

## Installing FreeBSD

Install a standard installation with ports from the boot cd menu.

### 1. Install Perl

1. Install from FreeBSD ports:  
**cd /usr/ports/lang/perl5.8;**  
**make -D ENABLE\_SUIDPERL=yes install clean;**

### 2. Install mail-toaster

1. Create the source directory to build everything in.  
**mkdir -p /usr/local/src**  
**cd /usr/local/src**  
**fetch <http://mail-toaster.org/Mail-Toaster.tar.gz>**
2. Once downloaded, extract the archive and install it much like you would any other perl module.  
**tar -xzf Mail-Toaster.tar.gz**  
**cd Mail-Toaster-\***  
**perl Makefile.PL**  
**make deps \*\***  
**make test**  
**make install**

### 3. Installing Ports

1. **bin/toaster\_setup.pl -s ports**
  1. **portsdb -Fu(y/n)** Say no to this by pressing n and <ENTER>
  2. The next step requires openssl support so install it as follows:  
**pkg\_add -r openssl**  
**cp /usr/local/openssl/openssl.cnf.sample /usr/local/openssl/openssl.cnf**  
**ln -s /usr/local/lib/libcrypto.so.5 /usr/local/lib/libcrypto.so.4**  
OR (at the time of this writing, the openssl-stable port is broken)  
**cd /usr/ports/security/openssl-stable**  
**make && make install**
3. Also, apache doesn't build in one of the script down the line so lets get apache out of the way.  
**cd /usr/ports/www/apache22**  
**make && make install clean**
  1. apache22 menu: choose defaults
4. Installing/Editing Config Files
  1. If you're only interested in making the required changes, you can use the "quick start" method:  
**cd /usr/local/src/Mail-Toaster-\***  
**bin/toaster\_setup.pl -s config**  
**bin/toaster\_setup.pl -s ssl**

2. After running the config step (above), you are encouraged to look through the following two files and adjust their contents to suit your preferences.

**nano -w /usr/local/etc/toaster.conf**

**nano -w /usr/local/etc/toaster-watcher.conf**

## 5. Installing Programs

1. Start building the toaster! (run one at a time)

1. **hash(bash) or rehash(csh)**

2. **cd /usr/local/src/Mail-Toaster-\***  
**bin/toaster\_setup.pl -s pre**  
**qmail menu: choose all patches**  
**daemontools menu: choose defaults**

3. **bin/toaster\_setup.pl -s mysql**  
**say (n) at first prompt to build from source**

4. **bin/toaster\_setup.pl -s apache**

5. **bin/toaster\_setup.pl -s webmail**  
**rsync menu: choose defaults**  
**y/n menu: yes to overwrite /usr/local/www/toaster**

6. **bin/toaster\_setup.pl -s ucspi**  
**yes to any y/n prompts**

7. **bin/toaster\_setup.pl -s ezmlm**

8. **bin/toaster\_setup.pl -s vpopmail**

9. **bin/toaster\_setup.pl -s maildrop**

10. **bin/toaster\_setup.pl -s qmailadmin**

11. **bin/toaster\_setup.pl -s qmail**

## 6. Install Web/POP3/IMAP servers

1. Install dovecot

1. **bin/toaster\_setup.pl -s dovecot**  
**dovecot menu: enable vpopmail support**

(I chose dovecot because it has proven more stable and handles malformed messages with more grace aka less customer calls)

2. Also a webmail client or three:  
**bin/toaster\_setup.pl -s squirrelmail**  
**bin/toaster\_setup.pl -s sqwebmail**  
sqwebmail menu: enable auth\_vchkw  
**bin/toaster\_setup.pl -s roundcube**  
roundcube menu: add spell check

## 7. Install Filtering

1. **bin/toaster\_setup.pl -s filter**

## 8. Install logging

1. Set up logging  
**bin/toaster\_setup.pl -s maillogs**  
**bin/toaster\_setup.pl -s supervise**  
**bin/toaster\_setup.pl -s rrdutil**  
**rrdutil menu: choose defaults**  
**ucd\_snmp menu: choose defaults**  
**ucd\_snmp questions: hit enter for all**

## 9. Install Cronjobs

1. **crontab -u root -e**  
**40 \* \* \* \* /usr/local/share/sqwebmail/cleancache.pl**  
**\*/5 \* \* \* \* /usr/local/sbin/toaster-watcher.pl**  
**\*/5 \* \* \* \* /usr/local/www/cgi-bin/rrdutil.cgi -a update**

## 10. Configuration

1. Edit /usr/local/etc/rc.d/apache22/extras/httpd-ssl.conf
  1. change the crt and key files to exist inside ssl.crt and ssl.key subdirectories.
2. Add a domain  
**~vpopmail/bin/vaddomain example.com [password]**  
**~vpopmail/bin/vadduser user@example.com [password]**
3. Test
  1. **bin/toaster\_setup.pl -s test**  
**run all tests: ignore failures**
4. Send some email  
**mail [user@example.com](mailto:user@example.com)**  
**Subject: test**  
**test**  
  
**ls ~vpopmail/domains/example.com/user/Maildir/\***

You should see a file in the new directory. If not, open up another terminal to the mail server and watch the mail log files with tail:

```
tail -f /var/log/mail/send/current
```

```
tail -f /var/log/maillog
```

## 5. Setup SimScan

1. Edit /usr/local/vpopmail/etc/tcp.smtp

Change the following line:

```
:allow
```

to look like this

```
:allow,QMAILQUEUE="/var/qmail/bin/simscan"
```

save the file

```
qmailctl cdb && qmailctl restart
```

(I chose simscan because it is a c binary not a script that uses craploads of memory. Also simscan will scan the email as it enters the server, rejecting it at the smtp level never letting excessive resources be used by mailscanning)

# 11. rc.conf configuration

1. Edit your rc.conf file and add these lines, removing any previous directives that are duplicates of these below.

```
apache22_enable="YES"
```

```
apache22ssl_enable="YES"
```

```
check_quotas="YES"
```

```
clamav_clamd_enable="YES"
```

```
clamav_freshclam_enable="YES"
```

```
courier_authdaemon_enable="YES"
```

```
courier_imap_imapd_enable="YES"
```

```
courier_imap_imapd_ssl_enable="YES"
```

```
courier_imap_imapdssl_enable="YES" #old
```

```
courier_imap_pop3d_ssl_enable="YES"
```

```
courier_imap_pop3dssl_enable="YES" #old
```

```
dovecot_enable="YES"
```

```
enable_quotas="YES"
```

```
hostname="fbsd.virtomain.local"
```

```
ifconfig_le0="DHCP"
```

```
ipf_rules="/etc/ipf.rules"
```

```
ipfilter_enable="YES"  
linux_enable="YES"  
mysql_enable="YES"  
mysqld_enable="YES"  
ntpdate="YES"  
sendmail_enable="NONE"  
snmpd_enable="YES"  
spamd_enable="YES"  
spamd_flags="-d -v -q -x -r /var/run/spamd.pid"  
spamd_flags="-v -x"  
sqwebmaild_enable="YES"  
sshd_enable="YES"  
stunnel_enable="YES"  
svscan_enable="YES"  
xntpd_enable="YES"  
xntpd_flags="-p /var/run/ntpd.pid"
```

## 12.Reboot