Server Security And Hack Detection

Amandeep Singh Sidhu
Post Graduate Student(M.E.)
Department of Computer Science & Engineering
Thapar University
Patiala, Punjab

Dr. Neeraj Kumar
Assistant Professor
Department of Computer Science & Engineering
Thapar Universirty
Patiala, Punjab

Abstract

Server machine can runs different configurations as per user requirement by default system is vulnerable to many threats. In this paper, We will learn how to enhance the security and improve control of Server for a particular website platform as well as harden the security of the website.

Server has benefits over shared hosting, as its gives complete control over the system. We get dedicated resources which is not available in shared. As most of the websites run on open source platform such as Wordpress using PHP. This make it vulnureable to attacks as code is known to the hacker. To counter that We will learn how simple script can check website from hack and protect our code. How important it is to have different privileges for different user groups.

1. Introduction

The Virtual Private Server alternative is often chosen by small businesses that need a customized Web site but cannot afford a dedicated server. Another useful aspect of VPS technology is the ability of a single subscriber to maintain multiple websites. By default these system are set up with default configurations which are vulnerable to attacks. In this project we are going to increase the security of the system by modifying the default configuration [1]. The concept of network security situation awareness refers to the operational picture that consolidates all available information to identify attacks and select and apply appropriate countermeasures [2].

Redmine is a web based project management application. We will install Redmine with only packages that are required to run Redmine fully and strip out the extra service from VPS [3]. By using tools like putty, portable keepass and portable Firefox to maintain secure access to VPS.

We will make system fully compatible and optimise for Wordpress. Install multiple copy of Wordpress for different client and maintain logs. With basic steps to maintain security and safe guard files. Check site for hack and improve access methods by using Secure Sockets Layer (SSL) [4].

2. Requirement

We will discuss all the things that we will to run a website and make it secure.

2.1 Virtual Private Server(VPS)

A virtual private server (VPS) is a virtual machine that runs different configurations+ as per user requirement. It appears to the user as a dedicated server but is actually installed on a computer serving multiple VPS or Web sites.

Virtualization extends this basic concept to the computer as a whole. In the traditional model, the operating system shares access to the resources, but there is still a single machine being shared. In the virtual server model, the virtualization software instead provides the illusion of more than one computer, hard drive, printer, etc. Although the resources are still shared, as under the time-sharing model, virtualization provides a higher level of security as the individual virtual servers are isolated from each other. Each virtual server can run own full-fledged operating system and can be independently rebooted. This is valuable as it allowed businesses to run their legacy applications on older versions of an operating system on the same server as newer applications [4].

Virtual private servers provide you with your own virtual server environment for your application complete with root access, your own database instance, and your own webserver. These plans support all of the same great features as our developer plans including Rails hosting using Passenger, FastCGI, Mongrel, or Lighttpd. VPS servers are powered by OpenVZ technology and running on premium dell serves with Dual Quad Core processors with a RAID 10 SAS 15k

hard drive configuration connected over a 100mbps connection to our GigE multiple provider internet connection. It also includes nightly file system backups for no additional charge [6].

With Rails VPS Cent OS 6 or Ubuntu 12.04 LTS Virtualmin Control Panel image, everything need to run a Ruby on Rails or a PHP application is preinstalled. Virtualmin also provides you with a web based control panel for managing your domains, databases, ftp accounts, email accounts and more, all for no additional cost [6].

Plan Specifications:

Storage Space	10 GB
Bandwidth	100 GB/Month
Guaranteed RAM	384MB
vSwap RAM	384 MB

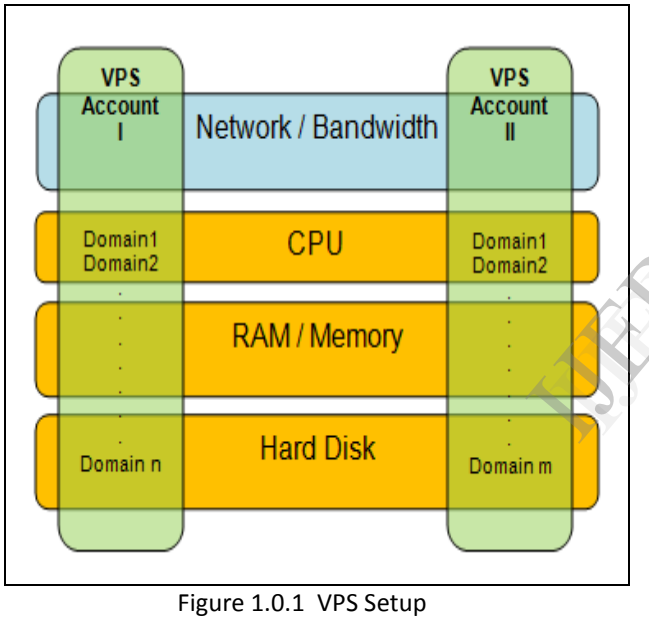

Figure 1.0.1 VPS Setup

2.2 PHP Parser

Rails are a web application development framework written in the Ruby language. It is designed to make programming web applications easier by making assumptions about what every developer needs to get started. It allows you to write less code while accomplishing more than many other languages and frameworks [7].

Rails are opinionated software. It makes the assumption that there is a "best" way to do things, and it's designed to encourage that way – and in some cases to discourage alternatives. If you learn "The Rails Way" you'll probably discover a tremendous increase in productivity [7].

2.3 Wordpress

WordPress started in 2003 with a single bit of code to enhance the typography of everyday writing and with fewer users than you can count on your fingers and toes. Since then it has grown to be the largest self-hosted blogging tool in the world, used on millions of sites and seen by tens of millions of people every day.

Everything you see here, from the documentation to the code itself, was created by and for the community. WordPress is an Open Source project, which means there are hundreds of people all over the world working on it. (More than most commercial platforms.) It also means you are free to use it for anything from your cat's home page to a Fortune 500 web site without paying anyone a license fee and a number of other important freedoms.

On this site you can download and install a software script called WordPress. To do this you need a web host who meets the minimum requirements and a little time. WordPress is completely customizable and can be used for almost anything. There is also a service called WordPress.com which lets you get started with a new and free WordPress-based blog in seconds, but varies in several ways and is less flexible than the WordPress you download and install yourself.

WordPress started as just a blogging system, but has evolved to be used as full content management system and so much more through the thousands of plugins, widgets, and themes, WordPress is limited only by your imagination. (And tech chops.)

3 Tools

We will discuss tools that are required to operate.

3.1 Keepass

Keepass is a free open source password manager, which helps you to manage your passwords in a secure way. You can put all your passwords in one database, which is locked with one master key or a key file. So you only have to remember one single master password or select the key file to unlock the whole database. The databases are encrypted using the best and most secure encryption algorithms currently known (AES and Twofish) [9].

3.2 Putty

PuTTY is a free and open source terminal emulator application which can act as a client for the SSH, Telnet, rlogin, and raw TCP computing protocols and as a serial console client. PuTTY was originally written for Microsoft Windows, but it has been ported to

various other operating systems. Official ports are available for some Unix-like platforms, with work-in-progress ports to Classic Mac OS and Mac OS X, and unofficial ports have been contributed to platforms such as Symbian and Windows Mobile [10].

Some features of Putty are:

• The storing of hosts and preferences for later use.
• Control over the SSH encryption key and protocol version.
• Command-line SCP and SFTP clients, called "pscp" and "psftp" respectively.
• Control over port forwarding with SSH (local, remote or dynamic port forwarding), including built-in handling of X11 forwarding.
• Emulates most xterm, VT102 control sequences, as well as much of ECMA-48 terminal emulation.
• Supports 3DES, AES, Arcfour, Blowfish, DES.
• Public-key authentication support (no certificate support).
• Support for local serial port connections.
• Self-contained executable requires no installation.

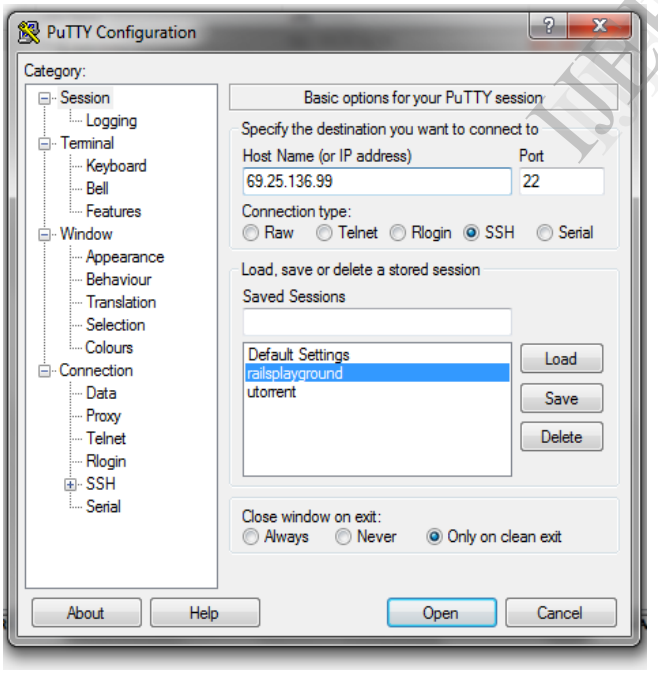

Figure 3.1 Putty

4 Implementation Seting

4.1 VPS

Consistent, continual IT security hardening is your enterprise's most valuable security control. It minimizes network vulnerabilities, reduces the attack surface, and helps your organization avoid becoming a victim of zero-day exploits. Yet most security solutions simply try to limit outside access to the system where your sensitive data resides. This perimeter-centric approach to security leaves your infrastructure vulnerable to attack and compromise [11].

With this VPS we have setup 3 virtual host entries:

1 http://redmine.aspiresoftnet.com/
2 http://69.25.136.99/

Host File for apache

<VirtualHost *:80>
ServerAdmin webmaster@dummy-host.example.com
DocumentRoot /var/www/main/html
ServerName 69.25.136.99
ErrorLog logs/dummy-host.example.com-error_log
CustomLog logs/dummy-host.example.com-access_log common
</VirtualHost>
<VirtualHost *:80>
ServerAdmin webmaster@dummy-host.example.com
DocumentRoot /var/www/main/html
DocumentRoot /var/www/redmine.aspiresoftnet.com/redmine/public
ServerName redmine.aspiresoftnet.com
ErrorLog logs//redmine.aspiresoftnet.com-error_log
CustomLog logs//redmine.aspiresoftnet.com-access_log common
</VirtualHost>

Redmine.aspiresoftnet.com

This is a demo as well test site to test any new changes in redmine or its plugins.

http://69.25.136.99/

It is a blank page to check that VPS is running or not. It's not linked to any domain.

Figure 4.1 Server Security

4.2 User Rights

4.2.1 Restricting Root

Create a separate user to login to VPS via putty i.e admin with less permission on the system, use root only to create new user or to do critical updates [3].

4.2.2 Password Policies

Strong passwords should be used. A strong password should have mixed case, special characters, numbers, and be longer than 8 characters. Password complexity requirements should be in place to enforce strong password usage. Passwords should be changed reasonably regularly. Some folks argue the value of changing passwords, however the longer you have a password, the longer someone has to break it. Conversely, if you're frequently changing passwords, your users will tend to use weaker passwords in order to remember them. You should to find a happy medium that suits your organization. Passwords shouldn't be changed more than once a day.

4.2.3 User Groups

Create different user group permission for each type of user who is accessing the VPS via web browser or via putty check their permission, level.

4.2.4 Detecting Listening Network Ports

One of the most important tasks is to detect and close network ports that are not needed. To get a list of listening network ports (TCP and UDP sockets), you can run the following command:

 # netstat -tulp

Close the ports that are not require to listen, for example Send mail should not listen for incoming network connections unless the server is a mail or relay server [12].

4.2.5 Disable Wi-Fi and usb

We should disable Wi-Fi and USB ports on the system to increase security.

4.3 System Security

4.3.1 Disabling Runlevel System Services

One of the most important tasks is to remove any network services from the system start up process that are not needed. Use following command to check:-

chkconfig --list |grep on

4.3.2 Removing Unnecessary Software Package

A very important step in securing a Linux system is to determine the primary function or role of the Linux server. You should have a detailed knowledge of what is on your system. Otherwise you will have a difficult time to understand what needs to be secured and hence securing your Linux systems proactively won't be that effective. Therefore, it is very critical to look at the default list of software packages and remove unneeded packages or packages that don't comply with your security policy. If you do that you will have fewer packages to update / maintain when patches are released [12].

```
[root@redmine ~]# netstat -lptu
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State       PID/Program name
tcp        0      0 *:dnp                       *:*                         LISTEN      1111/perl
tcp        0      0 *:imaps                     *:*                         LISTEN      653/dovecot
tcp        0      0 *:pop3s                     *:*                         LISTEN      653/dovecot
tcp        0      0 *:mysql                     *:*                         LISTEN      631/mysqld
tcp        0      0 *:submission                *:*                         LISTEN      1035/master
tcp        0      0 *:pop3                      *:*                         LISTEN      653/dovecot
tcp        0      0 localhost:39087             *:*                         LISTEN      14953/Rack
tcp        0      0 *:imap                      *:*                         LISTEN      653/dovecot
tcp        0      0 *:http                      *:*                         LISTEN      12834/httpd
tcp        0      0 *:ndmp                      *:*                         LISTEN      1118/perl
tcp        0      0 localhost:36790             *:*                         LISTEN      5062/Rack
tcp        0      0 *:ssh                       *:*                         LISTEN      474/sshd
tcp        0      0 *:smtp                      *:*                         LISTEN      1035/master
tcp        0      0 *:https                     *:*                         LISTEN      12834/httpd
tcp        0      0 *:imaps                     *:*                         LISTEN      653/dovecot
tcp        0      0 *:pop3s                     *:*                         LISTEN      653/dovecot
tcp        0      0 *:submission                *:*                         LISTEN      1035/master
tcp        0      0 *:pop3                      *:*                         LISTEN      653/dovecot
tcp        0      0 *:imap                      *:*                         LISTEN      653/dovecot
tcp        0      0 *:ftp                       *:*                         LISTEN      1044/proftpd
tcp        0      0 *:ssh                       *:*                         LISTEN      474/sshd
tcp        0      0 *:smtp                      *:*                         LISTEN      1035/master
udp        0      0 *:783                       *:*                                     446/portreserve
udp        0      0 *:ndmp                      *:*                                     1118/perl
udp        0      0 *:dnp                       *:*                                     1111/perl
[root@redmine ~]#
```

Figure 4.2 Service List

4.3.2 Logs

Maintain log for critical system have a close look at them, you can compare if there is any unusual behaviour of any user who is accessing the website. Check server log for denied permission.

4.3.3 System Update

Do not update whole of the system together set a cron job or manually update each package separately. As updating whole system together consume lots of resources. Avoid using following command:

yum update

4.3.4 Monitor

Monitor VPS control panel for excessive use of RAM, disk space or processing power this can help you to see if there is any suspicious activity.

4.4 Database Security

4.4.1 Secure Installation

Make secure installation for mysql root should have complex password, so that it's not easy to crack.

4.4.2 Different user and database

Create unique database for each website and assign different user with privileges to work on only one database to enhance security of the database.

4.4.3 Run As Localhost

If we do not need, make database available only via localhost. This means we should not allow database to be accessed by any other server by broadcasting database server.

4.4 Wordpress

4.1 Files

Make all the files outside of the webroot folder, to avoid direct access. Except public_html (web root) of redmine files which are require for theme and other functions.

4.2 File Permission

Only image and cache folder should have right folder/file permission by web user to protect over writing of any application/code file with malicious script.

4.5 Login

4.5.1 Complex Password

Admin/User should use complex password so that it is hard to crack.

4.5.2 SSL Access

Create a secure encryption communication between Webserver and client, to increase security from "Man in Middle" attacks.

4.6 Security Tools

4.6.1 Keepass Portable

I have used portable keepass to maintain my entire password as some passwords are 140-bit. You can keep simple yet complex password for your keepass. It file is saved in encrypted for so it's not easy to crack it. In addition to that you can store it on your pen drive and use it when require avoid it getting stolen from laptop [9].

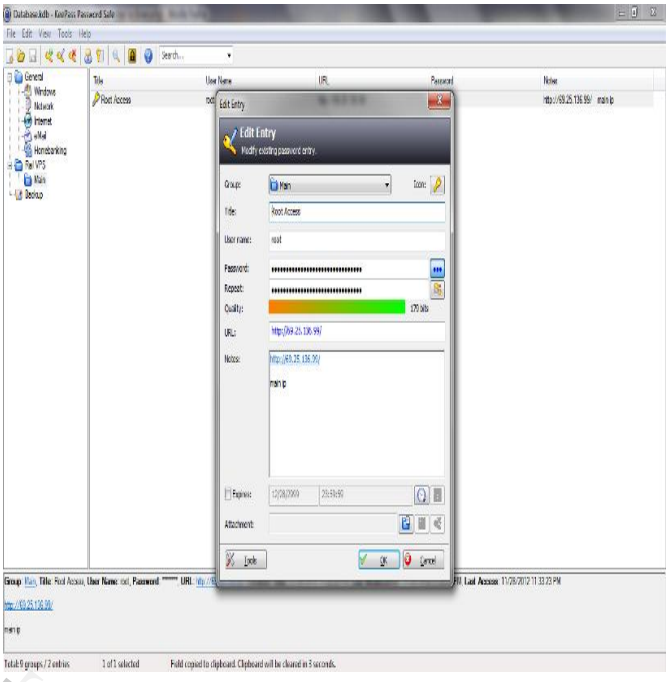

Figure 4.3 Keepass

4.6.2 Putty Portable

Putty give me shell access to my VPS as there is no graphical user interface to access my VPS except the control panel where we can only monitor resources or reboot/shutdown/restart system [10].

4.5 Hack Checks

Most of the hacker used decoded form of code when they put malicious code, so that it's easy to detect. We have setup a script and run it as a cron job.

Step 1:-

scanner.sh

cd ../stoneig.com

find . -type f | xargs grep base64_decode > ../scan/boo

cd ../scan

mail -s "Site Scan Results scanner.sh" amansidhu07@gmail.com < master_boo

Step 2:-

check_code.sh

cd ../stoneig.com

find . -type f | xargs grep base64_decode > ../scan/boo

cd ../scan

mail -s "Site Scan Results scanner_bak01.sh" aman.sidhu@stoneig.com < `diff master_boo boo`

rm boo

In step one, we make a default master_boo file that check from “decode” keyword in the code. As system might be using it to, so that it’s not mixed with hacker’s code. We only run this script once to create a master boo file.

In step two, we are searching again saving data in boo file then emailing the difference in both the files to email. We should set this script to run every 6 hours. Its output should be blank if there is not code.

REFRENCES

[1] Z. Du, Y. Chen, and X. Wang Z. Cheng, "SOAVM: A Service-Oriented Virtualization Management System with Automated Configuration," *IEEE Int’l Workshop on Service-Oriented System Engineering*, pp. 251-256, 2008.

[2] Roland Bueschkes, Ali Fessi Richard Kemmerer, "Outcome working group---situation assessment," in *Network Attack Detection and Defense, 2008*.

[3] National Security Agency. [Online]. HYPERLINK "http://www.nsa.gov" http://www.nsa.gov

[4] L. He, Q. Wang, and R. Willenborg C. Sun, "Simplifying Service Deployment with Virtual Appliances," *IEEE Int’l Conf on Services Computing*, pp. 265-272, 2008.

[5] Ajith Prasad Edassery. Dollar Shower. [Online]. HYPERLINK "http://www.dollarshower.com/what-is-vps-hosting-and-when-to-move-to-vps/" http://www.dollarshower.com/what-is-vps-hosting-and-when-to-move-to-vps/

[6] Rails Playground VPS. [Online]. HYPERLINK "http://railsplayground.com/plans-products/vps/" http://railsplayground.com/plans-products/vps/

[7] Ruby On Rails. [Online]. HYPERLINK "http://guides.rubyonrails.org/getting_started.html" http://guides.rubyonrails.org/getting_started.html

[8] Ruby Gems. [Online]. HYPERLINK "http://guides.rubygems.org/what-is-a-gem/" http://guides.rubygems.org/what-is-a-gem/

[9] KeePass Password Safe. [Online]. HYPERLINK "http://keepass.info/" http://keepass.info/

[10] Putty User Manual. [Online]. HYPERLINK "http://the.earth.li/~sgtatham/putty/0.62/htmldoc/" http://the.earth.li/~sgtatham/putty/0.62/htmldoc/

[11] Sotiris Ioannidis. Design and Implementation of Virtual Private Services.

[12] Trip Wire. [Online]. HYPERLINK "http://www.tripwire.com/data-security/security-hardening/" http://www.tripwire.com/data-security/security-hardening/

[13] M. Gaggero, and S. Manca P. Anedda, "A general service oriented approach for managing virtual machines allocation," *ACM Symposium on Applied Computing*, pp. 2154-2161, 2009.

[14] DMTF, "Open Virtualization Format," DMTF, White Paper DSP 2017 v1.0.0, 2007.

[15] C. Sapuntzakis et al., "Virtual Appliance for deploying and Maintaining Software," in *The 17th USENIX Conf on System Administration*, 2003, pp. 181-194.

[16] VMware. VMware OVF Tool. [Online]. HYPERLINK "http://communities.vmware.com/community/developer/ovf" http://communities.vmware.com/community/developer/ovf

[17] V. Bourassa, and E. Selberg A. Berman, "Process-Specific File Protection for the UNIX Operating System," in *Proceedings of the USENIX 1995 Technical Conference*, New Orleans, Louisiana, January, 1995.