANOTHER THING *DOMINATOR* DIDN'T CATCH)

Source of [redacted] ...

File Edit View Help

```
<html>

<title>&lt;img src=0 onerror=alert(0)
&gt;</title>

<body>

    <div id='demo'> Hi</div>
    <script>
        function changeLink(){
            document.getElementById('demo').innerHTML =
            document.title ;
        }
    </script>

    <input type="button"
    onclick="changeLink()" value="Change Hi">

</body>

</html>
```

Change Hi

0

OK

Myth#2 : I encoded server-side right?

- Exception. When DOM and HTML are mixed they tend to explode
- HTML->DOM->HTML means switching of context and browser auto decoding

THANKS FOLKS...



bish@route13.in
twitter:b1shan



yukinying@gmail.com
twitter: yukinying