ACKNOWLEDGEMENT

I would like to express my gratitude to Prof. Abhishek Tripathi (IIT Varanasi) for his valuable guidance. His rational feedbacks are responsible for the inclusion of innovative developments in the project and its timely completion. I am grateful to the professor because, I have observed and learnt many qualities like perseverance and passion for the subject from him.

TABLE OF CONTENTS:

ABSTRACT

In the cases where proxy servers are deployed in explicit mode, most of the software applications (regardless of the platform) do not work. This is because the apps have been designed under the assumption that there is an uninterrupted path out to the Internet, thus if the device is configured to use an explicit proxy, either the app does not use this global setting, or the app itself has no provision to be configured to use an explicit proxy. Therefore, the aim is to provide a **Proxy less Internet access**, which means that there should be no proxy configuration required on the client side. Transparent proxy deployments typically resolve these issues, as far as the application or devices are concerned, it has an uninterrupted connection out to the internet, and therefore work as intended. But, the usual authentication procedure (challenging the user for valid credentials before they can use the proxy) does not work for transparent proxy deployments. This is because the client does not know about the presence of a proxy server. Therefore, a captive portal may be used to validate a user. Thus, the proposed solution involves using a transparent proxy in conjunction with a captive portal to get the applications working.

MOTIVATION FOR THE PROJECT

Today, the software applications are used to satisfy a wide array of requirements. They have become an integral part of our lives. Various software development companies have come up with great software applications providing novel services. A student uses mobile applications to gain extra knowledge on the subjects, keep abreast with latest news and announcements related to his career. A musician uses mobile phone to track down the latest trend in music industry. Mobile application technology over the years has helped increase the client base by increasing the number of solutions and services, thereby improving the profitability of business organizations. But in certain institutions/ offices these apps don’t work due to the problems inherent to the explicit proxy deployment.

INTRODUCTION AND OVERVIEW

- The purpose of the project is to provide a "Proxy less Internet access". That is, the user should be able to access the Internet without configuring any proxy settings in the web browser.
  This, however, does not mean that there can't be any proxy server. In fact, the proposed solution makes use of a transparent proxy server (which does not require any proxy configuration on the web browser). The proxy configuration is being avoided because some apps are not designed to use these settings and therefore, fail to work.

  We are using pfSense firewall and configuring squid3 (an installed package) to act as a transparent proxy server.

- The task of user authentication is now accomplished using a captive portal. pfSense provides an integrated platform which also allows the configuration of a captive portal.

- All the user logs can be seen and analyzed using LightSquid (an installed package for log parsing). The LightSquid generates reports on the bases of date, IP address and browsed websites.

NETWORK SECURITY SYSTEMS:

1. FIREWALL

A Firewall is software or hardware-based network security system that controls the incoming and outgoing network traffic by analyzing the data packets and determining whether they should be allowed through or not, based on applied rule set. Firewalls can be defined in many ways according to your level of understanding. A firewall establishes a barrier between a trusted, secure internal network and another network (e.g., the Internet) that is not assumed to be secure and trusted.

Types of firewalls: -

- Stateless Firewall (Packet Filter):

Packet filters act by inspecting the "packets" which are transferred between computers on the Internet. If a packet matches the packet filter's set of filtering rules, the packet filter will drop (silently discard) the packet or reject it (discard it, and send "error responses" to the source).This type of packet filtering pays no attention to whether a packet is part of an existing stream of traffic (i.e. it stores no information on connection "state"). Instead, it filters each packet based only on information contained in the packet itself (most commonly using a combination of the packet's source and destination address, its protocol, and, for TCP and UDP traffic, the port number).

E.g., deny TCP port 23 (telnet). This Firewall can allow/disallow many types of services and destinations.

- Stateful Firewall:

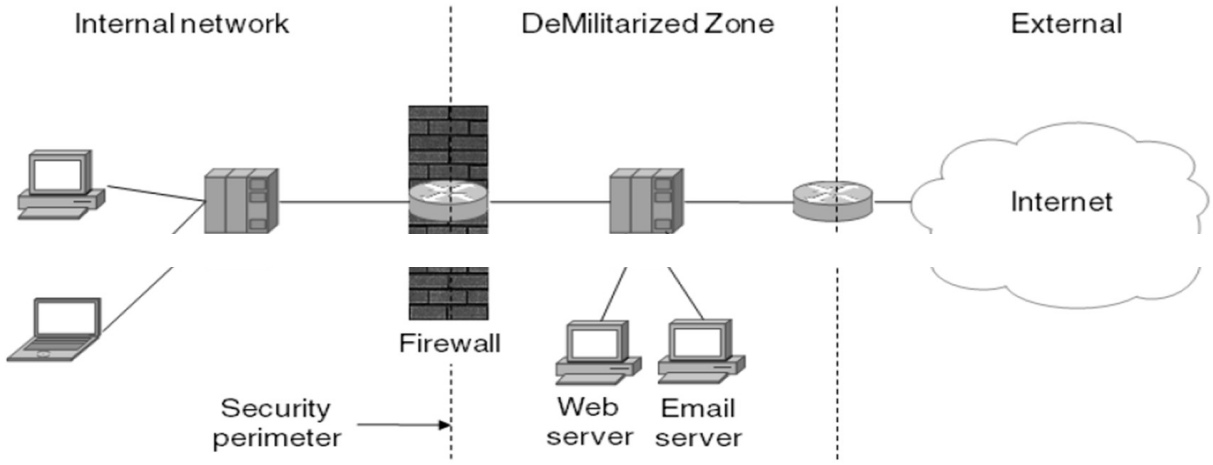
It is a step up from the stateless Firewall. It implements stateful packet filter rules that track packet exchanges.

E.g., accept incoming TCP packets after internal host connects.

- Application layer Firewall: –It is another improvement which implements rules based on app usage and content.

E.g., inspect content for viruses– Tries to look beyond packets by emulating higher layers, e.g., by reassembling application messages

A Firewall is placed around internal/external boundary. The classic setup includes DMZ (DeMilitarized Zone) to put busy internet hosts on the outside for better separation. The DMZ includes the sensitive servers which should also be protected from the malicious packets as if these servers get infected the hosts attached to these servers will also get infected.



Disadvantages of a traditional Firewall:

Traditional stateful inspection firewalls have effectively become obsolete because of two significant limitations. First, they don't inspect the data payload of network packets. Second, while more and more network traffic uses Web protocols including legitimate business applications, non-business applications and attacks,

traditional firewalls don't have the fine-grained intelligence to distinguish one kind of Web traffic from another and enforce business policies.

2. UNIFIED THREAT MANAGEMENT (UTM) SYSTEMS

Unified Threat Management (UTM) systems perform multiple security functions such as intrusion prevention and deep packet inspection, to detect malware or exploits in network traffic, in a single platform.

However, there are drawbacks to this approach. Adding more devices can also add more latency, as packets are passed from one appliance to the next. New devices also add operational overhead because each needs to be monitored and managed separately. As for UTMs, they tend to use separate internal engines to perform individual security functions. This means a packet may be examined several times by different engines to determine whether it should be allowed into the network. That round-robin approach adds latency, which may affect network performance.

In general, with a UTM, security administrators must work to find an acceptable balance between performance and protection.

3. NEXT GENERATION FIREWALLS (NGFWs)

This category of product attempts to address the traffic inspection and application awareness drawbacks of stateful inspection firewalls, without hampering performance.

The most significant difference between NGFWs and traditional firewalls is that NGFWs are application-aware; they use a variety of techniques to identify applications, including Web applications. Thus, instead of allowing all traffic coming in via typical Web ports, a NGFW can distinguish between specific applications and then apply policies based on business rules.

NGFWs also use deep packet inspection techniques to examine traffic for anomalies and known malware. However, these devices are optimized so that

packets need to be examined only once, rather than processed through multiple engines.

At minimum an NGFW should provide:

- Standard first-generation firewall capabilities, such as network-address translation (NAT), stateful protocol inspection (SPI) and virtual private networking (VPN)

- IPS engine

- Application awareness

- Ability to incorporate information from outside the firewall, such as directory-based policy, blacklists and whitelists.

THE GOLDEN SHIELD PROJECT

HISTORY:

The Golden Shield Project colloquially referred to as the Great Firewall of China is a censorship and surveillance project operated by the Ministry of Public Security (MPS) division of the government of China. The project was initiated in 1998 and began operations in November 2003.

Superseding the political ideologies of the Cultural Revolution, an economic reform led China towards a market economy and opened up the market for foreign investors. Nonetheless, the economic freedom, values, and political ideas of the Communist Party of China have had to be protected by "swatting flies" of other unwanted ideologies.

The Internet in China arrived in 1994, as the inevitable consequence of and supporting tool for the "socialist market economy". Gradually, while Internet availability has been increasing, the Internet has become a common communication platform and tool for trading information.

The Ministry of Public Security took initial steps to control Internet use in 1997, when it issued comprehensive regulations governing its use. The key sections, Articles 4-6, are: "Individuals are prohibited from using the Internet to: harm national security; disclose state secrets; or injure the interests of the state or society. Users are prohibited from using the Internet to create, replicate, retrieve, or transmit information that incites resistance to the PRC Constitution, laws, or administrative regulations; promotes the overthrow of the government or socialist system; undermines national unification; distorts the truth, spreads rumors, or destroys social order; or provides sexually suggestive material or encourages gambling, violence, or murder. Users are prohibited from engaging in activities that harm the security of computer information networks and from using networks or changing network resources without prior approval.

In 1998, the Communist Party of China feared that the China Democracy Party (CDP) would breed a powerful new network that the party elites might not be able to control. The CDP was immediately banned, followed by arrests and imprisonment. That same year, the Golden Shield project was started. The first part of the project lasted eight years and was completed in 2006. The second part began in 2006 and ended in 2008. On 6 December 2002, 300 people in charge of the Golden Shield project from 31 provinces and cities throughout China participated in a four-day inaugural “Comprehensive Exhibition on Chinese Information System”. At the exhibition, many western high-tech products, including Internet security, video monitoring and human face recognition were purchased. It is estimated that around 30,000-50,000 police are employed in this gigantic project.

It has been nicknamed "the Great Firewall" (a term that first appeared in a Wired magazine article in 1997) in reference to its role as a network firewall and to the ancient Great Wall of China. A major part of the project includes the ability to block content by preventing IP addresses from being routed through and consists of standard firewalls and proxy servers at the six Internet gateways. The system also selectively engages in DNS cache poisoning when particular sites are requested. The government does not appear to be systematically examining Internet content, as this appears to be technically impractical. Because of its disconnection from the larger world of IP routing protocols, the network contained within the Great Firewall has been described as "the Chinese autonomous routing domain".

During the 2008 Summer Olympics, Chinese officials told Internet providers to prepare to unblock access from certain Internet cafés, access jacks in hotel rooms and conference centers where foreigners were expected to work or stay.

PURPOSE:

In September 2002, Li Runsen, the technology director at Ministry of Public Security and member of the Golden Shield leadership, further explained this broad definition to thousands of police nationwide at a meeting in Beijing called "Information Technology for China’s Public Security".

In October 2001, Greg Walton of the International Centre for Human Rights and Democratic Development published a report; he wrote:

Old style censorship is being replaced with a massive, ubiquitous architecture of surveillance: the Golden Shield. Ultimately, the aim is to integrate a gigantic online database with an all-encompassing surveillance network – incorporating speech and face recognition, closed-circuit television, smart cards, credit records, and Internet surveillance technologies.

The empirical study by the OpenNet Initiative (collaboration between Harvard Law School, University of Toronto Citizen Lab, and Cambridge Security Program) found that China has the most sophisticated content-filtering Internet regime in the world. Compared to similar efforts in other countries, Chinese Government effectively filters content by employing multiple methods of regulation and technical controls.

BLOCKING METHODS:

Some commonly used technical methods for censoring are:

Method	Description
IP blocking	The access to a certain IP address is denied. Some large Web sites allocated additional IP addresses to circumvent the block, but later the block was extended to cover the new addresses.
DNS filtering and redirection	Doesn't resolve domain names, or returns incorrect IP addresses. A typical circumvention method is to find a domain name server that resolves domain names correctly, but domain name servers are subject to blockage as well, especially IP blocking. Another workaround is to bypass DNS if the IP address is obtainable from other sources and is not blocked. An example is typing the IP address instead of the domain name in a Web browser.

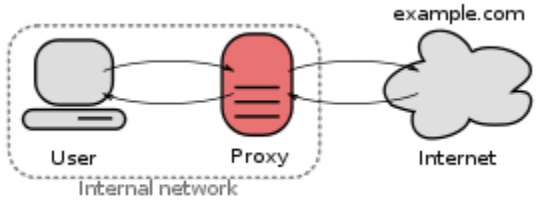
URL filtering	Scan the requested Uniform Resource Locator (URL) string for target keywords regardless of the domain name specified in the URL. This affects the Hypertext Transfer Protocol. A typical circumvention method is to use encrypted protocols such as VPN and SSL.
Packet filtering	Terminate TCP packet transmissions when a certain number of controversial keywords are detected. Typical circumvention methods are to use encrypted protocols such as VPN and SSL, to escape the HTML content, or reducing the TCP/IP stack's MTU, thus reducing the amount of text contained in a given packet.
Connection reset	If a previous TCP connection is blocked by the filter, future connection attempts from both sides will also be blocked for up to 30 minutes. Depending on the location of the block, other users or Web sites may be also blocked if the communications are routed to the location of the block. A circumvention method is to ignore the reset packet sent by the firewall

Whether a website is accessible in China or not can be found using http://www.greatfirewallofchina.org/

PROXY SERVERS:

A proxy server is a computer that acts as an intermediary between the user's computer and the Internet. It allows client computers to make indirect network connections to other network services. Client computers connect to the proxy server, requesting some resources like web pages, games, videos, mp3, e-books, any other resources which are available from various servers over Internet. As soon as getting such request, the proxy server will seek for the resources from the cache in its local hard disk. If the resources have been cached before, the proxy server will return them to the client computers. If not cached, it will connect to the relevant servers and request the resources on behalf of the client computers. Then it 'caches' resources from the remote servers, and returns subsequent requests for the same content directly. A proxy server may optionally alter the clients' requests or the servers' response for some potential purposes.

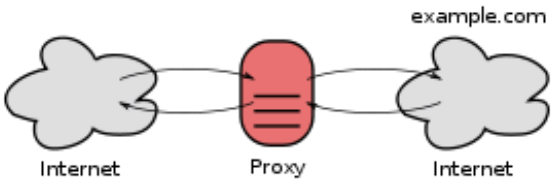
FORWARD PROXIES:



A forward proxy taking requests from an internal network and forwarding them to the Internet.

Forward proxies are proxies in which the client server names the target server to connect to. Forward proxies are able to retrieve from a wide range of sources (in most cases anywhere on the Internet).

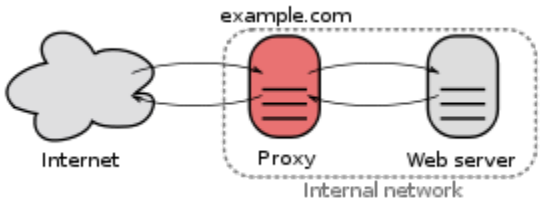
OPEN PROXIES:



An open proxy forwarding requests from and to anywhere on the Internet.

An open proxy is a forwarding proxy server that is accessible by any Internet user. An anonymous open proxy allows users to conceal their IP address while browsing the Web or using other Internet services. There are varying degrees of anonymity however, as well as a number of methods of 'tricking' the client into revealing itself regardless of the proxy being used.

REVERSE PROXIES:



A reverse proxy taking requests from the Internet and forwarding them to servers in an internal network. Those making requests connect to the proxy and may not be aware of the internal network.

A reverse proxy (or surrogate) is a proxy server that appears to clients to be an ordinary server. Requests are forwarded to one or more proxy servers which handle

the request. The response from the proxy server is returned as if it came directly from the origin server, leaving the client no knowledge of the origin servers.

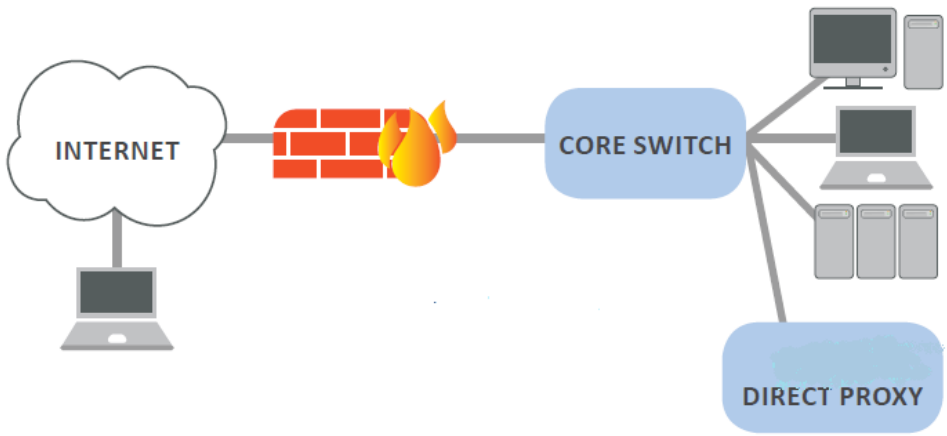
Reverse proxies are installed in the neighborhood of one or more web servers. All traffic coming from the Internet and with a destination of one of the neighborhood's web servers goes through the proxy server. The use of "reverse" originates in its counterpart "forward proxy" since the reverse proxy sits closer to the web server and serves only a restricted set of websites.

METHODS OF DEPLOYMENT OF PROXY SERVERS:

1. Explicit Proxy / Direct Proxy:

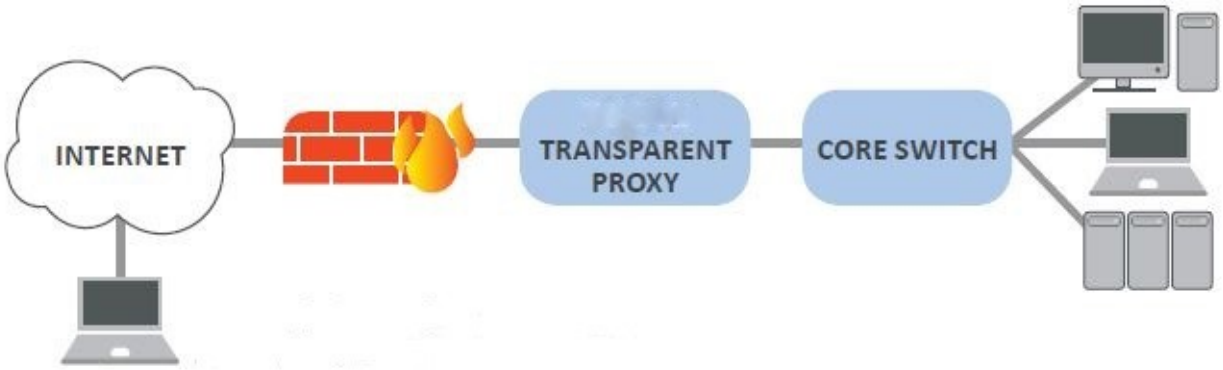
For explicit proxy deployment, individual client browsers have to be manually configured to send requests directly to the proxy. They may also be configured to download proxy configuration instructions from a Proxy Auto-Configuration (PAC) file.

Disadvantages of explicit proxy deployment include a user's ability to alter an individual client configuration and bypass the proxy. To counter this, one can configure the firewall to allow client traffic to proceed only through the proxy. The explicit proxy server may alter the client's request or the server's response. Sometimes the proxy may serve the request directly from its local storage/cache, without contacting the requested server.



2. Transparent Proxy:

In a transparent proxy deployment, the user's client software (typically a browser) is unaware that it is communicating with a proxy. Users request Internet content as usual, without any special client configuration, and the proxy serves their requests. The proxy establishes a connection with the origin server and returns requested content to the client. The requested content is returned as if it came directly from the origin server. The transparent proxy typically caches some or all of the content. A transparent proxy is normally placed in-line between the client and the Internet acting as either arouting or bridged device.



CAPTIVE PORTAL:

The captive portal technique forces an HTTP client on a network to see a special web page (usually for authentication purposes) before using the Internet normally. A captive portal turns a Web browser into an authentication device. This is done by intercepting most packets, regardless of address or port, until the user opens a browser and tries to access the web. At that time the browser is redirected to a web page which may require authentication and/or payment, or simply display an acceptable use policy and require the user to agree. Captive portals are used at many Wi-Fi hotspots, and can be used to control wired access (e.g. apartment houses, hotel rooms, business centers) as well.

Since the login page itself must be presented to the client, either that login page is locally stored in the gateway, or the web server hosting that page must be "whitelisted" via a walled garden to bypass the authentication process.

IMPLEMENTATION:

Redirection by HTTP

If an unauthenticated client requests a website, DNS is queried by the browser and the appropriate IP resolved as usual. The browser then sends an HTTP request to that IP address. This request, however, is intercepted by a firewall and forwarded to a redirect server. This redirect server responds with a regular HTTP response which contains HTTP status code 302 to redirect the client to the Captive Portal. To the client, this process is totally transparent. The client assumes that the website actually responded to the initial request and sent the redirect.

Redirection by DNS

When a client requests a website, DNS is queried by the browser. The firewall will forward all DNS requests by the unauthenticated clients to the DNS server(s) provided by DHCP. This DNS server will return the IP address of the Captive Portal page as a result of all DNS lookups.

In order to perform redirection by DNS the captive portal is using DNS poisoning to perform a man-in-the-middle attack.

VIRTUAL TEST BED SETUP

We are using Oracle VM VirtualBox for creating virtual machines for the virtual network setup.

We installed pfSense 2.1, ubuntu 14.04 and windows 8 as virtual machines in VirtualBox.

- **pfSense:**

For pfSense we enable two network adapters one set to NAT and the other to internal network (intnet).

Configure the firewall. We are not using any VLANs. The WAN interface name is set to em0 and the LAN interface name is set to em1.

- **Ubuntu (Guest Machine)**

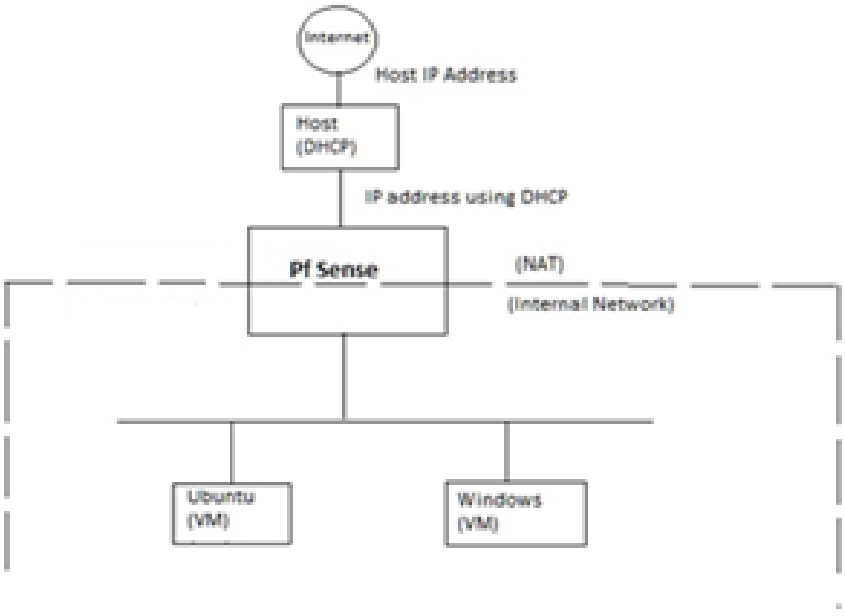
Only one network adapter is enabled and it is set to internal network (intnet).

- **Windows (Guest Machine)**

Only one network adapter is enabled and it is set to internal network (intnet).

NETWORK DIAGRAM:

The Host Machine connects directly to the internet. A DHCP Server runs on the host machine, which assigns an IP address to the pfSense's NAT interface. The other interface (internal network interface) gets a private IP address due to NAT. The two VMs (Ubuntu and Windows) in the same internal network as the second pfSense interface. These three together form the LAN.



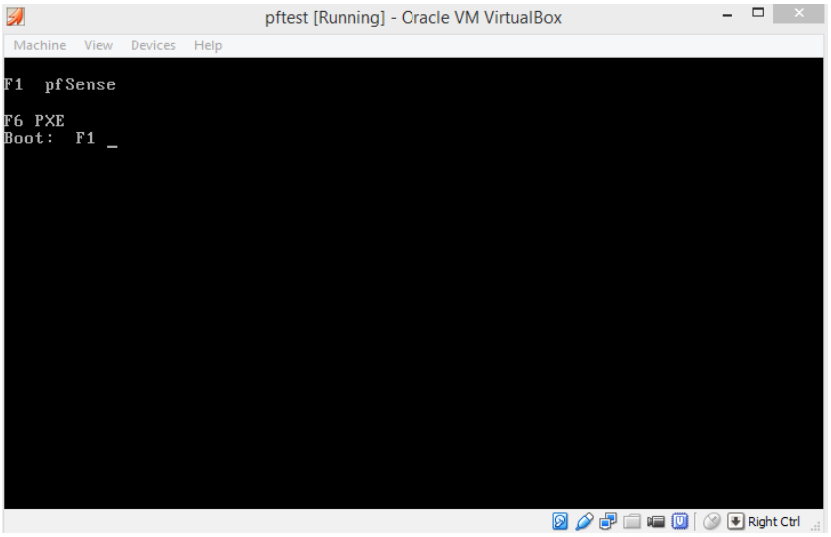
The Web GUI of the pfSense firewall can be accessed by entering the IP address of the LAN interface (default: 192.168.1.1) in the web browser's address bar.

From the GUI, the required packages can be installed. We require Squid3, Squidgaurd3 and Lightsquid for our setup.

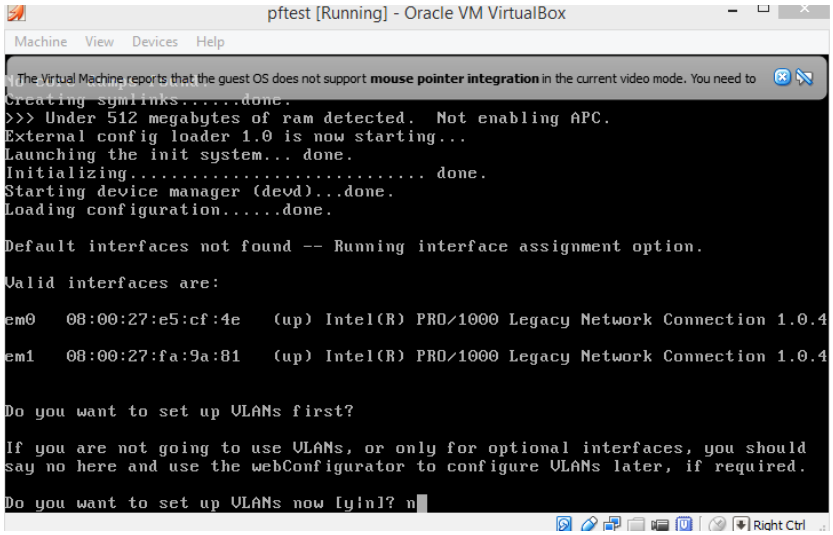
PFSENSE: OVERVIEW AND BASIC SETUP

pfSense is an open source firewall/router computer software distribution based on FreeBSD. It is installed on a computer to make a dedicated firewall/router for a network and is noted for its reliability and offering features often only found in expensive commercial firewalls. It can be configured and upgraded through a web-based interface, and requires no knowledge of the underlying FreeBSD system to manage. pfSense is commonly deployed as a perimeter firewall, router, wireless access point, DHCP server, DNS server, and as a VPN endpoint.

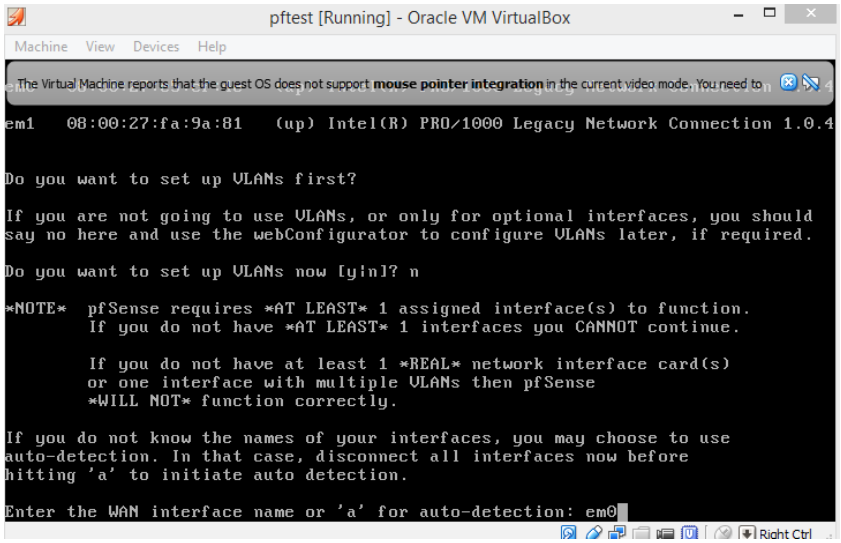
On launching the installed pfSense VM for the first time, the following screen is displayed where F1 has to be pressed for booting pfSense.



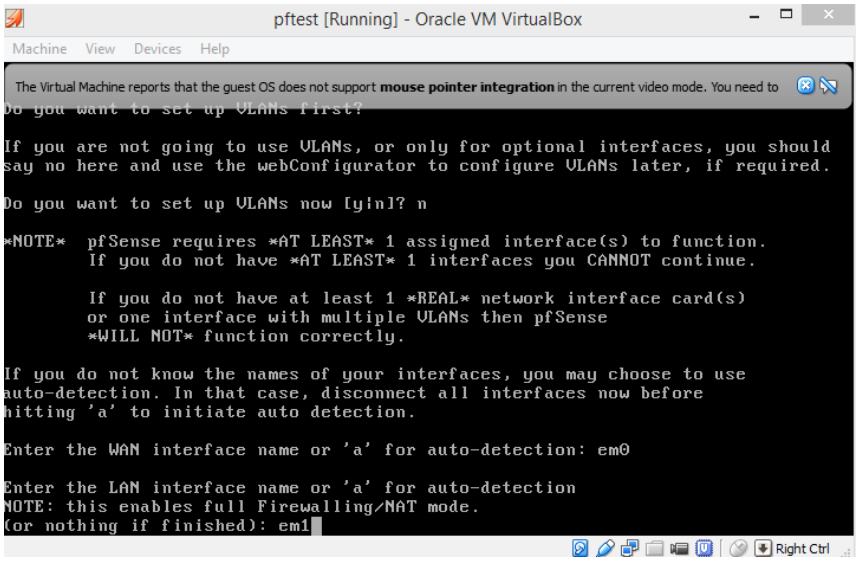
Next, we enter ‘n’ as we do not want to set up VLANs.



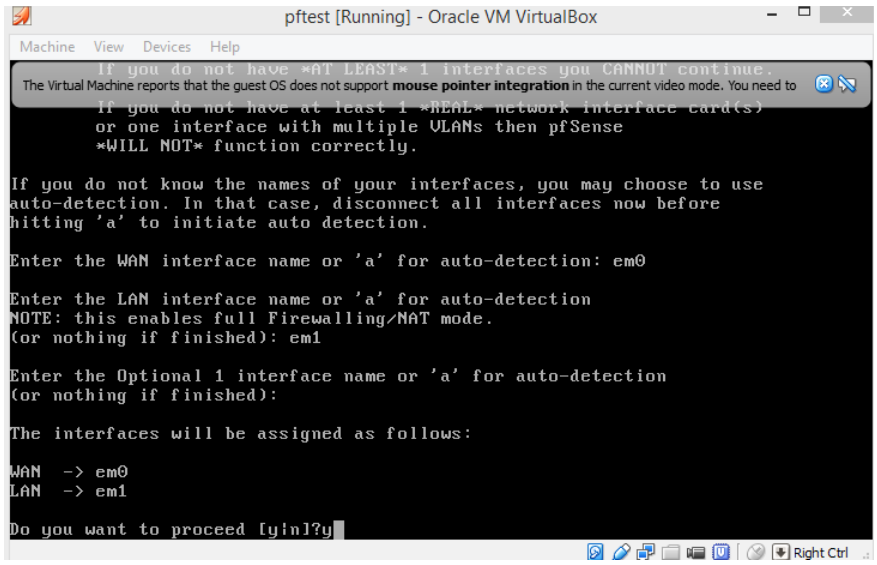
Next, we enter the WAN interface name as ‘em0’.



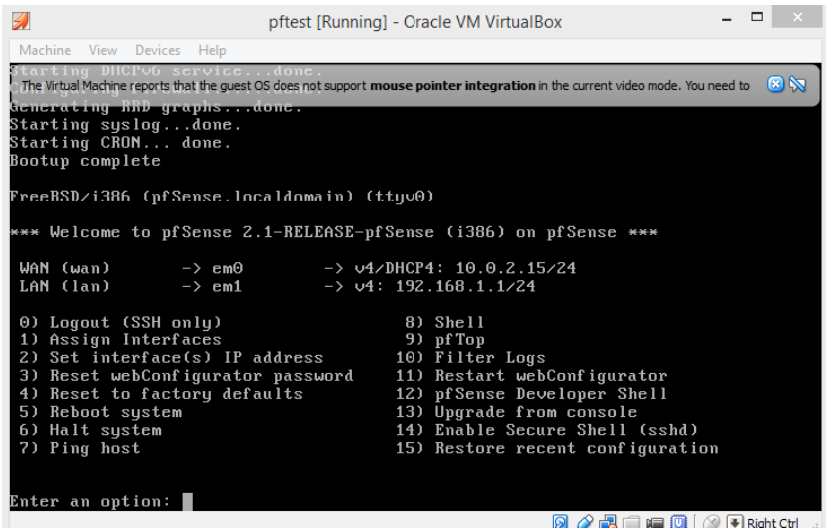
Further, the LAN interface name is entered as ‘em1’.



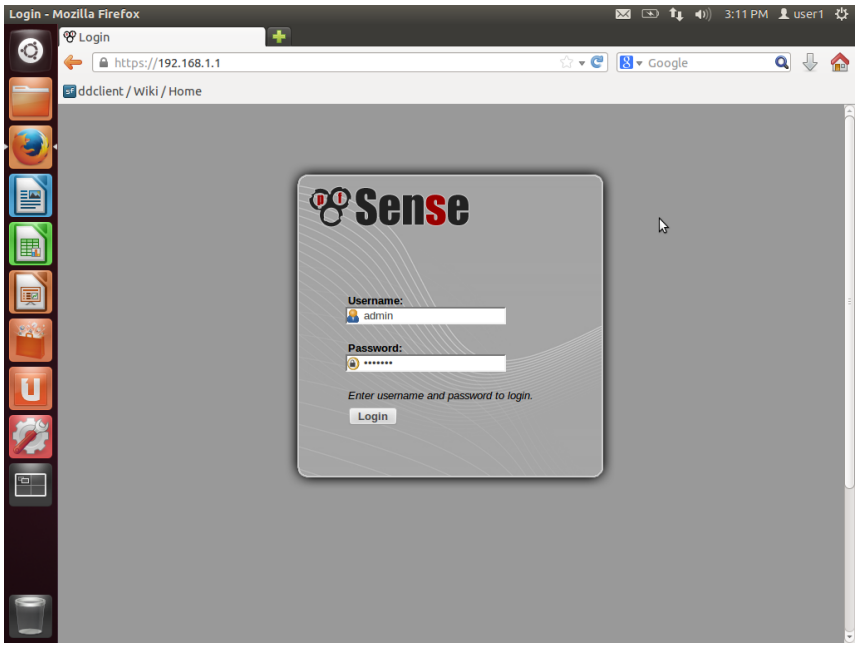
Next, we enter ‘y’ to proceed further.



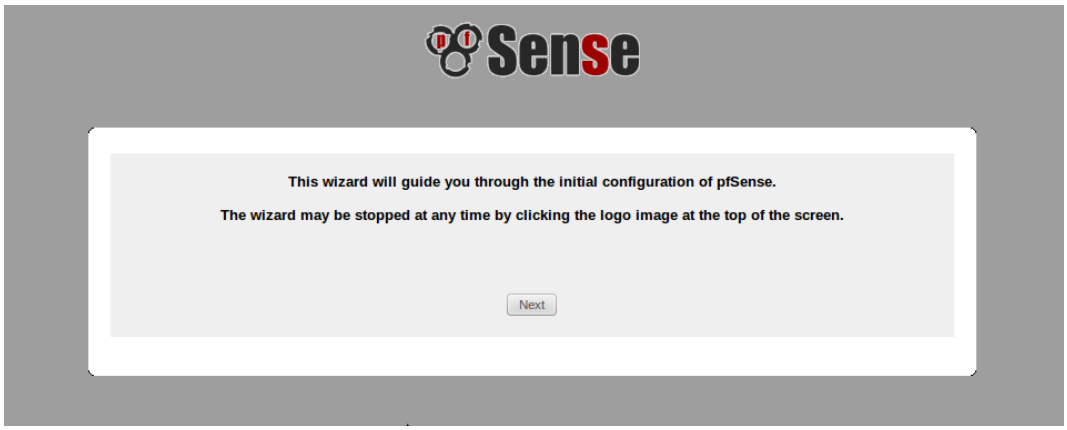
After the complete of pfsense bootup the following screen is displayed, that contains all the necessary configuration options. We will use the pfSense Web GUI for all the configurations though.



As mentioned earlier the pfSense Web Configurator (Web GUI) can be launched by typing the lan interface ip address (192.168.1.1) in the web browser's address bar. The following is the login page for the web GUI. The default **username** and **password** are 'admin' and 'pfsense' respectively. The **password** can, however, be changed later on.

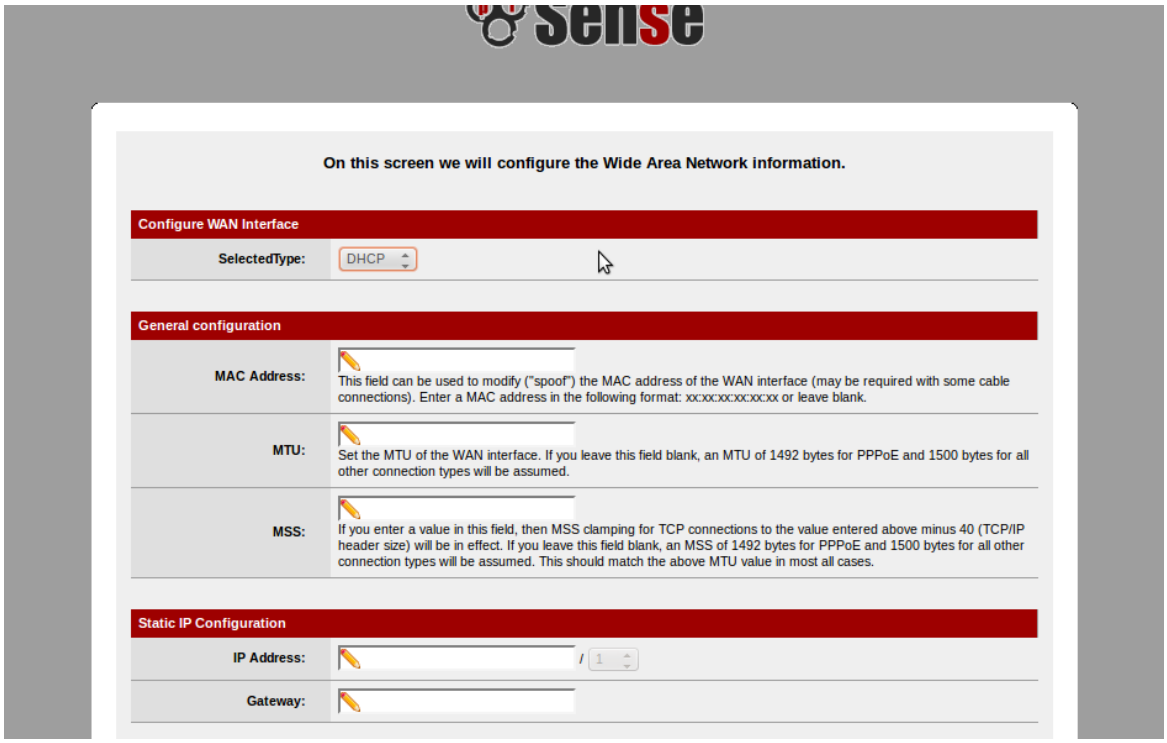


On successful first login, the user is presented with a wizard that helps complete the basic configuration.

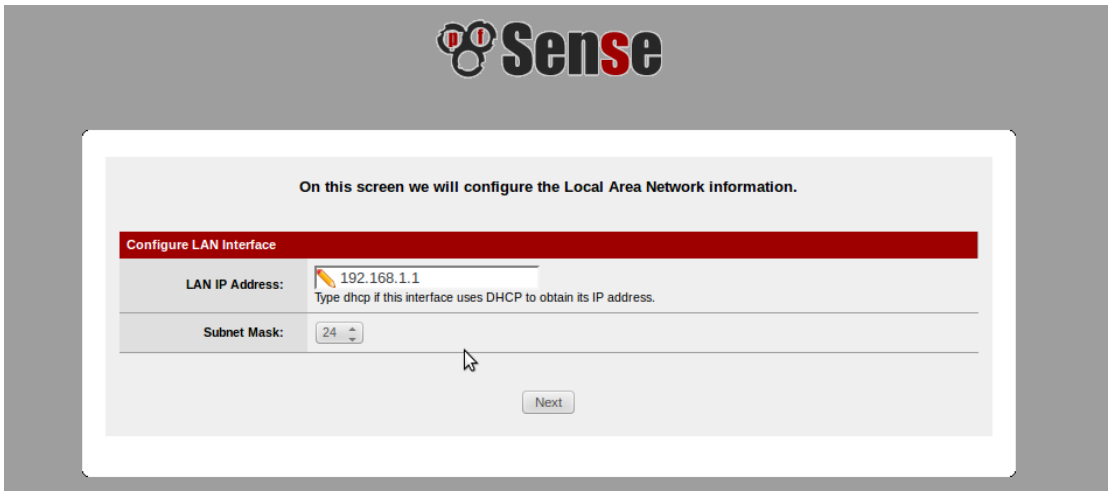


Next, the user is asked to select the timezone from a dropdown list. The Time server hostname is left default.

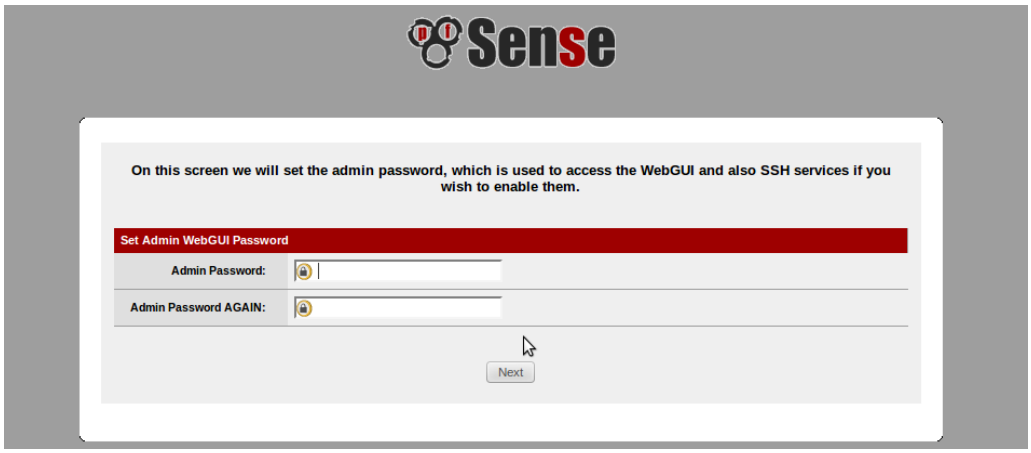
Next is the basic WAN configuration. WAN interface can be set to ‘DHCP’ or ‘Static’. In the latter case, static ip (or a subnet) has to be given as well. In our case, we are using DHCP for automatic ip address assignment. All the other fields can be left unfilled/unchanged.



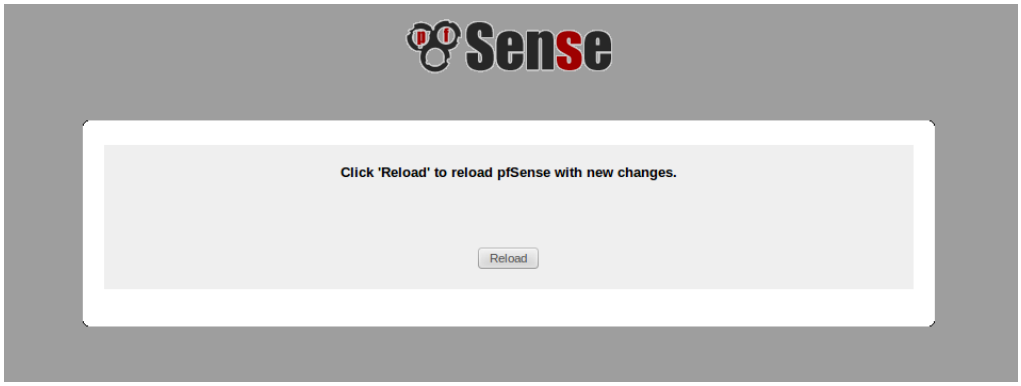
Next, the user is prompted with options for LAN setup. Here, a suitable LAN subnet can be entered by the user.



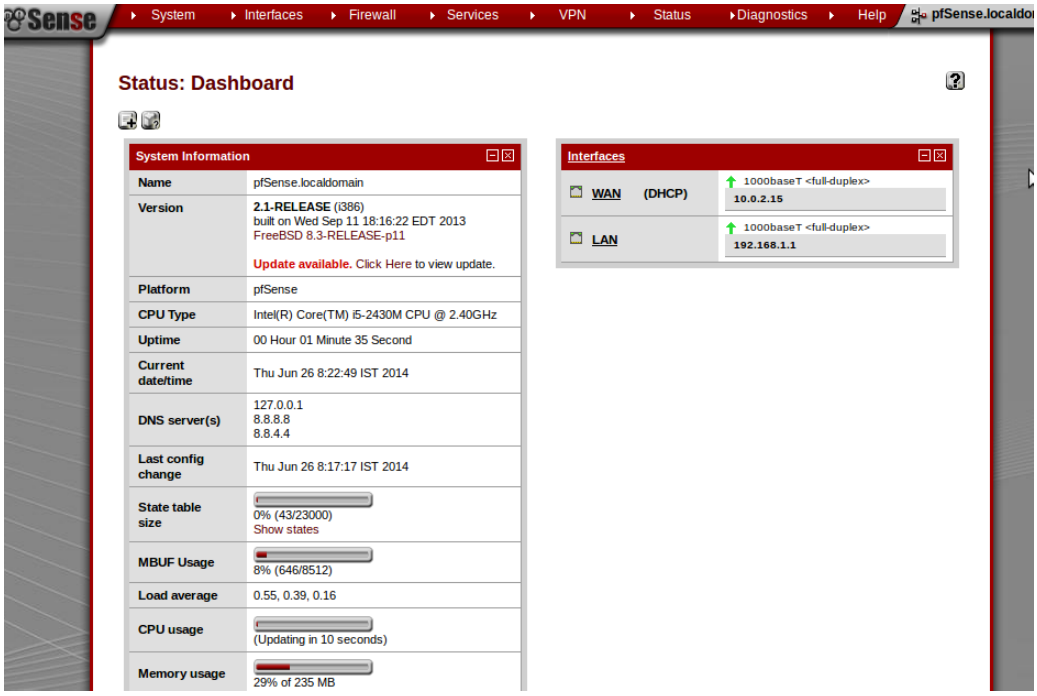
The admin password can now be changed by the user or left unchanged.



The basic configuration is now complete and pfSense has to be reloaded mandatorily.

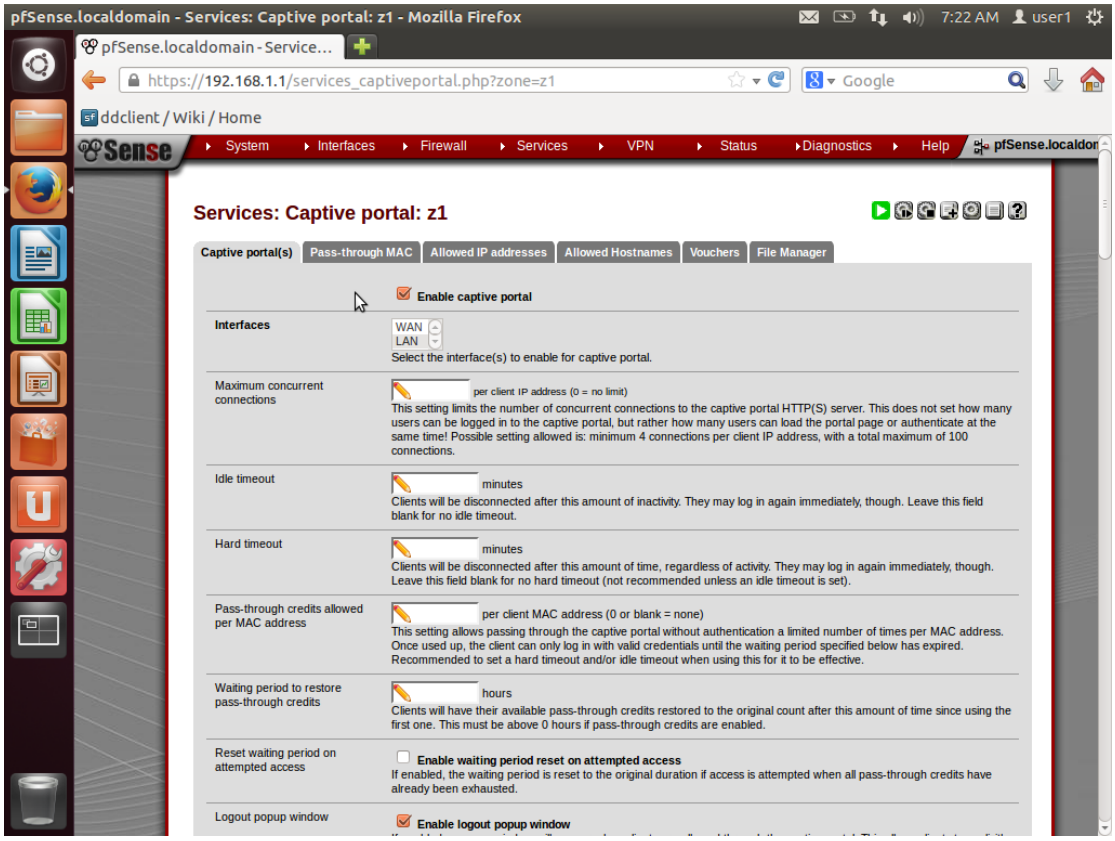


After the reload, the pfSense dashboard is presented to the user.

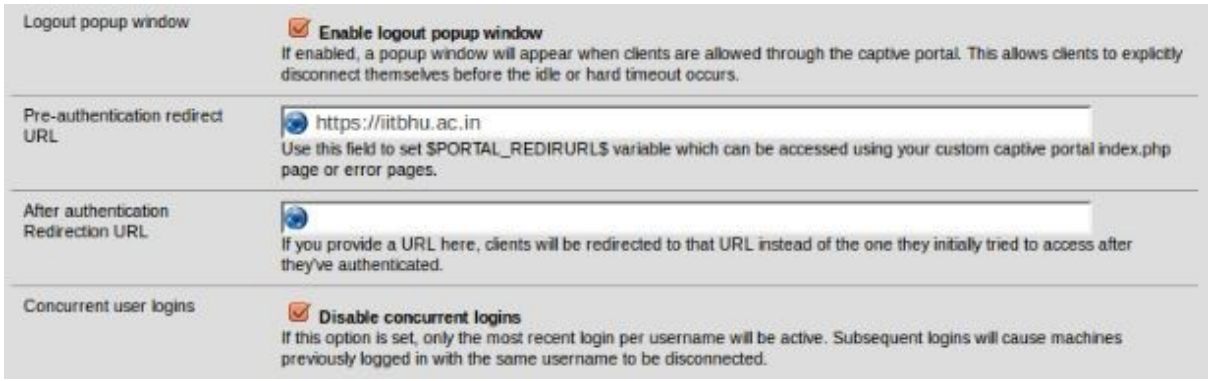

Sense
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help
pfSense.localdo
Status: Dashboard
System Information
Name
pfSense.localdomain
Version
2.1-RELEASE (i386)
built on Wed Sep 11 18:16:22 EDT 2013
FreeBSD 8.3-RELEASE-p11
Update available. Click Here to view update.
Platform
pfSense
CPU Type
Intel(R) Core(TM) i5-2430M CPU @ 2.40GHz
Uptime
00 Hour 01 Minute 35 Second
Current date/time
Thu Jun 26 8:22:49 IST 2014
DNS server(s)
127.0.0.1
8.8.8.8
8.8.4.4
Last config change
Thu Jun 26 8:17:17 IST 2014
State table size
0% (43/23000)
Show states
MBUF Usage
8% (646/8512)
Load average
0.55, 0.39, 0.16
CPU usage
(Updating in 10 seconds)
Memory usage
29% of 235 MB
Interfaces
WAN
(DHCP)
1000baseT <full-duplex>
10.0.2.15
LAN
1000baseT <full-duplex>
192.168.1.1

SETTING UP THE CAPTIVE PORTAL

To set up the captive portal in pfsense, we go to the services tab in the web GUI and select “Captive Portal”. We have to create a zone and then edit it. The options available for a created zone (z1) are as shown. The captive portal is enabled.

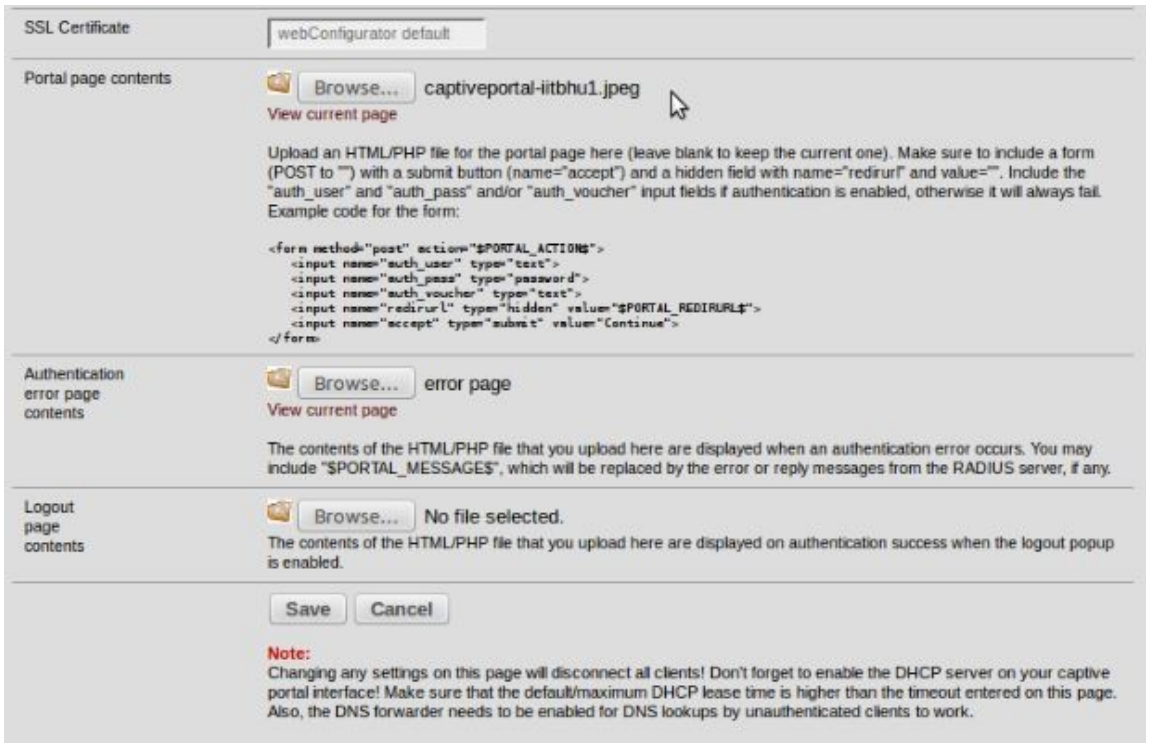


We want to have a logout window available for the user to log out of the captive portal; therefore, we enable the popup logout window option. We also do not want to allow multiple users to log in using the same username and password and use the Internet simultaneously. Additionally, we want to restrict a user from logging in more than once at a time (using a different web browser). To enforce these two restrictions we disable the concurrent login option. We have also set the variable $PORTAL_REDIURL$ to https://iitbhu.ac.in. All the users that log in will be redirected to this page after logging in.

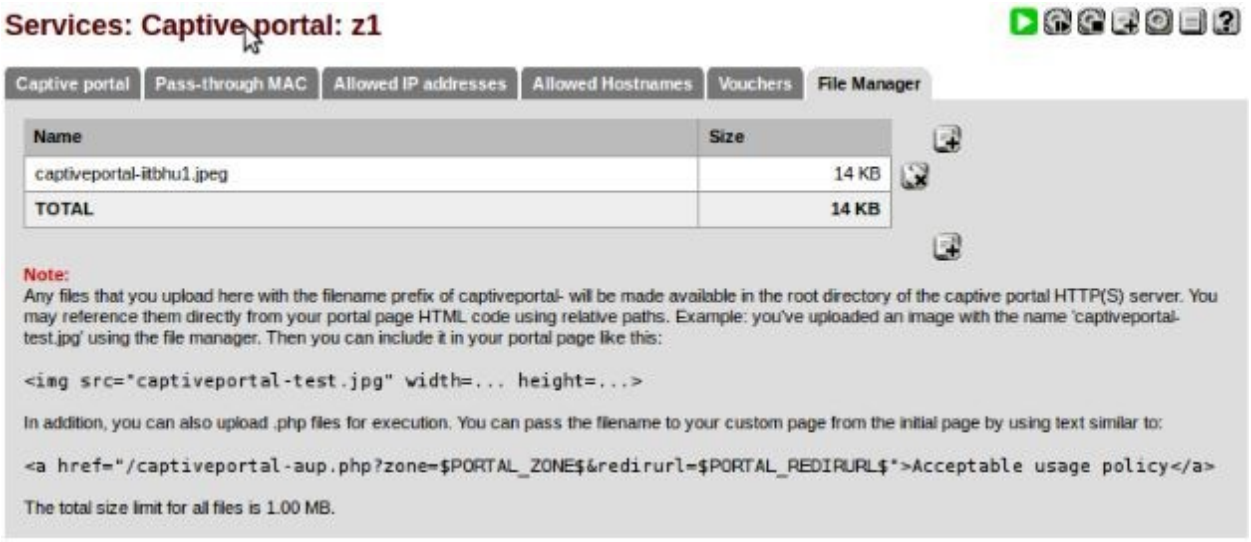


Next, we enable the 'Local User Manager' mode of authentication for the maintenance of a local database of users. Moreover, we want only those users to be able to log in, that have been given the privilege to do so, explicitly. Therefore, the next checkbox is also checked.

Now we upload our own html captive portal page and the error page that should get displayed in the course of action. All the settings are now saved.



Now under the ‘File Manager’ tab, we upload any images that are being used in the captive portal page / error page. It is saved to the root directory of the captive portal server and it is from this directory only that the image is retrieved for the portal page.



CAPTIVE PORTAL PAGE HTML CODE:

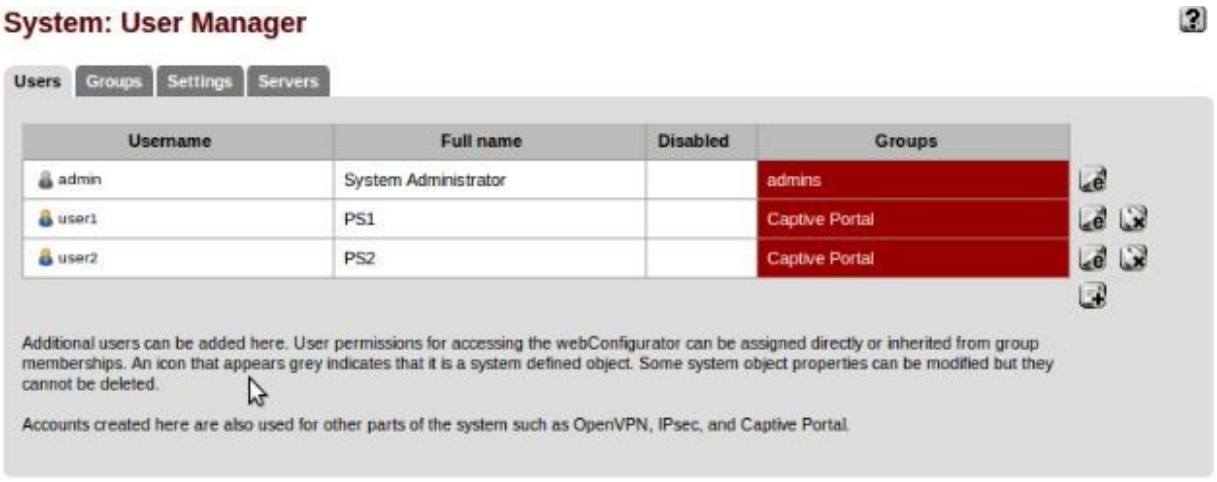
```
<html>

<head>

<title> "Welcome to NCCL's Network!"</title>

</head>

<body>

<img src="captiveportal-iitbhu1.jpeg"/>

<p>Enter your username and password.</p>

<form method="post" action="$PORTAL_ACTION$">

  <input name="auth_user" type="text">

  <input name="auth_pass" type="password">

  <input name="redirurl" type="hidden" value="$PORTAL_REDIRURL$">

  <input name="accept" type="submit" value="Continue">

</form>

</body>

</html>
```

ERROR PAGE HTML CODE:

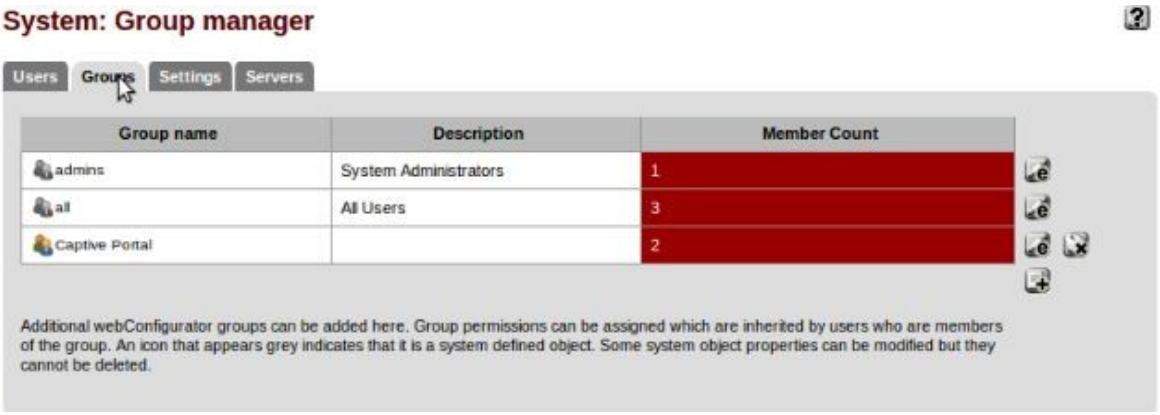
```
<html>

<head>

<title> "Incorrect credentials!"</title>

</head>

<body>

<img src="captiveportal-iitbhu1.jpeg"/>
```

```
<p><font color="red"><b>Incorrect credentials!</b></font></p>

<p>Enter your username and password.</p>

<form method="post" action="$PORTAL_ACTION$">

  <input name="auth_user" type="text">

  <input name="auth_pass" type="password">

  <input name="redirurl" type="hidden" value="$PORTAL_REDIRURL$">

  <input name="accept" type="submit" value="Continue">

</form>

</body>

</html>
```

Next, we go to **System -> User Manager** and create users with their respective usernames and passwords (Editing the created users)

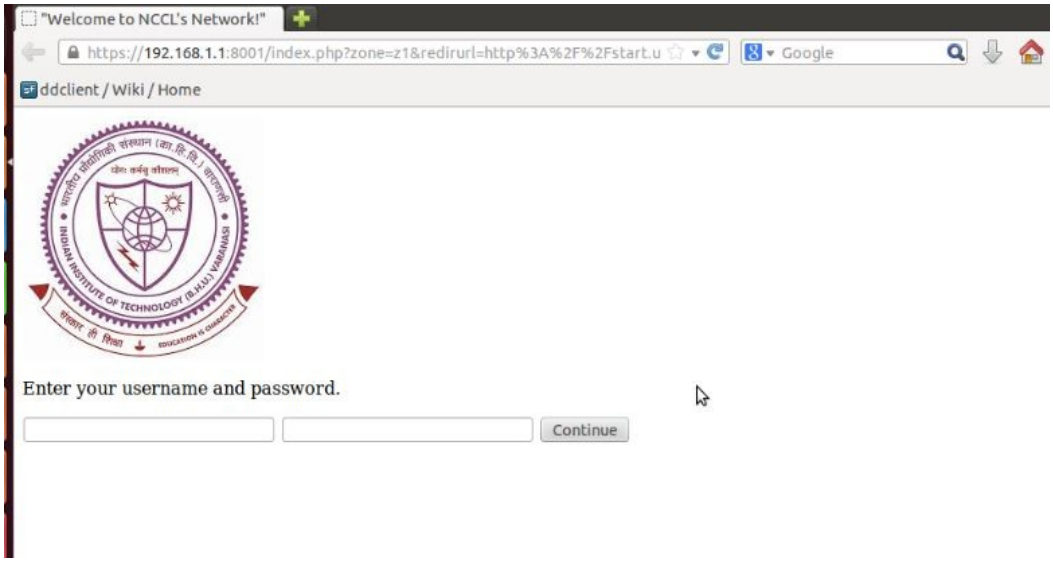


Next, in the **Groups tab**, we create a group ‘Captive Portal’ to define rules specific to the captive portal users only. We edit the group to shift the created users to this group and assign them privilege for captive portal login.



The captive portal configuration is now complete.

When the browser is opened, the captive portal page appears and it looks like this:



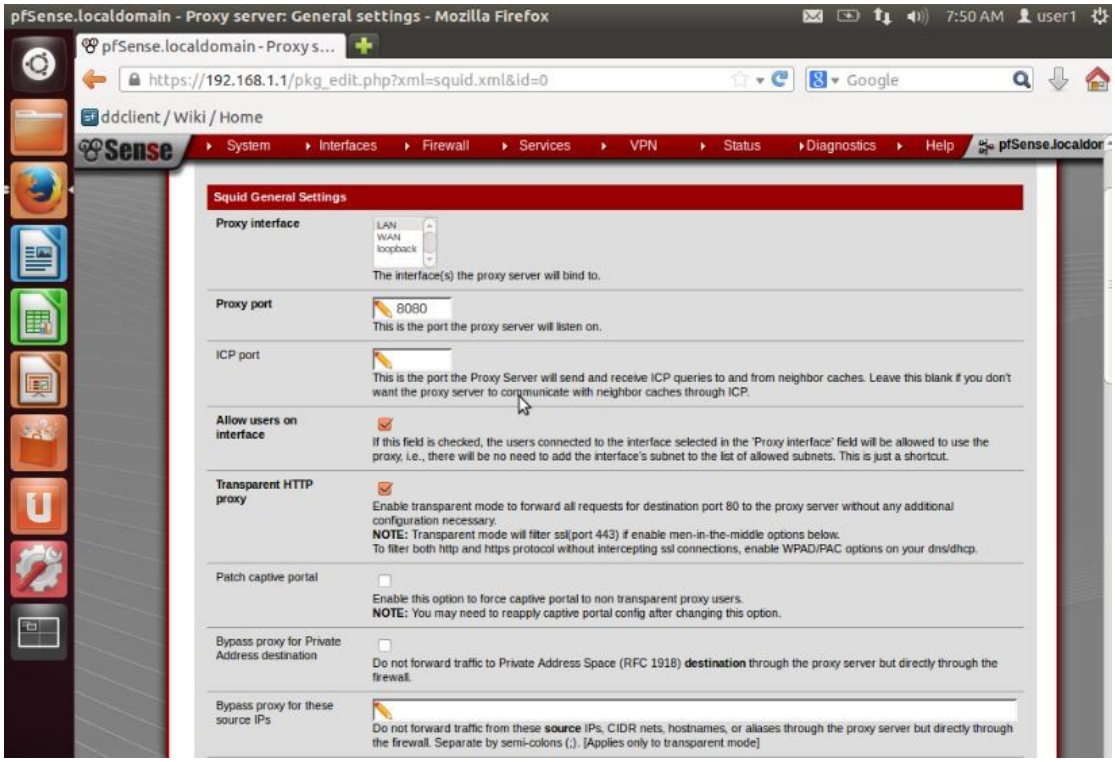
If invalid credentials are entered, the following error page is displayed:

Once the user successfully logs in the following logout window pops up and the user is redirected to https://www.iitbhu.ac.in

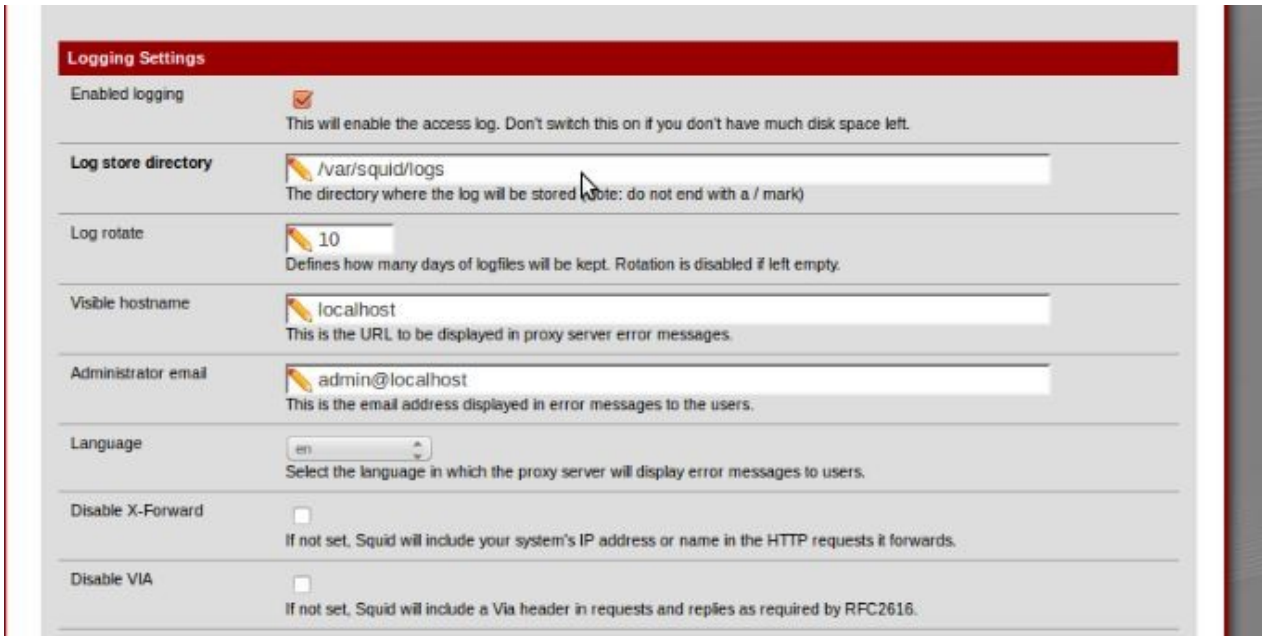
If a user has successfully logged into the captive portal, a temporary entry will be created for that user under **Status->Captive Portal.**

SETTING UP SQUID3 AS TRANSPARENT PROXY

We select the proxy interface as LAN with the port no. as 8080. We enable the transparent proxy by checking the relevant checkbox and we allow all the users on the LAN interface to use the proxy by checking “Allow users on interface”.



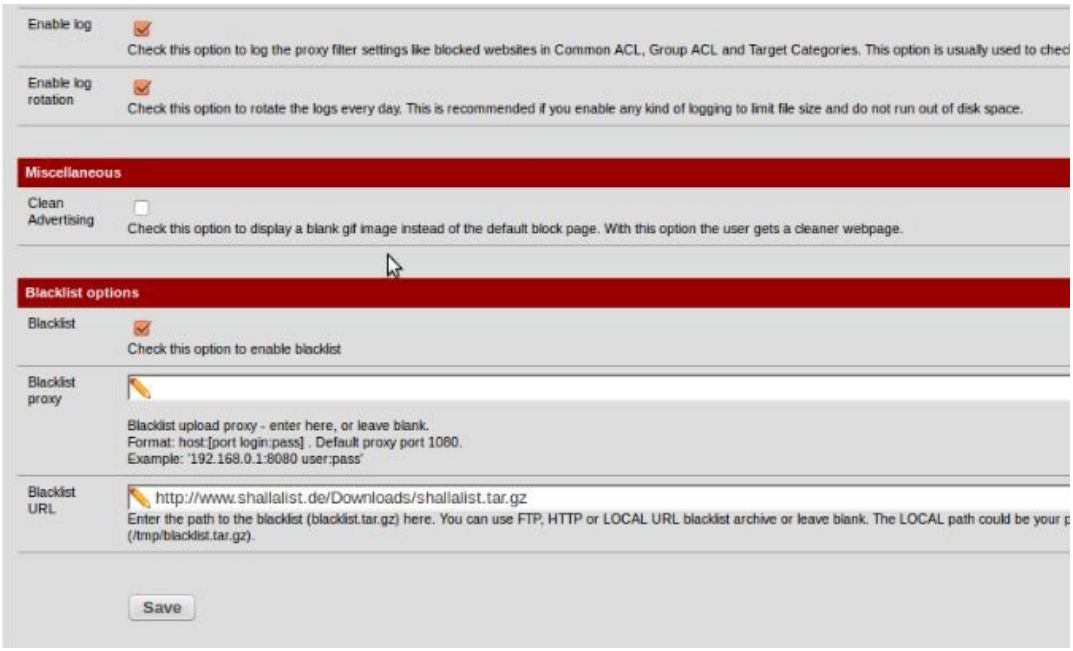
By enabling logging we allow squid to save logs of every client accessing the proxy server, which is a very important requirement for proxy server deployments.



Squid saves all the client logs in /var/squid/logs directory.

SETTING UP SQUIDGUARD3 (PROXY FILTER):

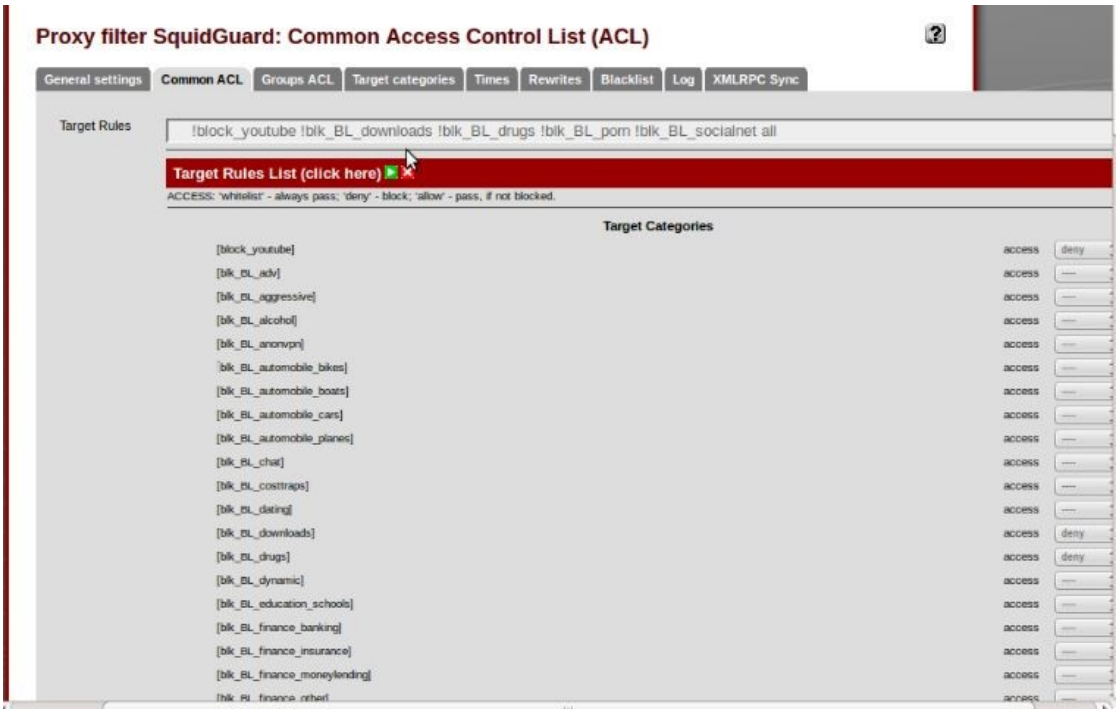
SquidGuard is a URL redirector. It is used to filter websites accessed through the proxy server based on their domain name. If access to a requested website is restricted, it redirects to a page displaying some error message. Since the filtering is URL based, it is unable to filter HTTPS traffic. SquidGuard is first enabled after navigating to the 'General settings' tab of SquidGuard.



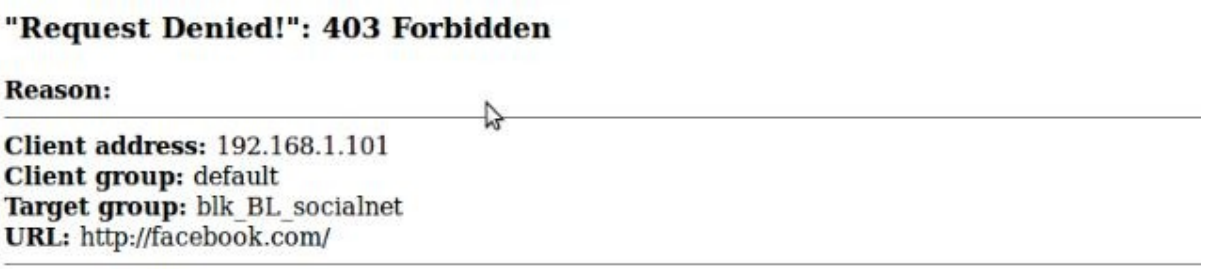
Enabling log saves all the log files in /var/log/squidGuard and gives us information of the blocked websites being tried to access, by the clients. Log rotation deletes the logs of users from one end every day and helps in saving disk space.

Standard blacklists are also available, that come with predefined categories. We can select which categories to block and which ones to allow. These blacklists contain long lists (almost complete) of the websites belonging to the given categories.

In the Common ACL tab, we may allow or deny the blacklist categories; therefore, we deny all the unwanted categories of websites and allow all others (default).



If a client tries to access a blocked website he/she will receive a message like this:



We can also block entire domains according to our needs, by going to the Target Categories tab and specifying the domains to block. In our case we have blocked ‘youtube.com’ domain.

FILTERING THE HTTPS TRAFFIC

There are many ways to filter https traffic; they are more or less related to filtering based on IP address, unless we try to attempt a MITM (Man In The Middle) Interception.

The ways are listed below:

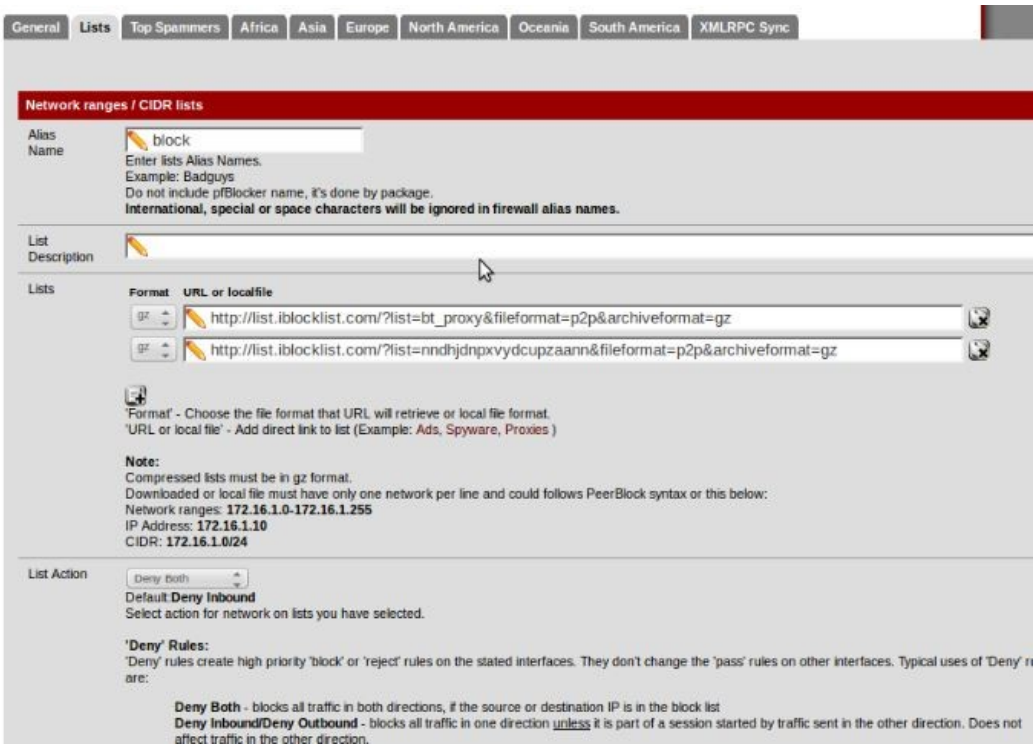
1) We can create aliases for all the websites that we want to block HTTPS traffic for, by finding their respective IP address ranges. We can then block the https traffic for this alias using firewall rules. Clearly, this is not a very efficient solution as the count of the websites to be blocked could be very high, thereby, making creation of aliases almost impossible. Other problem associated with this solution is that the websites keep expanding their ip address ranges thus, making frequent update of the aliases unavoidable.
2) We can block all the https traffic and then exempt only those websites, that we want to allow https traffic for. This will require creation of two rules. One for blocking all the https traffic and the other for the explicit bypass of the chosen few. Although, this solves the first problem of the previous method, it fails to solve the second problem.
3) We can use pfblocker, which is a firewall add-on. It can use pre-defined lists (containing the IP addresses of websites) of different categories. Moreover, there is a provision for the auto-update of the lists. So, this is the best among the three proposed methods.

We use the first method to block 'Facebook' and use pfblocker for blocking access to the proxy related websites (e.g. Tor)

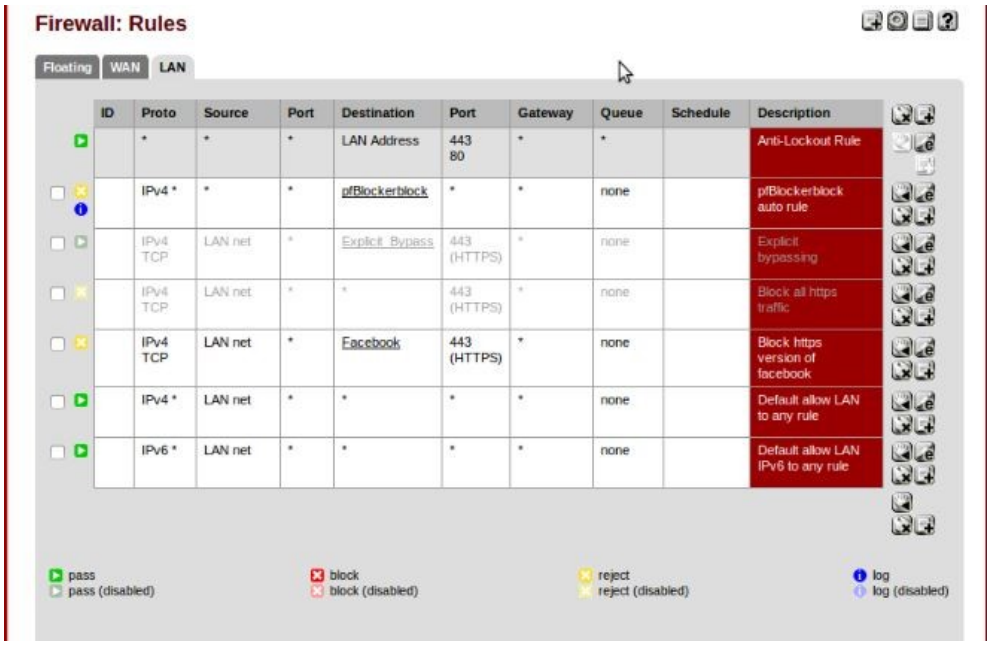
CONFIGURING PFBLOCKER

We navigate to **Firewall->pfblocker** and then to the General tab. We enable pfBlocker and also logs. All the other settings are left as default.

Under the Lists tab, we give an alias name and the URLs for the lists to be included. We, finally choose the ‘List Action’ as ‘Deny Both’. This will ensure that, there is no traffic from these IP addresses at all; neither to them nor from them.

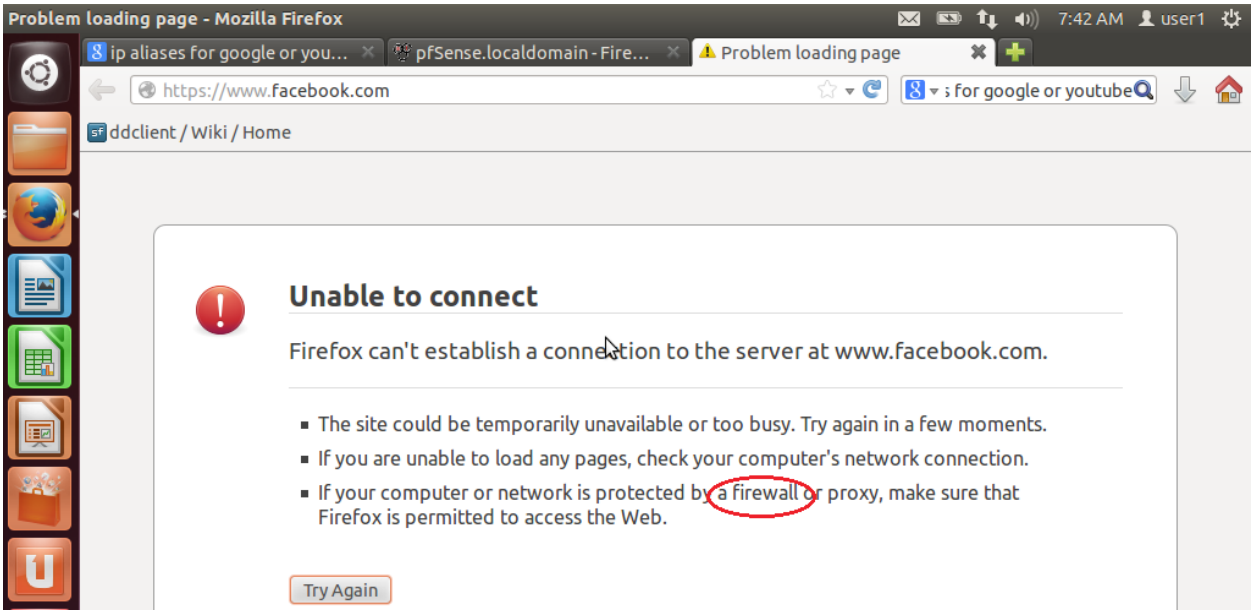


After these settings the firewall rule list looks like this:



Note that, the second rule is generated automatically by pfblocker.

Now when we try to access the https version of www.facebook.com the following page is displayed.

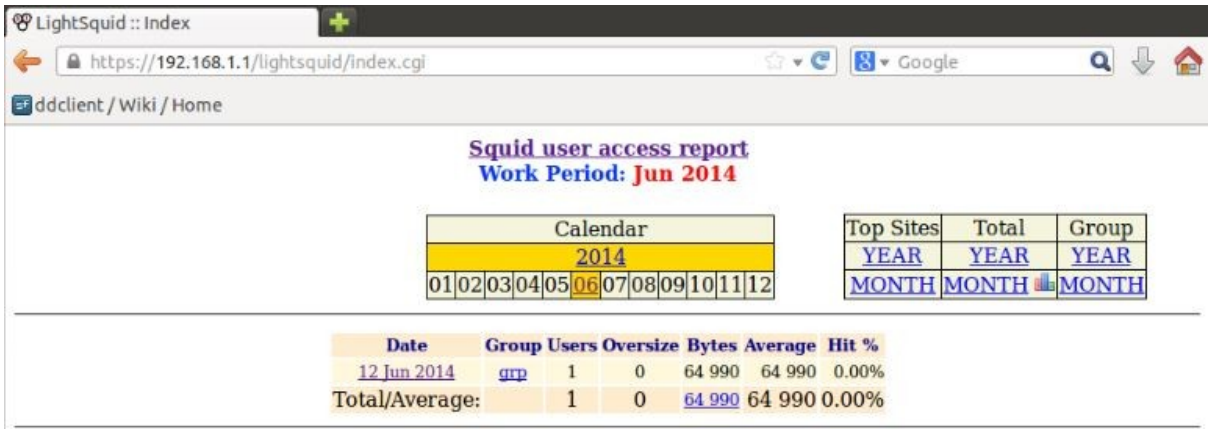


PROXY REPORT USING LIGHTSQUID

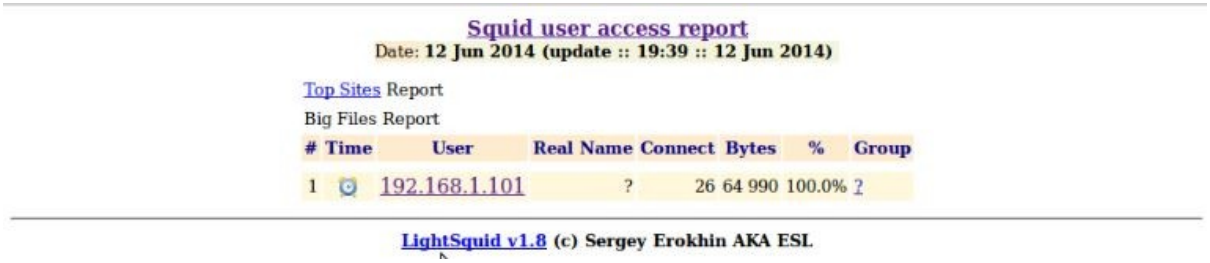
LightSquid is a Squid log analyzer that runs on pfSense. By parsing the proxy access logs, the package is able to produce web based reports that detail the URLs accessed by each user on the network. This package works well for both small and large networks. The reports have some useful features that allow you to see URL access by date and time, and top site reports.

Since LightSquid runs directly on your pfSense router, it is both centralized and stealthy. Users on the network have no way of knowing that their traffic is being logged and analyzed using this method.

Logs based on Date:



Logs based on client:



Squid user access report

User: 192.168.1.101 (?)
Group: ?
Date: **12 Jun 2014**

Total **64 990**

#	Accessed site	Connect	Bytes	Cumulative	%
1	clients1.google.com	6	14 613	14 613	22.4%
2	safebrowsing.clients.google.com	3	13 185	27 798	20.2%
3	ocsp.comodoca.com	4	12 816	40 614	19.7%
4	geoip.ubuntu.com	3	7 775	48 389	11.9%
5	gtglobal-ocsp.geotrust.com	3	5 700	54 089	8.7%
6	ocsp.usertrust.com	1	3 948	58 037	6.0%
7	gtssl-ocsp.geotrust.com	2	3 610	61 647	5.5%
8	ocsp.digicert.com	2	1 838	63 485	2.8%
9	www.iitbhu.ac.in	2	1 505	64 990	2.3%
Total			**64 990**		

EXTRA: WAN FAILOVER AND LOAD BALANCING IN PFSENSE

Failover:

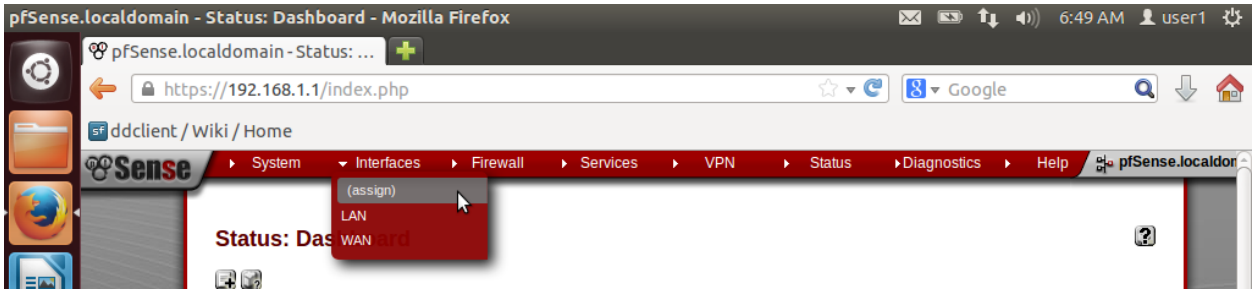
Failover refers to the ability to use only one WAN interface, but fail over to another WAN if the preferred WAN fails.

Load Balancing:

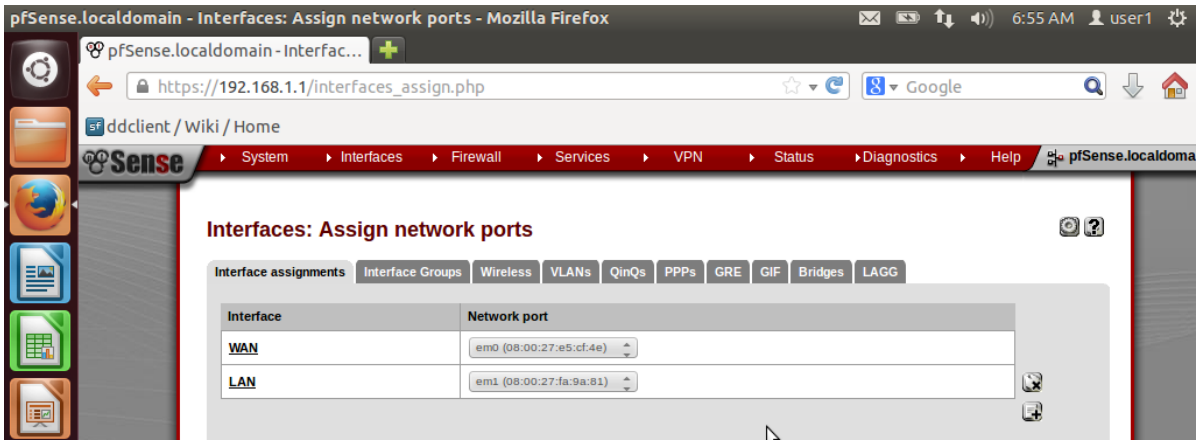
Load balancing refers to the ability to distribute load between multiple WAN interfaces. Note that load balancing and failover are not mutually exclusive. Load balancing automatically also provides failover capabilities, as any interface that is down is removed from the load balancing pool.

To configure WAN failover and Load balancing, we need to navigate to **Interfaces-> Assign**

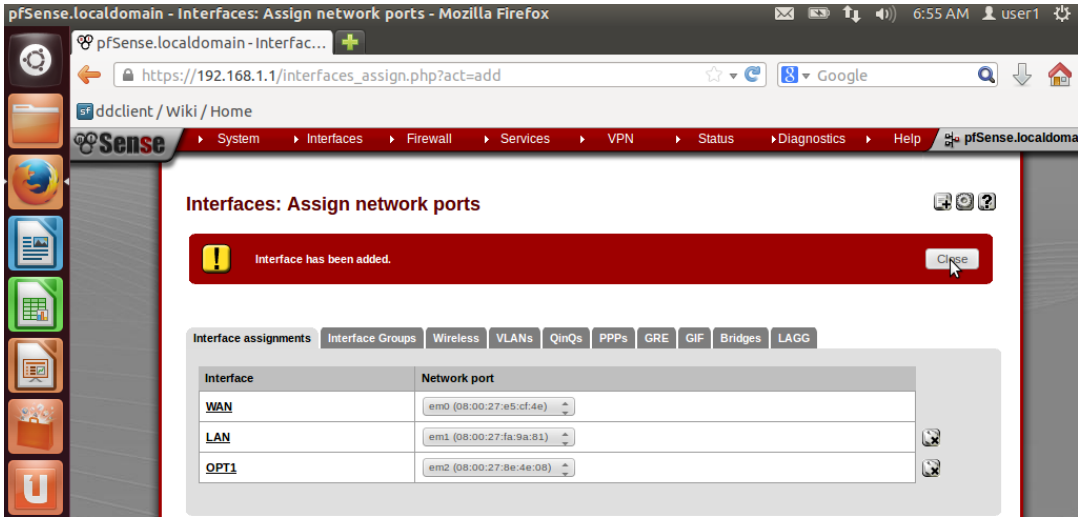
(Before that we have to make sure that we have enabled an additional adapter in VirtualBox, set as NAT)



The current existing interfaces get displayed.

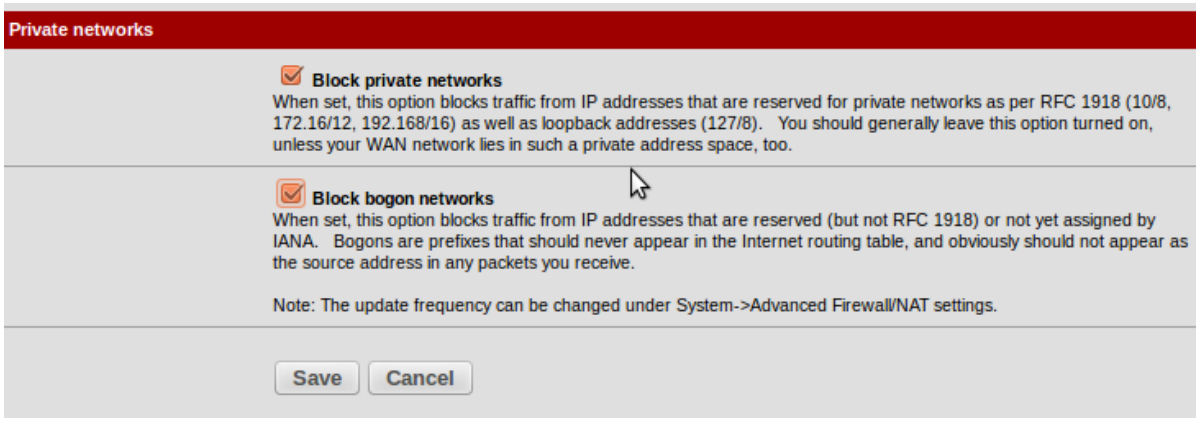


Now, we add another interface by clicking on the '+' symbol.



Now, we click on 'OPT1' to edit the interface. We change its description (name) to 'WAN2' and use DHCP for ip address assignment.

We also block private networks and bogon networks for the newly assigned WAN interface (WAN2).

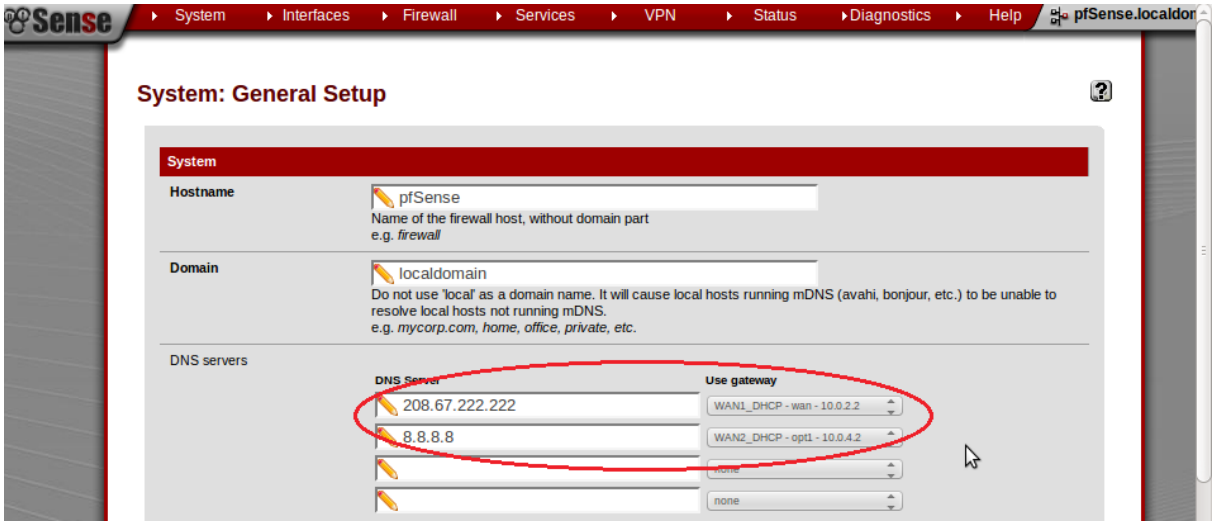


We similarly change the description (name) for the older WAN interface from 'WAN' to 'WAN1'.

Now, we have to assign public DNS servers for each of the gateways used.

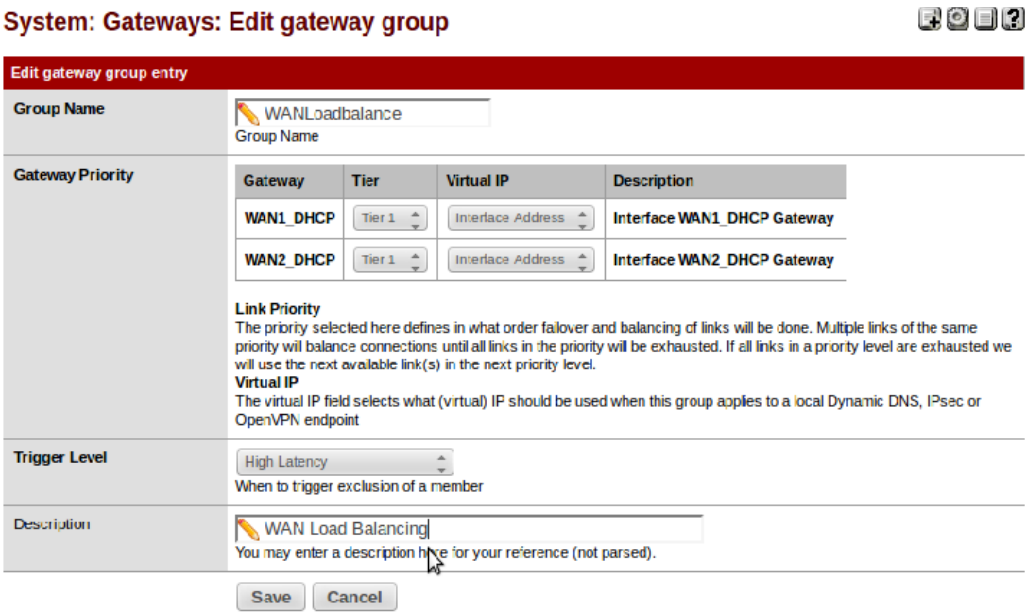
208.67.222.222 (OpenDNS Name server) - For WAN1 Gateway

8.8.8.8 (Google Name server) - For WAN2 Gateway

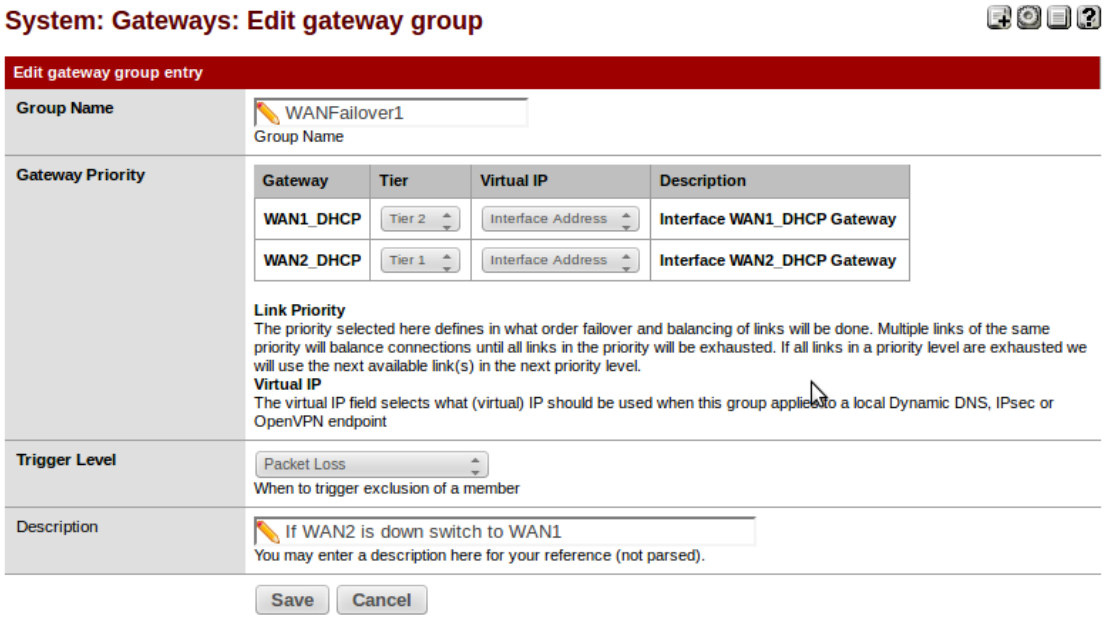


Now to create Gateway load balancing and failover groups, we go to **System->Routing->Groups.**

We add a new group for load balancing. Note that, the two gateways are at the same level of priority, and hence load will be balanced between these two. Moreover, we have configured the group such that a member will be excluded from the group, if there is high latency while using it.

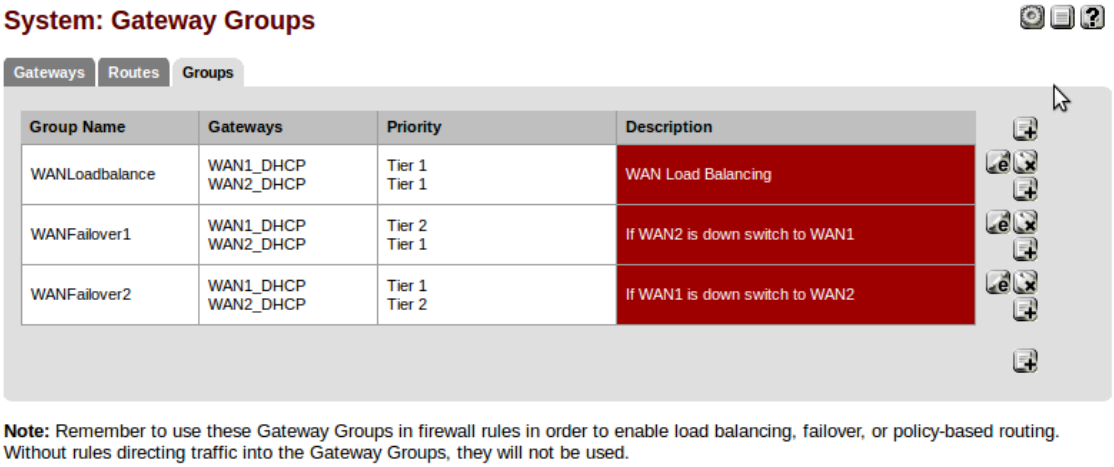


Next, we create the first WAN failover group. The configuration is done such that WAN2 is given more priority. Only when WAN2 goes down, WAN1 comes into play.



A Similar group is created for another failover mode, which has priorities swapped between these two WANs. Only when WAN1 goes down, WAN2 comes into play.

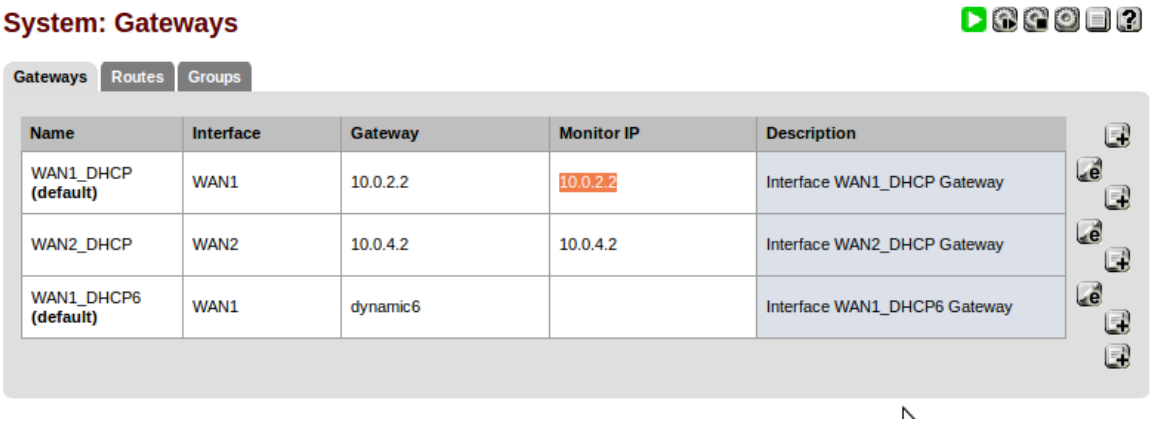
After creation of all the groups, the groups section would look like this.



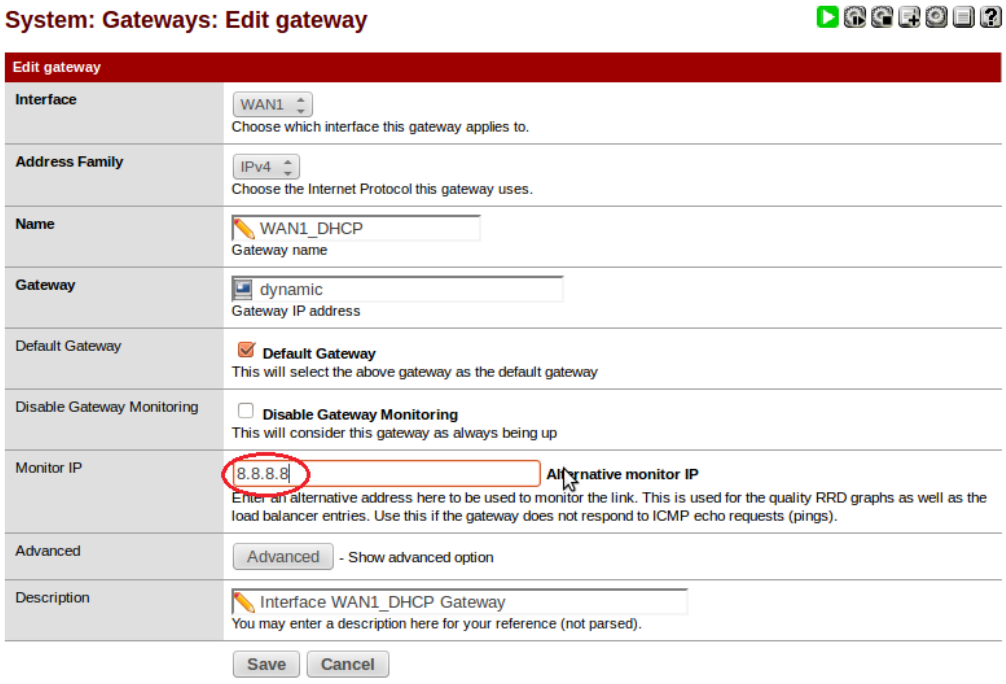
By default pfSense will ping a gateway to determine the quality of the WAN. In some cases, that is not an accurate measure. For instance, if the WAN gateway is actually a device that is local to the user, and not on the other side of user's ISP circuit, then the actual WAN link could be down and pinging the gateway would never show it. We can enter a custom IP address to monitor here that will be used to determine the WAN quality. We can use a public website, Google public DNS, or any IP on the internet that responds to pings. The downside is that should that IP ever go offline, or suffer a failure of its own, your WAN could be marked down when it's really up.

We will use the Google public DNS (8.8.8.8) and OpenDNS public DNS (208.67.222.222) as Monitor IPs.

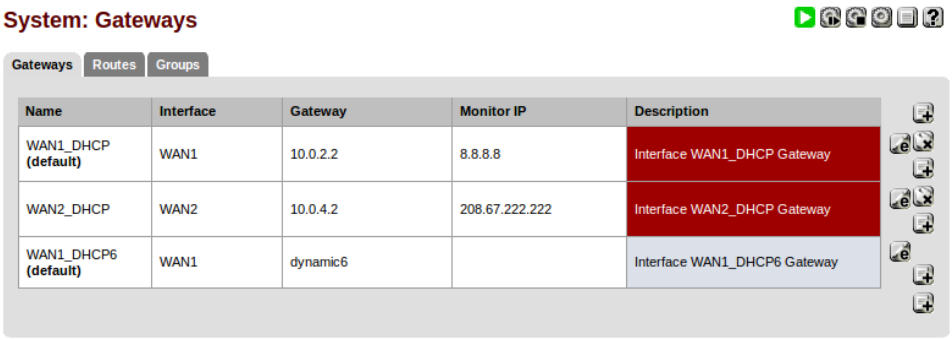
We first go to **System->Routing,** and then the gateways can be edited for changing their Monitor IPs.



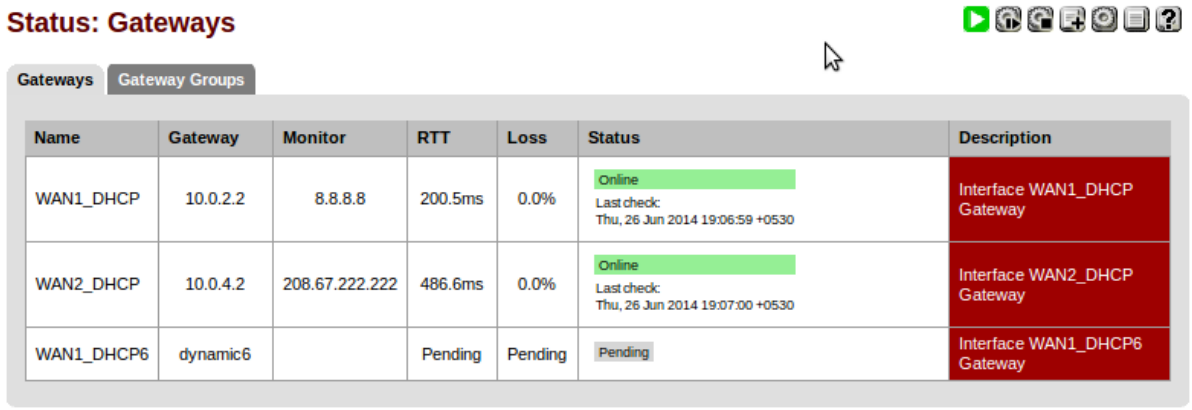
We change WAN1's Monitor IP to 8.8.8.8 and WAN2's Monitor IP to 208.67.222.222 (not shown).



After editing both the gateways, the **Gateways** section would look like this



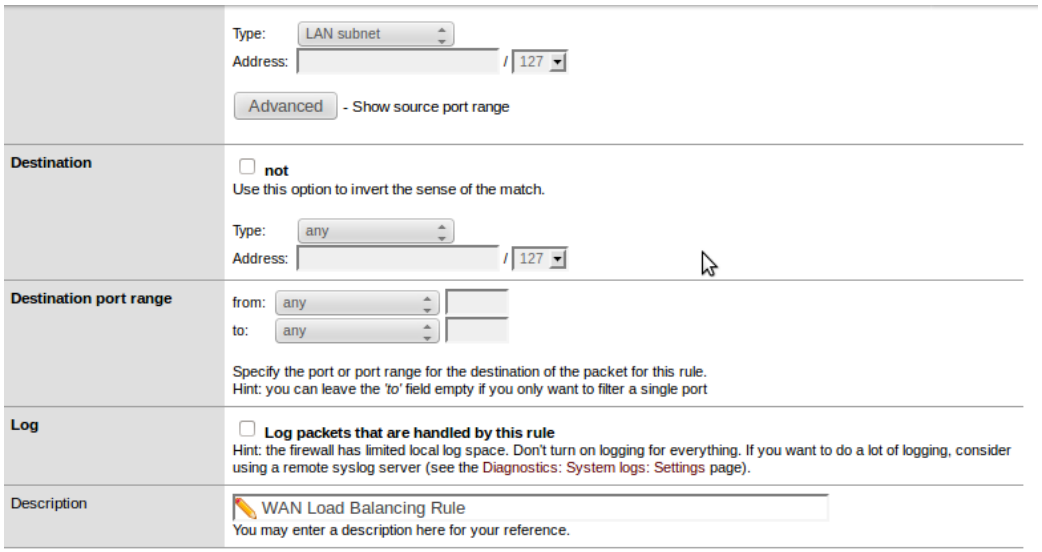
Now we can check whether both the gateways are up and working [**Status->Gateways**]. This is indeed the case.



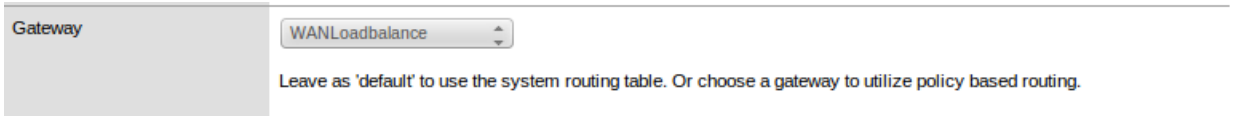
Now, we have to implement the firewall rules for WAN load balancing and failover. On the LAN interface we create three rules, one each for the previously created gateway groups.

- **<u>Load Balancing Rule</u>**:

We change the source field to ‘LAN subnet’.

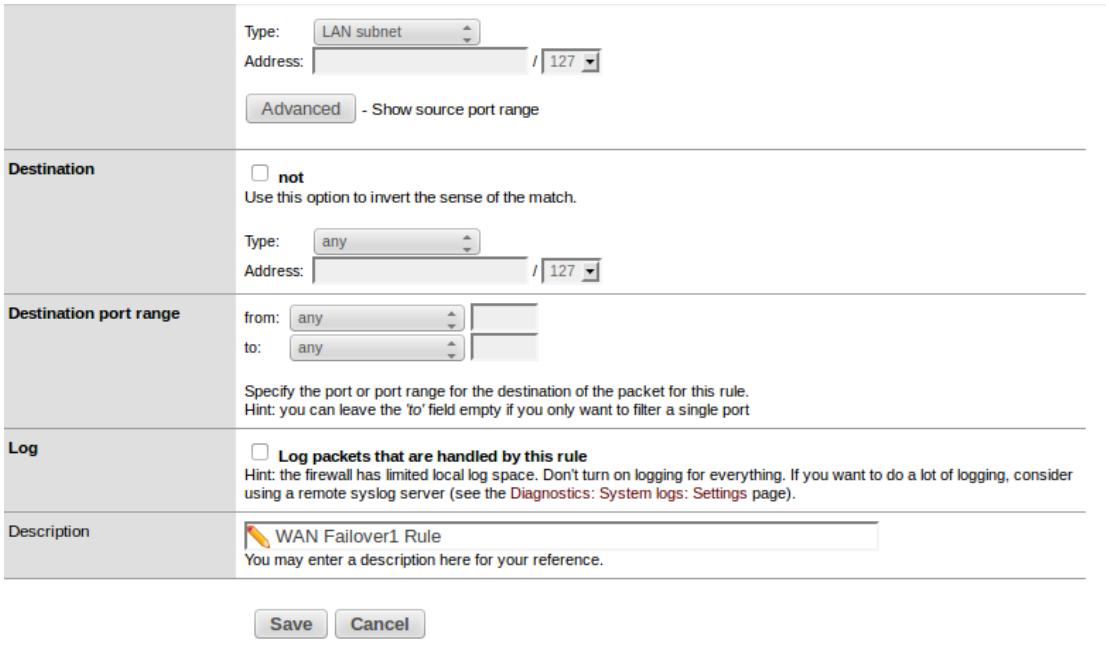


In the advanced settings, we choose the appropriate gateway group for policy based routing.

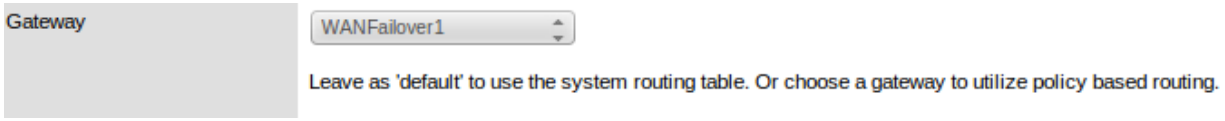


- **<u>First Failover Rule</u>**:

We change the source field to ‘LAN subnet’.

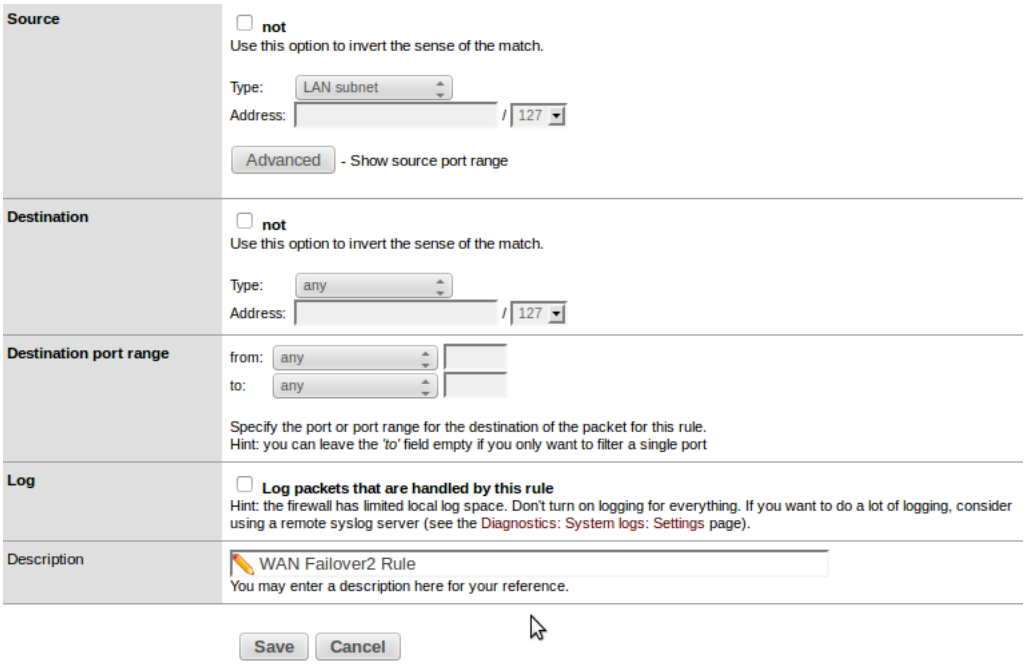


In the advanced settings, we choose the appropriate gateway group for policy based routing.

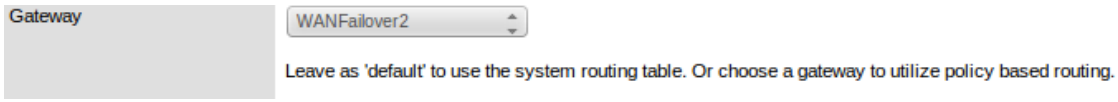


- **<u>Second Failover Rule</u>**:

We change the source field to ‘LAN subnet’.



In the advanced settings, we choose the appropriate gateway group for policy based routing.



We also have to disable the existing ‘default allow rules’ and after that the LAN rules section would look like this

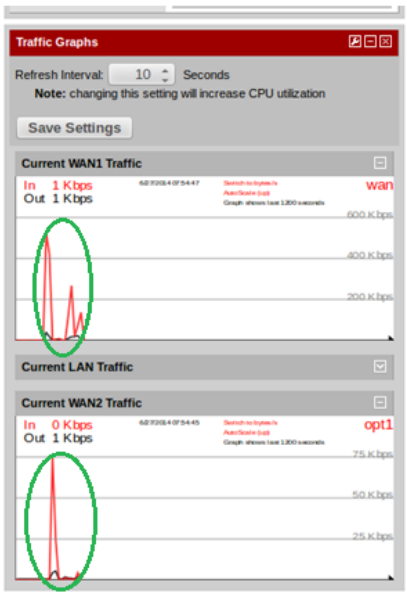
Firewall: Rules

Floating | WAN1 | LAN | WAN2

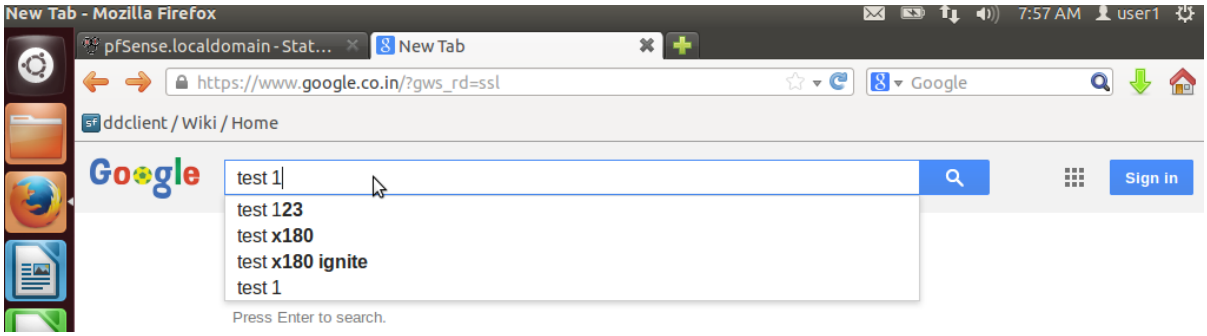
	ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
		*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule
		IPv4 *	LAN net	*	*	*	WANloadBalance	none		WAN Load Balancing rule
		IPv4 *	LAN net	*	*	*	WANFailover1	none		WAN Failover 1 rule
		IPv4 *	LAN net	*	*	*	WANFailover2	none		WAN Failover 2 rule
		IPv4 *	*	*	pfBlockerblock	*	*	none		pfBlockerblock auto rule
		IPv4 TCP	LAN net	*	Explicit_Bypass	443 (HTTPS)	*	none		Explicit bypassing
		IPv4 TCP	LAN net	*	*	443 (HTTPS)	*	none		Block all https traffic
		IPv4 TCP	LAN net	*	Facebook	443 (HTTPS)	*	none		Block https version of facebook
		IPv4 *	LAN net	*	*	*	*	none		Default allow LAN to any rule
		IPv6 *	LAN net	*	*	*	*	none		Default allow LAN IPv6 to any rule

Now, to verify the functionality of WAN Load Balancing, we simply browse the web, with both the WAN interfaces active. The traffic graphs for both WAN1 and WAN2 show activity during the browsing [We can enable the Traffic graphs on the dashboard by clicking on the '+' and selecting Traffic graph]. **This shows that load is being balanced between the two WAN connections.**

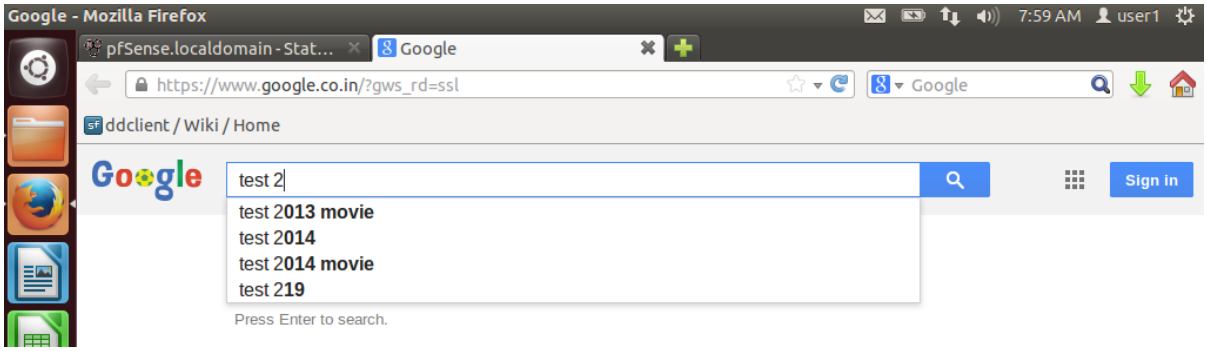
CPU Type	Intel(R) Core(TM) i5-2430M CPU @ 2.40GHz
Uptime	00 Hour 18 Minutes 52 Seconds
Current date/time	Thu Jun 26 19:24:48 IST 2014
DNS server(s)	127.0.0.1 8.8.8.8 218.248.241.3 8.8.4.4
Last config change	Thu Jun 26 19:16:29 IST 2014
State table size	1% (154/23000) Show states
MBUF Usage	11% (902/8512)
Load average	0.94, 0.90, 0.69
CPU usage	100%
Memory usage	31% of 235 MB
SWAP usage	0% of 256 MB
Disk usage	12% of 1.7G



To check the WAN failover functionality, we disable WAN2 interface and browse the web. Successful browsing **testifies the failover switching to WAN1**. [WAN Failover1 Rule]



To check the other failover mode, we disable WAN1, enable WAN2 and then browse the web. Successful browsing **testifies the failover switching to WAN2**. [WAN Failover2 Rule]

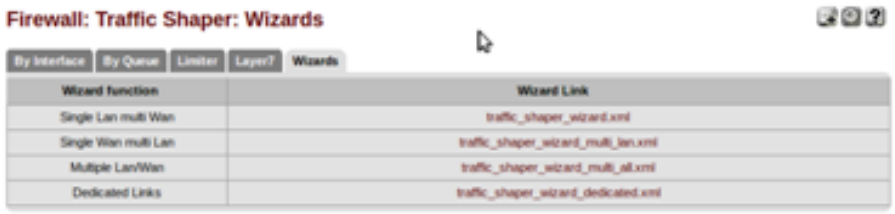


EXTRA: TRAFFIC SHAPING IN PFSENSE

Traffic shaping, also known as "packet shaping", is the practice of regulating network data transfer to assure a certain level of performance, quality of service (QoS) or return on investment (ROI). The practice involves delaying the flow of packets that have been designated as less important or less desired than those of prioritized traffic streams. Traffic shaping is a frequent topic of debate between advocates of Net neutrality and proponents of a two-tiered system. ISPs often target peer-to-peer (P2P) file sharing programs such as BitTorrent. Advocates of Net neutrality argue (among other things) that Internet data packets should be treated impartially, without regard to their content, destination or source and that it is difficult to delay some types of traffic without unintentionally hampering others. Proponents of a two-tiered system, on the other hand, argue that there have always been different levels of Internet service and that a two-tiered system would enable more freedom of choice and promote Internet-based commerce.

The pfSense Traffic shaper is available under Firewall->Traffic Shaper.

Now, we can select an appropriate wizard based on the number of LAN and WAN connections used.



There are several options available for handling VoIP call traffic. The choice, Prioritize Voice over IP traffic, is self-explanatory. It will enable the prioritization of VoIP traffic and this behavior can be fine-tuned by the other settings below.

"Penalty Box", is a place to which we can relegate misbehaving users or devices that would otherwise consume more bandwidth than desired. These users are assigned a hard bandwidth cap which they cannot exceed. Check the Penalize IP or Alias to enable the feature, we enter an IP or Alias in the Address box, and then enter upload and download limits in kilobits per second in their respective boxes.

To penalize P2P traffic, we first check Lower priority of Peer-to-Peer traffic. Many P2P technologies deliberately try to avoid detection. BitTorrent is especially guilty of this behavior. It often utilizes non-standard or random ports, or ports associated with other protocols. We can check the p2pCatchAll option which will cause any unrecognized traffic to be assumed as P2P traffic and its priority lowered

accordingly. We can set hard bandwidth limits for this traffic underneath the catchall rule.

Many games rely on low latency to deliver a good online gaming experience. If someone tries to download large files or game patches while playing, that traffic can easily swallow up the packets associated with the game itself and cause lag or disconnections. By checking the option to prioritize network gaming traffic, we can raise the priority of game traffic so that it will be transferred first and given a guaranteed chunk of bandwidth.

The last configuration screen of the shaper wizard, lists many other commonly available application and protocols. How these protocols are handled will depend

on the environment that this pfSense router will be protecting. Some of these may be desired, and others may not. For example, in a corporate environment, we may want to lower the priority of non-interactive traffic such as mail, where a slow down isn't noticed by anyone, and raise the priority of interactive services like RDP where poor performance is an impediment to people's ability to work. In a home, multimedia streaming may be more important, and other services can be lowered. If we enable the option for other networking protocols, we can pick and choose from the list.
There are more than 25 other protocols to choose from, and each can be given a **higher priority**, **lower priority**, or left at the **default priority**.

All of the rules and queues will now be created, but not yet in use. By pressing the Finish button on the final screen, the rules will be loaded and active. Shaping should now be activated for all new connections. Due to the stateful nature of the shaper, only new connections will have traffic shaping applied. In order for this to be fully active on all connections, you must clear the states. To do this, visit Diagnostics ->States, click the Reset States tab, check Firewall state table, then click Reset.

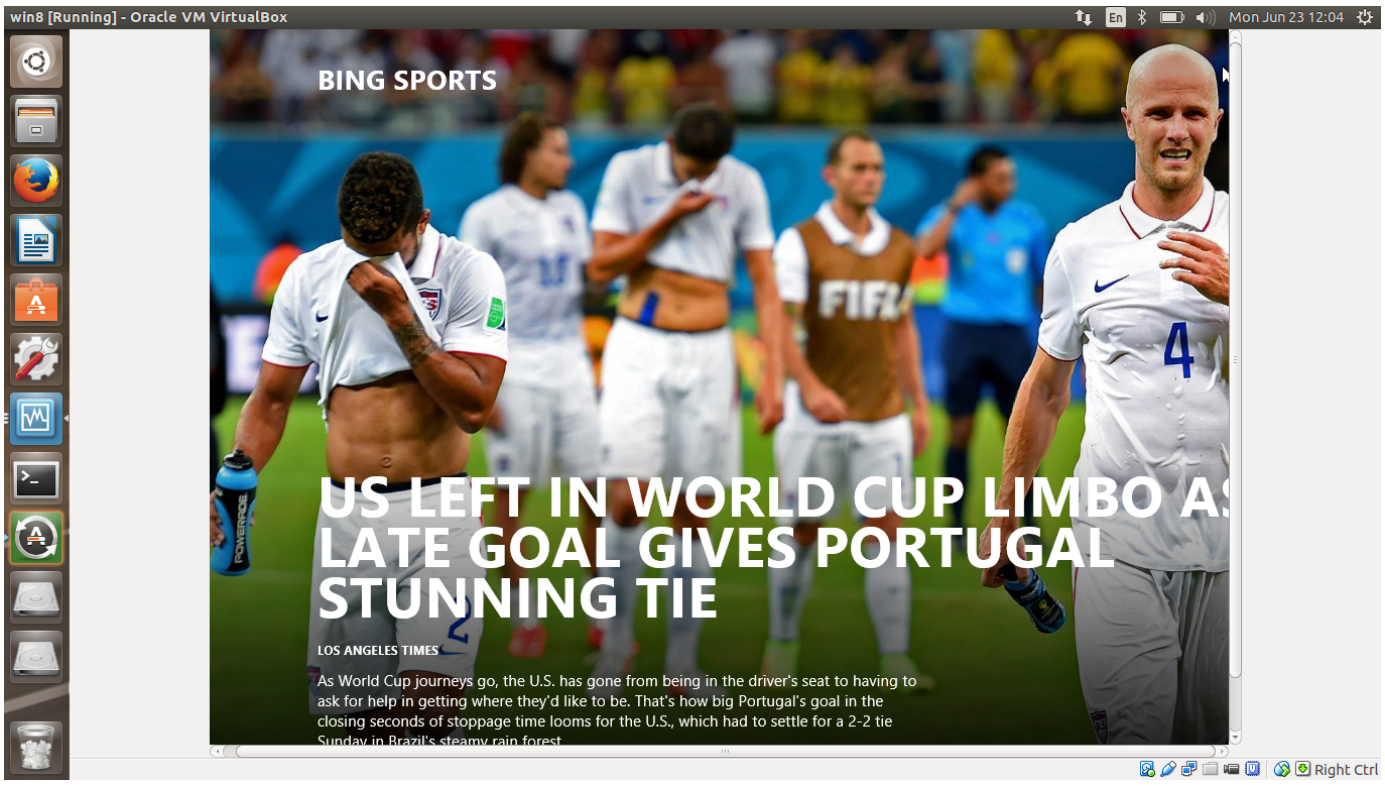
TEST: WINDOWS 8 APPS

It is a known problem that Windows 8 apps don't work in explicit proxy deployments, as the apps are not designed to use the proxy settings. This problem is however solved in Windows 8.1, where all the apps work seamlessly in this deployment.

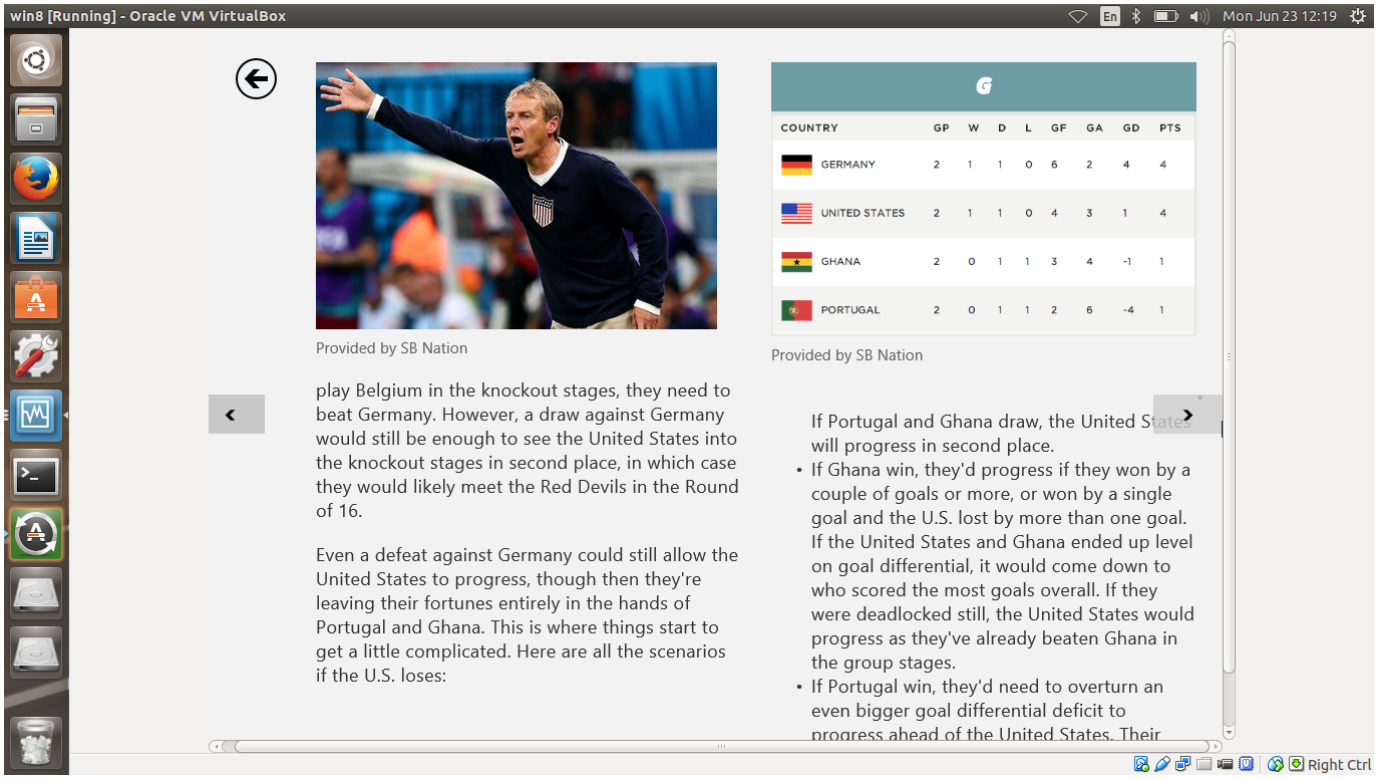
Our solution, which is, using a transparent proxy in conjunction with a captive portal, gets even the Windows 8 apps to work (without sacrificing authentication or filtering).

Following are the screenshots, testifying the working of Windows 8 sports app running on the windows 8 virtual machine in Oracle VM VirtualBox:

1)



2)


win8 [Running] - Oracle VM VirtualBox
Mon Jun 23 12:19
Provided by SB Nation
play Belgium in the knockout stages, they need to beat Germany. However, a draw against Germany would still be enough to see the United States into the knockout stages in second place, in which case they would likely meet the Red Devils in the Round of 16.
Even a defeat against Germany could still allow the United States to progress, though then they're leaving their fortunes entirely in the hands of Portugal and Ghana. This is where things start to get a little complicated. Here are all the scenarios if the U.S. loses:
G
COUNTRY GP W D L GF GA GD PTS
GERMANY 2 1 1 0 6 2 4 4
UNITED STATES 2 1 1 0 4 3 1 4
GHANA 2 0 1 1 3 4 -1 1
PORTUGAL 2 0 1 1 2 6 -4 1
Provided by SB Nation
If Portugal and Ghana draw, the United States will progress in second place.
If Ghana win, they'd progress if they won by a couple of goals or more, or won by a single goal and the U.S. lost by more than one goal. If the United States and Ghana ended up level on goal differential, it would come down to who scored the most goals overall. If they were deadlocked still, the United States would progress as they've already beaten Ghana in the group stages.
If Portugal win, they'd need to overturn an even bigger goal differential deficit to progress ahead of the United States. Their
Right Ctrl

CONCLUSION

We are deploying squid as a transparent proxy to resolve the issue of apps being unable to use the proxy setting. And since, the authentication procedure (challenging the user for valid credentials before they can use the proxy) does not work for transparent proxy deployments; a captive portal has been used to validate a user. Thus, the proposed solution involves using a transparent proxy in conjunction with a captive portal to get the apps to work seamlessly.

REFERENCES:

1) http://en.wikipedia.org/wiki/Proxy_server
2) http://whatis.techtarget.com/definition/proxy-server
3) https://publib.boulder.ibm.com/iseries/v5r2/ic2924/info/rzaie/rzaieproxytypes.htm
4) http://www.wisegeek.com/what-is-a-transparent-proxy-server.htm
5) http://netboxblue.com/sites/2012.netboxblue.com/files/Application%20Brief%20-%20Direct%20vs%20Transparent%20Proxy-v1.1-Nov13.pdf
6) http://en.wikipedia.org/wiki/Captive_portal
7) http://en.wikipedia.org/wiki/Firewall_(computing)
8) http://en.wikipedia.org/wiki/Golden_Shield_Project
9) http://www.greatfirewallofchina.org/
10) https://blogs.oracle.com/fatbloke/entry/networking_in_virtualbox1
11) http://en.wikipedia.org/wiki/PfSense
12) http://hubpages.com/hub/How-to-Set-Up-a-Captive-Portal-Using-pfSense
13) https://forum.pfsense.org/index.php?topic=43837.0
14) http://stackoverflow.com/questions/8608777/transparent-proxy-giving-ssl-error-when-accessing-via-https
15) http://stackoverflow.com/questions/10440690/pros-and-cons-of-using-a-http-proxy-v-s-https-proxy
16) http://skear.hubpages.com/hub/How-to-Configure-pfBlocker-An-IP-Block-List-and-Country-Block-Package-for-pfSense
17) https://www.iblocklist.com/lists.php
18) http://thwack.solarwinds.com/thread/32118
19) http://www.youtube.com/watch?v=omuklZrzopM