NGONGANG GUY MOLLET

CLOUD COMPUTING SECURITY

Helsinki Metropolia University of Applied Sciences
Bachelor Of Engineering Degree

Information Technology

Thesis

April 11, 2011

Metropolia

Abstract

Author(s) Title Number of Pages Date	NGONGANG GUY MOLLET Cloud computing Security 34 pages + 2 appendices April 11, 2011
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Network security
Instructor	Erik Pätynen, Senior lecturer

This project aimed to show how possible it is to use a network intrusion detection system in the cloud. The security in the cloud is a concern nowadays and security professionals are still finding means to make cloud computing more secure.

First of all the installation of the ESX4.0, vCenter Server and vCenter lab manager in server hardware was successful in building the platform. This allowed the creation and deployment of many virtual servers. Those servers have operating systems and applications, which are permanently used and stored valuable information, which attracts malware and intruders. The intruders are permanently looking for vulnerabilities found in the applications and networked system to steal and destroy sensitive data.

In order to mitigate those threats, an open-source network intrusion detection system was installed: Snort. The snort sensor was then configured in order to monitor the network activity. It sends an alert when it founds malicious traffic with the same pattern as those stored in its signature database. The system was built to prevent malware or intruders invasion, which wants to use the security weaknesses found in the applications and the operating system. In the cloud, the vulnerabilities are associated with the service being followed. Snort was able to detect the malicious traffic and stored it in the log file.

This study can be implemented in any virtual data center to mitigate the possible risk. By using a network intrusion detection system, one can track down individual hacker after the investigation by watching the attacks that occur and the vulnerabilities that need to be addressed.

Keywords	Cloud computing, Security, Malware, Alert, Compliance

CONTENTS

Appendices

1 Introduction

Service providers build their own data centers to serve their customers. A data centre is a place where companies keep their physical hardware. It is usually composed of many servers, routers and switches interconnected together. Usually data center requires a lot of space, consuming resources and energy. Redundant links exist between those allowing the companies to use their services via the Internet or private IP-based networks at any time and anywhere. Thus customers do not need to know where their data are being kept. However adding more and more equipment to their rack system saturates the data centre and consumes energy. Looking ahead there will be an increase in the management costs and resource utilization in the near future. To overcome those limitations, the virtualization technology was introduced.

Now many virtual servers can be run in the same hardware while using small resources. Applications and software are developed and deployed through the web interface in this virtual data center. Cloud computing is the ability to use applications and software on the Internet which stores and protect the data while providing a service [1]. It improves energy efficiency and lows management costs. In this final year project, I aim to build a cloud computing network and provide a security service through the platform. I chose this topic because today the traditional system is being replaced with the cloud. It has introduced some major changes in the way enterprises are handling their information system and assets. A free virtual server and the applications, which are built inside, are allocated to a company, which will permanently use the services.

Customers will be billed only when the virtual resources dedicated to them are being used. The network is running and needs to be maintained in a case of an incident to satisfy the availability of the services. One can think at this step, installing a network intrusion detection system will be a good idea. The NIDS (network intrusion detection system) will monitor the activities of the virtual servers and all the applications, which are built in those appliances. It then generates an alert every time there will be a malicious activity, which matches up with its signatures´ database.

Snort is an open-source intrusion detection system developed by Sourcefire [2]. The Snort sensor will monitor the network traffic and alarms one whenever there is a malicious activity.

2 Cloud computing

2.1 Concept

Like the meteorological phenomena from which it takes its name, cloud computing cannot be easily defined. There are many definitions, which share the same common denominator: the Internet. Cloud computing is a way to use the Internet in the daily life from a single machine or single room, using all the tools installed on computers. It is also the ability to use shared computing resources with local servers handling applications. With cloud computing users do not worry about the location and the storage of their data. They just start using the services anywhere and at anytime. The main driver of this technology is virtualization (Hypervisor) and virtual appliance [3].

Virtualization provides a means to separate the physical hardware and the operating system and applications by simulating software [4]. The software called hypervisor is uploaded inside the computer. That software in turn also uploads files that define a virtual computer, a virtual machine. A virtual appliance is an application that is grouped together with all the components that it needs in order to run with an operating system. The virtualization of computers and operating systems hides the physical characteristics of computers to the users. The hypervisor is a part of virtualization, which allows many virtual operating systems to run on the same physical machine instantaneously [5].

In the same server hardware, one was able to install many instances of virtual servers, which are connected together via the virtual switch. This architecture allows the creation of a virtual data center with the same functionalities as a real rack system environment.

The redundancy of this system allows the applications to be available to the users at anytime and everywhere.

2.2 Cloud type

The users have fast access to diverse types of applications regardless of the device they are using. It can be a computer, a smartphone or a personal digital assistant (PDA) as long as the device is connected to the Internet. There is no point in installing new software. The resources are stored in a pool of servers rather than running on a single dedicated server. There are four types of cloud computing: Public Cloud, Private Cloud, Hybrid Cloud and Community Cloud.

2.3 Public cloud

In public cloud computing, the provider makes the resources available to the customers over a public network like the Internet. It owns and runs the technology to deliver the service and the consumers have no control over the operations of the service. Usually the documents of the company which uses the public cloud are stored outside its premises by a third party which they trust. It raises concerns about the data privacy security since the computing infrastructure (computers, network and storage) is contained remotely outside the firewall of the company [6]. This leads to the issue of trusting the provider to do the work in the right way. This is one of the reasons why it is also named an external cloud. The public cloud is a web based access business solution where the consumers are billed only for what they use or what they have allocated for use. In this pay-per-use model, the difference between cloud storage and a dedicated hardware appliance is not the functional interface but the scalability that it offers to the consumers. Adding resources or managing the platform is not tied to only one tenant but spread over all tenants of the platform [7].

2.4 Private cloud

A private cloud allows the organization to manage its resources over its own private network. The company owns the service and defines which users can access it. The security risks are reduced because everything is managed inside the enterprise firewall allowing a fair used of the applications and the network bandwidth [8]. This internal cloud remains behind the firewall. Enterprises can deploy security protocols and monitor the levels of access to the information. In the private cloud, the tenant has relative flexibility on policies and procedures for provisioning, usage and security. The owner also controls the maintenance schedule and the upgrades. If hardware fails, the server is automatically booted on the remaining node. It provides direct access to the support team and helps to avoid the downtime. The private cloud works well for infrastructure when it comes to virtualizing servers. It is a good platform for organizations that want to implement the compliance.

2.5 Hybrid cloud

Hybrid cloud computing is a platform which interoperates between private cloud and public cloud. It is deployed by organizations, which do not want to put everything in the external cloud (public cloud) while hosting some servers in their own internal cloud infrastructure. The cloud providers are able to process applications which can work seamlessly between those boundaries [9]. In a case where the public cloud fails to handle an application, the request can be forwarded to the private cloud as shown in figure 1. The hybrid cloud validates the fact that not all information technology resources should remain in the public cloud today. When considering the security restrictions and the performance, the need of a private cloud is a fact today. Enterprises have to know which kind of data can be kept locally and what can be processed remotely.

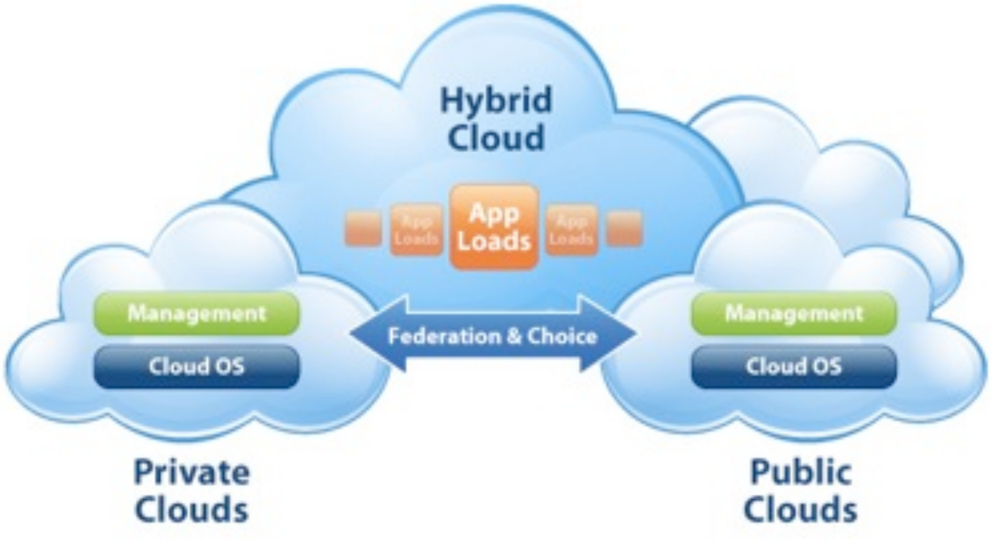


Figure 1 Hybrid Cloud computing [10]

Figure 1 Iillustrates the hybrid cloud.

2.6 Community cloud

A community cloud allows companies to share resources and infrastructure over a common cloud environment. These enterprises must be in the same field of operations and in the same industry. It is deployed by one company and used by the others or provided by a third party over the Internet [11]. In this model, several domains and computers are joined together to interoperate and compute a complex task. It is grid computing. The particularity of the community cloud is the distribution of its server functionality among the users machine, providing their resources into a virtual data centre. A node will be deployed as isolated virtual machine before proceeding to the resources exchange. The nodes have to be interconnected to form a peer-to-peer network with no failure. Upon successful task performance in this shared network, the computers should identify each other based on variable such as uptime and performance. If a service is not available in a local server, it can be copied and retrieved as needed [12].

3 Delivery model

3.1 Software as a Service (SaaS)

In software as a Service, an application is hosted by the service provider and accessed via the World Wide Web by the client [13]. An easy way to see the idea behind software as a service is to think about the web-based service offered by companies like Google (Google doc) and Microsoft (Microsoft Office 365). The vendor hosts all the programs and data in a central location, providing end users with access to the data and software through the World Wide Web. Software as a service offers the biggest cost savings over installed software by eliminating the need for enterprises to install and maintain hardware. The vendors have meticulous security audits.

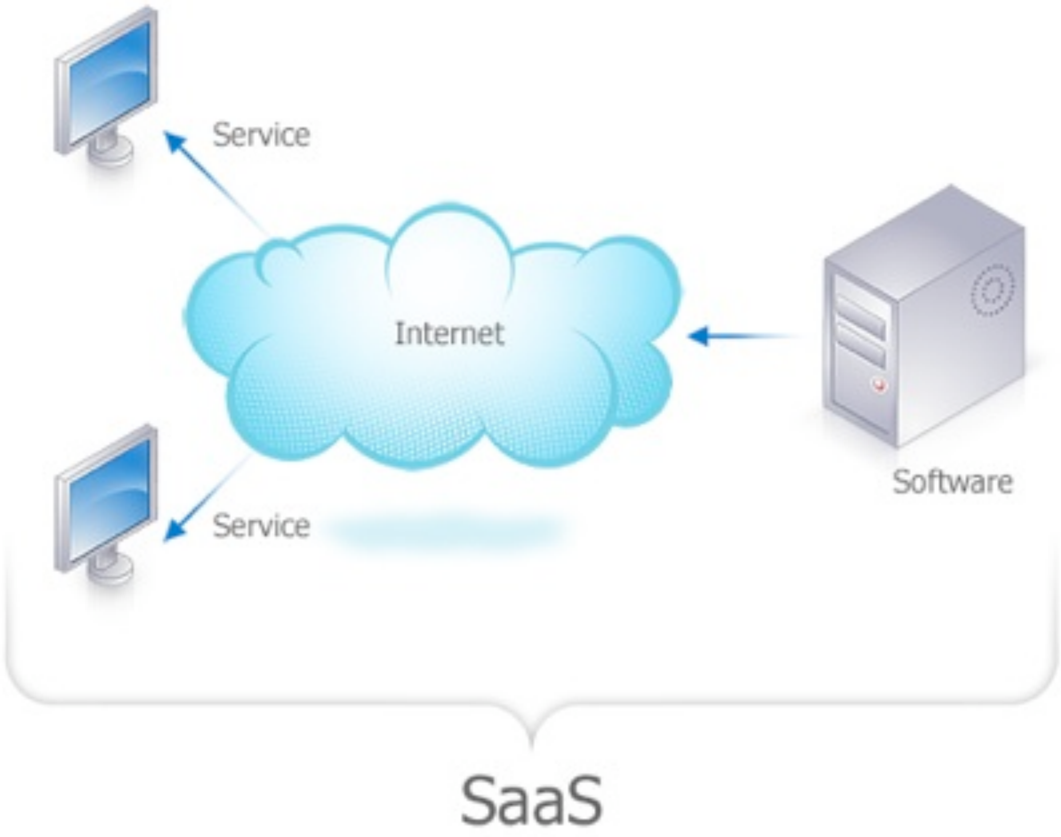


Figure 2 Software as a service [14]

Software as a service removes the need for an organization to handle the installation, setup and daily maintenance of its application. In this model, the resources are outsourced and accessed at anytime.

3.2 Platform as a Service (Paas)

Platform as a service is the delivery and deployment of computer applications over the Internet with the consideration that all the parties needed in the development process are already obtained [15]. An organization usually needs to worry about building its own virtual machine images, configuring the storage and network. Using Paas, network, storage and virtual machine images are already pre-built by the provider. Additionally the provider monitors the virtual machine for failure and initiate auto-recovery when needed.

An example of platform as a service is Ubuntu 10.10 Server in Amazon elastic cloud computing (EC2) in which a user is given an Internet protocol (IP) address and access the server in the cloud infrastructure using the secure shell protocol (SSH). Platform as a service provides the entire infrastructure needed to run applications over the Internet with security and scalability. Developers are now able to code, test to fix bugs, deploy and maintain applications in the same computing environment.

3.3 Infrastructure as a Service (Iaas)

Data infrastructure is moving towards a utility model just like electricity and telephone bill. Infrastructure as a service allows organizations to outsource the equipment used to support operations including storage, server, hardware and network component. A client purchases or rents computing power or disk space, which they can access through their mobile, or desktop PC. The provider hosts the client's web site while monitoring the availability of the service. Rather than purchasing software, a server, data centre space or network equipment, clients buy those resources as a fully outsourced services. Platform offered on top of the Infrastructure as a service can deliver rapid provisioning of applications allowing greater agility and reducing time for developers. It should scale to avoid limitations. A typical example of Iaas is Gogrid. [16]

3.4 Security as a Service

Cloud computing providers use security as a Service model to deliver real time security protection. The increased use of the Internet has raised the need to protect companies' assets against malware or data theft. Organizations and enterprises can then bring outsourcing to cyber security in order to secure their data, content and communication.

Security as a service shifts all the security inspection, enforcement and management processes from the customer's location to the cloud provider. The antivirus in this case is provided to the clients through the web browser. Using security as a Service, enterprises are certain to have the vulnerabilities in their system discovered and mitigated on time. A typical example is Zscale cloud services.[17].

Bitdefender Beta has a version called BitDefender Safego which acts like an application in the Facebook profile and checks for any infected link in the wall.[18]

4 Building the cloud

4.1 Hardware technology

Dell poweredge 6850 was used as hardware in this project [19]. It has a 32 GB of random access memory with 100 GB of hard disk space. It has a dual Gigabit interface network interface card (NIC) with redundant power supply. This server supports Microsoft Windows Server 2003 Enterprise Edition, Microsoft Windows Server 2003 Standard Edition and Red hat enterprise Linux.

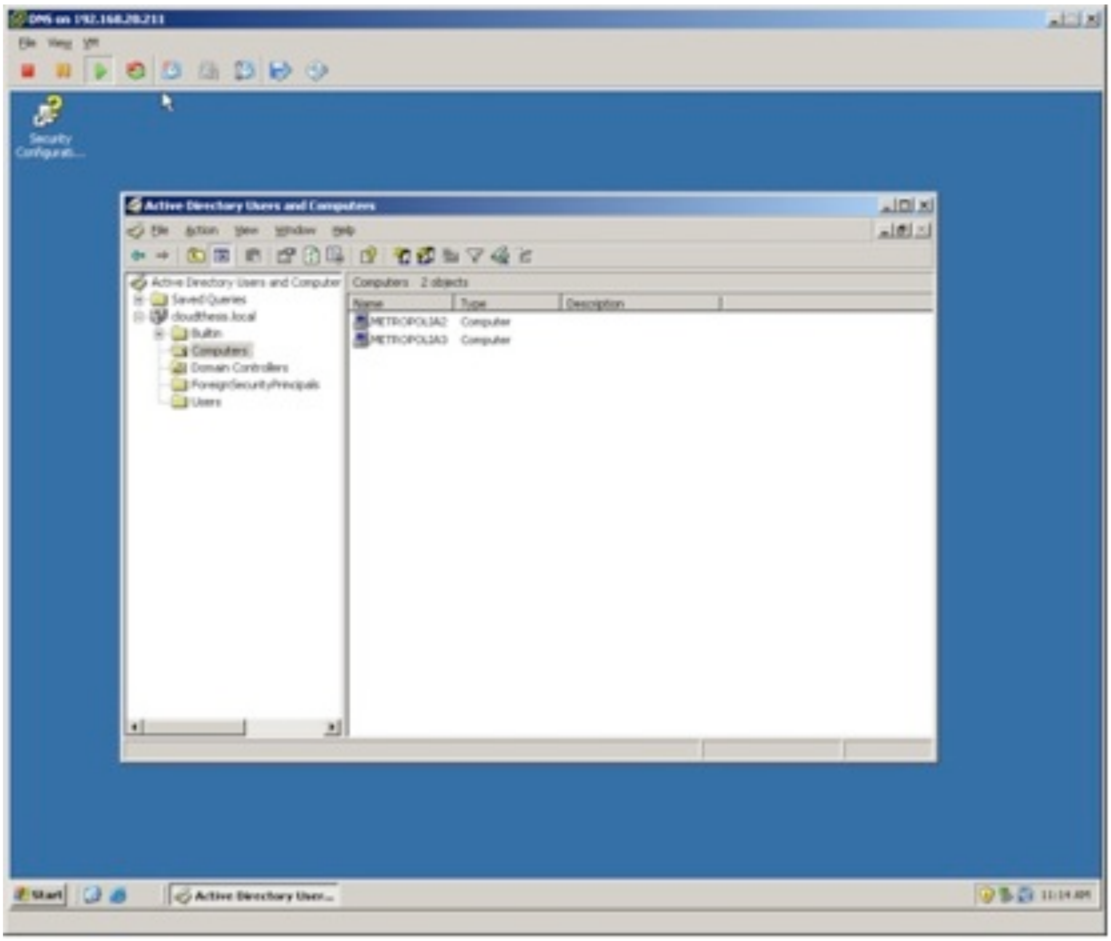
4.2 VMWARE ESX4.0

VMware ESX4.O is the hypervisor which is the virtualization layer that serves as the foundation for the rest of the product line. Two products that interact with each other to provide a virtualization environment make VMware ESX4.0: the service console and the VMkernel. The service console is the operating system used to interact with ESX4.0 and the virtual machines running on the servers. The VMkernel manages the access of the virtual machines to the physical hardware with a means of CPU scheduling, memory and virtual switch data processing. The software ESX4.0 runs only on 64 bits x86 CPU server and is directly installed in the system hardware.

The Dell Poweredge 6850 has up to 4 Intel Xeon processors, which provides support to 64 bit. It needs at least 2 GB of random access memory (RAM) and gigabit controllers as network adapters. More RAM can be used if we are running many applications. Data used by the virtual machines must be stored on physical disks. It is recommended to use many physical disks such as SCSI disks, Fibre channel LUN and RAID LUN. Prior to installing, we burn the ESX4.0 installation ISO image onto a DVD or USB media. We insert the DVD inside the server hardware and we run the installer in graphical mode. The main options provided during the installation are: keyboard layout (Finnish), vsphere license key, network adapter for the service console, IP address acquired by DHCP (dynamic host configuration protocol), root password, primary DNS (domain

name service), secondary DNS, installation location and time zone. After this step, one use a Windows desktop machine in which the IP address registered above has been entered and the Vsphere client downloaded to manage this ESX4.0 host. [20]
Filling the IP address and root password field in the Vsphere client allowed the connection inside the server and its management.

For the purpose of this project the Vmware Vcenter Server and Vmware Vcenter Lab manager were used. The requirement is that the both softwares must be installed in two different virtual machines under the same domain. For that reason, one installed the Windows Server 2003 Enterprise Edition x64 bits image. It was configured as a domain name system (DNS) server. Within the same server, one created an Active Directory for that domain and added the other guest Windows operating system in that Active Directory as shown in figure 3.



Computers 2 objects

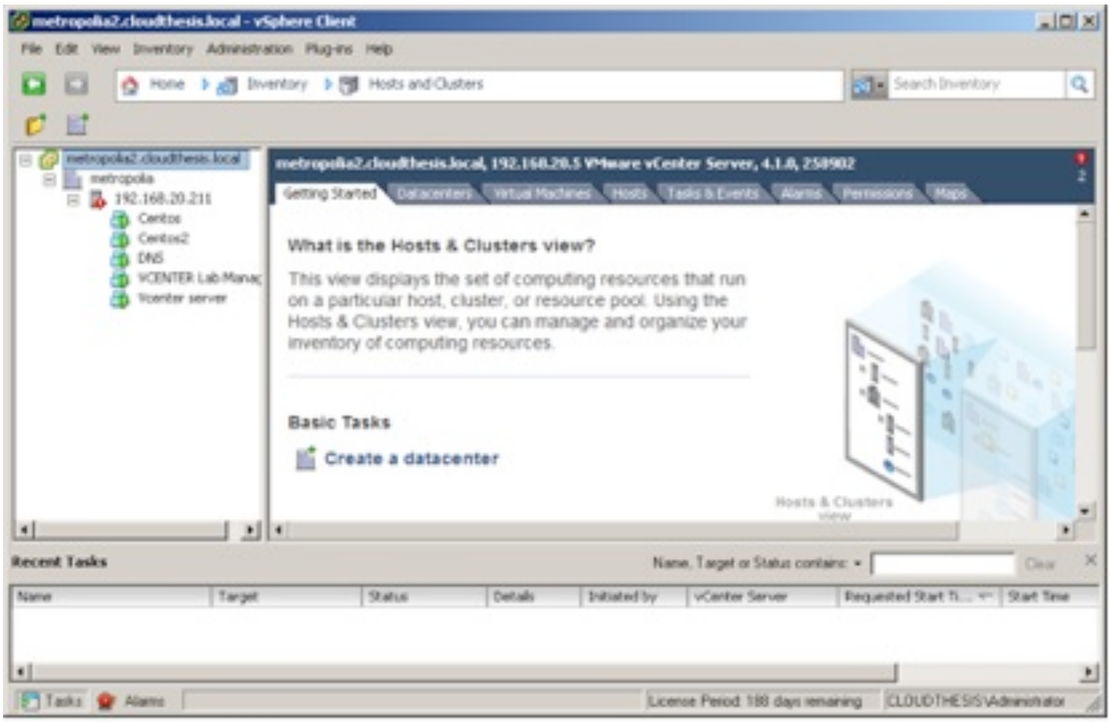
Name	Type	Description
METROPOLIA2	Computer	
METROPOLIA3	Computer	

Figure 3. Guest Operating System installed in the same domain

Active Directory (AD) is directory implementation in Microsoft computers used to store network services such as DNS, Kerberos-based authentication and Lightweight Directory Access protocol (LDAP). Since the both operating systems in the METROPOLIA2 (Microsoft Windows Server 2003 Enterprise Edition x64bits) and the METROPOLIA3 (Microsoft Windows Server Standard Edition SP2) are in the same domain, one can now installed the VMware Vcenter Server and the VMware Vcenter Lab manager.

4.3 Vmware vCenter server and vCenter Lab Manager

The vSphere client is software utility downloaded from the ESX4.0 installation, which facilitates the management of the ESX server. It has a graphical interface. The vSphere client is the best tool used when installing the server images. In the METROPOLIA2 computer, the vCenter Server was installed using the installer burned in a DVD media. This Windows server must use static assigned IP address. Vcenter Server is used to manage multiple instances of ESX host. This is why during the installation one was prompted to give the IP address of the ESX4.0. It is easy to create and deploy virtual machines under the same domain. This capability is made possible first by creating a datacenter and then a cluster, which will be the set of virtual servers being added and managed as in figure 4.



metropolia2.cloudthesis.local - vSphere Client

metropolia2.cloudthesis.local, 192.168.20.5 VMware vCenter Server, 4.1.0, 258902

What is the Hosts & Clusters view?

This view displays the set of computing resources that run on a particular host, cluster, or resource pool. Using the Hosts & Clusters view, you can manage and organize your inventory of computing resources.

Basic Tasks

Create a datacenter

Name	Target	Status	Details	Initiated by	vCenter Server	Requested Start Ti...	Start Time

License Period: 188 days remaining

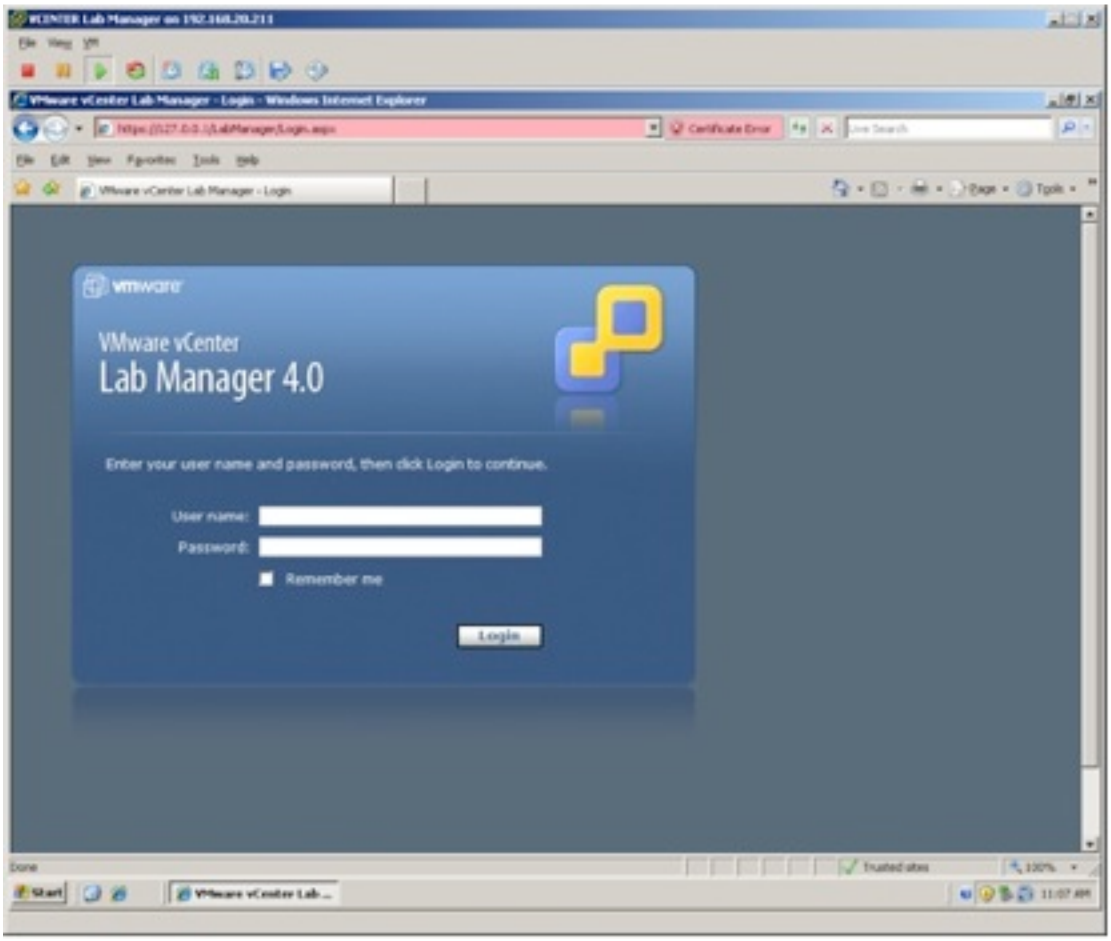
CLOUDTHESIS\Administrator

Figure 4. Vcenter Server

The performance monitoring view provides a detailed description of the graph of CPU (central processing Unit), memory, Disk I/O and network I/O to analyze the physical and virtual servers which are running.

The vCenter Lab Manager is an easy to use web application. Users can operate it using a browser (Internet Explorer or Firefox) to access the web console. It provides automated management of virtual environment. It creates, deploys and captures any system configuration in seconds. The vCenter Lab Manager has a library to store virtual machine images for its users and leverages the VMware vSphere and vCenter server to provide virtual infrastructure from a central location. It is a scalable solution with policy based access control enforced and provides users with a self-service access to these configurations for testing, support and training. The virtual or physical machine in which it is installed must also used a static IP address and should not contain any Vcenter Server installation. The main software requirements during the installation process are: Internet Information Services (IIS) 6.0 and Microsoft .NET framework 2.0(Service Pack 1 or later).

Internet Information Services (IIS) is a group of web servers with additional capabilities created by Microsoft for Windows server. Microsoft .NET framework is a software framework for Microsoft Windows Operating System, which includes a large library available to all the programming language that .NET supports. After burning the vCenter Lab Manager installer inside a DVD media, one can start the installation process.The VMware vCenter Lab Manager requires the IP address or the hostname of the vCenter Server in order to operate and after a successful set up we have figure 5


vCENTER Lab Manager on 192.168.20.211
File View VM
VMware vCenter Lab Manager - Login - Windows Internet Explorer
https://127.0.0.1/LabManager/Login.aspx
Certificate Error
Live Search
File Edit View Favorites Tools Help
VMware vCenter Lab Manager - Login
Page
Tools
vmware
VMware vCenter
Lab Manager 4.0

Enter your user name and password, then click Login to continue.

User name:
Password:
Remember me
Login
Done
Trusted sites
100%
Start
VMware vCenter Lab...
11:07 AM

Figure 5. vCenter Lab Manager

It is possible to log in using the administrator user name and password of the computer. The web console has many features on the deployment, the creation and the usage of virtual machines running a guest operating system. At this step, our cloud is built and ready to be used. The problem of security really matters in the cloud.

5 Threats to the cloud

5.1 Background

The virtual servers are created instantaneously in the cloud and used at the same time. In a public cloud the data of the customers are kept in the provider premises. The question of privacy is a real concern because there is no guarantee that illegitimated eyes could not have access to that sensitive information. Furthermore, because many services are deployed through the Internet via the virtual servers using software as a service (SaaS) there is a risk of malware infection and hacker penetration. In fact, a web server can be compromised and served to spread a bad URL (uniform resource locator) link and to redirect the requests to a fake page where the malicious code will be downloaded in order to infect and take control of the machines.

5.2 Malware

5.2.1 Viruses

A virus is a malicious code, which makes copies of itself and distribute those copies to other files and programs. It needs the user interaction to propagate. When viruses infect a program, they propagate to infect other programs on the system and other systems that use a common infected program. Viruses can also infect the MBR (master boot record) of the hard drive or a removable media.

The master boot record (MBR) of a hard drive is the unique location on the disk where a computer basic's input and output system can locate and load the boot program. If there is an infected disk in the drive when the computer boots, the virus can be loaded into the memory. Viruses exploit the vulnerabilities related to some applications document like word processing file and spreadsheet. Most of those software are writing using macro programming languages and the bad guys are taking advantage of those capabilities [21,5-1]. Macros viruses spread from application that uses macros such as Microsoft Office documents.

Email viruses travel as an attachment to email messages. They replicate by automatically mailing themselves to people in the victim's email book.

Most viruses are pretty harmless and sometimes the user might not notice them for years. The first virus which was able to hide without being discovered was called Brain. The Brain stealth virus hides itself in the memory by simulating all the DOS system call that normally detects viruses, causing them to return the information that the virus is absent [22].

5.2.2 Worms

A computer worm is a program that executes, reproduces independently and travels across network connection. It takes advantage of known vulnerabilities to spread. They are two types of worms: Network Service Worm and Mass Mailing Worms.

Network Services Worms exploits the common vulnerability found in network service associated with an operating system or an application. Once they have exploited the targeted protocol in the system they look for other possible systems over the same network by performing scanning. An example of such a worm is Sasser, which uses Server Message Block (SMB) and Local Security Authority Subsystem Service (LSASS) in Windows to spread [23].

Mass Mailing Worms infect system by searching for email addresses and sending a copy of itself to those addressees. Usually they use the system email client. Embedded in most network software, computer worms penetrates firewalls and other computer security measure.

5.2.3 Trojan horse

Trojan horse is an application which appears to be useful, downloaded from the Internet and in fact is malware. They do not spread and are separated into two parts: the server and the controlled computer. When the malicious program is loaded in the memory of the host, the attacker can take control of the computer by sending command. The client disguises itself and can spread via chat software such as Skype, yahoo messenger and file sharing website.

There exist many variants of Trojan: Remote access Trojan, password sending Trojan, key loggers, Destructive Trojans, and proxy Trojans.[24]

5.3 Web application and data security risk

5.3.1 Injection

Injection flaws allow an intruder to forward malicious code through the web application inside the system. Scripts written in Python, Perl or any other programming language can be injected and executed into the unsecure application. When the web application handles HTTP (hypertext transfer protocol) request through as part of an external request, it must be carefully examine otherwise a bad guy can inject special characters or malicious commands in the information which will certainly transfer these to the external system for execution. SQL injection is a widespread form of injection. In this type of attack, when the parameter that the application sends to the database is revealed, the attacker can append malicious SQL command into the content of that parameter and trick the web application to forward fake queries to the databases [25]. A successful SQL injection can lead to an authentication bypass allowing an unauthorized user to login to the application without supplying a valid username and password, information disclosure and remote command execution.

5.3.2 Security misconfiguration

The web server and application server are the backbone of a web application. They provide a number of services that the web application uses including directory service, data storage and mail. Failure to properly manage the configuration of these servers can lead to a wide variety of security breaches. Security misconfiguration can happen at the application stack, the framework, the web server, the custom code and the platform. External intruders and users with their own accounts can attempt to compromise the system. Attackers use the unpatched flaws, unprotected files and directories to have illegal access or knowledge of the system.

The defaults account must always be changed because the attacker can discover the standard admin page and log in with those defaults passwords [26].

The server can also generate an error message that displays information concerning its environment, users and associated data. The information may be useful for launching a deadly attack. If one attack fails, the attacker can still use the error information provided to launch a more focused attack.

5.3.3 Insecure cryptographic storage

In the cloud, the need to store sensitive information by the web application in the database or in the file system is important. The information can be a credit card number, social security number, account record and passwords. Therefore, the use of encryption is relevant. By simply not encrypting the data which deserves the encryption. There will be a flaw. Developers usually make a mistake when using encryption and the main areas where mistake are usually made are: failure to encrypt critical data, insecure storage of keys, certificates and passwords, improper storage of secrets in memory, poor choice of algorithm. Almost every application is connected to a database, the credentials used to make these connections should be encrypted to prevent easy access to these data storage systems. The web application must have cryptographic support [27]. In the case of the credit card number storage, a merchant should respect the compliance. The compliance is a set of regulations applied and enforced with the means of fines. Following the PCI DSS (payment card industry data security standard) compliance requirement 3, cardholder data must be protected. The personal account number, the cardholder’s name and the expiration date should be encrypted when transmitting across different network [28].

6 Threat mitigation

6.1 Symmetric cryptography

Cryptography is a method of storing and transmitting data in a form that only the recipient can read and process.

The mechanism that makes it up is to hide information from unauthorized individuals. It is an effective way to keep sensitive information, as it is stored on media.

Encryption is a method to convert readable data called plaintext into an unreadable format called ciphertext. Once it is transformed into ciphertext neither a human nor a machine can process it until it is decrypted.

In symmetric cryptography, the sender and the receiver use the same key for encryption and decryption. Symmetric keys are also called secret keys because this type of encryption requires each user to keep the key a secret and protected. The security of the symmetric encryption is completely dependent on how well users protect the key. If a key is compromised, all messages encrypted with that key can be decrypted and read by an attacker. [29,686-687]
The following are examples of symmetric cryptography: Data Encryption Standard (DES), Advanced Encryption Standard (AES) and Blowfish.

6.2 Asymmetric Cryptography

Asymmetric cryptography utilises the combination of two different keys, one public key and one private key. Everyone can know the public key but the private key is known and used only by the owner. The two keys are mathematically related. If someone gets the public key of another person, he or she could not be able to figure out the corresponding private key. When Bob encrypts data with his private key, the receiver Alice must have a copy of Bob's public key to decrypt it.

The receiver can reply also in an encrypted form. In that case, Alice encrypts the message using Bob's public key and the message will be decrypted at the other end using Bob's private key because he is the only person to have the private key. The both keys, public and private can be used to encrypt and decrypt a message.[29,688-691].
The following are examples of asymmetric key algorithms: Rivest-Shamir-Adleman (RSA), Diffie-Hellmann and Digital Signature Algorithm (DSA).[29,713-718]

6.3 Network intrusion detection system

An intrusion detection system aims to detect a security breach. Intrusion detection can be defined as a method to detect unauthorized use or attack to a computer, network or telecommunication system. The basic idea behind the intrusion detection system is to spot something suspicious happening on the network and sound an alarm. In a typical intrusion detection system product, the sensors collect traffic and user activity data and send them to an analyzer that looks for abnormal activities.

When the analyzer detects an activity, it sends an alert to the administrator interface. The network intrusion detection system uses sensors with a network interface card in a promiscuous mode [30,249]. When a network interface card is in a promiscuous mode, it collects all traffic, makes a copy of all packets, and then passes one copy to the TCP stack and one copy to the analyzer to look for specific types of patterns of known threats.

In the current project, Snort-2.9.0 was installed as a network intrusion detection system on a CentOS 5 server. Snort is an open-source network intrusion detection system which can be downloaded free of charge. It is made of five major components. The first component is the Packet Decoder that collects packets from different network interfaces and prepares them to be pre-processed. The second component, pre-processor is used to arrange and modify packets before being analyzed by the detection engine . The third component, the detection engine is responsible for analysing all the packets passing through it to detect any sign of attack by using a pre-defined set of rules. After the discovery of an intrusion by the detection engine, the logging and alerting system generates the alarm. The type of output produced by this alerting system is controlled by the fifth component called output modules or plug-ins. Some of the capabilities of the plug-in can be sending a message to a Syslog server.

The Snort installation requires Apache, MySql and PHP tool to work properly. The software were installed after logging as root

```
yum –y  install mysql mysql-bench mysql-server mysql-devel
mysqlclient10 php-mysql
httpd gcc pcre-devel php-gd gd mod_ssl glib2-devel gcc-c++ libp-
cap-devel php php-pear yum-utils
```

After updating the system using yum –y update, it is better to configure the server remotely using secure shell (SSH). SSH was secured by editing the /etc/ssh/sshd_config file change the following lines (if it is commented out remove the #):

```
Protocol 2
PermitRootLogin no
PermitEmptyPasswords no
```

As seen, login as root will no longer be accepted, so login as a simple user and su – to the root account. We start the required services

```
chkconfig httpd on
chkconfig mysqld on
service httpd start
service mysqld start [31,6-7]
```

Before downloading the needed file, the directory under `the /root called Snortinstall` was created

```
cd /root
mkdir snortinstall
```

From the directory /root/snortinstall, wget was used to have all the files inside.

```
Wget –c http://www.snort.org/downloads/808 will download the
source code.
```

```
tar –xvzf snort-2.9.0.4.tar.gz

cd snort-2.9.0.4
./configure --with-mysql --enable-dynamicplugin
make
make install
```

A group called snort was created [31,8].

```
groupadd snort  //will create a new group Snort
useradd -g snort snort –s /sbin/nologin
 mkdir /etc/snort
mkdir /etc/snort/rules
mkdir /etc/snort/so_rules
mkdir /var/log/snort
cd  etc/
cp * /etc/snort
```

In the /snort/snortinstall directory, the latest rule was downloaded. One need to be registered in the website in order to obtain the rules file.

```
tar  –xvzf  snortrules-snapshot-2860.tar.gz and we copy everything in the rules directory
cp * /etc/snort/rules
```

The network address was to monitor was assigned to Snort by editing the snort.conf file located in the /etc/snort [31,8].

```
var HOME_NET 192.168.0.0/16
change var RULE_PATH ../rules to var RULE_PATH /etc/snort/rules
change var SO_RULE_PATH ../so_rules to var SO_RULE_PATH /etc/snort/so_rules
Uncomment the last line from the output section in the /etc/snort/snort.conf
 # output log_unified: filename snort.log, limit 128
```

A MySql database must be set up for Snort to be able to log all the alerts inside [31,9]. For this purpose a new database snort was created in the database server. The password was set and the permission was granted.

```
mysql>  SET PASSWORD FOR root@localhost=PASSWORD('password');
//Set the password
>Query OK, 0 rows affected (0.25 sec)
mysql>  create database snort;
>Query OK, 1 row affected (0.01 sec)
mysql>  grant INSERT,SELECT on root.* to snort@localhost;
>Query OK, 0 rows affected (0.02 sec)
mysql>  SET PASSWORD FOR
snort@localhost=PASSWORD('pick_a_password);
>Query OK, 0 rows affected (0.25 sec)
mysql>  grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.*
to snort@localhost;
>Query OK, 0 rows affected (0.02 sec)
mysql>  grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.*
to snort;
>Query OK, 0 rows affected (0.02 sec)
mysql>  exit
>Bye
```

Everything is ready, Snort can be started from the command line to make sure it loads properly [32],

```
/usr/local/bin/snort –u snort-g snort –c /etc/snort/snort.conf –
i eth0.
```

After getting a confirmation message from the wizard if there is no software missing and misconfiguration, Snort can be run with ethernet 0 our sniffing interface. Snort is now running as linux daemon.

6.4 Reading a Snort alert

The alerts are stored in the /etc/log/snort/alert file and one of them is the following

```
[**] [1:17494:3] WEB-CLIENT Microsoft Internet Explorer Long URL
Buffer Overflow attempt [**]
[Classification: Attempted User Privilege Gain] [Priority: 1]
11/23-15:06:31.400618 89.207.18.81:80 -> 192.168.20.12:37439

TCP TTL:51 TOS:0x0 ID:46490 IpLen:20 DgmLen:835 DF
***AP*** Seq: 0x71C2285  Ack: 0x484DFFB3  Win: 0x7FFF  TcpLen:
20
[Xref   =>   http://cve.mitre.org/cgi-bin/cvename.cgi?name=2006-
3869][Xref => http://www.securityfocus.com/bid/19667]
```

The first number (1) represents the generator ID that tells the user which component of Snort, which generates this ID. The second number (17494) represents a signature ID and (3) is the revision ID [33]. This alarm is classified as Attempted User Privilege Gain.

The source address is 89.207.18.81 and the destination is 192.168.20.12 from port 37439 to port 80. One can see from the reference that this is a remote buffer overflow vulnerability founds in Microsoft Internet Explorer with the exploit code CVE-2006-3869 name [34,12]. This attack was successfully recorded in the log file. In the cloud, servers and applications have many security flows, which can be exploited to compromise their functionality. In the cloud services are accessed through the Internet and snort is monitoring every traffic on port 80 (HTTP).

7 Results

The intrusion detection system once running in the cloud, was able to monitor the whole network specified as 192.168.20.0. The alarms were triggered and stored in the log file. A thorough analysis of the log file lead to the following observations. At a glance all the alerts were ranked with a priority 3 meaning that the severity level of the bad traffic was not high.

The http_inspect is a preprocessor, which is associated with any malicious web traffic. It has detected a long client header in an http request to generate the [119:19:1] alert.

```
[**] [119:19:1] (http_inspect) LONG HEADER [**]
[Priority: 3]

11/23-15:06:31.832080 192.168.20.12:57561 -> 216.239.116.64:
```

The source address is our Cent0S 5 server and the destination addresses were outside our network. Any request made by a client to any webpage was inspected and classified accordingly [35]. They were many replies from the web servers with no content length and classified with [120:3:1].

```
[**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSFER-
ENCODING IN HTTP RESPONSE [**]
[Priority: 3]
11/23-15:06:31.400618 89.207.18.81:80 -> 192.168.20.12:37439
```

It is compulsory to include a content length or a transfer-Encoding in a request or a reply. Transfer-Encoding must be used to indicate any transfer-codings applied by an application to ensure safe and proper transfer of the message. It is a property of the message that may be added to a request or reply chain [36].

In our cloud environment, there is an Internet explorer installed. There was a Long URL Buffer overflow attempt to this web client, trying to exploit a known vulnerability in Internet Explorer. The alert is classified as [1.17494:3].

```
[**] [1:17494:3] WEB-CLIENT Microsoft Internet Explorer Long URL
Buffer Overflow attempt [**]
[Classification: Attempted User Privilege Gain] [Priority: 1]
11/23-15:06:32.025956 89.207.18.81:80 -> 192.168.20.12:37439
TCP TTL:51 TOS:0x0 ID:32773 IpLen:20 DgmLen:1230 DF
***AP*** Seq: 0x71C25A0  Ack: 0x484E0401  Win: 0x7FFF  TcpLen:
20
```

```
[Xref   =>   http://cve.mitre.org/cgi-bin/cvename.cgi?name=2006-
3869][Xref => http://www.securityfocus.com/bid/19667]
```

Buffer overflow occurs when a program or an application tries to input more data in the memory allocated for it [37]. By exploiting this vulnerability, a remote attacker could execute arbitrary code via a long URL on a web site. This alert has a priority 1 and is severe. If succeeding, this attack can lead to a denial of service, information disclosure, loss of integrity and a complete admin access .

8 Discussion

Snort monitors every packet in the network address specified in the snort.conf configuration file. It uses a sniffing interface where it captures the traffic before sending it to the detection engine. In this set up, our sniffing interface is Ethernet 0 with Internet protocol (IP) address 192.168.20.12. To check this out , the following command was typed `cd /etc/sysconfig/network-scripts/`

```
[root@metropolia2 network-scripts]# cat ifcfg-eth0
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:50:56:8D:00:00
IPV6INIT=yes
IPV6_AUTOCONF=yes
ONBOOT=yes
DHCP_HOSTNAME=metropolia2.cloudthesis.local
IPADDR=192.168.20.12
NETMASK=255.255.255.0
GATEWAY=192.168.20.254
TYPE=Ethernet
PEERDNS=yes
USERCTL=no
```

The pre-processors then analyzed the packet. Here it detected and normalized hypertext transfer protocol (HTTP), file transfer protocol (FTP) anomaly, fragmentation attack, internet control message protocol time to live (ICMP TTL), port scan and decoded application traffic. Later the pre-processor forwarded the packet to the detection engine which will look for any signature matching the received pattern and generated an alert.

This intrusion detection system was able to detect the ICMP (Internet control message protocol) ping generated by my Nmap scan with an hyperlink that gives more information about the attack.

```
**] [1:469:4] ICMP PING NMAP [**]
[Classification: Attempted Information Leak] [Priority: 2]
11/18-13:15:04.020752 192.168.115.46 -> 192.168.20.12
ICMP TTL:45 TOS:0x0 ID:29489 IpLen:20 DgmLen:28
Type:8 Code:0 ID:28711 Seq:0 ECHO
[Xref => http://www.whitehats.com/info/IDS162]
```

Implementing this solution gave good results and showed how reliable this intrusion detection system was. Any malicious traffic cannot escape this box.
Network forensics is the acquisition and analysis of a network event in order to trace the source of the intrusion and the activities made [38]. Attackers fingerprints remain in the firewall log, IDS/IPS log, web proxies and traffic capture. Snort can be used to open any packet capture (PCAP) using the Libpcap format. Packet capture is an application programming interface used for capturing network traffic, and it is implemented in a Linux system using the Libpcap library.

Snort can be used in support of other network forensic tools like Xplico [39], Wireshark [40] to understand the intrusion and trace the suspect activity. Once it opens the packet capture (Pcap), it directly detects and lists the vulnerabilities used by the criminal and the weaknesses found in the system that facilitates the operation.
The interrogation assessed now is how can I provide a real time protection to my system knowing those alerts?. A session can be terminated to prevent the attacker to trying again. Consider seriously the implication of an active response and intrusion prevention as it can create a denial of service against the syste. Just Imagine that an attacker is spoofing his attack using a regular customer Internet protocol (IP) address . Blocking this session from the firewall will cut the legitimate user off.

9 Conclusion

Cloud computing is being the emerging technology in use today. It is cost-effective because a user is charged only for what he or she uses. There are three different types of cloud computing: private cloud, public cloud and hybrid cloud. Those platforms are used to provide services to companies in a way that there is no need to download any additional software. A browser and an Internet connection are enough. Therefore permanent threat and attack facing this evolving technology exists. As a solution to mitigate those attacks a network intrusion detection system was installed in this project. This software will monitor the virtual server and the applications, which are built in and send an alert every time it will find a malicious activity matching the signature database.

In this project, Snort was installed with its rules engine. It was efficient because alerts were triggered. The alarms were generated based on the preprocessors. The results prove that by knowing those alarms, the system can be patched based on the vulnerability discovered to enhance the security. Snort was able to determine in advance the type of the attack, whether it is malware, vulnerability or an intruder. They are classified by priority, which assigns the severity level to the rule. These alerts are stored in plain text in the log file and can be sent to a network managenemnt system which supports simple network management protocol (SNMP) traps. As the alerts are stored in the MySql database, Snort can be integrated with a web interface to view the required data from the database.

Most of the alerts generated were ranked priority 3 meaning that they were not severe. Snort is then an efficient intrusion detection system. It is sufficient to provide enough protection when configured as an intrusion detection system because it does not stop an attack. Configuring as an intrusion detection system, Snort will only log packets matching a pre-defined attack signature and generate the alarm. Common rules can be obtained from the installation files themselves.

These signatures can be updated manually by downloading the recent rules in the snort installation and restarting the censor. Since it is not so efficient to manually download the new rules everytime,Oinkmaster provide an automatic updated of snort rules . Oinkmaster is a tool written in Perl that can be used to update and manage the vulnerability and research team licensed rules, the community rules and the Bleeding Snort rules. It also helps to enable or disable the signatures based on our environment. There is a commercial version of Snort which provides real time access to the updated signatures developed by the Sourcefire Vulnerability and Research team.

For future development of this environment, a script that captures the alerts based on their priority and sends them as an email message to the corresponding email address will be advantageous. The log file contains the different steps of any attack . Choosing the alert being sent among all the triggered ones based on the severity level instead of sending the whole log file will reduce the time when looking for an evidence. An intrusion detection system must be combined with a network intrusion prevention system, which will identify, stop and report the abnormal traffic.

References

1.Introduction to cloud computing architecture [online]. Sun Microsystems USA: Santa Clara:Sun Microsystems 2009.
URL :http:// http://www.scribd.com/doc/17274860/Introduction-to-Cloud-Computing-Architecture. Accessed 08 June 2010.

2. Snort [online]. Sourcefire : 2010
URL: http://www.snort.org/ .Accessed March 17,2011

3. Application architecture for cloud computing [online]. Path: Raleigh NC 27607; Path 2008.
URL:http://www.rpath.com/corp/images/stories/white_papers/WP_ArchitectureForCloudComputing.pdf cloud computing pdf. Accessed 18 June 2010.

4.Virtualization is it right for you?[online].LAD enterprises Inc .URL:http:// www.ladenterprizes.com/pdf/Virtualization.pdf . Accessed June 22,2010.

5.Hypervisor [online].Techtarget.
URL: http://searchservervirtualization.techtarget.com/definition/hypervisor.
Accessed June 01,2011.

6. Laying the ground work for Public cloud and Private Cloud [online]. Dell inc: 15 June 2010.
URL: http://whitepapers.businessweek.com/data/memberRegister.do. Accessed July 06,2010.

7. Public cloud Storage [online] .Techtarget .
URL:http://searchcloudstorage.techtarget.com/definition/public-cloud-storage
.Accessed June 01,2011

8. Cloud Computing Use Cases White Paper [online]. SA(shared alike): July 2,2010. URL: http://www.scribd.com/doc/18172802/Cloud-Computing-Use-Cases-Whitepaper. Accessed July 21,2010.

9. Cloud computing [online]. Infosys Technology limited: 2010.
URL:
http://www.infosysblogs.com/cloudcomputing/2009/05/hybrid_approach_for_cloud_comp_1.html. Accessed July 23,2010.

10.Reprinted from Acute system consulting [online]. Acute system consulting: 2010. URL:http://www.acutesys.com/wp-content/uploads/2009/10/virtualize-why-choose-hybrid-cloud-dg-en-full.jpg. Accessed July21, 2010.

11. A community approach to cloud computing [online]. Orange Business Services: July-August 2010.
URL:http://www.orangebusiness.com/en/mnc2/footer/news/enterprise_briefing/summer2010/cloud-computing.jsp . Accessed July 23,2010.

12. Digital Ecosystems in the Clouds : Towards Community Cloud Computing [on-line].Gerard Briscoe,Alexandros Marino :5th Octobre 2009.
URL: www.arxiv.org/pdf/0903.0694 .Accessed April 28,2011

13.What is Saas and What are its advantages?[online]. Metaquote Software corp: 2008-2011.URL:http://www.teamwox.com/en/teamwox/tutorials/71.
Accessed September 15,2010.

14. Teamwoks ® Online Team Collaboration Software [online]. Groupware Software: 2008-2010.
URL: http://www.teamwox.com/en/teamwox/tutorials/71
Accessed September 25,2010.

15. Cloud computing [online].Cloud computing Asia: 2010.
URL: http://www.cloudcomputinglive.com/asia/platform-as-a-service.html. Accessed December 08,2010.

16. GoGrid [online] : Gogrid . URL: http://www.gogrid.com/ .Accessed May 1,2011.

17. Cloud computing-Evaluating security as a service [online].Quinstreet Inc:2011.
URL:http://www.cioupdate.com/trends/article.php/3893521/Cloud-Computing---Evaluating-Security-as-a-Service.htm .Accessed March 17,2011.

18. Bitdefender Safego [online].bitdefender
.URL: http://www.bitdefender.com/media/html/facebook/safego/. Accessed March 26,2011.

19.Technical specifications, Dell poweredge 6850 system's user Guide[online]. Dell;2010.
URL: http://support.dell.com/support/edocs/systems/pe6850/en/ug/f2332aa.htm .Accessed December 17,2010.

20. ESX and vCenter Server Installation guide [online].Vmware Inc, USA: Palo Alto :2009-2010
URL: www.vmware.com/pdf/vsphere4/r41/vsp_41_esx_vc_installation_guide.pdf. Accessed January 24,2011.

21. Computer Security Incident handling Guide [online].National Institute Of Standard and technology, Gaithersburg USA : Mach 2008
URL:
http://csrc.nist.gov/publications/nistpubs/800-61-rev1/SP800-61rev1.pdf.Accessed February 07,2011.

22.Virus and threat description Virus: Brain [online]. F-secure corporation :2009
URL: http://www.f-secure.com/v-descs/brain.shtml .Accessed May 1,2011.

23. Microsoft Security Information [online] .Microsoft Corporation :2011.
URL: http://www.microsoft.com/cze/security/incident/sasser_info.mspx.
Accessed May,01,2011.

24.Trojan Virus [online],TopBits URL: http://www.tech-faq.com/trojan-virus.html . Accessed March 08,2011.

25. A6 2004 Injection Flaws-The Open Web Application Security Project [online],Mediawiki : 10 October 2008.
URL: http://www.owasp.org/index.php/A6_2004_Injection_Flaws. Accessed March 09,2011

26. Top 10 2010-A6-Security misconfiguration-The Open Web Application Security Project [online]. Mediawiki : 14 june 2010 .
URL:http://www.owasp.org/index.php/Top_10_2010-A6-Security_Misconfiguration .Accessed March 11,2011.

27. A8 2004 Insecure Storage –Open Web Application Security project [online]. Mediawiki : 10 October 2008 .
URL:http://www.owasp.org/index.php/A8_2004_Insecure_Storage.
Accessed March 11,2011.

28. Payment Card Industry Data Security Standard –Navigating the PCI DSS [online]. PCI Security Standard Council LLC: October 2010.
URL: https://www.pcisecuritystandards.org/documents/navigating_dss_v20.pdf .Accessed May 1,2011

29. Shon Harris .CISSP. USA:McGraw-Hill ;2010. p.686-687

30. Snort_Enterprise_Install [online].Patrick Harper : June 22.2007
URL:.www.internetsecurityguru.com/documents/Snort_Base_Barnyard_CentOS_5.pdf
Accessed March 22,2011.

31. Snort 2.8.6 on CentOS5.5 Installation and Configuration Guide [online].
Ken Schar :June 27,2010.
URL: https://www.snort.org/assets/145/Install_Snort_2.8.6_on_CentOS_5.5.pdf
Accessed March 22, 2011.

32. SNORT ® 2.9.0 user manual.[Online]. Sourcefire Inc: 2003-2011
URL: http://www.snort.org/assets/166/snort_manual.pdf. Accessed March 26,2011

33. Microsoft Internet Explorer HTTP 1.1 and compression Long URI Buffer Overflow Vulnerability [online]. SecurityFocus: 2010.
URL: http://www.securityfocus.com/bid/19667/info. Accessed March26, 2011

34. Snort [online]. Sourcefire: 2010. URL: http://www.snort.org/search/sid/119-19 .Accessed April 14,2011.

35. HTTP/1.1. HTTP message [online] . RFC 2616.
URL: http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html.Accessed April 14,2011.

36. Buffer Overflow-Open web application security project. [online]. Owasp foundation :2011. URL: https://www.owasp.org/index.php/Buffer_Overflow .Accessed April 14,2011.

37. Snort [online]. Sourcefire 2010. URL: http://www.snort.org/search/sid/17494?r=1 .Accessed April 14,2011.

38. What is Network Forensics [online]. SearchSecurity.com :October 28,2002.
URL: http://searchsecurity.techtarget.com/definition/network-forensics .
Accessed May 02.2011.

39. Xplico-Network forensic Analysis tool [online]. Wordpress :2007-2011.
URL: http://www.xplico.org/ .Accessed May 02,2011.

40. Wireshark [online].Wireshark . URL: http://www.wireshark.org/ .Accessed May 02.2011.

Appendices

Snort configuration file

/etc/snort/snort.conf

```
#     False Positive reports:    fp@sourcefire.com
#     Snort bugs:                bugs@snort.org
#
#     Compatible with Snort Versions:
#     VERSIONS : 2.9.0
#
#     Snort build options:
#     OPTIONS : --enable-ipv6 --enable-gre --enable-mpls --enable-targetbased
--enable-decoder-preprocessor-rules  --enable-ppm  --enable-perfprofiling  --
enable-zlib
#---------------------------------------------------

####################################################
# This file contains a sample snort configuration.
# You should take the following steps to create your own custom configura-
tion:
#
#  1) Set the network variables.
#  2) Configure the decoder
#  3) Configure the base detection engine
#  4) Configure dynamic loaded libraries
#  5) Configure preprocessors
#  6) Configure output plugins
#  7) Customize your rule set
####################################################
```

```
####################################################
# Step #1: Set the network variables.  For more information, see README.variables
####################################################

# Setup the network addresses you are protecting
var HOME_NET [192.168.20.0/24]

# Set up the external network addresses. Leave as ″an y″ in most situations
var EXTERNAL_NET !$HOME_NET

# List of DNS servers on your network
var DNS_SERVERS $HOME_NET

# List of SMTP servers on your network
var SMTP_SERVERS $HOME_NET

# List of web servers on your network
var HTTP_SERVERS $HOME_NET

# List of sql servers on your network
var SQL_SERVERS $HOME_NET

# List of telnet servers on your network
var TELNET_SERVERS $HOME_NET

# List of ssh servers on your network
var SSH_SERVERS $HOME_NET

# List of ports you run web servers on
```

```
portvar HTTP_PORTS
[80,311,591,593,901,1220,1414,2301,2381,2809,3128,3702,7777,7779,8000,8008,80
28,8080,8118,8123,8180,8243,8280,8888,9443,9999,11371]

# List of ports you want to look for SHELLCODE on.
portvar SHELLCODE_PORTS !80

# List of ports you might see  oracle attacks on
portvar ORACLE_PORTS 1024:

# List of ports you want to look for SSH connections on:
portvar SSH_PORTS 22

# other variables, these should not be modified
var AIM_SERVERS
[64.12.24.0/23,64.12.28.0/23,64.12.161.0/24,64.12.163.0/24,64.12.200.0/24,205
.188.3.0/24,205.188.5.0/24,205.188.7.0/24,205.188.9.0/24,205.188.153.0/24,205
.188.179.0/24,205.188.248.0/24]

# Path to your rules files (this can be a relative path)
# Note for Windows users:  You are advised to make this an absolute pat h,
# such as:  c:¥snort¥rules
var RULE_PATH /etc/snort/rules
var SO_RULE_PATH /etc/snort/so_rules
var PREPROC_RULE_PATH ../preproc_rules

####################################################
# Step #2: Configure the decoder.  For more information, see  README.decode
####################################################

# Stop generic decode events:
```

```
config disable_decode_alerts

# Stop Alerts on experimental TCP options
config disable_tcpopt_experimental_alerts

# Stop Alerts on obsolete TCP options
config disable_tcpopt_obsolete_alerts

# Stop Alerts on T/TCP alerts
config disable_tcpopt_ttcp_alerts

# Stop Alerts on all other TCPOption type events:
config disable_tcpopt_alerts

# Stop Alerts on invalid ip options
config disable_ipopt_alerts

# Alert if value in length field (IP, TCP, UDP) is greater th elength of the packet
# config enable_decode_oversized_alerts

# Same as above, but drop packet if in Inline mode (requires enable_decode_oversized_alerts)
# config enable_decode_oversized_ drops

# Configure IP / TCP checksum mode
config checksum_mode: all

# Configure maximum number of flowbit references.  For more information, see README.flowbits
# config flowbits_size: 64
# Configure ports to ignore
# config ignore_ports: tcp 21 6667:6671 1356
# config ignore_ports: udp 1:17 53


###################################################
# Step #3: Configure the base detection engine.  For more information, see README.decode
###################################################

# Configure PCRE match limitations
config pcre_match_limit: 1500
config pcre_match_limit_recursion: 1500

# Configure the detection engine  See the Snort Manual, Configuring Snort - Includes - Config
config detection: search-method ac-split search-optimize max-pattern-len 20

# Configure the event queue.  For more information, see README.event_queue
config event_queue: max_queue 8 log 3 order_events content_length

# Configure Inline Resets.  See README.INLINE
# config layer2resets: 00:06:76:DD:5F:E3

###################################################
# Inline latency enforcement
# For more information see README.ppm
###################################################

# Per Packet latency configuration
#config ppm: max-pkt-time 250, ¥
```

```
#   fastpath-expensive-packets, ¥
#   pkt-log

# Per Rule latency configuration
#config ppm: max-rule-time 200, ¥
#   threshold 3, ¥
#   suspend-expensive-rules, ¥
#   suspend-timeout 20, ¥
#   rule-log alert

####################################################
# Step #4: Configure dynamic loaded libraries.
# For more information, see Snort Manual, Configuring Snort  - Dynamic Modules
####################################################

# path to dynamic preprocessor libraries
dynamicpreprocessor directory /usr/local/lib/snort_dynamicpreprocessor

# path to base preprocessor engine
dynamicengine /usr/local/lib/snort_dynamicengine/libsf_engine.so

# path to dynamic rules libraries
# dynamicdetection directory /usr/local/lib/snort_dynamicrules

####################################################
# Step #5: Configure preprocessors
# For more information, see the Snort Manual, Configuring Snort - Preprocessors
####################################################

# Target-based IP defragmentation.  For m ore inforation, see README.frag3
```

```
preprocessor frag3_global: max_frags 65536
preprocessor frag3_engine: policy windows detect_anomalies overlap_limit 10
min_fragment_length 100 timeout 180

# Target-Based stateful inspection/stream reassembly.  For more inforation,
see README.stream5
preprocessor stream5_global: max_tcp 8192, track_tcp yes, track_udp yes,
track_icmp no
preprocessor stream5_tcp: policy windows, detect_anomalies, require_3whs 180,
¥
   overlap_limit 10, small_segments 3 bytes 150, timeou t 180, ¥
    ports client 21 22 23 25 42 53 79 109 110 111 113 119 135 136 137 139 143
¥
        161 445 513 514 587 593 691 1433 1521 2100 3306 6665 6666 6667 6668
6669 ¥
        7000 32770 32771 32772 32773 32774 32775 32776 32777 32778 32779,  ¥
    ports both 80 311 443 465 563 591 593 636 901 989 992 993 994 995 1220
1414 2301 2381 2809 3128 3702 6907 7702 7777 7779  ¥
        7801 7900 7901 7902 7903 7904 7905 7906 7908 7909 7910 7911 7912 7913
7914 7915 7916 ¥
        7917 7918 7919 7920 8000 8008 8028 8080 8118 8123 8180 8243 8280 8888
9443 9999 11371
preprocessor stream5_udp: timeout 180

# performance statistics.  For more information, see the Snort Manual, Con-
figuring Snort - Preprocessors - Performance Monitor
# preprocessor perfmonitor: t ime 300 file /var/snort/snort.stats pktcnt 10000

# HTTP normalization and anomaly detection.  For more information, see RE-
ADME.http_inspect
preprocessor http_inspect: global iis_unicode_map unicode.map 1252
preprocessor http_inspect_server: server default ¥
    chunk_length 500000 ¥
    server_flow_depth 0 ¥
    client_flow_depth 0 ¥
    post_depth 65495 ¥
        oversize_dir_length 500 ¥
    max_header_length 750 ¥
    max_headers 100 ¥
    ports { 80 311 591 593 901 1220 1414 2301 2381 2809 3128 3702 7777 7779 8000 8008 8028 8080 8118 8123 8180 8243 8280 8888 9443 9999 11371 } ¥
    non_rfc_char { 0x00 0x01 0x02 0x03 0x04 0x05 0x06 0x07 } ¥
    enable_cookie ¥
    extended_response_inspection ¥
    #inspect_gzip ¥
    apache_whitespace no ¥
    ascii no ¥
    bare_byte no ¥
        directory no ¥
        double_decode no ¥
        iis_backslash no ¥
        iis_delimiter no ¥
        iis_unicode no ¥
        multi_slash no ¥
        non_strict ¥
        u_encode yes ¥
        webroot no

# ONC-RPC normalization and anomaly detection.  For more information, see the Snort Manual, Configuring Snort - Preprocessors - RPC Decode
```

```
preprocessor rpc_decode: 111 32770 32771 32772 32773 32774 32775 32776 32777
32778 32779 no_alert_multiple_requests no_alert_large_fragments
no_alert_incomplete

# Back Orifice detection.
preprocessor bo

# FTP / Telnet normalization and anomaly detection.  For more information,
see README.ftptelnet
preprocessor ftp_telnet: global inspection_type stateful  encrypted_traffic no
preprocessor ftp_telnet_protocol: telnet ¥
    ayt_attack_thresh 20 ¥
    normalize ports { 23 } ¥
    detect_anomalies
preprocessor ftp_telnet_protocol: ftp server default ¥
    def_max_param_len 100 ¥
    ports { 21 2100 3535 } ¥
    telnet_cmds yes ¥
    ignore_telnet_erase_cmds yes ¥
    ftp_cmds { ABOR ACCT ADAT ALLO APPE AUTH CCC CDUP }  ¥
    ftp_cmds { CEL CLNT CMD CONF CWD DELE ENC EPRT }  ¥
    ftp_cmds { EPSV ESTA ESTP FEAT HELP LANG LIST LPRT }  ¥
    ftp_cmds { LPSV MACB MAIL MDTM MIC MKD MLSD MLST }  ¥
    ftp_cmds { MODE NLST NOOP OPTS PASS PASV PBSZ PORT }  ¥
    ftp_cmds { PROT PWD QUIT REIN REST RETR RMD RNFR }  ¥
    ftp_cmds { RNTO SDUP SITE SIZE SMNT STAT STOR STOU }  ¥
    ftp_cmds { STRU SYST TEST TYPE U SER XCUP XCRC XCWD } ¥
    ftp_cmds { XMAS XMD5 XMKD XPWD XRCP XRMD XRSQ XSEM }  ¥
    ftp_cmds { XSEN XSHA1 XSHA256 } ¥
    alt_max_param_len 0 { ABOR CCC CDUP ESTA FEAT LPSV NOOP PASV PWD QUIT
REIN STOU SYST XCUP XPWD } ¥
```

```
    alt_max_param_len 200 { ALLO APPE CMD HELP NLST RETR RNFR STOR STOU XMKD
} ¥
    alt_max_param_len 256 { CWD RNTO }  ¥
    alt_max_param_len 400 { PORT }  ¥
    alt_max_param_len 512 { SIZE }  ¥
    chk_str_fmt { ACCT ADAT ALLO APPE AUTH CEL CLNT CMD }  ¥
    chk_str_fmt { CONF C WD DELE ENC EPRT EPSV ESTP HELP }  ¥
    chk_str_fmt { LANG LIST LPRT MACB MAIL MDTM MIC MKD }  ¥
    chk_str_fmt { MLSD MLST MODE NLST OPTS PASS PBSZ PORT }  ¥
    chk_str_fmt { PROT REST RETR RMD RNFR RNTO SDUP SITE }  ¥
    chk_str_fmt { SIZE SMNT STAT  STOR STRU TEST TYPE USER }  ¥
    chk_str_fmt { XCRC XCWD XMAS XMD5 XMKD XRCP XRMD XRSQ }  ¥
    chk_str_fmt { XSEM XSEN XSHA1 XSHA256 }  ¥
    cmd_validity ALLO < int [ char R int ] >  ¥
    cmd_validity EPSV < { char 12|string } >  ¥
    cmd_validity  MACB < string > ¥
    cmd_validity MDTM < [ date nnnnnnnnnnnnnn[.n[n[n]]] ] string >  ¥
    cmd_validity MODE < char ASBCZ >  ¥
    cmd_validity PORT < host_port >  ¥
    cmd_validity PROT < char CSEP >  ¥
    cmd_validity STRU < char FRPO [ string ] >  ¥
    cmd_validity TYPE < { char AE [ char NTC ] | char I | char L [ number ] }
>
preprocessor ftp_telnet_protocol: ftp client default  ¥
    max_resp_len 256 ¥
    bounce yes ¥
    ignore_telnet_erase_cmds yes  ¥
    telnet_cmds yes
```

```
# SMTP normalization and anomaly detection.  For more information, see README.SMTP
preprocessor smtp: ports { 25 465 587 691 } ¥
    inspection_type stateful ¥
    normalize cmds ¥
    normalize_cmds { ATRN AUTH BDAT CHUNKING DATA DEBUG EHLO EMAL ESAM ESND ESOM ETRN EVFY } ¥
    normalize_cmds { EXPN HELO HELP IDENT MAIL NOOP ONEX QUEU QUIT RCPT RSET SAML SEND SOML } ¥
    normalize_cmds { STARTTLS TICK TIME TURN TURNME VERB VRFY X-ADAT X-DRCP X-ERCP X-EXCH50 } ¥
    normalize_cmds { X-EXPS X-LINK2STATE XADR XAUTH XCIR XEXCH50 XGEN XLICENSE XQUE XSTA XTRN XUSR } ¥
    max_command_line_len 512 ¥
    max_header_line_len 1000 ¥
    max_response_line_len 512 ¥
    alt_max_command_line_len 260 { MAIL } ¥
    alt_max_command_line_len 300 { RCPT } ¥
    alt_max_command_line_len 500 { HELP HELO ETRN EHLO } ¥
    alt_max_command_line_len 255 { EXPN VRFY ATRN SIZE BDAT DEBUG EMAL ESAM ESND ESOM EVFY IDENT NOOP RSET } ¥
    alt_max_command_line_len 246 { SEND SAML SOML AUTH TURN ETRN DATA RSET QUIT ONEX QUEU STARTTLS TICK TIME TURNME VERB X-EXPS X-LINK2STATE XADR XAUTH XCIR XEXCH50 XGEN XLICENSE XQUE XSTA XTRN XUSR } ¥
    valid_cmds { ATRN AUTH BDAT CHUNKING DATA DEBUG EHLO EMAL ESAM ESND ESOM ETRN EVFY } ¥
    valid_cmds { EXPN HELO HELP IDENT MAIL NOOP ONEX QUEU QUIT RCPT RSET SAML SEND SOML } ¥
    valid_cmds { STARTTLS TICK TIME TURN TURNME VERB VRFY X-ADAT X-DRCP X-ERCP X-EXCH50 } ¥
    valid_cmds { X-EXPS X-LINK2STATE XADR XAUTH XCIR XEXCH50 XGEN XLICENSE
XQUE XSTA XTRN XUSR } ¥
    xlink2state { enabled }

# Portscan detection.  For more information, see README.sfportscan
# preprocessor sfportscan: proto  { all } memcap { 10000000 } sense_level {
low }

# ARP spoof detection.  For more information, see the Snort Manual - Config-
uring Snort - Preprocessors - ARP Spoof Preprocessor
# preprocessor arpspoof
# preprocessor arpspoof_detect_host: 192.168.40.1 f0:0f:00:f0:0f:00

# SSH anomaly detection.  For more information, see README.ssh
preprocessor ssh: server_ports { 22 } ¥
                  autodetect ¥
                  max_client_bytes 19600 ¥
                  max_encrypted_packets 20 ¥
                  max_server_version_len 100 ¥
                  enable_respoverflow enable_ssh1crc32 ¥
                  enable_srvoverflow enable_protomismatch

# SMB / DCE-RPC normalization and anomaly detection.  For more information,
see README.dcerpc2
preprocessor dcerpc2: memcap 102400, events [co ]
preprocessor dcerpc2_server: default, policy WinXP, ¥
    detect [smb [139,445], tcp 135, udp 135, rpc -over-http-server 593], ¥
    autodetect [tcp 1025:, udp 1025:, rpc -over-http-server 1025:], ¥
    smb_max_chain 3

# DNS anomaly detection.  For more information, see README.dns
```

```
preprocessor dns: ports { 53 } enable_rdata_overflow

# SSL anomaly detection and traffic bypass.  For more information, see README.ssl
preprocessor ssl: ports { 443 465 563 636 989 992 993 994 995 7801 7702 7900 7901 7902 7903 7904 7905 7906 6907 7908 7909 7910 7911 7912 7913 7914 7915 7916 7917 7918 7919 7920 }, trustservers, n oinspect_encrypted

# SDF sensitive data preprocessor.  For more information see README.sensitive_data
preprocessor sensitive_data: alert_threshold 25

###################################################
# Step #6: Configure output plugins
# For more information, see Snort Manual, Configuring Snort - Output Modules
###################################################

# unified2
# Recommended for most installs
# output unified2: filename merged.log, limit 128, nostamp,
#mpls_event_types, vlan_event_types

# Additional configuration for specific types of installs
# output alert_unified2: filename snort.alert, limit 128, nostamp
 output log_unified2: filename snort.log, limit 128, nostamp

# syslog
# output alert_syslog: LOG_AUTH LOG_ALERT

# pcap
# output log_tcpdump: tcpdump.log
# database
# output database: alert,mysql, user=snort dbname=snort host=localhost pass-
word=snort
#output database: log, mysql, user=snort   dbname=snort host=localhost pass-
word=snort


# prelude
# output alert_prelude

# metadata reference data.  do not modify these lines
include classification.config
include reference.config


####################################################
# Step #7: Customize your rule set
# For more information, see Snort Manual, W riting Snort Rules
#
# NOTE: All categories are enabled in this conf file
####################################################

# site specific rules
include $RULE_PATH/local.rules

include $RULE_PATH/attack-responses.rules
include $RULE_PATH/backdoo r.rules
include $RULE_PATH/bad-traffic.rules
include $RULE_PATH/chat.rules
include $RULE_PATH/content-replace.rules
include $RULE_PATH/ddos.rules
include $RULE_PATH/dns.rules
include $RULE_PATH/dos.rules
include $RULE_PATH/exploit.rules
include $RULE_PATH/finger.rules
include $RULE_PATH/ftp.rules
include $RULE_PATH/icmp.rules
include $RULE_PATH/icmp-info.rules
include $RULE_PATH/imap.rules
include $RULE_PATH/info.rules
include $RULE_PATH/misc.rules
include $RULE_PATH/multimedia.rules
include $RULE_PATH/mysql.rules
include $RULE_PATH/netbios.rules
include $RULE_PATH/nntp.rules
include $RULE_PATH/oracle.rules
include $RULE_PATH/other-ids.rules
include $RULE_PATH/p2p.rules
include $RULE_PATH/policy.rules
include $RULE_PATH/pop2.rules
include $RULE_PATH/pop3.rules
include $RULE_PATH/rpc.rules
include $RULE_PATH/rservices.rules
# include $RULE_PATH/scada.rules
include $RULE_PATH/scan.rules
include $RULE_PATH/shellcode.rules
include $RULE_PATH/smtp.rules
include $RULE_PATH/snmp.rules
include $RULE_PATH/specific-threats.rules
include $RULE_PATH/spyware-put.rules
include $RULE_PATH/sql.rules
include $RULE_PATH/telnet.rules
include $RULE_PATH/tftp.rules
include $RULE_PATH/virus.rules
include $RULE_PATH/voip.rules
# include $RULE_P ATH/web-activex.rules
include $RULE_PATH/web -attacks.rules
include $RULE_PATH/web -cgi.rules
include $RULE_PATH/web -client.rules
include $RULE_PATH/web -coldfusion.rules
include $RULE_PATH/web -frontpage.rules
include $RULE_PATH/web -iis.rules
include $RULE_PATH/web-misc.rules
include $RULE_PATH/web -php.rules
include $RULE_PATH/x11.rules

####################################################
# Step #8： Customize your preprocessor and decoder alerts
# For more information， see README.decoder_preproc_ru les
####################################################

# decoder and preprocessor event rules
# include $PREPROC_RULE_PATH/preprocessor.rules
# include $PREPROC_RULE_PATH/decoder.rules
# include $PREPROC_RULE_PATH/sensitive -data.rules

####################################################
# Step #9： Customize your Shared Object Snort Rules
# For more information， see http://vrt-sourcefire.blogspot.com/2009/01/using-
vrt-certified-shared-object-rules.html
####################################################
```

```
# dynamic library rules
# include $SO_RULE_PATH/bad-traffic.rules
# include $SO_RULE_PATH/chat.rules
# include $SO_RULE_PATH/dos.rules
# include $SO_RULE_PATH/exploit.rules
# include $SO_RULE_PATH/icmp.rules
# include $SO_RULE_PATH/imap.rules
# include $SO_RULE_PATH/misc.rules
# include $SO_RULE_PATH/multimedia.rules
# include $SO_RULE_PATH/netbios.rules
# include $SO_RULE_PATH/nntp.rules
# include $SO_RULE_PATH/p2p.rules
# include $SO_RULE_PATH/smtp.rules
# include $SO_RULE_PATH/sql.rules
# include $SO_RULE_PATH/web-activex.rules
# include $SO_RULE_PATH/web-client.rules
# include $SO_RULE_PATH/web-iis.rules
# include $SO_RULE_PATH/web-misc.rules

# Event thresholding or suppression commands. See threshold.conf
include threshold.conf
[root@metropolia2 snortinstall2]#
```

NIC in the PIG

```
 cd /etc/sysconfig/network-scripts/
```

Sniffing Interface

```
[root@metropolia2 network-scripts]# cat ifcfg-eth0
```

```
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth0
BOOTPROTO=none
HWADDR=00:50:56:8D:00:00
IPV6INIT=yes
IPV6_AUTOCONF=yes
ONBOOT=yes
DHCP_HOSTNAME=metropolia2.cloudthesis.local
IPADDR=192.168.20.12
NETMASK=255.255.255.0
GATEWAY=192.168.20.254
TYPE=Ethernet
PEERDNS=yes
USERCTL=no
```

Management interface:

```
[root@metropolia2 network-scripts]# cat ifcfg-eth1
# Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
DEVICE=eth1
BOOTPROTO=none
HWADDR=00:50:56:8D:00:01
ONBOOT=yes
DHCP_HOSTNAME=metropolia2.cloudthesis.local
IPADDR=192.168.20.13
NETMASK=255.255.255.254
GATEWAY=192.168.20.254
TYPE=Ethernet
USERCTL=no
IPV6INIT=no
```

```
PEERDNS=yes
```

Snort alert file

```
[**] [119:19:1] (http_inspect) LONG HEADER [**]
[Priority: 3]
11/23-15:06:29.934392 192.168.20.12:38109 -> 216.239.116.154:80
TCP TTL:64 TOS:0x0 ID:40875 IpLe n:20 DgmLen:983 DF
***AP*** Seq: 0x480CE684  Ack: 0x84E7D320  Win: 0x16D0  TcpLen: 20

[**] [119:19:1] (http_inspect) LONG HEADER [**]
 [Priority: 3]
11/23-15:06:30.180363 192.168.20.12:37463 -> 216.239.116.48:80
TCP TTL:64 TOS:0x0 ID:55307 IpLen:20 D gmLen:1325 DF
***AP*** Seq: 0x47B6A21E  Ack: 0xF7F6ABF3  Win: 0x16D0  TcpLen: 20

[**] [119:19:1] (http_inspect) LONG HEADER [**]
[Priority: 3]
11/23-15:06:30.400652 192.168.20.12:47535 -> 193.166.4.72:80
TCP TTL:64 TOS:0x0 ID:55843 IpLen:20 DgmLen:97 7 DF
***AP*** Seq: 0x47D9A1B8  Ack: 0xAA67A611  Win: 0x2E  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1861385114 394076487

[**] [119:19:1] (http_inspect) LONG HEADER [**]
[Priority: 3]
11/23-15:06:30.454676 192.168.20.12:37465 -> 216.239.116.48:80
TCP TTL:64 TOS:0x0 ID:48485 IpLen:20 DgmLen:1325 DF
***AP*** Seq: 0x4887C642  Ack: 0x362249EE  Win: 0x16D0  TcpLen: 20

[**] [119:19:1] (http_inspect) LONG HEADER [**]
[Priority: 3]
```

```
11/23-15:06:30.463153 192.168.20.12:37466 -> 216.239.116.48:80
TCP TTL:64 TOS:0x0 ID:32431 IpLen:20 DgmLen:1327 DF
***AP*** Seq: 0x47FA0586  Ack: 0xC744281A  Win: 0x16D0  TcpLen: 20

 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSFER-ENCODING IN HTTP RESPONSE [**]
[Priority: 3]
11/23-15:06:30.363053 216.239.116.48:80 -> 192.168.20.12:37463
TCP TTL:64 TOS:0x0 ID:55310 IpLen:20 DgmLen:612 DF
***A**** Seq: 0xF7F6ABF3  Ack: 0x47B6AC39  Win: 0x1D50  TcpLen: 20

 [**] [119:19:1] (http_inspect) LONG HEADER [**]
[Priority: 3]
11/23-15:06:31.289457 192.168.20.12:5492 1 -> 193.166.4.71:80
TCP TTL:64 TOS:0x0 ID:871 IpLen:20 DgmLen:999 DF
***AP*** Seq: 0x47BB7D9C  Ack: 0xAD59B8C9  Win: 0x2E  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1861385987 471523742

[**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSFER-ENCODING IN HTTP RESPONSE [**]
[Priority: 3]
11/23-15:06:30.636144 216.239.116.48:80 -> 192.168.20.12:37465
TCP TTL:64 TOS:0x0 ID:48488 IpLen:20 DgmLen:612 DF
***A**** Seq: 0x362249EE  Ack: 0x4887D05D  Win: 0x1D50  TcpLen: 20

[**] [1:17494:3] WEB-CLIENT Microsoft Internet Explorer Long URL Buffer Overflow attempt [**]
[Classification: Attempted User Privilege Gain] [Priority: 1]
11/23-15:06:31.400618 89.207.18.81:80 -> 192.168.20.12:37439
TCP TTL:51 TOS:0x0 ID:46490 IpLen:20 DgmLen:835 DF
***AP*** Seq: 0x71C2285  Ack: 0x484DFFB3  Win: 0x7FFF  TcpLen: 20
```

[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=2006-3869][Xref => http://www.securityfocus.com/bid/19667]