High Availability

Consider how many users and if you really need HA. If you're running 1000 or fewer users (one might in fact argue this number to be more like 2000 or 3000 users) and on a solid virtual environment that can do snapshotting and automatic host migration consider just a single server. Saves complexity and it's not needed from a load perspective.

Design Considerations

For High Availability (HA), 3 servers are required. This is a requirement of MongoDB (Database (MongoDB)).

Media relay and sipXbridge do not work properly in HA configuration. You'll additionally need a Session Border Controller (Session Border Controllers (SBCs)).

In sipXcom voicemail and media services are not HA (they are in our commercial openUC product) but they can be moved to different servers.

For servers 4GB of RAM is a bare minimum. If you need your system to be robust enough and be highly available, consider not going with bare minimums on any servers... 2 to 4 cores and 8 GB of RAM on each server with at least 80GB of disk on each.

DNS is critical to this setup. Consider that you'll probably need to run 'inside' dns and 'outside' dns (i.e., split dns) if you need connectivity from the Internet. Make sure you understand DNS concepts for sipXcom (DNS Concepts for sipXcom)

High Level Overview

(Somebody please expand on this with Screenshots)
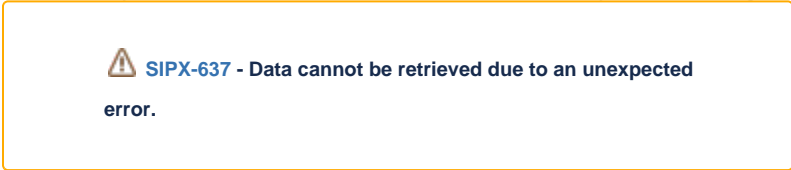
First Server

Pick a SIP domain that you can reference from inside and outside (if needed).

Install the first server from ISO (or RPM if you are more familiar with the installation procedures). Tell the setup script that this is the first server.

Get into the Admin GUI.

Add 2 additional servers (servers can not have NAT between them). Note that I personally add one server at a time.

Note: Primary server should be rebooted and NTP should be sync'd and configured before adding secondaries (

> ⚠️ **SIPX-637 - Data cannot be retrieved due to an unexpected error.**

)

Additional Servers

Install servers 2 and 3.

During setup on those nodes you'll need to indicate that these are not the first servers in the cluster and you'll need to provide the server number that was assigned to these servers when you added them in the admin GUI.

Once the servers are all up they should show in the Admin GUI (System -> Servers).

Distribute Services

Now you can go about distributing your services across the servers... Start with DNS and database.

Make sure that DNS is running on all 3 servers.

The next piece of order is to add databases to each of the servers (either databases on all 3 or databases on 2 and an arbiter on the third, I usually do the former unless my arbiter server is tiny).

Note: An arbiter is a voting member of the MongoDB cluster. A majority of servers must vote to elect a new master if one of the servers is to go down.

Now, distribute services across the servers. You'll want Proxy and Registrar to be on servers with databases (never on an arbiter). They (particularly Registrar) use the database heavily so having it local to those services saves A LOT of network traffic.

What to do Next

If you are using something other than sipXcom's DHCP, make sure that multiple DNS servers are handed out. Note that sipXcom's DHCP services are not HA.

Get some phones to register to the cluster. If you are using manually configured phones you'll always want to register against the SIP domain and not an individual server (kinda kills the idea of high availability).

Lastly, consider your trunking and remote user scenarios. The highest available system in the world is no good if you rely on a single SBC and single path in for your trunking. Also, if availability is so critical to you... don't use the Internet for your trunking. The quality of your connection to your trunking provide determines the quality of your SIP trunking solution.