

LDAP Task

We need Dependencies, Spring Configuration(Connection Configurations and Security Configurations) and Controller Route Mappings to achieve integration using Lightweight Directory Access Protocol (LDAP). This can be done by making some changes to the pom.xml file and adding following lines to the existing code.

Dependencies:

pom.xml

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-ldap</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-security</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.ldap</groupId>
    <artifactId>spring-ldap-core</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.security</groupId>
    <artifactId>spring-security-ldap</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-thymeleaf</artifactId>
  </dependency>
  <dependency>
    <groupId>org.thymeleaf.extras</groupId>
    <artifactId>thymeleaf-extras-springsecurity4</artifactId>
  </dependency>
</dependencies>
```

Spring configuration:

Connection configurations

```
ldap.urls=ldap://ldap.forumsys.com:389/  
ldap.base.dn=dc=example,dc=com  
ldap.username=cn=read-only-admin,dc=example,dc=com  
ldap.password=password ldap.user.dn.pattern =  
uid={0}
```

Security configuration

@Configuration

```
public class LDAPSecurityConfig extends WebSecurityConfigurerAdapter  
{
```

```
    @Value("${ldap.urls}")  
    private String ldapUrls;  
    @Value("${ldap.base.dn}")  
    private String ldapBaseDn;  
    @Value("${ldap.username}")  
    private String ldapSecurityPrincipal;  
    @Value("${ldap.password}")  
    private String ldapPrincipalPassword;  
    @Value("${ldap.user.dn.pattern}")  
    private String ldapUserDnPattern;
```

@Override

```
protected void configure(HttpSecurity http) throws Exception  
{
```

```
    http  
        .authorizeRequests()  
        .antMatchers("/login**").anonymous()  
        .antMatchers("/resources/**").permitAll()  
        .antMatchers("/assets/**").permitAll()  
        .antMatchers("/").authenticated()  
        .antMatchers("/home").authenticated()  
        .and()  
        .formLogin()  
        .loginPage("/login")  
        .permitAll()  
        .and()  
        .logout()  
        .permitAll()  
        .and()  
        .csrf()  
        .disable();
```

```

    }

    @Override
    protected void configure(AuthenticationManagerBuilder auth) throws Exception
    {
        Auth
        .ldapAuthentication()
        .contextSource()
        .url(IdapUrls + IdapBaseDn)
        .managerDn(IdapSecurityPrincipal)
        .managerPassword(IdapPrincipalPassword)
        .and()
        .userDnPatterns(IdapUserDnPattern);
    }
}

```

Controller Mappings:

```

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
@Controller
public class AppController
{
    @RequestMapping(value = "/login")
    public String login()
    {
        return "signin";
    }

    @RequestMapping(value = "/home")
    public String home()
    {
        return "index";
    }

    @RequestMapping(value = "/")
    public String home1()
    {
        return "index";
    }
}

```