

# Securing Your Open Source Geospatial Stack with Single Sign On

Ian Turton

# Introduction

I'm Ian Turton, I work for Astun Technology providing coding, support and training for a range of open source programs:

- GeoServer
- QGis
- MapServer
- Python for GIS
- PostGIS

In my spare time I'm a moderator at [gis.stackexchange.com](https://gis.stackexchange.com) and a regular contributor to GeoTools and GeoServer.

## Get the slides



# The Problem

- Provide a single login for QGIS users to WMS layers in GeoServer and PostGIS data tables
- Restrict access by team or user
- GUI for administration team
- Use existing Azure Active Directory

## What is Single Sign On (SSO)?

*Single sign-on (SSO) is a session and user authentication service that permits a user to use one set of login credentials – for example, a username and password – to access multiple applications. This is managed by providing a **federated identity system**.*

## Implementation

- Both GeoServer and PostGIS can make use of an LDAP service to provide authentication via a federated identity system
- LDAP provides the central store of identities and GeoServer and PostGIS pass username/password pairs to the LDAP server to check if the user is known to the system.

# GeoServer

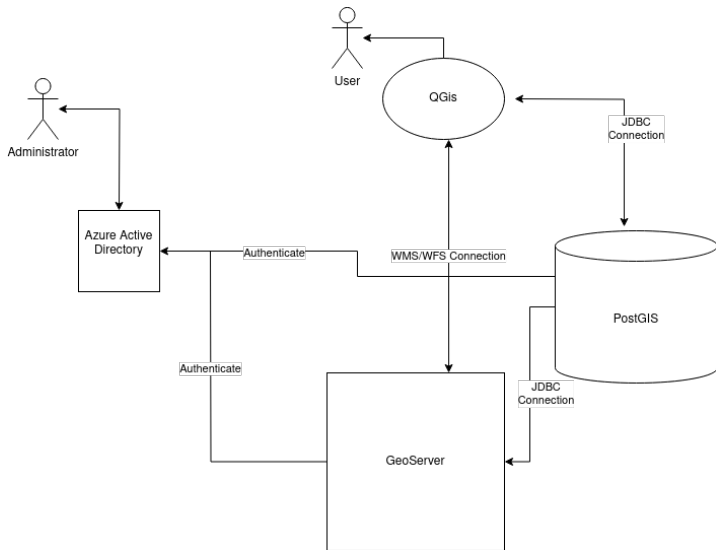
- GeoServer uses the built in LDAP Authentication Provider
- allows the use of users and groups for authorization
- same configuration as any other custom authentication provider (GUI)

## PostGIS

*This authentication method operates similarly to password except that it uses LDAP as the password verification method. **LDAP is used only to validate the user name/password pairs.** Therefore **the user must already exist in the database** before LDAP can be used for authentication.*

Add a line to the `pg_hba.conf` file in the same way as any other authentication method (e.g. peer, md5)





# The Pivot

- Corporate IT team inform client there is no way that the GIS team are altering entries in the Active Directory!
- Back to the drawing board
- Can we replace an expensive piece of Microsoft code with some open source tools.

# Providing LDAP

- OpenLDAP (<https://www.openldap.org/>)
  - Provides a complete LDAP server
  - allows the import of users and groups (from exports from AD)
  - Comes pre-dockerized for easy installation

## Add users to LDAP Server

```
# cat adam.ldif
dn: uid=adam,ou=users,dc=tgs,dc=com
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
cn: adam
uid: adam
uidNumber: 16859
gidNumber: 100

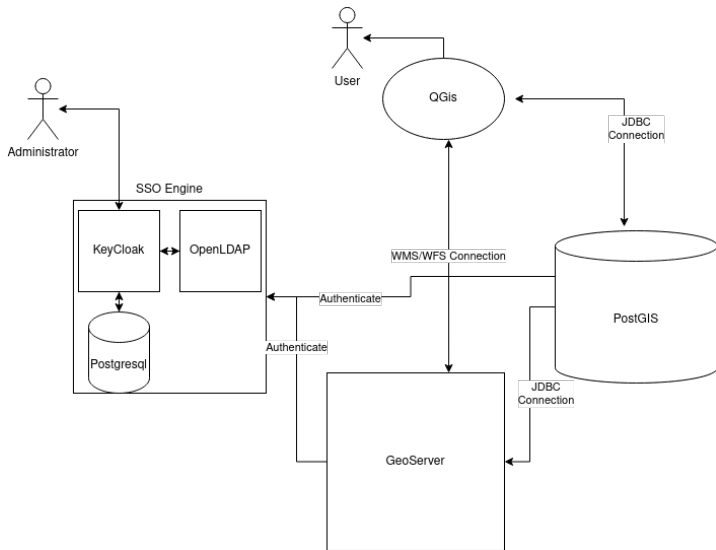
ldapadd -x -W -D "cn=ramesh,dc=tgs,dc=com" -f adam.ldif
```

## Providing a “nice” GUI for Administrators

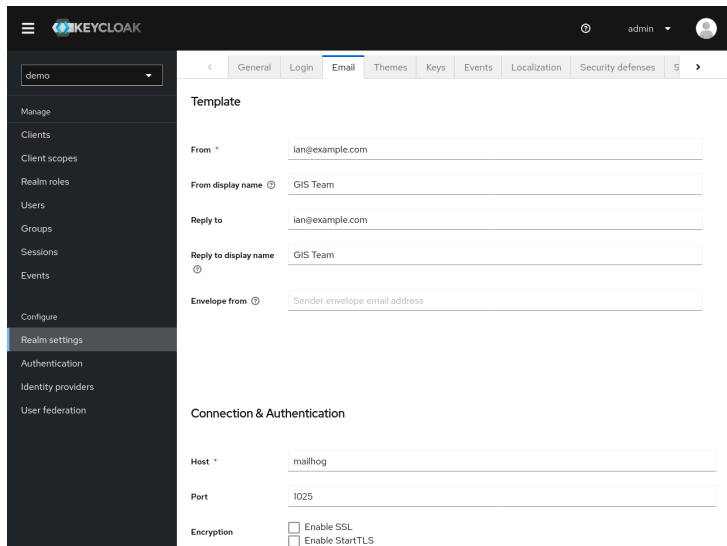
- These guys are GIS experts not IT experts
- So no CLI and python scripts :-(  
■ Deploy KeyCloak to manage the human to LDAP interactions

# KeyCloak

- KeyCloak (<https://www.keycloak.org/>)
- Can be used to handle all your identity provider needs
- GeoServer support with an extension
- No PostGIS support!
- So use in federation mode, as a front end to OpenLDAP
- Comes pre-dockerized (with PostgreSQL in a container)



# Set up Keycloak Administration



The screenshot displays the Keycloak Administration Console interface. The top navigation bar shows the Keycloak logo, a user profile icon for 'admin', and a search icon. The left sidebar contains a menu with options: Manage, Clients, Client scopes, Realm roles, Users, Groups, Sessions, Events, Configure, Realm settings (highlighted), Authentication, Identity providers, and User federation. The main content area is titled 'Email' and includes a breadcrumb trail: < General Login Email Themes Keys Events Localization Security defenses S >. The 'Email' configuration is divided into two sections: 'Template' and 'Connection & Authentication'. The 'Template' section contains fields for 'From' (ian@example.com), 'From display name' (GIS Team), 'Reply to' (ian@example.com), 'Reply to display name' (GIS Team), and 'Envelope from' (Sender envelope email address). The 'Connection & Authentication' section contains fields for 'Host' (mailhog), 'Port' (1025), and 'Encryption' options (Enable SSL and Enable StartTLS, both unchecked).

**Template**

From \*

From display name ⓘ

Reply to

Reply to display name ⓘ

Envelope from ⓘ

**Connection & Authentication**

Host \*

Port

Encryption  Enable SSL  Enable StartTLS



# Connect to the LDAP server

The screenshot displays the Keycloak Admin Console interface for configuring an LDAP identity provider. The left sidebar shows the navigation menu with 'User federation' selected. The main content area is titled 'Connection and authentication settings' and includes the following fields and controls:

- Vendor:** A dropdown menu set to 'Other'.
- Connection URL:** An empty text input field.
- Enable StartTLS:** A toggle switch currently set to 'Off'.
- Use Truststore SPI:** A dropdown menu set to 'Only for Idaps'.
- Connection pooling:** A toggle switch currently set to 'Off'.
- Connection timeout:** An empty text input field with a clear button.
- Test connection:** A blue button to verify the connection.
- Bind type:** A dropdown menu set to 'simple'.
- Bind DN:** A text input field containing 'cn=admin,dc=demo,dc=co,dc=uk'.
- Bind credentials:** An empty text input field with a visibility toggle.
- Test authentication:** A blue button to test the authentication process.
- Save / Cancel:** Buttons at the bottom of the configuration area.

On the right side, a 'Jump to section' menu lists various configuration categories, with 'General options' currently selected.

The screenshot displays the Keycloak administration interface. The top navigation bar includes the Keycloak logo, a user profile icon labeled 'admin', and a search icon. The left sidebar contains a menu with the following items: 'demo' (selected), 'Manage', 'Clients', 'Client scopes', 'Realm roles', 'Users', 'Groups', 'Sessions', 'Events', 'Configure', 'Realm settings', 'Authentication', 'Identity providers', 'User federation', and 'User federation' (highlighted). The main content area is titled 'LDAP searching and updating' and contains the following settings:

- Edit mode**: WRITABLE
- Users DN**: ou=users,dc=demo,dc=co,dc=uk
- Username LDAP attribute**: uid
- RDN LDAP attribute**: uid
- UUID LDAP attribute**: entryUUID
- User object classes**: person, organizationalPerson, inetOrgPerson
- User LDAP filter**: (empty)
- Search scope**: One Level
- Read timeout**: (empty)
- Pagination**: Off

At the bottom of the settings area are 'Save' and 'Cancel' buttons. On the right side, there is a 'Jump to section' menu with the following options: 'General options', 'Connection and authentication settings', 'LDAP searching and updating' (selected), 'Synchronization settings', 'Kerberos integration', 'Cache settings', and 'Advanced settings'.

# Configure Rules for mapping between KeyCloak and LDAP

The screenshot displays the Keycloak Admin Console interface. The top navigation bar shows the Keycloak logo and the user 'admin'. The left sidebar contains a menu with options like 'Manage', 'Clients', 'Client scopes', 'Realm roles', 'Users', 'Groups', 'Sessions', 'Events', 'Configure', 'Realm settings', 'Authentication', 'Identity providers', and 'User federation'. The main content area is titled 'Create new mapper' and shows the configuration for a 'group-ldap-mapper'. The configuration fields are as follows:

- Name:** group-copy
- Mapper type:** group-ldap-mapper
- LDAP Groups DN:** ou=groups,dc=demo,dc=co,dc=uk
- Group Name LDAP Attribute:** cn
- Group Object Classes:** groupOfUniqueNames
- Preserve Group Inheritance:** On
- Ignore Missing Groups:** Off
- Membership LDAP Attribute:** uniqueMember
- Membership Attribute Type:** DN
- Membership User LDAP Attribute:** cn

# Create Group and User “Organizations” in LDAP

The screenshot shows the phpLDAPadmin interface. On the left is a navigation tree with the following structure:

- dc=demo, dc=co, dc=uk (2)
  - ou=groups (3)
    - cn=default-roles-demo
    - cn=freddy
    - cn=ian-group
  - ou=users (1)
    - cn=ian

The main content area displays the details for the entry **ou=groups**. The header indicates the server is **ldap**, the distinguished name is **ou=groups,dc=demo,dc=co,dc=uk**, and the template is **Default**. A list of actions is available:

- Refresh
- Switch Template
- Copy or move this entry
- Rename
- Create a child entry (Hint: To delete an attribute, empty the text field and click save.)
- View 3 children (Hint: To view the schema for an attribute, click the attribute name.)
- Show internal attributes
- Export
- Delete this entry
- Compare with another entry
- Add new attribute
- Export subtree

The schema section shows the following attributes:

- objectClass** (required):
  - organizationalUnit (structural)
  - top (add value)
- ou** (required, otn):
  - groups (add value, rename)

An **Update Object** button is located at the bottom right of the main content area.

1.2.5

# Create groups in LDAP or KeyCloak

The screenshot displays the phpLDAPadmin interface for managing an LDAP directory. The main panel shows the configuration for the entry `cn=energy`. The server is identified as `ldap` with a distinguished name of `cn=energy,ou=groups,dc=demo,dc=co,dc=uk`. The entry is currently using the `Default` template.

On the left sidebar, the directory tree shows the following structure:

- dc=demo, dc=co, dc=uk (2)
  - ou=groups (4)
    - cn=energy
    - cn=forestry
    - cn=rural
    - cn=sales\_and\_lettings
    - Create new entry here
  - ou=users (63)
    - Create new entry here

The main configuration area for `cn=energy` includes the following fields and options:

- cn** (required, rdn): energy (add value, rename)
- objectClass** (required):
  - groupOfUniqueNames (structural)
  - top (add value)
- uniqueMember** (required): cn=empty-membership-placeholder

Available actions for this entry include: Refresh, Switch Template, Copy or move this entry, Rename, Create a child entry, Show internal attributes, Export, Delete this entry, Compare with another entry, and Add new attribute. Hints are provided for deleting attributes and viewing schemas.

# Create Users in LDAP via KeyCloak or import

**KEYCLOAK** admin

Users > Create user

**Create user**  Enabled Ac

**Username \***

**Email**

**Email verified**  Off

**First name**

**Last name**

**Required user actions**

**Groups**

# A user in LDAP (GDPR redacted)

**cn** required

(add value)

**Email** alias

(add value)

**givenName**

(add value)

**objectClass** required

person

inetOrgPerson (structural)

organizationalPerson

top

(add value)

**sn** required

(add value)

**User Name** alias, rdn

(add value) (rename)

# Configure GeoServer Authentication Provider

The screenshot shows the GeoServer web interface. At the top, it says "Logged in as admin." with a "Logout" button and a language dropdown set to "en". The left sidebar contains a navigation menu with sections: "About & Status", "Data", "Services", "Settings", "Tile Caching", and "Security". The "Security" section is expanded, showing "Authentication" as the active sub-section.

The main content area is titled "LDAP Authentication Provider ldap-demo" and includes the subtitle "Authentication via Lightweight Directory Access Protocol server". The configuration is organized into several sections:

- Name:** ldap-demo
- LDAP Settings:**
  - Server URL: ldap://localhost/dc=demo,dc=co,dc=uk
  - TLS
  - User lookup pattern: udi={0},ou=users
  - Filter used to lookup user: [empty field]
  - Format used for user login name: [empty field]
- Authorization:**
  - Use LDAP groups for authorization
  - Bind user before searching for groups
  - Group search base: ou=groups
  - Group search filter: (memberOf={0})
  - Group to use as ADMIN: admins
  - Group to use as GROUP\_ADMIN: admins
  - Enable Hierarchical groups search
  - Max. depth for hierarchical groups search: 10
  - Nested group search filter: (members={1})

On the right side, there are input fields for "Username" and "Password", and a "Test Connection" button. At the bottom of the configuration area, there are "Save" and "Cancel" buttons.



# Configure GeoServer Role Provider

Role service stored in LDAP repository

**Settings** Roles

Name  
ldap-demo-roles

Administrator role  
ROLE\_ADMIN

Group administrator role  
ROLE\_ADMIN

**LDAP Settings**

Server URL  
ldap://localhost

TLS

Group search base  
ou=groups,dc=demo,dc=co,dc=uk

Group user membership search filter  
uniqueMember=uid=0,ou=users,dc=demo,dc=co,dc=uk

All groups search filter

Filter used to lookup user

Role prefix  
ROLE\_

Convert Role To Upper Case

**Authentication**

Authenticate to extract roles

Username  
cn=admin,dc=demo,dc=co,dc=uk

Password  
\*\*\*\*\*

**Hierarchical groups options**

Enable Hierarchical groups search

Save Cancel

# Configure GeoServer Groups

The screenshot shows the GeoServer web interface. At the top, it says "Logged in as admin." with a "Logout" button and a language dropdown set to "en". The main content area is titled "LDAP User Group Service ldap-demo-groups" and includes a description: "User group service provider which sources its information from a LDAP directory". There are three tabs: "Settings", "Users", and "Groups", with "Settings" selected. The settings are organized into sections:
 

- Names:** A text input field containing "ldap-demo-groups".
- Passwords:**
  - Password encryption: "Strong PBE" (dropdown menu).
  - Recode existing passwords: "default" (dropdown menu).
- LDAP Settings:**
  - Server URL: "ldap://localhost" (text input).
  - TLS:  (checkbox).
- Group lookup options:**
  - Group search base: "ou=groups,dc=demo,dc=co,dc=uk" (text input).
  - Filter to search all groups (leave blank to derive from attribute): "(objectClass=groupOFUniqueNames)" (text input).
  - Filter to search group by name (leave blank to derive from attribute): (text input).
  - Attribute which contains the name of the group (leave blank to derive from name filter): (text input).
- User/group mapping options:**
  - Query format to retrieve the user/group mapping (leave blank to derive from attribute): "(uniqueMember=uid={0},ou=users,dc=demo,dc=co,dc=uk)" (text input).
  - Attribute name to retrieve the user/group mapping (leave blank to derive from filter): (text input).
- User lookup options:**
  - User search base: "ou=users,dc=demo,dc=co,dc=uk" (text input).

 The left sidebar contains a navigation menu with categories: "About & Status", "Data", "Services", "Settings", "Tile Caching", and "Security".

# All the LDAP groups are imported automatically



Logged in as admin.

Logout

en

## About & Status

- Server Status
- GeoServer Logs
- Contact Information
- About GeoServer
- Process status

## Data

- Layer Preview
- Workspaces
- Stores
- Layers
- Layer Groups
- Styles

## Services

- WFS
- WCS
- WMS
- WMTS
- WPS

## Settings

- Global

## LDAP User Group Service ldap-demo-groups

User group service provider which sources its information from a LDAP directory

Settings Users Groups

**i** This user group service is read-only.

Group list

<< < 1 > >> Results 1 to 5 (out of 5 items)

<input type="checkbox"/>	Groupname	Enabled
<input type="checkbox"/>	admins	✓
<input type="checkbox"/>	energy	✓
<input type="checkbox"/>	forestry	✓
<input type="checkbox"/>	rural	✓
<input type="checkbox"/>	sales_and_lettings	✓






<< < 1 > >> Results 1 to 5 (out of 5 items)

# Don't forget to make the role service active




Log

## About & Status

-  Server Status
-  GeoServer Logs
-  Contact Information
-  About GeoServer
-  Process status

## Data

-  Layer Preview
-  Workspaces
-  Stores

## Security Settings

Configure security settings

Active role service

ldap-demo-roles ▾

### Encryption

Encrypt web admin URL parameters



Password encryption

Weak PBE ▾

 Strong cryptography available

# Limit access to a workspace (as per usual)

er

Logged in as admin. [Logout](#)  en 

## New data access rule

Configure a new data access rule

Global layer group rule

Workspace

Layer and groups

Access mode

### Roles

Grant access to any role

Available Roles	Selected Roles
<input type="checkbox"/> Admin	<input type="checkbox"/> ROLE_ENERGY
<input type="checkbox"/> ROLE_ANONYMOUS	<input type="checkbox"/> GROUP_ADMIN
<input type="checkbox"/> ROLE_AUTHENTICATED	<input type="checkbox"/> ROLE_ADMINS
<input type="checkbox"/> ROLE_FORESTRY	
<input type="checkbox"/> ROLE_RURAL	
<input type="checkbox"/> ROLE_SALES_AND_LETTINGS	

[Save](#) [Cancel](#)

# Configure ldap-sync to keep PostGIS up to date

```
ldap_connection:  
  host: localhost  
  port: 389  
  auth:  
    method: :simple  
    username: CN=admin,DC=demo,DC=co,DC=uk  
    password: admin
```

```
# Search parameters for LDAP users which should be synchronized
```

```
ldap_users:
```

```
base: ou=users,DC=demo,DC=co,DC=uk
```

```
# LDAP filter (according to RFC 2254)
```

```
# defines to users in LDAP to be synchronized
```

```
filter: (&(objectClass=person)(cn=*))
```

```
# this attribute is used as PG role name
```

```
name_attribute: uid
```

```
# lowercase name for use as PG role name
```

```
lowercase_name: true
```

```
# Add lowercase name *and* original name for use as PG role names (useful)
```

```
bothcase_name: false
```

```
# Search parameters for LDAP groups which should be synchronized
```

```
ldap_groups:
```

```
base: ou=groups,dc=demo,dc=co,dc=uk
```

```
filter: objectClass=groupOfUniqueNames
```

```
# this attribute is used as PG role name
```

```
name_attribute: cn
```

## Configure a cron job

```
*/1 * * * * pg_ldap_sync -c ldap-sync-config.yaml >>  
/var/log/sync.log 2>&1
```

- run the sync job every minute to copy any new LDAP users or groups to PostgreSQL



```
I, [2022-11-24T16:53:01 #24436] INFO -- : found user-dn: uid=iturton,cn=u
I, [2022-11-24T16:53:01 #24436] INFO -- : found user-dn: uid=test,cn=user
I, [2022-11-24T16:53:01 #24436] INFO -- : found group-dn: cn=users,dc=gal
I, [2022-11-24T16:53:01 #24436] INFO -- : found pg-user: "iturton"
I, [2022-11-24T16:53:01 #24436] INFO -- : found pg-user: "test"
I, [2022-11-24T16:53:01 #24436] INFO -- : found pg-group: "users" with me
I, [2022-11-24T16:53:01 #24436] INFO -- : user stat: create: 0 drop: 0 ke
I, [2022-11-24T16:53:01 #24436] INFO -- : group stat: create: 0 drop: 0 k
```

## Modify the `pg_hba.conf` file to let PostgreSQL know who to trust

```
hostssl    all    all    127.0.0.1/32    ldap ldapserver=ldap-machine  
           ldapbinddn="cn=admin,dc=demo,dc=co,dc=uk" ldapbindpasswd=secret!  
           ldapbasedn="ou=users,dc=demo,dc=co,dc=uk"
```

```
psql -U astun -h demo-db -d "dbname=postgres  
sslmode=require"
```

# Problems

- managing email authentication
- not being flagged as phishing attempts by Microsoft Outlook
- custom KeyCloak event triggers so we can send custom emails
- Automating initial import of users, write a python script

## Conclusions

- It is possible to provide a single sign on service to QGIS users to GeoServer and PostGIS
- KeyCloak is a very simple and intuitive interface to LDAP

## Get the slides

