# Интеграция HashiCorp Vault со службой каталога ALD Pro


02/16/2026

# Содержание

# 1 Введение

HashiCorp Vault - инструмент для безопасного хранения и управления секретами, разработанный компанией HashiCorp и распространяемый с открытым исходным кодом. Под секретами здесь понимается любая конфиденциальная информация: пароли, токены, API-ключи, SSH-доступы, сертификаты и другие чувствительные данные.

Интеграция позволяет добавить в Vault такие методы аутентификации, как Kerberos и LDAP, а также добавить ротацию секретов для заданных учетных записей, используя LDAP Secret Engine.

Для демонстрации интеграции использовалось.

- dc-1.ald.company.lan - контроллер домена ALD Pro 3.1
- vault.ald.company.lan – сервер, введенный в доменПодготовка учетных записей

# 2 Сервисная учетная запись

Для работы с LDAP нужно создать сервисную учетную запись. Для этого на контроллере домена создайте файл **service.ldif** со следующей конфигурацией, предварительно заменив значения параметров **dn, uid и userPassword**на желаемые.

```
dn: uid=vault_srv,cn=sysaccounts,cn=etc,dc=ald,dc=company,dc=lan

objectClass: account

objectClass: simplesecurityobject

uid: vault_srv

userPassword: <SecretPass>

passwordExpirationTime: 20380119031407Z

nsIdleTimeout: 0
```

Примените конфигурацию, выполнив следующую команду:

```
ldapadd -H ldaps://dc-1.ald.company.lan -D "cn=Directory Manager" -W -f service.ldif
&& rm service.ldif
```

Для работы Kerberos создайте учетную запись для керберизированного сервиса:

```
ipa service-add HTTP/vault.ald.company.lan
```

Далее на сервере vault.ald.company.lan загрузите ключ сервиса

```
ipa-getkeytab -p HTTP/vault.ald.company.lan -k /etc/vault.keytab
```

Проверка полученного ключа:

```
klist -k /etc/vault.keytab -e
```

**Важно!**

Для ключа необходимо выставить права той учетной записи, из-под которой у вас запущен vault. В данном демонстрационном варианте используется пользователь root.

```
chown root:root /etc/vault.keytab

chmod 700 /etc/vault.keytab
```

# 3 Настройка Kerberos-аутентификации

Первым шагом нужно активировать модуль Kerberos-аутентификации:

```
vault auth enable -passthrough-request-headers=Authorization -allowed-response-
headers=www-authenticate kerberos
```

Проверка примененных параметров:

```
vault read sys/auth/kerberos/
```

Преобразовать ранее полученный Kerberos-ключ в base64:

```
base64 /etc/vault.keytab > vault.keytab.base64
```

Запись Kerberos-ключа и имени Kerberos-принципала в конфигурацию vault:

```
vault write auth/kerberos/config keytab=@vault.keytab.base64 service_account="HTTP/
vault.ald.company.lan"
```

где

- keytab – передача преобразованного в base64 Kerberos-ключа;
- service_account – имя кербезированной службы vault.

Добавление учетной записи параметров LDAP, с которым vault будет извлекать данные об аутентифицированных пользователях:

```
vault write auth/kerberos/config/ldap \
binddn="uid=vault_srv,cn=sysaccounts,cn=etc,dc=ald,dc=company,dc=lan" \
bindpass="<SecretPass>" \

groupattr="cn" \

groupdn="cn=groups,cn=accounts,dc=ald,dc=company,dc=lan" \

groupfilter="(&(objectClass=ipausergroup)(member={{.UserDN}}))" \

userdn="cn=users,cn=accounts,dc=ald,dc=company,dc=lan" \

userattr=uid \

url=ldaps://dc-1.ald.company.lan \

userfilter="(&(objectClass=person)(uid={{.Username}}))" \

use_token_groups="false" \

upndomain=""
```

где

- binddn – путь в LDAP-каталоге до сервисной записи, которая была создана на этапе подготовки учетных записей;
- bindpass – пароль от учетной записи;
- groupattr — атрибут, идентифицирующий группу;
- groupdn - путь в LDAP-каталоге до места хранения групп пользователей;
- groupfilter – фильтр, с которым применяется поиск соответствующих пользователю групп;
- userdn – путь в LDAP-каталоге до места хранения пользователей;
- userattr – атрибут, идентифицирующий пользователя;
- url – тип подключения к серверу LDAP;
- userfilter – фильтр, применяемый при поиске пользователей;
- use_token_groups – требуется установить значение false;
- upndomain – требуется установить пустое значение.

**Важно!**

Изменения, внесенные в конфигурации LDAP, могут неочевидным образом менять структуру LDAP-запроса к каталогу. Например, установка значения у параметра **upndomain** или **use_token_groups** принудительно применяет запрос, сформированный специально для Active Directory, и такой запрос не будет принят ALD Pro.

Если требуется, чтобы policy назначались согласно группам LDAP, нужно их добавить в строгом соответствии с названием группы в каталоге (ее атрибуту cn)

Пример добавления группы ipausers и связывание ее с политикой default:

```
vault write auth/kerberos/groups/ipausers policies="default"
```

Проверка примененной конфигурации:

```
vault read auth/kerberos/config/ldap
```

```
vault login -method=kerberos username=<DomainUser> service=HTTP/vault.ald.company.lan
realm=ALD.COMPANY.LAN keytab_path=user.keytab

krb5conf_path=/etc/krb5.conf

disable_fast_negotiation=false
```

где

- method – метод аутентификации;
- username – пользователь, который проходит аутентификацию;
- service – Service Principal Name;
- realm – Kerberos realm;
- keytab_path – путь до keytab **пользователя;**
- krb5conf_path – путь до конфигурации Kerberos-клиента;

- disable_fast_negotiation – статус FAST-бронирования.

Также проверить корректную настройку можно, используя утилиту curl:

```
curl -X POST --negotiate -u : https://vault.ald.company.lan:8200/v1/auth/kerberos/
login
```

Результатом выполнения команды, должен быть переданный токен доступа, с назначенными policy:


```
{
 "request_id": "69511451-70cc-83ef-2693-67ce07510c2f",
 "lease_id": "",
 "renewable": false,
 "lease_duration": 0,
 "data": null,
 "wrap_info": null,
 "warnings": null,
 "auth": {
   "client_token": "SECRET_TOKEN",
   "accessor": " SECRET_TOKEN ",
   "policies": [
     "default"
   ],
   "token_policies": [
     "default"
   ],
   "metadata": {
     "domain": "ALD.COMPANY.LAN",
     "user": "user"
   },
   "lease_duration": 2764800,
   "renewable": false,
   "entity_id": "175b8bdf-a510-bee9-6765-75f0bae83a0f",
   "token_type": "service",
   "orphan": true,
   "mfa_requirement": null,
   "num_uses": 0
 },
 "mount_type": ""
}
```

# 4 Настройка LDAP-аутентификации

Активация LDAP-аутентификации:

```
vault auth enable ldap
```

Добавление учетной записи параметров LDAP, с которым vault будет извлекать данные об аутентифицированных пользователях:

```
vault write auth/ldap/config

url="ldaps://dc-1.ald.company.lan"
 binddn="uid=vault_srv,cn=sysaccounts,cn=etc,dc=ald,dc=company,dc=lan"
 bindpass="<SecretPass>" userdn="cn=users,cn=accounts,dc=ald,dc=company,dc=lan"

userattr="uid"

groupdn="cn=groups,cn=accounts,dc=ald,dc=company,dc=lan"

groupattr="cn" groupfilter="(&(objectClass=posixGroup)(member={{.Username}}))"

starttls=false
```

где

- url – тип подключения к серверу ldap;
- binddn – путь в LDAP-каталоге до сервисной записи, которая была создана на этапе подготовки учетных записей;
- bindpass – пароль от учетной записи;
- userdn – путь в LDAP-каталоге до места хранения пользователей;
- userattr – атрибут, идентифицирующий пользователя;
- groupdn - путь в LDAP-каталоге до места хранения групп пользователей;
- groupattr — атрибут, идентифицирующий группу;
- groupfilter – фильтр, с которым применяется поиск соответствующих пользователю групп;
- starttls – использование подключения с использованием Start-TLS.

Пример LDAP-аутентификации продемонстрирован на рисунке 1.

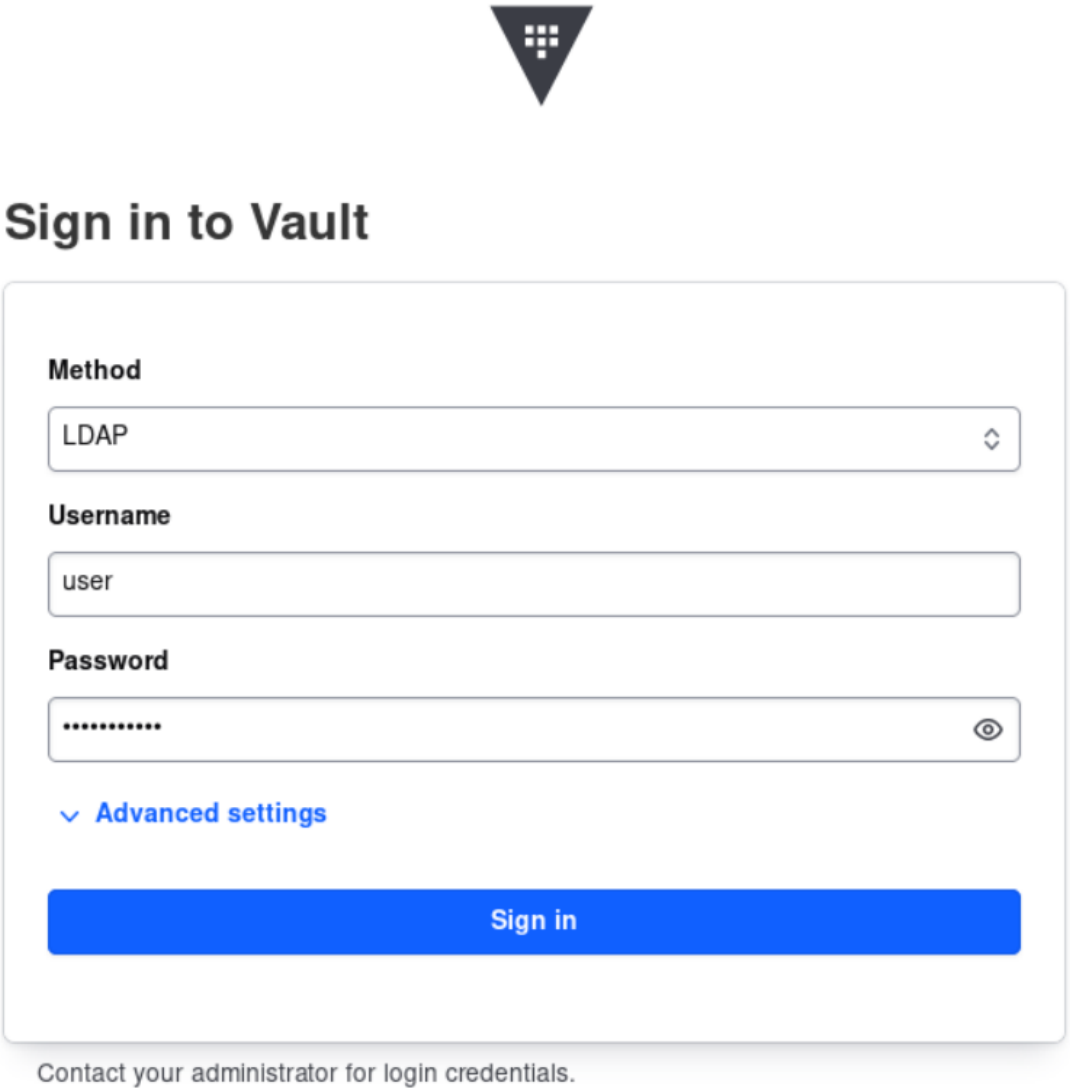


Рисунок 1 – Форма ввода учетных данных LDAP

# 5 Настройка LDAP Secret Engine

В HashiCorp Vault присутствует функционал Secret Engine. Secret Engine – это компоненты системы управления секретами Hashicorp Vault, которые хранят, генерируют или шифруют данные. Такие компоненты могут быть разными хранилищами, одно из которых LDAP.

> Примером использования такого хранилища может быть практика, когда в своем рабочем окружении администратор использует учетную запись с пониженными правами доступа, а для выполнения административных действий использует специальные административные учетные записи или предоставляет специально подготовленные записи на группы пользователей. Например, helpdesk или разработчикам. Благодаря Secret Engine на таких учетных записях можно настроить более строгую ротацию паролей.

Настройка будет продемонстрирована в графическом интерфейсе Vault.

В первую очередь нужно зайти в раздел Secret Engine и активировать в нем LDAP, как изображено на рисунках 2-4.

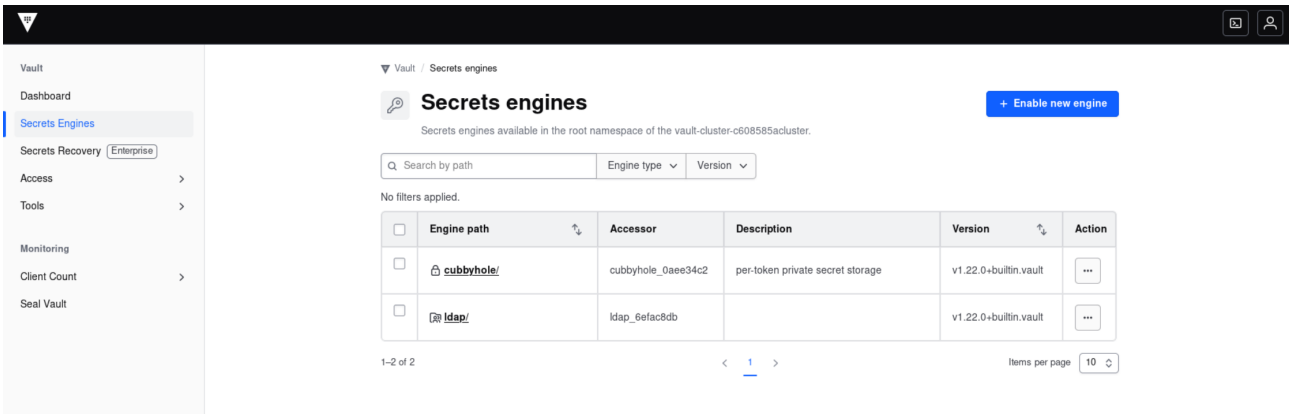


Рисунок 2 – Меню Secrets engines

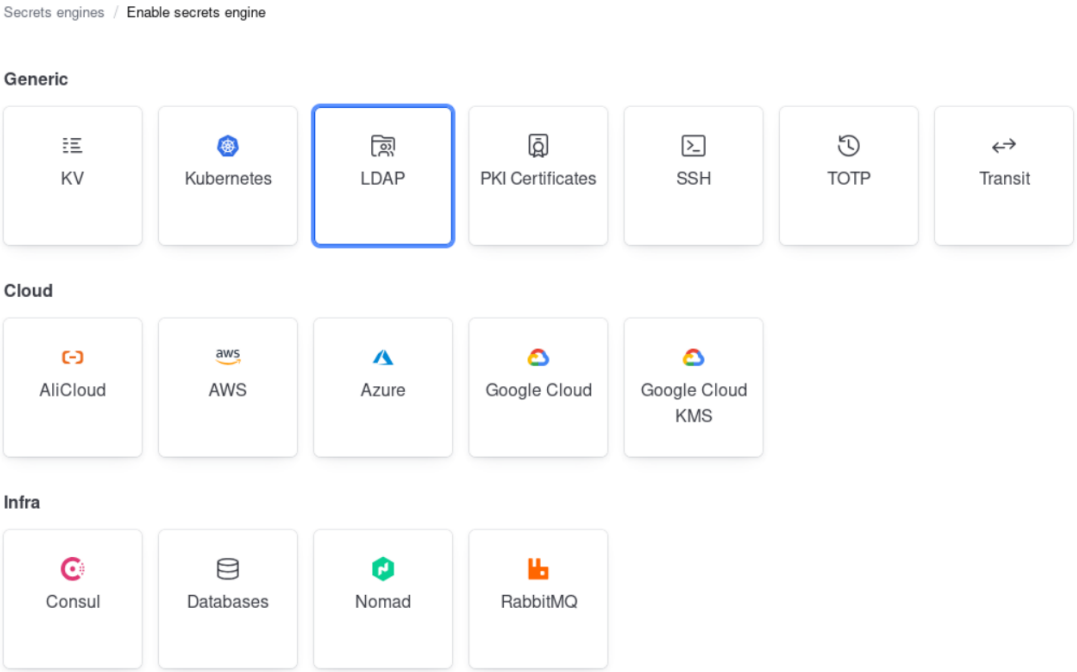


Рисунок 3 – Выбор модуля LDAP

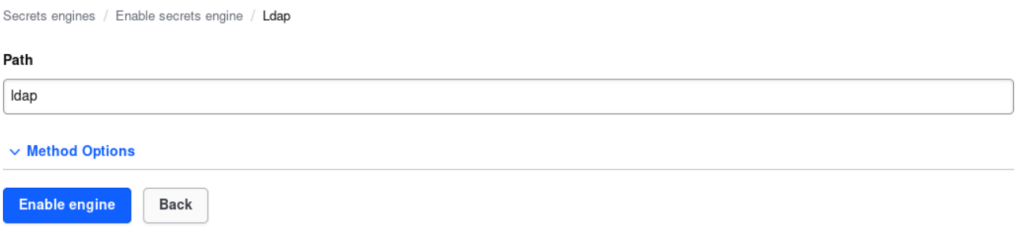


Рисунок 4 – Активация модуля LDAP

После перехода в LDAP engine pass нужно зайти на станицу конфигурации (рис. 5, 6).

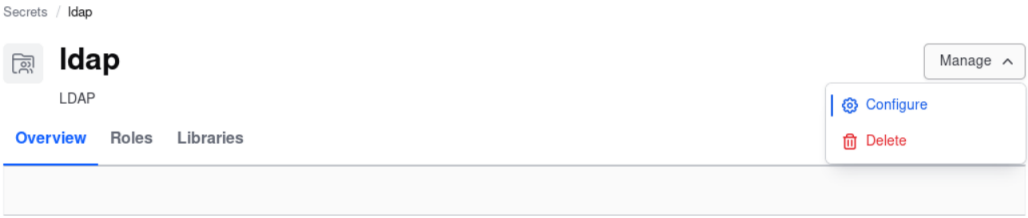


Рисунок 5 – Переход в меню конфигурации LDAP

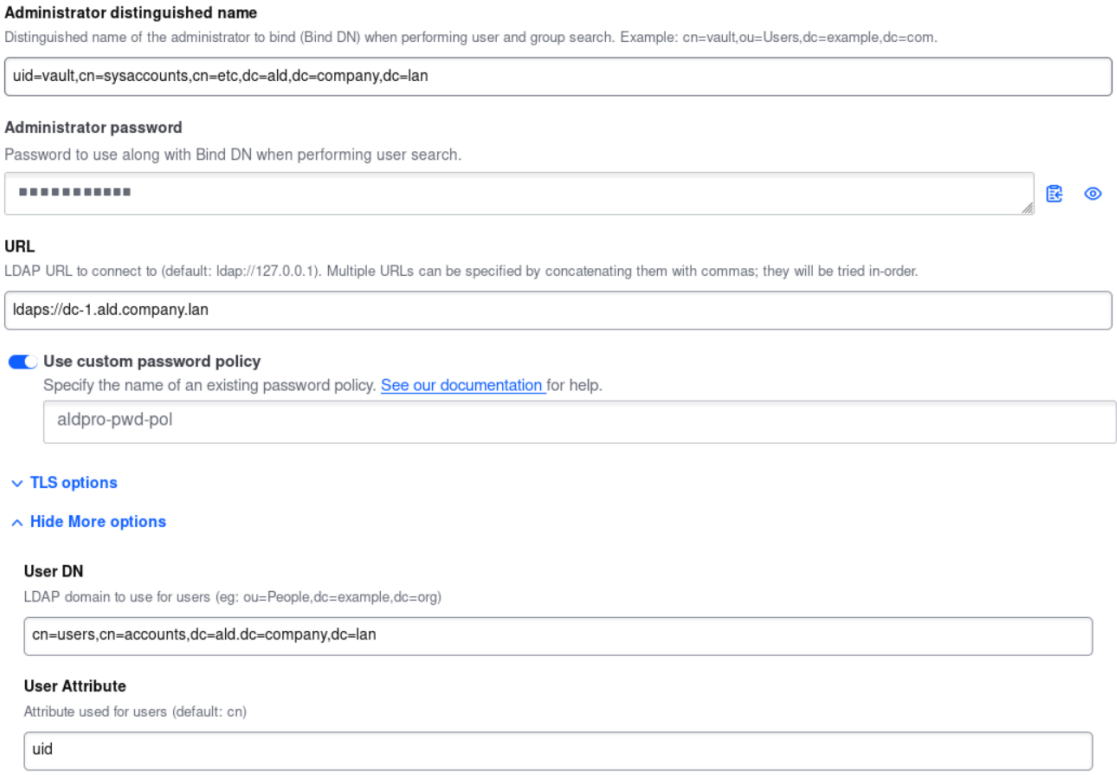


Рисунок 6 – Конфигурация подключения LDAP

Разъяснения:

- Administrator distinguished name – путь к административной учетной записи в LDAP;
- Administrator password – пароль от административной учетной записи в LDAP;
- URL – LDAP URL контроллера домена, с которым будет устанавливаться соединение ldaps;
- Use custom password policy – указывает, какую политику паролей будет использовать данный secret engine. Пример конфигурации будет рассмотрен на следующем шаге;
- User DN – путь, по которому расположены пользователи в LDAP;
- User Attribute – имя атрибута, идентифицирующее пользователя в LDAP.

# 6 Настройка политики паролей

В Vault для разных Secret Engine можно настраивать различные парольные политики. Расширенное описание того, как она формируется, можно посмотреть в документации по ссылке

https://developer.hashicorp.com/vault/docs/concepts/password-policies

Для создания политики паролей потребуется выполнить следующие действия:

1. Создать файл с расширением .hcl:

```
vim aldpropassword-policy.hcl
```

2. Добавить текст политики:

```
length = 64
rule "charset" {
 charset = "abcdefghijklmnopqrstuvwxyz"
 min-chars = 20
}

rule "charset" {
 charset = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
 min-chars = 20
}

rule "charset" {
 charset = "0123456789"
 min-chars = 5
}

rule "charset" {
 charset = "!@#$%"
 min-chars = 5
}
```

3. Загрузить политику:

vault write sys/policies/password/aldpro-pwd-pol=@aldpropassword-policy.hcl

где

- aldpro-pwd-pol - это имя создаваемой политики,
- aldpropassword-policy.hcl – имя файла с текстом политики.

# 7 Создание роли

LDAP Secret Engine поддерживает два режима работы ролей: динамический и статический. В данном руководстве будет рассмотрен только статический способ хранения ролей, поскольку динамический способ создания ролей не соответствует заявленным функциональным возможностям и требует доработок.

В меню создания роли выбрать Static Role и заполнить поля согласно рисунку 7.

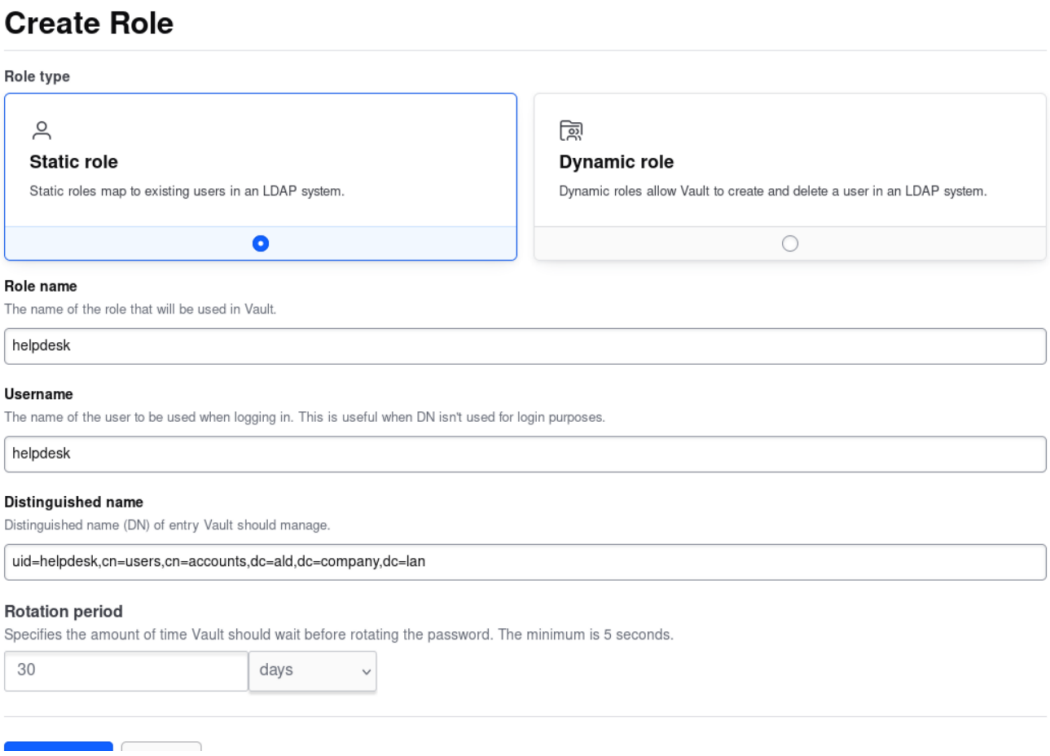


Рисунок 7 – Создание статической роли

Разъяснения:

- Role name – имя, которое создается в Vault;
- Username – значение атрибута uid настраиваемой учетной записи;
- Distinguished name – полный путь до настраиваемой учетной записи в LDAP;
- Rotation period – время ротации пароля.

**Важно!**

Каталог LDAP по своей архитектуре не предназначен для частых изменений записей. Поэтому рекомендуется устанавливать продолжительные периоды ротации паролей для атрибута Rotation period.

После создания роли пользователи, которым будет доступна эта роль, смогут запросить пароль от этой учетной записи, нажав кнопку Get credentials, как изображено на рисунках 8-9. Важно отметить, что пароли будут сгенерированы согласно ранее загруженной политике паролей, а для учетных записей, имеющих krbPrincipalName, будут автоматически обновлены ключи Kerberos.

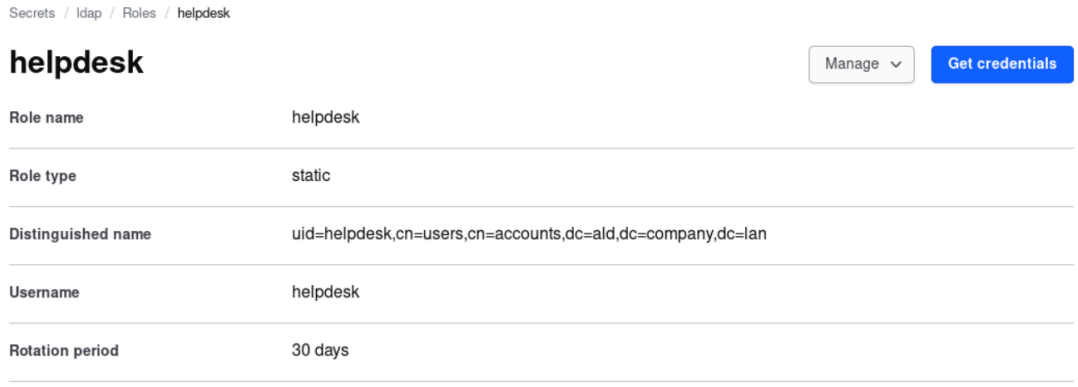


Рисунок 8 – Страница запроса учетных данных

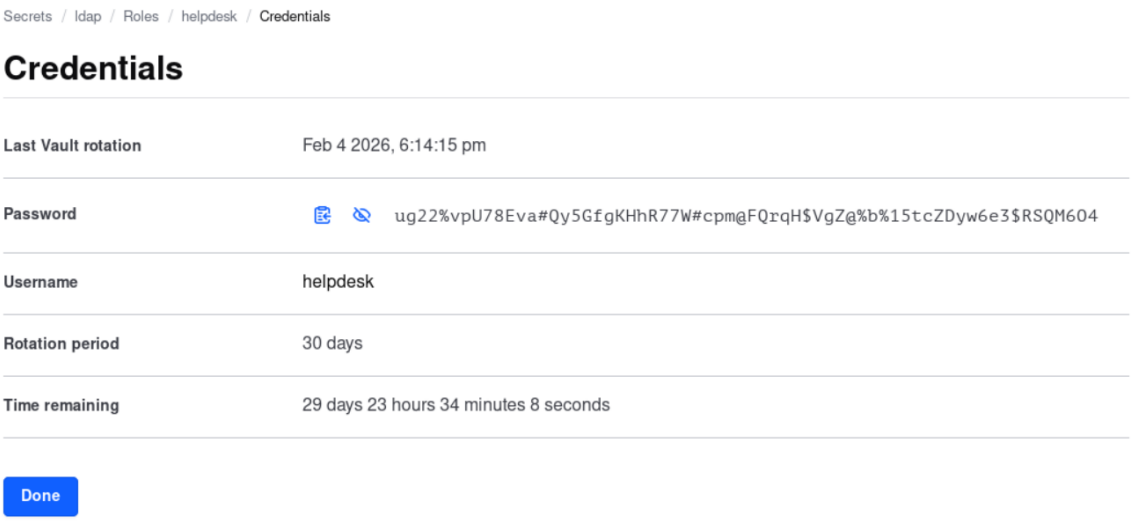


Рисунок 9 – Демонстрация получения пароля