

Protecting Exchange 2000 Via Squid and Postfix

1. Introduction

The purpose of this how to is to not create experts in Squid and Postfix, but to help novices like myself in the process of setting up a Linux server to provide a buffer to the Internet for their Exchange servers. This is a very basic configuration and a work in progress. If you have suggestions please forward them to me so I can update this document. I am setting this up on a SuSE 9.0 Professional system with all the current updates via Yast2. The first step is to install Squid which comes with the SuSE 9.0 installation. At the time of this writing, Squid 3.0 is still in beta. The 2.5 version is more complicated to configure per the squid-cache.org user list. It is highly recommended that you build this from source so that it is tuned to your needs. For directions on accomplishing this, please look at Duane Wessels book: Squid The Definitive Guide.

Postfix was written with security and ease of configuration in mind. The process of setting up Postfix to relay mail for your internal Exchange server is quite simple, but I encourage you to learn more than what is needed to avoid security pitfalls and difficulty troubleshooting issues down the line. I highly recommend Postfix: The Definitive Guide by Kyle D. Dent. This is an easy and quick read and very helpful during the process.

2. Installation

a. Squid

You can either download the latest release from <http://www.squid-cache.org/Versions/v3/3.0/> or you can use the Squid 3.0.PRE3 version that comes with SuSE 9.0. Note: you will need to add the following lines in your conf file if you use the SuSE version since it was built with these parameters.

```
cache_replacement_policy heap
memory_replacement_policy heap
```

Next, make sure that OpenSSL is installed on your Linux box. I used the default version that is included with SuSE 9.0. You will need to use this to create your SSL certificate for the OWA proxy. When the certificate and key are created, combine them together and place them in the /etc/squid directory. The following is an excellent link to a step-by-step guide to creating certificates with OpenSSL:

<http://www.electica.ca/howto/ssl-cert-howto.php>

You'll also need to edit your /etc/hosts file to include your FQDN of the Exchange server so that Squid knows where to send traffic. This should be pointed to your INTERNAL IP address.

```
Mail.company.org      192.168.1.2
```

b. Postfix

More than likely, your distribution either came pre-loaded with Postfix or has it on one of the installation discs. You can either compile from source or install from an RPM depending on your distribution. I am using version 2.0.14 of Postfix. Make sure you install any updates or patches as well to ensure you are up to date.

3. Configuration

a. Squid

The process of configuring Squid for this purpose is rather straight forward once you get it working correctly. There are significant changes between 2.5 and 3.0 regarding how you configure the reverse proxy (accelerator). Here is a snip from <http://www.squid-cache.org/Versions/v3/3.0/squid-3.0-PRE3-20040301-RELEASENOTES.html>

For a description of the changes, please refer to the squid.conf comments.

- accelerator mode cleaned up, using the design from the rproxy development branch
 - The `httpd_accel_*` directives is now gone, replaced by `http(s)_port` options and `cache_peer` based request forwarding.
 - The `http(s)_port` options has a list of new options for controlling the type and mode of port created with respect to
 - transparent proxying
 - plain acceleration
 - host header based acceleration
 - normal proxying (default)
 - To enforce a reasonable level of security in accelerators, accelerated requests are denied to go direct unless forced by `always_direct`.

There are many settings within the squid.conf file that need to be configured. I've focused on the settings needed to get Squid functioning as an accelerator (reverse proxy) for OWA. You'll notice that Squid will terminate SSL sessions to itself and then communicate via http through the LAN. For some environments you may want to research the possibility of creating full https through the whole connection to your Exchange server. I've included the two configuration entries that are associated with setting up Squid as an accelerator to your Exchange OWA connection. I've added comments from my squid.conf to help explain what each section does.

[begin]

```
# TAG: https_port
#     Usage:  [ip:]port cert=certificate.pem [key=key.pem] [options...]
#
#     The socket address where Squid will listen for HTTPS client
#     requests.
#
```

```

# This is really only useful for situations where you are
# running squid in accelerator mode and you want to do the SSL
# work at the accelerator level.
#
# cert=      Path to SSL certificate (PEM format)
#
# key=       Path to SSL private key file (PEM format)
#             if not specified, the certificate file is
#             assumed to be a combined certificate and
#             key file
#
#             * Combine key per http://www.eclectica.ca/howto/ssl-cert-howto.php
#
# defaultsite= The name of the https site presented on
#               this port. DO NOT USE THE PATH "/exchange" enter this
#               at your browser.

```

```

https_port 443 cert=/etc/squid/key-cert.pem defaultsite= mail.company.com

```

```

# TAG: cache_peer
# To specify other caches in a hierarchy, use the format:
#
#       cache_peer hostname type http_port icp_port [options]
#       type: either 'parent', 'sibling', or 'multicast'.
#
# proxy_port: The port number where the cache listens for proxy
#             requests.
#
# icp_port:   Used for querying neighbor caches about
#             objects. To have a non-ICP neighbor
#             specify '7' for the ICP port and make sure the
#             neighbor machine has the UDP echo port
#             enabled in its /etc/inetd.conf file.
#
#             use 'proxy-only' to specify that objects fetched
#             from this cache should not be saved locally.
#
#             use 'no-query' to NOT send ICP queries to this
#             neighbor.
#
#             use 'no-digest' to NOT request cache digests from
#             this neighbor.
#
#             use front-end-https to enable the "Front-End-Https: On"
#             header needed when using Squid as a SSL frontend in front
#             of Microsoft OWA. See MS KB document Q307347 for details
#             on this header. If set to auto then the header will
#             only be added if the request is forwarded as a https://
#             URL.
#
#             use 'login=PASS' if users must authenticate against
#             the upstream proxy. This will pass the users credentials
#             as they are to the peer proxy. This only works for the
#             Basic HTTP authentication scheme. Note: To combine this
#             with proxy_auth both proxies must share the same user
#             database as HTTP only allows for one proxy login.

```

```
#           Also be warned that this will expose your users proxy
#           password to the peer. USE WITH CAUTION

cache_peer 192.168.1.2 parent 80 0 proxy-only no-query no-digest front-end-
https=on login=pass
```

[End]

Although there are many more options to setting up squid securely, I have focused on getting these two directives working properly. I will continue to add to this document as I get the ACL tested and reviewed.

Here is the final squid.conf

```
#squid.conf
https_port 443 cert=/etc/squid/key-cert.pem defaultsite=mail.company.org
cache_peer 192.168.1.2 parent 80 0 proxy-only no-query no-digest front-end-https=on login=pass
hierarchy_stoplist cgi-bin ?
acl QUERY urlpath_regex cgi-bin \?
no_cache deny QUERY
cache_replacement_policy heap
memory_replacement_policy heap
#auth_param basic children 5
#auth_param basic realm Squid proxy-caching web server
#auth_param basic credentialsttl 2 hours
# ACLs
# Base ACLs
acl all src 0.0.0.0/0.0.0.0
acl localhost src 127.0.0.1/255.255.255.255
acl http proto http
acl port80 port 80
acl https proto https
acl port443 port 443

# Only Allow cachemgr access from localhost
acl manager proto cache_object
http_access allow manager localhost
http_access deny manager

# Allow access to our servers
acl Exchangebox dstdomain mail.company.org
http_access allow https port443 Exchangebox

# Deny all other access to this proxy
http_access deny all

# Disable ICP
icp_port 0
```

b. Postfix

Postfix will be used as a mail relay and not a backup MX so make sure your DNS settings are pointing to the main MX record (i.e., Postfix). In my case, my MX record

will point to my firewall which is pointing SMTP traffic to Postfix that relays mail to Exchange.

[SMTP REQUESTS] -> [FIREWALL] -> [POSTFIX] -> [EXCHANGE]

I am not using Postfix to relay mail out since the likelihood of a security issue there is less of a factor. This also makes it much easier to simply redirect my firewall if there is a problem with the Postfix Box. There are many more configuration settings involved than this, so refer to the main.cf file or read the book noted above for more information. I will focus on the settings I needed to change out of the box as well as some that caused me some confusion. I will list each in the order I found them in the main.cf file. Keep in mind that the syntax is parameter = value. This make the configuration process easier, but it can cause some confusion.

Relay_domains = company.org

This parameter restricts what domains or destinations that the server will relay mail to. By default, Postfix will relay mail from clients that are located on the same subnet as the Postfix server and from external clients to destinations listed here. By default, postfix includes the mydestination parameter, so if you've included your domain as a value this setting may be redundant. I never got it work that way though.

Relay_recipient_maps = hash:/etc/postfix/relay_recipients

This parameter specifies the users that are allowed to pass through Postfix and thus relay to Exchange. The file format looks like the following and you need to perform the postmap /etc/postfix/relay_recipients command to turn it into a .db file

user2@company.org	any_value
user3@company.org	any_value

mydestination = \$myhostname, localhost.\$mydomain, \$mydomain

This parameter lists all the domains your server will accept messages for and process them for local delivery. This is where you'd enter the value \$mydomain. This would then include your domain for the relay_domains parameter. I was unable to get this to work so I just used the above relay_domains setting to specify the domain and dropped the \$mydomain value.

Mydomain = company.org

This specifies your local domain name, which is referenced in the previous values. Again, this did not work, but I may not have things dialed in correctly.

Transport_maps = hash:/etc/postfix/transport

This parameter relay's mail regardless of how DNS MX records are configured. It overrides the default transport mechanism by referencing transport lookup tables.

Again you develop a file that will need to be converted to a .db file (see above). I used the following entry in my transport file.

```
Company.org      smtp:[192.168.1.3]
```

Reference the book or documentation for detailed options of this entry. Basically, I am saying for company.org use the smtp transport to host at 192.168.1.3 using the standard port of 25.

The parameters entered above still need peer review of a postfix guru, but they work. If you find any in error, please forward your comments and I will fix this document. Once you've saved the main.cf, perform a postfix reload to not interrupt service. Go to zoneedit.com and use their SMTP testing page to see if you can get mail relayed to your internal server.

That's it.

Here are the parameters and values I found to work.

```
Relay_domains = company.org
Relay_recipient_maps = hash:/etc/postfix/relay_recipients
Transport_maps = hash:/etc/postfix/transport
mydestination = $myhostname, localhost.$mydomain
```

Setup asynchronous logging to improve performance. Set by default on SuSE 9.0.

```
/etc/syslog.conf
mail.* -/var/log/maillog
```

SPAM reduction – Edit the main.cf file. Reference the man pages for details on these settings.

```
Smtpd_helo_required = yes
Smtpd_helo_restrictions = permit_mynetworks, reject_invalid_hostname
Smtpd_sender_restriction = reject_non_fqdn_sender, reject_unknown_sender_domain
Smtpd_recipient_restrictions = permit_mynetworks, reject_unauth_destination,
reject_non_fqdn_recipients
```

Here is the final main.cf:

```
# Postfix main.cf
queue_directory = /var/spool/postfix
command_directory = /usr/sbin
daemon_directory = /usr/lib/postfix
mail_owner = postfix
unknown_local_recipient_reject_code = 450
relay_domains = company.org
relay_recipient_maps = hash:/etc/postfix/relay_recipients
debug_peer_level = 2
debugger_command =
```

```
PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
xxgdb $daemon_directory/$process_name $process_id & sleep 5
sendmail_path = /usr/sbin/sendmail
newaliases_path = /usr/bin/newaliases
mailq_path = /usr/bin/mailq
setgid_group = maildrop
manpage_directory = /usr/share/man
sample_directory = /usr/share/doc/packages/postfix/samples
readme_directory = /usr/share/doc/packages/postfix/README_FILES
mail_spool_directory = /var/mail
canonical_maps = hash:/etc/postfix/canonical
virtual_maps = hash:/etc/postfix/virtual
relocated_maps = hash:/etc/postfix/relocated
transport_maps = hash:/etc/postfix/transport
sender_canonical_maps = hash:/etc/postfix/sender_canonical
masquerade_exceptions = root
masquerade_classes = envelope_sender, header_sender, header_recipient
myhostname = gateway
program_directory = /usr/lib/postfix
inet_interfaces = all
masquerade_domains =
mydestination = $myhostname, localhost.$mydomain
defer_transports =
disable_dns_lookups = no
relayhost =
content_filter =
mailbox_command =
mailbox_transport =
smtpd_sender_restrictions = reject_non_fqdn_sender, reject_unknown_sender_domain
smtpd_client_restrictions =
smtpd_helo_required = yes
smtpd_helo_restrictions = permit_mynetworks, reject_invalid_hostname
strict_rfc821_envelopes = no
smtpd_recipient_restrictions = permit_mynetworks, reject_unauth_destination, reject_non_fqdn_recipient
smtp_sasl_auth_enable = no
smtpd_sasl_auth_enable = no
smtpd_use_tls = no
smtp_use_tls = no
alias_maps = hash:/etc/aliases
mailbox_size_limit = 0
message_size_limit = 10240000
```

4. Getting Help

There are numerous resources for getting help with your Squid implementation. I highly suggest you first look locally (Linux Users Group). I was fortunate enough to have volunteers help me through various parts of this setup. Other helpful links include the following.

Open SSL - <http://www.eclectica.ca/howto/ssl-cert-howto.php>

Squid Website - <http://www.squid-cache.org>

Squid Mailing Lists (squid-users) - <http://www.squid-cache.org/mailling-lists.html>

Changes from Squid 2.5 -> 3.0 - <http://www.squid-cache.org/Versions/v3/3.0/squid-3.0-PRE3-20040224-RELEASENOTES.html>

Squid book - <http://squidbook.org/index-two.html>

Postfix Website = <http://www.postfix.org>

Postfix Mailing List – <http://www.postfix.org/lists.html>

Postfix Book – <http://www.oreilly.com/catalog/postfix/>