CHAPTER 3

Preparing for Installation

This chapter describes the tasks to be performed before you install Prime Home. Prime Home installation depends on various third-party components. You need to set up these components prior to installing Prime Home in your network infrastructure.

This chapter includes the following sections:

Configuring the RHEL Platform

The Red Hat Enterprise Linux (RHEL) operating system is installed on the ACS hosts and database server with the following specifications:

- The SSH access must be set up for Prime Home user for remote installation of Prime Home.
- The appropriate file system must be set up to support Prime Home installation. Table 3-1 describes the minimum disk space required for the file system.

Table 3-1 File System Size

File System	Minimum Size
/opt	64 GB
/var	128 GB
/home—Only for ACS hosts	128 GB
/data—Only for database server	256 GB

Note The */data* file system should be an XFS file system. It is used for MongoDB or MySQL data storage.

After you install RHEL, you must configure the RHEL platform to support Prime Home installation.

To configure the RHEL platform:

Step 1 Log into the ACS host and database server as root.

Step 2 Modify the config file to disable SELinux using the following commands:

```
# perl -p -i -e 's/^SELINUX=.*$/SELINUX=disabled/g' /etc/selinux/config
# perl -p -i -e 's/^SELINUXTYPE=.*$/SELINUXTYPE=targeted/g' /etc/selinux/config
```

The config file controls the state of SELinux on the system. In this file, set the value of SELINUX to disabled and SELINUXTYPE to targeted.

Step 3 Reboot the ACS host and database server using the following command:

```
# reboot
```

Step 4 Wait for 30 seconds and re-login to the ACS host and database server.

Step 5 On the ACS host, create the Prime Home user account using the following command:

```
# useradd -c "Prime Home User" -m -G wheel clearvision
```

The configurator tool uses the Prime Home user account to log into the load balancer and application server nodes, and install the necessary components on the host servers.

Step 6 Configure the Sudo facility on the ACS host:

a. Open the vi editor using the following command:

```
# visudo
```

b. Comment out the requiretty default using the following command:

```
# Defaults requiretty
```

c. Add the PATH variable to the new environment:

```
...
Defaults env_keep += "LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY"
Defaults env_keep += "PATH"
...
```

d. Disable the password prompts for the clearvision user using the following command:

```
clearvision ALL=(ALL) NOPASSWD: ALL
```

Step 7 Configure the SSH keys on the ACS host:

Note
- The SSH keys are generated on an administrative system, and their public keys are loaded onto the ACS host. This allows the administrative system to perform an SSH access into the ACS host, and get authenticated by the public keys instead of a password. The configurator tool uses this SSH facility to connect to the ACS host, and configure the Prime Home components.
- The administrative system user should not use a blank password for SSH keys.

a. Generate the SSH key pair on the administrative system using the following commands:

```
# ssh-keygen -t rsa -b 2048 -C "<User/Identifying Comment>" -f ~/.ssh/id_rsa
# chmod 700 ~/.ssh
# chmod 600 ~/.ssh/id_rsa
```

b. Set up agent on the administrative system using the following commands:

```
# exec ssh-agent bash -l
# ssh-add
```

c. Set up public or private key login on the ACS server using the following command:

```
# ssh clearvision@ACSHOST "mkdir -p ~/.ssh; touch ~/.ssh/authorized_keys; chmod 700
~/.ssh; chmod 600 ~/.ssh/authorized_keys; echo \"`cat ~/.ssh/id_rsa.pub`\" >>
~/.ssh/authorized_keys"
```

You can also log into the ACS host and copy the SSH keys if the following conditions are met:

- SSH agent is running
- SSH agent forwarding is enabled on the ACS host
- Remote .ssh directory already exists

To log into the ACS host and copy the SSH keys, run the following commands:

```
# ssh clearvision@acshost
# ssh-add -L >> ~/.ssh/authorized_keys
# chmod 600 ~/.ssh/authorized_keys
```

Note Ensure that the administrative SSH public keys are distributed to all Apache and Prime Home nodes. Repeat this step to distribute SSH keys to all the Apache and Prime Home nodes in your network infrastructure.

Step 8 On the ACS host and database server, configure OS limits and sysctl using the following commands:

```
# echo '# DO NOT USE' > /etc/security/limits.d/90-nproc.conf
# perl -p -i -e 's/^net\.ipv4\.tcp_syncookies[ \t]=.*$/net.ipv4.tcp_syncookies = 0/g'\
/etc/sysctl.conf
# cat <<EOF > /etc/security/limits.conf
* soft nofile 8192
* hard nofile 16384
* soft data unlimited
* hard data unlimited
* soft stack unlimited
* hard stack unlimited
* soft rss unlimited
* hard rss unlimited
* soft nproc 32768
* hard nproc 65535
EOF
```

Note The OS limits are configured for the Prime Home components to improve the host's performance. For a heavily loaded system with single server deployment, configure the OS limit and sysctl as described in this step.

Step 9 Configure time on the ACS host and database server:

a. Modify the /etc/sysconfig/clock file using the following command:

```
# perl -p -i -e 's/^ZONE=.*$/ZONE=UTC/g' /etc/sysconfig/clock
```

b. Change the system link using the following command:

```
# ln -sf /usr/share/zoneinfo/UTC /etc/localtime
```

c. Enable the time setting using the following commands:

```
# ntpdate pool.ntp.org
# chkconfig ntpd on
# service ntpd start
```

Note In virtualized environment, if the ACS host server uses Network Time Protocol (NTP) and the clock is synchronized with guest virtual machines, the installation of NTP is not required on the guest virtual machines.

Step 10 Configure iptables for the ACS host server:

Note It is assumed that em2 is the internal NIC and em1 is the external NIC on the host server.

a. Modify the /etc/sysconfig/iptables file using the following command:

```
# vi /etc/sysconfig/iptables
```

b. Set the parameters as follows:

- UDP for STUN—3478 or 3479
- TCP for ActiveMQ—7400
- TCP for Solr—7700
- To support validation of configurator ports for Apache or Tomcat TCP ports, add the following:

```
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -i em2 -j ACCEPT
-A INPUT -i em1 -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -i em1 -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -i em1 -p tcp -m tcp --dport 1080 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

c. If IPv6 support is available, add the following to the /etc/sysconfig/ip6tables file:

```
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
```

```
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p ipv6-icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
#-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp6-adm-prohibited
-A FORWARD -j REJECT --reject-with icmp6-adm-prohibited
COMMIT
```

Step 11 On the ACS host and database server, remove the default RHEL platform packages using the following command:

```
# yum -y erase java-*-openjdk libgcj mod_perl mod_wsgi
```

Step 12 Configure Java for the ACS host and database server:

a. Download the 64-bit Java6 jdk RPM version; jdk-6u37-linux-x64-rpm.bin, from http://www.oracle.com/technetwork/java/javase/downloads/jdk6u37-downloads-1859587.html.

b. Install Java6 jdk using the following command:

```
# sh jdk-6u34-linux-x64-rpm.bin
```

c. Verify the Java version using the following command:

```
# java -version
```

d. Add the Sun JDK to /etc/alternatives file using the following command:

```
# alternatives --install /usr/bin/java <path_to_JDK_bin/java> 20000
```

Step 13 On the ACS host and database server, install RHEL platform packages for Prime Home using the following commands:

```
# yum install ruby ruby-devel rubygems mysql-devel gcc zlib-devel
# yum install screen telnet logwatch lynx
```

Step 14 Install Ruby gems on the machine that is used for system configuration and database migration, using the following commands (you may also install Ruby gems on the ACS host):

```
# gem sources -a http://gems.rubyforge.org
# gem install -v=0.9.2.2 rake
# gem install -v=0.9.2.2 rails
# gem install mysql ruby-mysql
```

Step 15 If you have Oracle database in your network infrastructure, install the Oracle specific set of ruby gems.

Note To improve the backup capability of the host, you can also install the pigz package, along with the lbzip2 or pbzip2 package.

Observations - RHEL Setup

Table 3-2 is a worksheet that you must fill out after you configure RHEL on the ACS host server and database server.

Table 3-2 Worksheet - RHEL Setup

RHEL Setup	Sample Observation	Your Observation
Prime Home host user or ACS host user	clearvision	
Prime Home host password or ACS host password	clearvision	

Setting Up the Apache Web Server

The Apache web server acts as a proxy server, and is used to redirect northbound requests to the correct components in Prime Home. The Apache web server is deployed in the network infrastructure to provide the following functionalities for Prime Home:

- Direct the CPE, API, and Prime Home UI requests to the correct service port in the Prime Home platform's Tomcat container.
- Provide a simple means of managing SSL certificates for Customer Premises Equipment (CPE) and Prime Home web URLs.
- Limit the Prime Home platform access during maintenance, and allow the maintenance activity to be tracked.

You can analyze the log files of the Apache web server to determine CPE behavior and ACS performance.

Note SSL certificates are loaded in the Apache web server with specific certificate keys. You must restart the Apache web server when a new certificate key is added. If the certificate is renewed and the associated certificate key is unchanged, only a reload of Apache web server is needed.

The Apache web server is installed on the ACS host.

To set up the Apache web server:

Step 1 Log into the ACS host as root.

Step 2 Install Apache httpd using the following command:

```
# yum install httpd mod_ssl
```

Step 3 Remove the Perl and WSGI module of the Apache web server using the following command:

```
# yum erase mod_perl mod_wsgi
```

Step 4 Delete the php.conf file using the following command:

```
# rm -f /etc/httpd/conf.d/php.conf
```

Step 5 Modify the /etc/httpd/conf/httpd.conf configuration file using the following commands:

```
# perl -p -i -e 's/^Timeout .*$/Timeout 360/g' /etc/httpd/conf/httpd.conf
# perl -p -i -e 's/^KeepAlive .*$/KeepAlive On/g' /etc/httpd/conf/httpd.conf
# perl -p -i -e 's/^MaxKeepAliveRequests .*$/MaxKeepAliveRequests 1000/g'
/etc/httpd/conf/httpd.conf
```

```
# perl -p -i -e 's/^KeepAliveTimeout .*$/KeepAliveTimeout 300/g'
/etc/httpd/conf/httpd.conf
# sed -n '1h;1!H;${;g;s/<IfModule prefork.c>[^<]*</<IfModule prefork.c>\nStartServers
20\nMinSpareServers 20\nMaxSpareServers 50\nServerLimit 1024\nMaxClients
768\nMaxRequestsPerChild 0\n</g;p;}' /etc/httpd/conf/httpd.conf >
/etc/httpd/conf/httpd.conf.temp
# sed -n '1h;1!H;${;g;s/<IfModule worker.c>[^<]*</<IfModule worker.c>\nStartServers
5\nMaxClients 750\nMinSpareThreads 25\nMaxSpareThreads 75\nThreadsPerChild
25\nMaxRequestsPerChild 0\n</g;p;}' /etc/httpd/conf/httpd.conf.temp >
/etc/httpd/conf/httpd.conf
# rm -f /etc/httpd/conf/httpd.conf.temp
```

Depending upon the expected load, the worker may be the preferred worker at higher loads.

Step 6 Create the directory, home/clearvision/vhosts, on the Apache web server using the following commands:

```
# mkdir -p /home/clearvision/vhosts
# chown clearvision /home/clearvision/vhosts
```

Note You have to specify the location of this directory in the Prime Home configuration file.

Ensure that the appropriate permissions are provided to the Prime Home user to access this directory.

Step 7 Include the directory, home/clearvision/vhosts, in the httpd.conf file using the following commands:

```
# perl -p -i -e 's/^Include \/home\/clearvision\/vhosts.*$//g' /etc/httpd/conf/httpd.conf
# perl -p -i -e 's/^Include conf\.d\/\*\.conf.*$/Include conf\.d\/\*\.conf\nInclude
\/home\/clearvision\/vhosts\/\*\.conf/g' /etc/httpd/conf/httpd.conf

# perl -p -i -e 's/^Listen 81$//g' /etc/httpd/conf/httpd.conf
# perl -p -i -e 's/^Listen 80$/Listen 80\nListen 81/g' /etc/httpd/conf/httpd.conf
```

Step 8 Create the /etc/httpd/conf.d/proxy.conf file. The proxy.conf file enables NameVirtualHosts to route requests correctly. Set the proxy.conf file as follows:

```
# Proxy Config
cat <<EOF > /etc/httpd/conf.d/proxy.conf
<IfModule mod_proxy.c>
ProxyRequests Off
<Proxy *>
AddDefaultCharset off
Order deny,allow
Deny from all
Allow from all
</Proxy>
# Enable/disable the handling of HTTP/1.1 "Via:" headers.
# ("Full" adds the server version; "Block" removes all outgoing Via: headers)
# Set to one of: Off | On | Full | Block
ProxyVia On
ProxyTimeout 300
NameVirtualHost *:80
NameVirtualHost *:443
</IfModule>
EOF
```

Step 9 Remove the _default_ virtual host section in /etc/httpd/conf.d/ssl.conf file.

Step 10 Test the configuration using the following command:

```
# apachectl -t
```

You may ignore the warnings received for no virtual host existence. The configurator adds these virtual hosts in the Prime Home setup.

Note The **apachectl -t** command can only check syntax errors. If you have specified incorrect file paths or missing files, the error can be detected only during runtime.

Step 11 If the configuration is correct, start the httpd service using the following command:

```
# service httpd start
```

Step 12 If new SSL private keys are defined in the new configuration file, restart the httpd service using the following command:

```
# service httpd restart
```

Step 13 Verify the location of the log files and vhost files:

- /var/log/httpd/<acsname>.log —Log file for user interface NBI
- /var/log/httpd/<acsname>.cpe.log—Log file for CPE responder
- /home/clearvision/vhosts/hostname/—Directory where the configurator-generated vhost files are stored

Observations - Apache Web Server Setup

Table 3-3 is a worksheet that you must fill out after you set up Apache web server.

Table 3-3 Worksheet - Apache Web Server Setup

Apache Web Server Setup	Sample Observation	Your Observation
loadBalancers		
hostname	acs-host or http-host.internal.net	
loadBalancerId	http-public	
location	/home/clearvision/httpd	
type	apache	
URL	acshost.clearaccess.com	
nodeIds	main	
user	clearvision	
nodes		
hostname	acs-host	
location	/home/clearvision/acshost/main	
nodeId	main	
catalina_logDir	/home/clearvision/acshost/logs/main	
catalina_logNamePrefix	catalina-main	
log4j_logFile	/home/clearvision/acshost/logs/main/acs.log	
ports_tc_ajp_external	8082 443 - If SSL is enabled	

Table 3-3 Worksheet - Apache Web Server Setup (continued)

Apache Web Server Setup	Sample Observation	Your Observation
ports_tc_ajp_internal	8083	
ports_tc_http_external	8080 443 - If SSL is enabled	
ports_tc_http_internal	8081	
ports_tc_shutdown	8084	
tc_route	main	
uuid **Note** You can generate the UUID using the system command, **uuidgen**.	12B78A44-4F11-4142-8F5B-5990F9592C21	
URL_user_internal_root	http://acshost.clearaccess.com:80/prime-home	
clearprobe_rrd_directory	/home/clearvision/acshost/logs/main/rrd	

For information on the fields listed in the worksheet, see Table 5-1.

Setting Up a MongoDB Server

MongoDB provides a robust data storage engine that can be easily scaled. It can run on a single server as a standalone database to support smaller setups with only two nodes and an arbiter. MongoDB is deployed in the network infrastructure to provide the following functionalities for Prime Home:

- Permanent storage of data associated with system audit, Taze, and bandwidth monitoring.
- Limited storage of configuration data. The configuration data is modified only when you change the Prime Home setup. If MongoDB is used for storing the configuration data, you can deploy MongoDB on a single server as the data transaction rate is very less for configuration data.
- A highly available peer network that can be tracked easily in real time.

Note Memory utilization is high for MongoDB. Hence, we recommend that you deploy MongoDB on a separate server or virtual machine.

Installing and Configuring a MongoDB Server

To install and configure a MongoDB server:

Step 1 Log into the MongoDB host as root.

Step 2 Add the 10gen repository to your local repository by creating the file /etc/yum.repos.d/10gen.repo with the following contents (the example text shown is for a 64-bit system):

```
cat <<EOF > /etc/yum.repos.d/10gen.repo
```

```
[10gen]
name=10gen Repository
baseurl=http://downloads-distro.mongodb.org/repo/redhat/os/x86_64
gpgcheck=0
enabled=1
EOF
```

Step 3 Install the MongoDB packages, mongo-10gen and mongo-10gen-server, using the following command:

```
# yum -y install mongo-10gen mongo-10gen-server
```

Note You must install the MongoDB package mongo-10gen on the components that access the MongoDB server. In this release, the components are the ACS server and the optional bandwidth monitoring server only.

Step 4 Modify the /etc/mongod.conf file using the following commands:

```
# perl -p -i -e 's/^[#]*dbpath[= ].*$/dbpath=\/data\/mongo/g' /etc/mongod.conf
# perl -p -i -e 's/^[#]*logpath[= ].*$/logpath=\/var\/log\/mongo\/mongod.log/g'
/etc/mongod.conf
# perl -p -i -e 's/^[#]*logappend[= ].*$/logappend=true/g' /etc/mongod.conf
# perl -p -i -e 's/^[#]*fork[= ].*$/fork=true/g' /etc/mongod.conf
```

Verify that you have created directories for dbpath and logpath with appropriate permissions, or create them using the following commands:

```
# mkdir -p /data/mongo /var/log/mongo
# chown mongod:mongod -R /data/mongo /var/log/mongo
```

Step 5 Verify the MongDB configuration and restart the Mongo service using the following commands:

```
# chkconfig mongod on
# service mongod start
```

Step 6 Initialize MongoDB to create the databases and collections for STUN and system audit.

A confirmation message informs you when the setup is complete.

Creating Replica Sets for MongoDB

If MongoDB is used for permanent storage of system audit, Taze, and bandwidth monitoring data, we recommend that you create replica sets for MongoDB. You must deploy MongoDB on multiple servers and configure three nodes with sufficient data storage.

To create replica sets for MongoDB:

Step 1 Deploy MongoDB on an odd number of nodes greater than one.

Step 2 Modify the /etc/mongod.conf file to add the replSet and oplogSize directives using the following command:

```
# perl -p -i -e 's/^[#]*replSet[= ].*$/replSet=replicaSetName/g' /etc/mongod.conf
# perl -p -i -e 's/^[#]*oplogSize[= ].*$/oplogSize=100/g'
/etc/mongod.conf
```

Specify the replica set name and oplogSize as follows:

```
replSet = replicaSetName
oplogSize = 100
```

Note The oplogSize directive is the maximum size of the replication buffer, in megabytes. It should be set to approximately 5% of the file system size.

Step 3 Initialize the MongoDB replica sets using mongo shell as follows:

```
rs.initiate({
"_id" : "rs01",
"members" : [
{"_id" : 1, "host" : "mongohost1.domain.com"},
{"_id" : 2, "host" : "mongohost2.domain.com"},
{"_id" : 3, "host" : "mongohost3.domain.com"}]
})
```

Observations - MongoDB Server Setup

Table 3-4 is a worksheet that you must fill out after you set up the MongoDB server.

Table 3-4 Worksheet - MongoDB Server Setup

MongoDB Server Setup	Sample Observation	Your Observation
db_directory on MongoDB server	/data/mongo	
Hosts on which MongoDb is configured	mongohost1.domain.com, mongohost2.domain.com, and mongohost3.domain.com	

Setting Up Apache Solr

Apache Solr provides full text search capabilities within the Prime Home platform. When objects are added to the MySQL database, Apache Solr creates an index for the descriptors associated with the objects. The different parts of the Prime Home platform use this index to search the descriptors.

Apache Solr can be installed on the ACS host or on a separate host server.

After you set up the Apache Solr component, you can perform the following functions using the Apache Solr admin UI:

- View the counts of objects in the database and index.
- Rebuild the complete or partial index.

To set up Apache Solr:

Step 1 Log into the Apache Solr host or ACS host as root.

Step 2 Install solr-jetty-core package using the following commands:

```
# mkdir -p /opt/clearvision/packages/solr
# cd /opt/clearvision/packages/solr
# tar -xzf /path/to/solr-jetty-core-3.5.0.tgz
```

```
# useradd -d /opt/clearvision/packages/solr -s /bin/bash -G clearvision solr
ln -sf /opt/clearvision/packages/solr/solr-jetty-core-3.5.0/bin/solr-jetty-core-3.5.0
/etc/init.d/solr-jetty-core
# chkconfig solr-jetty-core on
# cd /opt/clearvision
# ln -s packages/solr/solr-jetty-core-3.5.0 solr
# chown -R solr:clearvision solr packages/solr
```

Step 3 Modify the /opt/clearvision/solr/solr/conf/solrconfig.xml file based on your network infrastructure, using the following command:

```
# vi /opt/clearvision/solr/solr/conf/solrconfig.xml
```

For multiple node with single ACS host, use the following pattern for solrconfig.xml file

```
<!-- remove the <lst name="master"> section if this is just a slave -->
<!-- remove the <lst name="slave"> section if this is just a master -->
<requestHandler name="/replication" class="solr.ReplicationHandler" >
<lst name="master">
<str name="enable">${enable.master:false}</str>
<str name="replicateAfter">commit</str>
<str name="replicateAfter">startup</str>
<str name="confFiles">schema.xml,stopwords.txt</str>
</lst>
</requestHandler>
```

For High Availability installation, use the following pattern for solrconfig.xml file:

```
<!-- remove the <lst name="master"> section if this is just a slave -->
<!-- remove the <lst name="slave"> section if this is just a master -->
<requestHandler name="/replication" class="solr.ReplicationHandler" >
<lst name="master">
<str name="enable">${enable.master:false}</str>
<str name="replicateAfter">commit</str>
<str name="replicateAfter">startup</str>
<str name="confFiles">schema.xml,stopwords.txt</str>
</lst>
<lst name="slave">
<str name="enable">${enable.slave:false}</str>
<str name="masterUrl">http://MASTERHOSTNAME:7700/solr/corename/replication</str>
<str name="pollInterval">00:00:60</str>
</lst>
</requestHandler>
```

Step 4 Start the Apache Solr service using the following command:

```
# service solr-jetty-core start
```

Step 5 Create specific cores using the following script:

```
# /opt/clearvision/packages/solr/solr-jetty-core-3.5.0/bin/create-core.sh corename
```

Step 6 Confirm that Apache Solr was successfully set up on the host server by opening the admin UI using the following URL:

```
http://host:7700/solr/corename/admin
```

Observations - Apache Solr Setup

Table 3-5 is a worksheet that you must fill out after you set up Apache Solr.

Table 3-5 Worksheet - Apache Solr Setup

Apache Solr Setup	Sample Observation	Your Observation
solr_url	http://MASTERHOSTNAME:7700 /solr/corename	

For information on the fields listed in the worksheet, see Table 5-1.

Setting Up Fuse ActiveMQ

Fuse ActiveMQ provides Java Messaging Services (JMS) to various components of Prime Home. It acts as the central communication system for Prime Home.

The following two components of Prime Home require access to Fuse ActiveMQ's JMS queues:

- ACS host server
- Taze STUN server

If the Prime Home components are individually hosted on separate servers, access to Fuse ActiveMQ's JMS queue is required for all components. For larger distributed topologies, we recommend that you set up access to Fuse ActiveMQ's JMS queue for all Prime Home components.

The Fuse ActiveMQ component is always hosted on the ACS server or a separate standalone server.

To set up Fuse ActiveMQ:

Step 1 Log into the Fuse ActiveMQ host or ACS host as root.

Step 2 Create a Fuse ActiveMQ folder using the following command:

```
# mkdir -p /opt/clearvision/packages/activemq
# cd /opt/clearvision/packages/activemq
```

Step 3 Obtain the ActiveMQ Fuse tarball from RedHat. For example, the filename should be similar to apache-activemq-5.*X.X*-fuse-*XX*-*XX*-bin.tar.gz.

Step 4 Decompress and unpack the installation package using the following command:

```
# tar xzf /path/to/apache-activemq-5.5.1-fuse-09-16-bin.tar.gz
```

Step 5 Create the Active MQ user and set up ActiveMQ using the following commands:

```
# useradd -d /opt/clearvision/packages/activemq -s /bin/bash -G clearvision activemq
# ln -sf /opt/clearvision/packages/activemq/apache-activemq-5.5.1-fuse-09-16/bin/activemq
/etc/init.d/activemq
# chkconfig activemq on
# cd /opt/clearvision
# ln -s packages/activemq/apache-activemq-5.5.1-fuse-09-16 activemq
# chown -R activemq:clearvision activemq packages/activemq
```

Step 6 Configure basic data directories using the following command:

```
# mkdir -p /var/db/activemq
# chown -R activemq:clearvision /var/db/activemq
```

Step 7 Modify the conf/activemq.xml file to configure Fuse ActiveMQ transport connectors as follows:

```
<transportConnector name="nio" uri="nio://acshost.domain://0.0.0.0:7400"/>
```

For performance reasons, change the ActiveMQ data store to a different disk using the following command:

```
# vim /opt/clearvision/activemq/conf/activemq.xml
```

Note

- The transport connectors create listening interfaces for the Fuse ActiveMQ service. You can define multiple transport connectors with different listening ports. When you define transport connectors, ensure that you specify the correct hostname and listening port. If you are using Taze STUN, the listening port must exist on the network that is available to Taze STUN.
- The transport connector defines the broker URI in the configuration file.

Step 8 Start the Fuse ActiveMQ service using the following command:

```
# chkconfig activemq on
# service activemq start
```

Step 9 Launch the Fuse ActiveMQ admin UI to verify that it is processing the JMS Queues.

High Availability Setup

In High Availability setup, the Fuse ActiveMQ component is most commonly deployed in a primary-secondary environment where:

– The client uses a failover URI. For example:

```
failover:(nio://masterhost:7400,nio://slavehost:7400)?randomize=false
```

– The basic configuration of primary ActiveMQ is modified. The activemq.xml file of primary ActiveMQ must have a broker section with a waitForSlave="true" parameter. For example:

```
<broker xmlns="http://activemq.apache.org/schema/core"
brokerName="CPHmaster"
waitForSlave="true"
dataDirectory="/var/db/activemq">
```

This ensures that both primary ActiveMQ and secondary ActiveMQ will replicate all messages.

– The secondary ActiveMQ uses the masterConnectorURI, which is specified in its broker definition. For example:

```
<broker xmlns="http://activemq.apache.org/schema/core"
brokerName="CPHslave"
masterConnectorURI="nio://amqmaster:7400"
shutdownOnMasterFailure="false"
dataDirectory="/var/db/activemq">
```

This ensures that when the secondary ActiveMQ starts, it registers with the primary ActiveMQ and the topics that are used for replication are created.

Note When using Fuse ActiveMQ in a primary-secondary configuration, the failover occurs one way only, and the down time is required to restore the primary-secondary configuration.

Observations - Fuse ActiveMQ Setup

Table 3-6 is a worksheet that you must fill out after you set up Fuse ActiveMQ.

Table 3-6 Worksheet - Fuse ActiveMQ Setup

Fuse ActiveMQ Setup	Sample Observation	Your Observation
transportConnector name **Note** In the Prime Home configuration file, this field is specified as jms_brokerUrl.	"nio" uri="nio://acshost.domain://0.0.0.0:7400"/	

For information on the fields listed in the worksheet, see Table 5-1.

Setting Up Cloudera Flume

Cloudera Flume provides a platform to transfer a large volume of data from the Prime Home components to MongoDB. MongoDB is the data storage solution for STUN and system audit data.

Before deploying Cloudera Flume in your network infrastructure, be sure that you have the appropriate network planning information for Prime Home functions. The network planning information includes whether you are setting up:

- Prime Home system auditing application
- Taze STUN application

If you set up these applications in a Prime Home setup that manages a large number of devices, the volume of data transfer and rate of notifications between these applications and MongoDB will be high. You need to set up dedicated flows from these applications to MongoDB. Prime Home provides the package required to allow Cloudera Flume flows to connect to MongoDB.

Cloudera Flume is installed on all Prime Home components and the database server. You can deploy Cloudera Flume in both Multiple node and High Availability environments. For High Availability setup, you can deploy Cloudera Flume agents in peer arrangements on hosts in the network.

To set up Cloudera Flume:

Step 1 Log into the ACS hosts and database server as root.

Step 2 Download the Cloudera Flume repository definition from http://archive.cloudera.com/redhat/6/x86_64/cdh/cloudera-cdh3.repo.

Step 3 Add the Cloudera Flume repository definition to the local repository /etc/yum.repos.d/ of all ACS hosts and the database server, using the following command:

```
# cat <<EOF > /etc/yum.repos.d/cloudera-cdh3.repo
[cloudera-cdh3]
name=Cloudera´s Distribution for Hadoop, Version 3
mirrorlist=http://archive.cloudera.com/redhat/6/x86_64/cdh/3/mirrors
gpgkey = http://archive.cloudera.com/redhat/6/x86_64/cdh/RPM-GPG-KEY-cloudera
gpgcheck = 1
EOF
```

Step 4 Install the Cloudera Flume packages, flume-node and flume-master, on the database server, using the following command:

```
# yum -y install flume-node flume-master
```

On the ACS hosts, install only flume-node package using the following command:

```
# yum -y install flume-node
```

Step 5 Modify the Flume master server configuraton to include the database host details. Using the example text shown here, do the following:

- Replace dbhost.domain with the <*Database host*>
- Include the flume.plugin.classes property in the configuration

```
# mkdir -p /var/flume && chown flume:flume /var/flume
# cat <<EOF > /etc/flume/conf/flume-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl"  href="configuration.xsl"?>
<!--
 Licensed to Cloudera, Inc. under one
 or more contributor license agreements.  See the NOTICE file
 distributed with this work for additional information
 regarding copyright ownership.  Cloudera, Inc. licenses this file
 to you under the Apache License, Version 2.0 (the
 "License"); you may not use this file except in compliance
 with the License.  You may obtain a copy of the License at
     http://www.apache.org/licenses/LICENSE-2.0
 Unless required by applicable law or agreed to in writing, software
 distributed under the License is distributed on an "AS IS" BASIS,
 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 See the License for the specific language governing permissions and
 limitations under the License.
-->
<!-- site specific configuration variables should go here. -->
<configuration>

  <property>
    <name>flume.master.servers</name>
    <value>dbhost.domain</value>
    <description>This is the address for the config servers status
    server (http)
    </description>
  </property>

  <property>
    <name>flume.agent.logdir</name>
    <value>/var/flume/flume-\${user.name}/agent</value>
    <description> This is the directory that write-ahead logging data
    or disk-failover data is collected from applications gets
    written to. The agent watches this directory.
    </description>
  </property>

  <property>
    <name>flume.collector.dfs.dir</name>
    <value>file:///var/flume/flume-\${user.name}/collected</value>
    <description>This is a dfs directory that is the the final resting
    place for logs to be stored in.  This defaults to a local dir in
    /tmp but can be hadoop URI path that such as hdfs://namenode/path/
    </description>
  </property>

  <property>
    <name>flume.master.zk.logdir</name>
    <value>/var/flume/flume-\${user.name}-zk</value>
```

```
    <description>The base directory in which the ZBCS stores data.</description>
  </property>

  <property>
    <name>flume.plugin.classes</name>
<value>com.clearaccess.clearsight.flume.MongoDBAppendSink,com.clearaccess.clearsight.flume
.MongoDBSummaryUpdateSink</value>
    <description></description>
  </property>

</configuration>
EOF
```

Step 6 Start the Flume service on the database server using the following commands:

```
# chkconfig flume-master on
# service flume-master start
```

Step 7 Connect to the Flume master and configure bindings on the database server.

In the following example, replace:

- dbhost.domain with the <*Database host*>
- acshost.domain with the <*ACS host*>

```
# flume shell
# connect dbhost.domain

    # Audit Logging
    exec map dbhost.domain dbhost.domain.collector.audit
    exec map dbhost.domain dbhost.domain.agent.audit
    exec map acshost.domain acshost.domain.agent.audit
    exec config dbhost.domain.collector.audit audit.flow 'collectorSource(35854)'
'collector(10000) {
csMongoAppendSink("mongodb://dbhost.domain/cv.audit.log?maxpoolsize=17", "16") }'
    exec config dbhost.domain.agent.audit audit.flow 'avroSource(12346)'
'agentE2ESink("dbhost.domain", 35854)'
    exec config acshost.domain.agent.audit audit.flow 'avroSource(12346)'
'agentE2ESink("dbhost.domain", 35854)'
```

(Optional) If you have deployed the STUN server in your network infrastructure, run the following commands else skip this part:

```
    # STUN
        exec map dbhost.domain dbhost.domain.collector.stun
        exec map dbhost.domain dbhost.domain.agent.stun
        exec map acshost.domain acshost.domain.agent.stun

        exec config dbhost.domain.collector.stun stun.flow 'collectorSource(35853)'
    'collector(10000) {
    csMongoUpdateSink("mongodb://dbhost.domain/cv.stun.summary?maxpoolsize=17",
    "cvClusterId,oui,sn", "16") }'
        exec config acshost.domain.agent.stun stun.flow 'avroSource(12345)'
    'agentE2ESink("dbhost.domain", 35853)'
        exec config dbhost.domain.agent.stun stun.flow 'avroSource(12345)'
    'agentE2ESink("dbhost.domain", 35853)'
```

Step 8 Obtain the flume.tar file that contains the Flume library, and unpack it to /usr/lib/flume/lib using the following commands:

```
# dir=`pwd`
# cd /usr/lib/flume/lib
# tar -xf "${dir}/flume.tar"
# cd "${dir}"
```

Step 9 Open a web browser on the Flumemaster host, and launch Cloudera Flume using the following URL:

```
http://localhost:35871
```

Note If the web browser is not available on the Flumemaster host, you must forward the port 35871 over SSH to a system on which the web browser is available and launch Cloudera Flume.

Observations - Cloudera Flume Setup

Table 3-7 is a worksheet that you must fill out after you set up Cloudera Flume.

Table 3-7 Worksheet - Cloudera Flume Setup

Cloudera Flume Setup	Sample Observation	Your Observation
Flume_URL	http://flumemaster:35871	

Setting up Cisco Taze

You can deploy Cisco Taze in your network infrastructure to provide STUN functionalities. Session Traversal Utilities for NAT (STUN) helps you to manage the devices behind a NAT gateway.

Before you deploy Cisco Taze in your network infrastructure, verify that the following prerequisites have been met:

- A pair of publicly routable IP addresses that Cisco Taze uses to listen and respond to is available.
- A pair of UDP ports, 3478 and 3479, on each of the publicly routable IP addresses is available.
- Devices can access the publicly routable IP addresses with UDP ports 3478 and 3479.
- Network connectivity between Cisco Taze and Fuse ActiveMQ is available. This allows the Cisco Taze to access the JMS queues on the Fuse ActiveMQ servers, and process solicit requests between the ACS and the CPE.
- Cloudera Flume is deployed. Cloudera Flume captures binding request messages and transfers the message data to MongoDB. For information on how to set up Cloudera Flume, see Setting Up Cloudera Flume, page 3-15.

You can deploy Cisco Taze in multiple node or High Availability environments, but only one host at a given time can be the active Cisco Taze host.

To set up the Cisco Taze:

Step 1 Log into Cisco Taze host as root.

Step 2 Copy the Cisco Taze package into the /opt/clearvision/taze directory.

Step 3 Copy the Cisco Taze init script to /etc/init.d.

Step 4 Enable the Cisco Taze startup using the following command:

```
chkconfig taze on
```

Step 5 Modify the taze.conf file using the following command:

```
# vi taze.conf
```

The value of avroport in the taze.conf file must match the avrosource value in the Flume master configuration because the Flume node listens on the avroport.

The taze.conf file must include the following settings:

```
stun.primary.address=19.15.45.15
stun.secondary.address=19.15.45.16
stun.primary.port=3478
stun.secondary.port=3479
activeMQ.brokerURL=failover:(nio://amqhost1:7400,nio://amqhost2:7400)
clearprobe.rrds-path=rrds
clearprobe.port=9090
clearsight.enabled=true
clearsight.eventHost=stunhost.fqdn.com
clearsight.stun.avroHost=localhost
clearsight.stun.avroPort=12345
clearsight.stun.pool.maxActive=50
clearsight.stun.pool.maxIdle=10
clearsight.stun.pool.minIdle=10
clearsight.stun.pool.maxWait=5000
```

Observations - Cisco Taze Setup

Table 3-8 is a worksheet that you must fill out after you set up Cisco Taze.

Table 3-8 Worksheet - Cisco Taze Setup

Cisco Taze Setup	Sample Observation	Your Observation
taze_broker_maxConnections	50	
taze_broker_url	nio://amqhost:7400	
taze_enabled	True	
Check whether STUN binding requests are logged into MongoDB.	Yes	
Check whether the Cisco Taze service is able to respond to UDP connection requests from remote hosts.	Yes	
If the audit function is enabled, verify whether binding requests from devices appear in the Cisco Prime Home log.	Yes	

For information on the fields listed in the worksheet, see Table 5-1.