Подключение к LDAP-совместимому внешнему источнику учетных записей

Оглавление

Настройка функциональности

Настройка параметров подключения

RooX UIDM поддерживает одновременную работу с несколькими источниками записей по LDAP. Например, пользователи могут находиться в разных кустах одного домена, на разных серверах одного леса, либо на серверах с различным ПО: часть в MS AD, а часть в OpenLDAP.

Настройка нескольких подключений LDAP на данный момент не поддержана через helm chart. Такая возможность будет добавлена позднее

Для настройки нескольких источников необходимо использовать названия конфигурационных параметров вида

```
com.rooxteam.uidm.ldapv3repository.{source_id}.{property_name}={property_value}
```

где `{source_id}` - условное название для данного источника учетных записей. Например, `main` и `secondary`.

Если предполагается работа только с одним набором УЗ, то указание `{source_id}` можно опустить.

Описание настроек в таблице ниже приведено в варианте без указания `{source_id}`.

При установке на виртуальные машины используйте параметры с пометкой `env.properties`. При установке в k8s - с пометкой `Values`.

Описание	Включение LDAP-совместимого источника учетных записей
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.enabled`
Параметр Values	`.Values.rx.uidm.ldapv3repository.enabled`
Пример	`true`

Описание	Имя пользователя от имени которого sso-server будет выполнять поиск данных
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.superuserId`
Параметр Values	`.Values.rx.uidm.ldapv3repository.superuserId`
Пример	`cn=uidm_super_user\,cn=Users\,dc=demo\,dc=exmaple\,dc=com`

Описание	Пароль для пользователя от имени которого sso-server будет выполнять поиск данных
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.superuserPassword`
Параметр Values	`.Values.rx.uidm.ldapsv3repository.superuserPassword`
Пример	`SuperUserPa$$w0rd`

Описание	Имя сервера
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.ldapServerName`
Параметр Values	`.Values.rx.uidm.ldapv3repository.ldapServerName`
Пример	`dc.example.com`

Описание	Порт для подключения к серверу
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.ldapPort`
Параметр Values	`.Values.rx.uidm.ldapv3repository.ldapPort`
Пример	`389`

Описание	Ветка, от которой начинается поиск данных
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.baseDN`
Параметр Values	`.Values.rx.uidm.ldapv3repository.baseDN`
Пример	`ou=Users\,dc=example\,dc=com`

Описание	Имя атрибута AD, по которому будет выполняться поиск пользователя
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.namingAttribute`
Параметр Values	`.Values.rx.uidm.ldapv3repository.namingAttribute`

Пример	`userPrincipalName`

Описание	LDAP-filter, которая будет использоваться при поиске УЗ в LDAP-репозитории. Значение должно формировать валидный фильтр при заключении в скобки.
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.userSearchFilter`
Параметр Values	`.Values.rx.uidm.ldapv3repository.userSearchFilter`
Пример	`| (memberOf=CN=RRMSFO_BIUSER) (memberOf=CN=RRMSFO_APPADMIN)`

Описание	Флаг использования защищенного соединение. Если флаг установлен в true, также необходимо выполнить действия, описанные в разделе Настройка подключения по LDAPS
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.ssl`
Параметр Values	`.Values.rx.uidm.ldapv3repository.ssl.enabled`
Пример	`false`

Настройка правил работы с атрибутами учетной записи

RooX UIDM позволяет настраивать, какие атрибуты учетной записи из AD должны быть запрошены и сохранены в УЗ во внутренней БД UIDM. Такие атрибуты в дальнейшем могут быть переданы в токене доступа пользователя в соответствующих клеймах.

Описание	Список аттрибутов AD, которые sso-server будет запрашивать
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.profileAttributesToFetch`
Параметр Values	`.Values.rx.uidm.ldapv3repository.profileAttributesToFetch`
Пример	`displayName,memberOf,mail,mobile,sAMAccountName,userPrincipalName`

Описание	Перечень атрибутов, которые должны быть сохранены в профиль пользователя. Перечень допустимых значений приведен в таблице ниже. Если атрибут был указан в параметре `profileAttributesToFetch`, но для него не был задана настройка в данном параметре, значение атрибута будет добавлено в `extendedAttributes` в профиле учетной записи.
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.attributes`
Параметр Values	`.Values.rx.uidm.ldapv3repository.attributes`
Пример	`DISPLAY_NAME_ATTRIBUTE,EMAIL_ATTRIBUTE,MOBILE_ATTRIBUTE,UID_ATTRIBUTE,ROLES_ATTRIBUTE`

Описание	После успешной аутентификации пользователя в AD, UIDM создаст (или обновит) информацию об учетной записи в своей БД. В качестве логина в этой учетной записи будет использовано значение

атрибута из ответа LDAP-сервера, указанного в данной настройке. Как правило, совпадает с атрибутом, заданным в параметре `namingAttribute`

Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.attributes.use_as_login`
Параметр Values	`.Values.rx.uidm.ldapv3repository.attributes_use_as_login`
Пример	`userPrincipalName`

В следующей таблице приведены возможные значения для параметра `attributes`. Если соответствующее значение из таблицы ниже встречается в параметре `attributes`, то UIDM скопирует значение указанного в таблице параметра учетной записи из AD в указанный параметр учетной записи в RooX UIDM. Если в attributes указано 2 значения, отвечающие за наполнение одного и того же атрибута RooX UIDM (например, `MOBILE_ATTRIBUTE` и `CONTACT_PHONE_ATTRIBUTE`) и из ldap получены оба атрибута-источника, результат не определен.

Значение параметра `attributes`	Атрибут AD	Атрибут RooX UIDM	в БД RooX UIDM
`DISPLAY_NAME_ATTRIBUTE`	`displayName`	`displayName`	`Person.displayNameNat`
`EMAIL_ATTRIBUTE`	`mail`	`contactEmail`	`Contact.address (type=email)`
`MOBILE_ATTRIBUTE`	`mobile`	`contactPhone`	`Contact.address (type=phone)`
`CONTACT_PHONE_ATTRIBUTE`	`telephoneNumber`	`contactPhone`	`Contact.address (type=phone)`
`PHONE_ATTRIBUTE`	`telephoneNumber`	`telephoneNumber`	`Principal.msisdn`
`FIRST_NAME_ATTRIBUTE`	`givenName`	`givenname`	`Person.firstNameNat`
`LAST_NAME_ATTRIBUTE`	`sn`	`sn`	`Person.lastNameNat`
`LOGIN_ATTRIBUTE`	`userPrincipalName`	`login`	`Credentials.login`
`UID_ATTRIBUTE`	`uid`	`uid`	`Principal.id`
`ROLES_ATTRIBUTE`	`memberOf`	`roles`	таблица `PrincipalRoles`

Настройка правил определения ролей и организационных ролей пользователя

RooX UIDM опционально позволяет настроить какие роли необходимо назначить пользователю в зависимости от его вхождения в группы в AD.

RooX UIDM также поддерживает механизм организационных ролей, позволяющий задавать одному пользователю разные роли в разных организациях. Организационные роли также могут быть назначены на основе данных из AD. В текущей версии документации эти возможности не описаны. При необходимости обратитесь в техническую поддержку RooX за консультацией.

RooX UIDM оперирует только с CN-компонентом от имени группы. Так для группы с distinguished name

`CN=Group1,DC=dc,DC=demo,DC=rooxteam,DC=com` UIDM будет использовать имя `Group1`

Описание	RegEx-шаблон для разрешенных имён групп. Опциональный параметр. По умолчанию разрешены любые имена (`.*`)
Параметр env.properties	`com.rooxteam.sso.autologin.agw.allowed_roles_pattern`
Параметр Values	В текущей версии чартов не поддерживается
Пример	`.*`

Описание	RegEx-шаблон для запрещённых имён групп. Опциональный параметр. По умолчанию запрещены имена системных ролей RooX UIDM
Параметр env.properties	`com.rooxteam.sso.autologin.agw.prohibited_roles_pattern`
Параметр Values	В текущей версии чартов не поддерживается
Пример	`(?i)^(role_)?(system|provision|antifraud)$`

Описание	Правила определения ролей по группам пользователя. Опциональный параметр. Подробное описание см. ниже.
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.groups_to_roles_mapping`
Параметр Values	`.Values.rx.uidm.ldapv3repository.groups_to_roles_mapping`
Пример	`rooxgroup1=rooxrole1;rooxrole2,rooxgroup_all=rooxrole_all`

Описание	Роли по умолчанию, которые добавляются всем пользователям. Опциональный параметр, по умолчанию пуст.
Параметр env.properties	`com.rooxteam.uidm.ldapv3repository.defaultRoles`
Параметр Values	В текущей версии чартов не поддерживается
Пример	`defaultRole1,defaultRole2`

При обработке полученных от AD данных UIDM производит фильтрацию групп, оставляя только те, которые подходят под регулярное выражение из параметра `allowed_roles_pattern` и не подходят под выражение из `prohibited_roles_pattern`.

Для оставшихся значений происходит определение ролей по именам группы согласно параметру `groups_to_roles_mapping`.

В качестве значения параметра `groups_to_roles_mapping` можно указать разделенный запятой список в следующим формате

```
<groupName1>=<roleName1>;<roleName2>..<roleNameN>,<groupName2>=<roleName3>;<roleName4>..
```

```
<roleNameM>
```

, где:

- `<groupNameX>` - имя группы в AD
- `<roleNameX>` - имя роли, которую надо присвоить пользователю

Таким образом, если пользователь является в AD членом группы `<groupName1>`, ему будут присвоены роли `<roleName1>, <roleName2>, .. <roleNameN>`

Если параметр `groups_to_roles_mapping` не задан, то пользователю будут присвоены роли, одноименные всем группам, которые остались после фильтрации.

Последним шагом пользователю присваиваются все роли по умолчанию, заданные в параметре `defaultRoles`

Настройка подключения по LDAPS

Для настройки подключения к серверу по защищенному протоколу, необходимо:

1. Задать параметр `com.rooxteam.uidm.ldapv3repository.ssl=true` (или аналогичный для чарта)
2. Указать порт, соответствующий протоколу LDAPS: `com.rooxteam.uidm.ldapv3repository.ldapPort=636` (или аналогичный для чарта)
3. Положить в хранилище сертификат сервера либо авторизационного центра, который выдал сертификат серверу. Для этого:
   - для версий, поставляемых как нативные сервисы,
     i. необходимо добавить сертификат в хранилище `keytool -importcert -file /tmp/ldap_certificate.cer -alias cert_name -keystore /etc/pki/java/cacerts -storepass chageit -noprompt`
   - для версий, поставляемых в helm-чартах
     i. `kubectl create secret generic roox-ca-certs --from-file=certfile=./ldap_certificate.cer`
     ii. при установке компонента roox-sso-server указать название созданного секрета: `.Values.rx.uidm.ldapv3repository.ssl.kubernetesSecretName: roox-ca-certs`