# vt-ldap to ldaptive migration for LDAP access



Many components of Grouper may optionally access LDAP

1. Subject API if your person subjects are stored in LDAP.
2. Grouper Loader if you load groups into Grouper from LDAP.
3. Grouper Web Services if authentication there is via LDAP BINDs.
4. PSPNG if you provision groups to LDAP.

In Grouper 2.3, #1-3 above used vt-ldap and #4 used ldaptive. In Grouper 2.4, all of the above uses ldaptive. Now in Grouper 2.4, #1-3 uses common configuration via grouper-loader.properties and uses an abstraction layer to make any future migrations much easier. And #4 still uses the separate configuration as it was used in Grouper 2.3, but will migrate to using the same configuration in the future.

Note that the migration to ldaptive is being done because vt-ldap is no longer supported and has been deprecated for a long time.

## Migration for Subject API

- Credentials are no longer stored in the subject.properties file (also formally the sources.xml file). So the following options are no longer valid in that file.
    - INITIAL_CONTEXT_FACTORY
    - PROVIDER_URL
    - SECURITY_AUTHENTICATION
    - SECURITY_PRINCIPAL
    - SECURITY_CREDENTIALS
    - subjectApi.source.*.param.ldapProperties_file.value (subject.properties no longer uses external property sources, this can be migrated to grouper-loader.properties)
    - VTLDAP_* (pooling config migrated to grouper-loader.properties)

- Instead you must specify a new property in subject.properties. "example" should be replaced with the name of your source. And "personLdap" should be replaced with what your ldap configuration is called in grouper-loader.properties.

```
subjectApi.source.example.param.ldapServerId.value = personLdap
```

- If you have trouble using the new ldaptive based subject source, you can revert to the vt-ldap based subject source used in Grouper 2.3 by using this configuration in subject.properties. (Though also inform the Grouper developers via Jira or email in case a fix is needed.)

```
subjectApi.source.example.adapterClass = edu.internet2.middleware.subject.provider.
LdapSourceAdapterLegacy
```

## Migration for Grouper Loader

- Changes may not be needed here since the loader was already using the grouper-loader.properties file. However, if you used vt-ldap specific properties, changes may be needed.
- If you have trouble using ldaptive, you can revert back to vt-ldap using this configuration in grouper.properties. (Though also inform the Grouper developers via Jira or email in case a fix is needed.)

```
ldap.implementation.className = edu.internet2.middleware.grouper.ldap.vtldap.VTLdapSessionImpl
```

## Migration for Grouper Web Services

- Changes may not be needed here since the web services were already using the grouper-loader.properties file. However, if you used vt-ldap specific properties, changes may be needed.
- If you have trouble using ldaptive, you can revert back to vt-ldap using this configuration in grouper.properties. (Though also inform the Grouper developers via Jira or email in case a fix is needed.)

```
ldap.implementation.className = edu.internet2.middleware.grouper.ldap.vtldap.VTLdapSessionImpl
```

## Configuration options

The following applies to the subject api, loader, and web services.

- Look at the grouper-loader.base.properties file for the latest configuration options.
- Your configuration should go in grouper-loader.properties.
- ldap.<connection name>.url is the only property that's required (unless you specify the url using the configFileFromClasspath configuration).
- The default pooling validator is SearchValidator.  And the validation is done on connection checkout by default.  ConnectLdapValidator is no longer valid.
- ldap.personLdap.validatorCompareSearchFilterString has been changed to ldap.personLdap.validatorCompareAttribute and ldap.personLdap.validatorCompareValue
- Search result handlers have changed, although there is a shim to map the old ones.

```
#################################
## LDAP connections
#################################
# specify the ldap connection with user, pass, url
# the string after "ldap." is the ID of the connection, and it should not have
# spaces or other special chars in it.  In this case is it "personLdap"

#note the URL should start with ldap: or ldaps: if it is SSL.
#It should contain the server and port (optional if not default), and baseDn,
#e.g. ldaps://ldapserver.school.edu:636/dc=school,dc=edu
#ldap.personLdap.url = ldaps://ldapserver.school.edu:636/dc=school,dc=edu

# load this ldaptive config file before the configs here.  load from classpath
#ldap.personLdap.configFileFromClasspath = ldap.personLdap.properties

#optional, if authenticated
#ldap.personLdap.user = uid=someapp,ou=people,dc=myschool,dc=edu

#optional, if authenticated, note the password can be stored encrypted in an external file
#ldap.personLdap.pass = secret

#optional, if you are using tls, set this to true.  Generally you will not be using an SSL URL to use TLS...
#ldap.personLdap.tls = false

#optional, if using sasl
#ldap.personLdap.saslAuthorizationId =
#ldap.personLdap.saslRealm =

#optional (note, time limit is for search operations, timeout is for connection timeouts),
#most of these default to ldaptive defaults.  times are in millis
#validateOnCheckout defaults to true if all other validate methods are false
#ldap.personLdap.batchSize =
#ldap.personLdap.countLimit =
#ldap.personLdap.timeLimit =
#ldap.personLdap.timeout =
#ldap.personLdap.minPoolSize =
#ldap.personLdap.maxPoolSize =
#ldap.personLdap.validateOnCheckIn =
#ldap.personLdap.validateOnCheckOut =
#ldap.personLdap.validatePeriodically =
#ldap.personLdap.validateTimerPeriod =
#ldap.personLdap.pruneTimerPeriod =
# if there is a max size limit on ldap server, then this will retrieve results in pages
#ldap.personLdap.pagedResultsSize =
# set to 'follow' if using AD and using paged results size and need this for some reason (generally you
shouldnt)
#ldap.personLdap.referral =
# validator setup, currently supports CompareLdapValidator and SearchValidator.  additional properties below
for CompareLdapValidator.
ldap.personLdap.validator = SearchValidator
#ldap.personLdap.validator = CompareLdapValidator
#ldap.personLdap.validatorCompareDn = ou=people,dc=example,dc=com
#ldap.personLdap.validatorCompareAttribute = ou
#ldap.personLdap.validatorCompareValue = people
# comma-delimited list of classes to process LDAP search results. Useful if AD returns a ranged attribute for
large
# groups (e.g., member;range=0-1499); include the GrouperRangeEntryHandler to handle progressive fetching.
#ldap.personLdap.searchResultHandlers=org.ldaptive.handler.DnAttributeEntryHandler,edu.internet2.middleware.
grouper.ldap.ldaptive.GrouperRangeEntryHandler
# comma-delimited list of result codes (org.ldaptive.ResultCode) to ignore, e.g. TIME_LIMIT_EXCEEDED,
SIZE_LIMIT_EXCEEDED, PARTIAL_RESULTS
#ldap.personLdap.searchIgnoreResultCodes=SIZE_LIMIT_EXCEEDED
```