
lowball_ad_auth_provider

Release 1.0.0

unknown

Jul 12, 2021

CONTENTS:

1	Installation	3
2	Pip	5
3	From Source	7
4	Implemented Interface	9
5	Auth Package	11
6	Configuration	13
7	Example Usage	15

lowball-ad-auth-provider is an AuthProvider implementation for a [Lowball](#) microservice, that works with Active Directory.

INSTALLATION

lowball-ad has been tested to work with only Python 3.6+

CHAPTER TWO

PIP

```
pip install lowball-ad-auth-provider
```

CHAPTER THREE

FROM SOURCE

```
git clone https://github.com/EmersonElectricCo/lowball-ad-auth-provider
cd ./lowball-ad-auth-provider
python setup.py install
```


IMPLEMENTED INTERFACE

lowball ad implements the following methods of a Lowball Authentication Provider

- authenticate - this is required for basic authentication with AD
- get_client - service account/password must be configured to work. Enables basic user lookup routes

The following authentication provider dependent builtin routes will be usable with this implementation. Routes related to the authentication database will be available as expected and implemented by the chosen authentication database.

- POST /builtins/auth (login)
- DELETE /builtins/auth (logout)
- GET /builtins/auth (whoami)
- POST /builtins/auth/tokens (create token) - for non admin users only if service account is configured
- GET /builtins/client (get authenticated client) - only if service account is configured
- GET /builtins/client/<client_id> (get client information) - only if service account is configured

AUTH PACKAGE

The Authentication Package which should be sent to *POST /builtins/auth* for authentication

```
{  
  "username": "ad_user",  
  "password": "ad_password"  
}
```


CONFIGURATION

The configuration for the ad auth provider goes under the *auth_provider* section of a lowball configuration

Mandatory Config Fields

base_dn base dn of the of the search path for users

hostname hostname or ip of the server to use

domain domain to prepend in front of user authentications

Optional Config Fields

roll_mappings a dictionary of role -> list of groups that would give a user that role

ignore_ssl_cert_errors true/false, whether or not to validate ssl. Unused if *use_ssl* is set to false

use_ssl true/false, whether or not to use ssl for the connection

service_account username of the service account used to lookup users. can be left empty, but users will not be able to look themselves up or create their own tokens

service_account_password password of the service account

Example Config

```
auth_provider:
  service_account: admin
  service_account_password: myComplexPassword
  base_dn: "dc=example, dc=org"
  domain: corp
  ignore_ssl_cert_errors: false
  use_ssl: true
  role_mappings:
    user:
      - CN=regular_user,OU=groups,DC=example,DC=org
      - CN=owners,OU=groups,DC=example,DC=org
    finance:
      - CN=accounting,OU=groups,DC=example,DC=org
      - CN=owners,OU=groups,DC=example,DC=org
```


EXAMPLE USAGE

```
from lowball_ad_auth_provider import ADAuthProvider
from lowball import Lowball, config_from_file

app = Lowball(config_from_file("/path/to/config"), auth_provider=ADAuthProvider)
```