Linux + Transparent Proxy + Content Filtering

Written by Nybbles
nybbles@nybbles.net
http://www.nybbles.net

version 0.1

I. Introduction

First off, let me state that I am still pretty new when it comes to anything beyond the basics of Linux. I have done my best to research and find the most standard and efficient ways to set things up. The method described below does work, but if you know better, you're free to deviate from my directions.

Ok, with that out of the way, let's get to what we came here for. Since you're reading this you probably want to set up a proxy server of sorts. There are several good uses for one! It can cache commonly viewed pages, much like your web browser does, except this would be on a larger, network-wide basis. You can also use a proxy to limit who can get access to the Internet, when they can get access, what they can get access to, and so on. Another great reason may simply be to log and view accesses to keep track of employee productivity.

This guide will walk through the setup and installation of Debian GNU/Linux and Squid-Cache proxy. Optionally, DansGuardian may be used in addition to provide URL blacklisting and content filtering.

II. The Concepts

It was events that happened with my current employer that brought about the events that led to this guide. Basically, employee's were using company resources to surf the Internet, reducing employee productivity, and introducing a slew of technical problems.

We first looked to various commercial solutions but found the cost of them to be prohibitive. Then we looked to free/open-source software and found Squid. Squid is the proxy server software itself and it does a great job at what it does. Well, the drawback in our initial Squid efforts was that each computer on the network needed to be configured to use the proxy. Who wants to go around and re-configure 100+ web-browsers?

In this guide, I will tell you how to set up a server that will sit on the network 'transparently', meaning it will require no extra configuration on the clients end.

III. The Setup Process

The setup process contains the following steps:

- Install Debian GNU/Linux
- Install our proxy and content filtering software
- Configure the Squid Proxy
- Configure the DansGuardian content filter
- Configure the 'transparent' aspect of it
- Install and configure some sort of logging/monitoring solution

These steps will only go so far as to get a basic, general-purpose system in place. The software has the ability to be configured for far more than just what we're doing here, but for that part, you're on your own.

IV. Installing Debian

- First, you will want to download the Debian-Installer ISO file. It can be found here (http://www.debian.org/devel/debian-installer/)
- Create a bootable CD by burning the ISO file to the CD. This can be done with any CD-Recording software.
- Boot the computer off of the CD. The installer will prompt you to choose your language, country, and keymap. If this part confuses you, take the CD out of the drive and turn off the computer.
- If all goes well, the Debian-Installer will automatically detect and install drivers for your network card as well as your SCSI controller(s) if you have any. If either is not detected, then refer to the Debian documentation, or Google as setting this up is beyond the scope of this document.
- If you have a DHCP server on your network, Debian should pull an IP address from it. If it cannot, it will ask you to input an IP address, subnet mask, and gateway, as well as set the hostname and domain name.
- Next we'll partition the hard drive. This is most likely where people will have differing opinions on how this should be done.

  On a computer with a 4.3 GB hard drive, I set the partitions up like this:

200	MB	primary	swap
1024	MB	primary	/
500	MB	logical	/squid-cache
100	MB	logical	/tmp
2500	MB	logical	/var

  On a server with an 18 GB RAID 5 array, I divided it up like this:

1	GB	primary	swap
2	GB	primary	/
1	GB	logical	/squid-cache
1	GB	logical	/tmp

13 GB logical /var

On both examples, the /var partition is intentionally kept large for two reasons; First, on even the 4 GB hard drive, the other partitions have more than enough so why not, and also, secondly, all logs and reports are kept under /var.
We've also chosen to set aside a dedicated partition to hold the Squid cache separate from the rest. To be honest, I'm still reading up and playing with values here, so I might change this setup in the future.

Once the partitioning is complete, the Debian-Installer will start copying the system onto the hard drive.

- When copying is complete, you will want to choose to install GRUB onto the Master Boot Record. Then you will reboot.
- If the system comes back up, you're in good shape. The installer will walk you through configuring your time zone as well as setting up a root password and a non-privileged account.
- Next we will choose HTTP as the installation method, then choose your country and a mirror-site.
- When the Debian Software Selection comes up, hit Esc. We don't want to use one of their preset collections of packages; we want to install only what we need and nothing else.
- The install will now ask you to configure Exim4; choose local delivery only unless you know what you're doing.
- You should be able to log into the root account now. The first thing you will want to do is **apt-get update; apt-get dist-upgrade**. This will download the latest and greatest packages for you.
- That should complete the install!
- Before we move on, we just need to install a few other extras and we'll be ready to start configuring! To do this, use **apt-get install apache squid**
- Optionally, if you plan to do content filtering, you will also want to install DansGuardian as well. To do this, run **apt-get install dansguardian**.

V. Configuring Squid

First, we need to prep the /squid-cache partition:

chown proxy:proxy /squid-cache

Then, we need to make some changes to the /etc/squid/squid.conf file. Open it in your favorite editor and add the following to the HTTPD_ACCELERATOR OPTIONS section:

httpd_accel_host virtual
httpd_accel_port 80

httpd_accel_with_proxy on
httpd_accel_uses_host_header on

Find the cache_dir section, and add the following line:

(NOTE: SET X AS STATED BELOW!)

cache_dir ufs /squid-cache x 16 256

The Squid docs recommend you replace 'x' with a value that is 80% of the total size of the /squid-cache partition. So if you made /squid-cache a full gigabyte, x would equal 800 because that is roughly 80% of 1000.

For example: cache_dir ufs /squid-cache 800 16 256

Also, find the line that says *log_fqdn* and set it to on. Then save the file and exit your editor. Now, we'll restart Squid with:

/etc/init.d/squid restart

If it restarts successfully, we can remove the old Squid cache directory. Be careful with this command because it does bad things when used incorrectly. You've been warned.

rm –rf /var/spool/squid

That should conclude the Squid setup!

VI. Setting up iptables

Ok, before we get to configuring, a little explaination is in order. Basically, this new machine will become the networks new Default Gateway. What this means in most basic configurations is that all data will flow through it on its way to the Internet. What we basically want to do is to grab outbound HTTP requests and send them to the content filter or proxy. Anything that is not an HTTP requestion, we just forward it on.

So first off, we need to tell the system to allow IP forwarding. We do this with the following:

echo 1 > /proc/sys/net/ipv4/ip_forward

Next we configure IP tables to handle the forwarding:

If you want to do content filtering use:

iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 8080

If you just want to proxy without content filtering, use:

iptables -t nat -A PREROUTING -i eth0 p tcp --dport 80 -j REDIRECT --to-port 3128

That redirects HTTP requests, now to forward on everything else:

iptables -A INPUT -i eth0 -j ACCEPT
iptables -A FORWARD -i eth0 -j ACCEPT

Then we save the iptables config:

iptables-save > /etc/iptables.conf

Now, we just need to make sure that ip_forwarding and the iptables config loads at every startup. To do this, we will create /etc/init.d/iptables and put the following into it.

echo 1 > /proc/sys/net/ipv4/ip_forward
iptables-restore < /etc/iptables.conf

Save the file, then make it executable:

chmod a+x /etc/init/iptables

Now, we link it to run when we startup in run level 2:

ln –s /etc/init.d/iptables /etc/rc2.d/S19iptables

VII. Setting up DansGuardian

If you chose to use content filtering as well, it's a very simple process with just

Open /etc/dansguardian/dansguardian.conf in your favorite editor.

Comment out 'UNCONFIGURED' by placing a # sign before it. Look for a line called 'reverseaddresslookups' and change it to 'on'. Right below that, also set 'reverseclientlookups' to 'on' as well.

Save the file and close it.

Then open /etc/dansguardian/dansguardianf1.conf.

Look for a line called 'naughtynesslimit'. The value here is the maximum score that a webpage can have before it is blocked by the content filter. Many words are banned outright, and there existence in a webpage will cause the content filter to block a page and return an error message. Other 'lesser evil' words are simply given a point value to

them. For each occurance of these words, the point value of the page is increased. Once the value exceeds the limit you set here, the page is blocked. This could be particularly useful if you are setting up DansGuardian for a school and need to protect children from accessing adult content.

The recommended values are as follows:

50 – young children
100 – older children
160 – young adults

I'm still playing with the values here so I cant recommend one from personal experience. However, with a value of 200, I've only had a single page get blocked due to content.

Once you have this file set up that way you'd like it, we need to restart DansGuardian:

/etc/init.d/dansguardian stop
/etc/init.d/dansguardian start

Note: At the time I wrote this document, running **/etc/init.d/dansguardian restart** didn't seem to work properly. I did not take the time to figure out why since simply stopping and restarting worked just as well.

This completes the DansGuardian setup!

VIII. Finishing Up

Now we'll want to make sure our network settings are correct. To do this, edit /etc/network/interfaces. Locate the lines for your network card which will typically be 'eth0'. If you have 'iface eth0 inet dhcp', then we'll definitely want to change it to 'iface eth0 inet static'. Basically, we want to specify which IP our server will use. We don't want it being auto-assigned due to the possibility of it changing. Also, make sure that the IP you choose does not conflict with any DHCP scopes on the network. As an example, my settings look like this:

auto eth0
iface eth0 inet static
address 10.1.10.253
netmask 255.255.255.0
gateway 10.1.10.254

Once your settings are configured, we need to apply them:

ifdown eth0; ifup eth0

The server should be ready to go. I would test it first by manually assigning it to use this server as its Default Gateway Once you confirm it's working, you can change all computers Default Gateways, or, if your network runs DHCP, simply change the Default Gateway in the scope that you're using and those settings will automatically be sent out as your clients renew their addresses. (Note that this renewal could take a couple days, but it will work. If you need it to happen sooner, tell your users to reboot, or if they run Win2k/XP, tell them to run a 'ipconfig /renew'.

IX. Logging and Reporting

I'm currently still evaluating several log file analysis and reporting tools for use with Squid and DansGuardian. If you only run Squid, I've pretty much settled on SARG (http://sarg.sourceforge.net). However, this tool is of limited use if you are also using DansGuardian. Since all HTTP traffic gets forwarded to 8080 (DansGuardian), and then DanGuardian makes the request to 3128 (Squid), all requests in the Squid logs show as coming from 127.0.0.1. So really, in this case, if you want to monitor individual users, you must analyze the DansGuardian logs. At this time, I do not have a recommendation on a good DansGuardian log analyzer.