
slapd Documentation

Release 0.1.4

python-ldap

Oct 01, 2023

CONTENTS

1	Table of contents	3
1.1	API	3
1.2	Changelog	6
2	Indices and tables	7
	Python Module Index	9
	Index	11

slapd helps you to run and control an OpenLDAP instance in a pythonic way. You can use it to test some python ldap-based applications, or set-up development environments.

TABLE OF CONTENTS

1.1 API

```
class slapd.Slapd(host=None, port=None, log_level=30, schemas=None, database='mdb',
                  suffix='dc=slapd-test,dc=python-ldap,dc=org', root_cn='Manager', root_pw='password',
                  configuration_template=None, datadir_prefix=None, debug=None)
```

Controller class for a slapd instance, OpenLDAP's server.

This class creates a temporary data store for slapd, runs it listening on a private Unix domain socket and TCP port, and initializes it with a top-level entry and the root user.

When a reference to an instance of this class is lost, the slapd server is shut down.

An instance can be used as a context manager. When exiting the context manager, the slapd server is shut down and the temporary data store is removed.

Parameters

- **schemas** – A list of schema names or schema paths to load at startup. By default this only contains *core*.
- **host** – The host on which the slapd server will listen to. The default value is *127.0.0.1*.
- **port** – The port on which the slapd server will listen to. If *None* a random available port will be chosen.
- **log_level** – The verbosity of Slapd. The default value is *logging.WARNING*.
- **suffix** – The LDAP suffix for all objects. The default is *dc=slapd-test,dc=python-ldap,dc=org*.
- **root_cn** – The root user common name. The default value is *Manager*.
- **root_pw** – The root user password. The default value is *password*.
- **configuration_template** – An optional initial database template.
- **datadir_prefix** – The prefix of the temporary directory where the slapd configuration and data will be stored. The default value is *python-ldap-test*.
- **debug** – Whether to launch slapd with debug verbosity on. When *True* debug is enabled, when *False* debug is disabled, when *None*, debug is only enable when *log_level* is *logging.DEBUG*. Default value is *None*.

init_tree()

Creates the organization and applicationProcess object.

ldapadd(*ldif*, *extra_args=None*, *expected=0*)

Runs ldapadd on this slapd instance, passing it the ldif content

Parameters

- **ldif** – The ldif content to pass to the *ldapadd* standard input.
- **extra_args** – Extra argument to pass to *ldapadd*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *ldapadd* execution data.

ldapdelete(*dn*, *recursive=False*, *extra_args=None*, *expected=0*)

Runs ldapdelete on this slapd instance, deleting 'dn'

Parameters

- **dn** – The distinguished name of the element to delete.
- **recursive** – Whether to delete sub-elements. Defaults to *False*.
- **extra_args** – Extra argument to pass to *ldapdelete*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *ldapdelete* execution data.

ldapmodify(*ldif*, *extra_args=None*, *expected=0*)

Runs ldapadd on this slapd instance, passing it the ldif content

Parameters

- **ldif** – The ldif content to pass to the *ldapmodify* standard input.
- **extra_args** – Extra argument to pass to *ldapmodify*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *ldapmodify* execution data.

ldapsearch(*filter*, *searchbase=None*, *extra_args=None*, *expected=0*)

Runs search on this slapd instance

Parameters

- **filter** – The search filter.
- **base** – The starting point for the search.
- **extra_args** – Extra argument to pass to *ldapdelete*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *ldapdelete* execution data.

ldapwhoami(*extra_args=None, expected=0*)

Runs ldapwhoami on this slapd instance

Parameters

- **extra_args** – Extra argument to pass to *ldapwhoami*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *ldapwhoami* execution data.

restart()

Restarts the slapd server with same data

slapadd(*ldif, extra_args=None, expected=0*)

Runs slapadd on this slapd instance, passing it the ldif content

Parameters

- **ldif** – The ldif content to pass to the *slapadd* standard input.
- **extra_args** – Extra argument to pass to *slapadd*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *slapadd* execution data.

slapcat(*extra_args=None, expected=0*)

Runs slapcat on this slapd instance, passing it the ldif content

Parameters

- **extra_args** – Extra argument to pass to *slapcat*.
- **expected** (*An integer or a list of integers*) – Expected return code. Defaults to 0.

Returns

A `subprocess.CompletedProcess` with the *slapcat* execution data.

start()

Starts the slapd server process running, and waits for it to come up.

stop()

Stops the slapd server, and waits for it to terminate and cleans up

wait()

Waits for the slapd process to terminate by itself.

slapd.combinedlogger(*log_name, log_level=30, syslogger_format='% (levelname)s %(message)s', consolelogger_format='% (asctime)s %(levelname)s %(message)s'*)

Returns a combined SysLogHandler/StreamHandler logging instance with formatters

1.2 Changelog

All notable changes to this project will be documented in this file.

The format is based on [Keep a Changelog](#), and this project adheres to [Semantic Versioning](#).

1.2.1 [0.1.4] - 2023-08-13

Added

- Implement *init_tree* method [#8](#)

Removed

- Stop supporting python 3.7

1.2.2 [0.1.3] - 2023-03-12

Added

- Python 3.11 support [#6](#)
- Increase initial value for *olcDbMaxSize* [#7](#)

1.2.3 [0.1.2] - 2021-11-24

Added

- Added *ldapsearch* command. [#4](#)
- The root DN can add schemas. [#5](#)

1.2.4 [0.1.1] - 2021-05-06

Added

- Customizable configuration template

1.2.5 [0.1.0] - 2021-04-02

Added

- Initial release.

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

S

slapd, [3](#)

INDEX

C

`combinedlogger()` (*in module slapd*), 5

I

`init_tree()` (*slapd.Slapd method*), 3

L

`ldapadd()` (*slapd.Slapd method*), 3

`ldapdelete()` (*slapd.Slapd method*), 4

`ldapmodify()` (*slapd.Slapd method*), 4

`ldapsearch()` (*slapd.Slapd method*), 4

`ldapwhoami()` (*slapd.Slapd method*), 4

M

module

 slapd, 3

R

`restart()` (*slapd.Slapd method*), 5

S

`slapadd()` (*slapd.Slapd method*), 5

`slapcat()` (*slapd.Slapd method*), 5

slapd

 module, 3

`Slapd` (*class in slapd*), 3

`start()` (*slapd.Slapd method*), 5

`stop()` (*slapd.Slapd method*), 5

W

`wait()` (*slapd.Slapd method*), 5