LDAP AUTHENTICATION

New in version Jewel.

You can delegate the Ceph Object Gateway authentication to an LDAP server.

HOW IT WORKS

The Ceph Object Gateway extracts the users LDAP credentials from a token. A search filter is constructed with the user name. The Ceph Object Gateway uses the configured service account to search the directory for a matching entry. If an entry is found, the Ceph Object Gateway attempts to bind to the found distinguished name with the password from the token. If the credentials are valid, the bind will succeed, and the Ceph Object Gateway will grant access.

You can limit the allowed users by setting the base for the search to a specific organizational unit or by specifying a custom search filter, for example requiring specific group membership, custom object classes, or attributes.

REQUIREMENTS

- **LDAP or Active Directory:** A running LDAP instance accessible by the Ceph Object Gateway
- **Service account:** LDAP credentials to be used by the Ceph Object Gateway with search permissions
- **User account:** At least one user account in the LDAP directory
- **Do not overlap LDAP and local users:** You should not use the same user names for local users and for users being authenticated by using LDAP. The Ceph Object Gateway cannot distinguish them and it treats them as the same user.

SANITY CHECKS

Use the `ldapsearch` utility to verify the service account or the LDAP connection:

```
# ldapsearch -x -D "uid=ceph,ou=system,dc=example,dc=com" -W \
-H ldaps://example.com -b "ou=users,dc=example,dc=com" 'uid=*' dn
```

Note: Make sure to use the same LDAP parameters like in the Ceph configuration file to eliminate possible problems.

CONFIGURING THE CEPH OBJECT GATEWAY TO USE LDAP AUTHENTICATION

The following parameters in the Ceph configuration file are related to the LDAP authentication:

- `rgw_ldap_uri`: Specifies the LDAP server to use. Make sure to use the `ldaps://<fqdn>:<port>` parameter to not transmit clear text credentials over the wire.
- `rgw_ldap_binddn`: The Distinguished Name (DN) of the service account used by the Ceph Object Gateway
- `rgw_ldap_secret`: The password for the service account
- `rgw_ldap_searchdn`: Specifies the base in the directory information tree for searching users. This might be your users organizational unit or some more specific Organizational Unit (OU).
- `rgw_ldap_dnattr`: The attribute being used in the constructed search filter to match a username. Depending on your Directory Information Tree (DIT) this would probably be `uid` or `cn`.
- `rgw_search_filter`: If not specified, the Ceph Object Gateway automatically constructs the search filter with the `rgw_ldap_dnattr` setting. Use this parameter to narrow the list of allowed users in very flexible ways. Consult the *Using a custom search filter to limit user access section* for details

USING A CUSTOM SEARCH FILTER TO LIMIT USER ACCESS

There are two ways to use the `rgw_search_filter` parameter:

SPECIFYING A PARTIAL FILTER TO FURTHER LIMIT THE CONSTRUCTED SEARCH FILTER

An example for a partial filter:

```
"objectclass=inetorgperson"
```

The Ceph Object Gateway will generate the search filter as usual with the user name from the token and the value of `rgw_ldap_dnattr`. The constructed filter is then combined with the partial filter from the `rgw_search_filter` attribute. Depending on the user name and the settings the final search filter might become:

```
"(&(uid=hari)(objectclass=inetorgperson))"
```

So user `hari` will only be granted access if he is found in the LDAP directory, has an object class of `inetorgperson`, and did specify a valid password.

SPECIFYING A COMPLETE FILTER

A complete filter must contain a `USERNAME` token which will be substituted with the user name during the authentication attempt. The `rgw_ldap_dnattr` parameter is not used anymore in this case. For example, to limit valid users to a specific group, use the following filter:

```
"(&(uid=USERNAME)(memberOf=cn=ceph-users,ou=groups,dc=mycompany,dc=com))"
```

Note: Using the `memberOf` attribute in LDAP searches requires server side support from you specific LDAP server implementation.

GENERATING AN ACCESS TOKEN FOR LDAP AUTHENTICATION

The `radosgw-token` utility generates the access token based on the LDAP user name and password. It will output a base-64 encoded string which is the access token.

```
# export RGW_ACCESS_KEY_ID="<username>"
# export RGW_SECRET_ACCESS_KEY="<password>"
# radosgw-token --encode --ttype=ldap
```

Note: For Active Directroy use the `--ttype=ad` parameter.

Important: The access token is a base-64 encoded JSON struct and contains the LDAP credentials as a clear text.

TESTING ACCESS

Use your favorite S3 client and specify the token as the access key.