389 Server Build Documentation

LDAP

Installing 389 on Centos or Redhat is actually very easy once Salt Minion is installed, because they both require the EPEL yum repository to be added. From there the salt file that installs 389-DS is very easy and accomplished with the package module which already knows how to install and uninstall rpm's from yum...

The 389-ds/init.sls file simply creates an ldap user, and installs 389-ds.

Commit Directory

It is when you want to install a directory into 389-ds where installation gets more involved. Fortunately, we can do much of it from the command line.

Essentially there are three parts to this, the setup perl script which is included in the 389-ds packages installed on the client, the setup.inf file which it uses to configure the new directory, and multi-master.ldif which initiates a dual multi-master connection between two nodes (one assumed to be ldap-dev1 and ldap-dev2).

Mullti-master.ldif uses Jinja templates to make slight alterations for whether it is going to ldap-dev1 or ldap-dev2.

In this way we can give our servers a sense of architecture, or where they fit in a much greater whole. The challenge, which is a challenge with any configuration management program, is how to describe that architecture in a dynamic way. How does the installation change if we add another ldap server so that they are all mutli-master? We can bypass this conundrum if we simplify the multi-master role to simply be two servers, and make any extension to them be copy-only. But only slightly, as the replication agreement is still requires actions on both members of that replication scheme.

Also perplexing at the moment is how to handle SSL files. We are using a script which will generate new self-signed certs for each installation. This would certainly be required for any additional ldap server iterated into the cluster. However those certs will need to be added all over the cluster, and to the java servers authenticating to ldap, once created. how to do that is an issue that we don't have a good answer to at the moment. One solution would be to create our own certificate authority, which would be the only cert required to add to every server.