

- [TebogoYungMercyKay - Practical Documentation](#)
- [Accessing phpLDSPAdmin:](#)
- [Practical Task:](#)
- [Example Code](#)
- [Creating A New User:](#)

# TebogoYungMercyKay - Practical Documentation

---

- COS332 - Computer Networking
  - Practical Assignment 5
- 

## Accessing phpLDSPAdmin:

---

- `sudo apt-get install phpldapadmin`
- `sudo systemctl start apache2`
- `sudo systemctl start apache2`
- `sudo ufw allow 80`
- `http://localhost/phpldapadmin`

## Practical Task:

---

Based on your assignment, you need to create an LDAP directory structure that mimics the DNS hierarchy for the **za** TLD, and then write a client that can query this directory. Here's a step-by-step plan:

1. **Create the Directory Information Tree (DIT):** Use phpLDAPAdmin to create the DIT on your LDAP server. The root of the tree should be **dc=za**. Under this root, create entries for second-level domains (like **dc=ac, dc=za, dc=co, dc=za, dc=gov, dc=za**). Under each second-level domain, create entries for individual organizations.
2. **Populate the DIT:** For each organization, add attributes to store the A, NS, and MX records. You can use the **nslookup** command to get this information for real organizations, or you can make up data for testing purposes.

3. **Write the client:** Your client should be a program that asks the user for an organization name, constructs an LDAP query based on this input, sends the query to the LDAP server, and displays the response. The client should connect to the server on port 389 and should not use any high-level libraries for handling the LDAP protocol. Instead, it should construct the query packet and parse the response packet using byte or string operations.

Please note that this is a complex assignment that requires a good understanding of the LDAP protocol, the DNS hierarchy, and network programming in general. You might need to refer to the LDAP-related RFCs and your programming language's documentation for more details.

## Example Code

---

```
import ldap3

class LdapClient:
    def __init__(self, server, user, password):
        self.server = ldap3.Server(server)
        self.conn = ldap3.Connection(self.server, user, password,
        auto_bind=True)

    def query(self, base_dn, query):
        self.conn.search(base_dn, query, attributes=ldap3.ALL_ATTRIBUTES)
        return self.conn.entries

# Add More Methods and Queries
```

## Creating A New User:

---

To create credentials for an LDAP server, you typically need to add an entry for the admin user to the server's directory. This process can vary depending on the specific LDAP server software you're using, but here's a general outline of the steps you might need to take:

1. **Create an LDIF file:** An LDIF (LDAP Data Interchange Format) file is a standard format for storing LDAP data. You can create an LDIF file to add a new entry to the LDAP directory. Here's an example of what this file might look like:

```
dn: cn=admin,dc=tebogoyungmercykay,dc=co,dc=za
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: Networks LDAP administrator
userPassword: tebogo63729013
```

## 1.2. Creating Command:

- `ldapadd -x -D "cn=admin,dc=tebogoyungmercykay,dc=co,dc=za" -w tebogo63729013 -f admin.ldif`

**1.2. Searching Command:** `ldapssearch -x -H ldap://localhost:389 -D "cn=admin,dc=tebogoyungmercykay,dc=co,dc=za" -w tebogo63729013 -b "dc=tebogoyungmercykay,dc=co,dc=za" "(objectClass=*)"`

This LDIF file creates a new entry with the DN `cn=admin,dc=example,dc=com`. The `userPassword` attribute is set to `{CLEARTEXT}password`, which means the password is `password`.

**2. Add the entry to the LDAP directory:** You can use the `ldapadd` or `ldapmodify` command to add the entry to the directory. Here's an example of how you might use this command:

```
ldapadd -x -D "cn=admin,dc=example,dc=com" -W -f admin.ldif
```

This command adds the entry defined in the `admin.ldif` file to the directory. The `-D` option specifies the DN to bind to the directory (in this case, the admin user), and the `-W` option prompts for the bind password.

Please note that these are general instructions and might not work for your specific LDAP setup. You should check the documentation for your LDAP server software for more detailed and accurate instructions.

The LDIF (LDAP Data Interchange Format) entries you need to create for your LDAP directory based on the steps you provided would look something like this:

### 1. Create the TLD:

```
dn: dc=za
objectClass: top
objectClass: domain
dc: za
```

## 2. Create the second-level domains (e.g., "co"):

```
dn: ou=co,dc=za
objectClass: top
objectClass: organizationalUnit
ou: co
```

## 3. Create the organizations (e.g., "tebogoyungmercykay"):

```
dn: o=tebogoyungmercykay,ou=co,dc=za
objectClass: top
objectClass: organization
o: tebogoyungmercykay
```

## 4. Add the DNS information (e.g., for "up.ac.za"):

```
dn: dc=up,dc=ac,dc=za,o=tebogoyungmercykay,ou=co,dc=za
objectClass: top
objectClass: domain
dc: up
```

You can create these LDIF files and then use the **ldapadd** command to add them to your LDAP directory. For example:

```
ldapadd -x -D "cn=admin,dc=za" -w tebogo63729013 -f tld.ldif
ldapadd -x -D "cn=admin,dc=za" -w tebogo63729013 -f second_level_domain.ldif
ldapadd -x -D "cn=admin,dc=za" -w tebogo63729013 -f organization.ldif
ldapadd -x -D "cn=admin,dc=za" -w tebogo63729013 -f dns_info.ldif
```

Replace "adminpassword" with your actual admin password, and replace "tld.ldif", "second\_level\_domain.ldif", "organization.ldif", and "dns\_info.ldif" with the actual paths to your LDIF files.