APPENDIX 1

MID-TERM EVALUATION REPORT

A TRAINING REPORT

Submitted by

SAKSHI

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

UNIVERSITY INSTITUTE OF ENGINEERING AND TECHNOLOGY
PANJAB UNIVERSITY,
CHANDIGARH 160025

APRIL 2022

ACKNOWLEDGEMENT

Firstly, I would like to thank Deloitte Risk and Financial Advisory for giving me the opportunity to do internship here. I would also like to extend my gratitude to my mentors for their valuable guidance and support whenever required for the technologies I learnt. I would also like to thank my mentors for providing me with tips and guidance from their experience on how to sustain and excel in corporate world.

I would also like to thank my teammates for their continuous encouragement and support throughout the internship period.

I would like to thank UIET, Panjab University for providing me this opportunity to do internship at Deloitte during my Bachelor of Engineering. I would like to express my deepest gratitude to the TPC team and faculty members Dr. Nidhi Garg and Dr. Garima Joshi for guiding me in completing this report.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

This report describes my internship at Deloitte. Deloitte is among the Big Four accounting firms across the world. It provides audit, consulting, financial advisory, risk advisory, tax, and legal services with approximately 334,800 professionals globally.

The scope of this document is to describe in detail about the things learnt and experience gained during my internship. During the first two months of my internship, I went through various bootcamps such as IAM Bootcamp, Okta Bootcamp and Java Bootcamp and got introduced to some new technologies, concepts, and frameworks. And in the remaining two months, I learnt about Web Access Management (WAM) and PingIdentity products such as Ping Directory, Ping Federate and Ping Access is used to implement WAM at the enterprise level.

About the Organization

Deloitte Touche Tohmatsu Limited, commonly referred to as Deloitte, is a multinational professional services network with offices in over 150 countries and territories around the world. Deloitte is one of the Big Four accounting organizations and the largest professional services network in the world by revenue and number of professionals, with headquarters in London, England. Deloitte provides audit, consulting, financial advisory, risk advisory, tax, and legal services with approximately334,800 professionals globally. In FY 2021, the network earned revenues of US$50.2 billion in aggregate. As of2020, Deloitte is the third-largest privately-owned company in the United States, according to Forbes.

CHAPTER 1: INTRODUCTION TO IAM AND WAM

1.1 What is IAM?

IAM stands for Identity and Access Management. Identity refers to the virtual identity of the user. User as identity trying to access the right resource at right time securely is defined as IAM. It basically ensures that the right people and job roles in an organization can access the tools they need to do their jobs. IAM systems acts as an interface between identity of the users and resources that users want to access.

Some important terminologies in IAM are:

- Digital Identity/Identity: It refers to the digital representation of a user, including a unique identifier, credentials and other attributes that makes the complete virtual entity of the user.
- Authentication (AuthN): It is the process of validating who the user is claiming to be.
- Authorization (AuthZ): It is the process of granting the right access of application and resource to the right person at right time.
- Provisioning: Complete management of creation of user accounts in information technology (IT) resources and providing appropriate access to those accounts is known as provisioning in IAM.
- Self-service: Allowing the end user to perform specific activities without any help and intervention of helpdesk.
- SSO: SSO stands for Single Sign On. In SSO, user when signs into one account can get access to all other apps under that organization. Using SSO, there is no need to remember different passwords for different apps. Also, it provides better security.
- Connector/Adapter: Technology used by IAM System to interface and interact with managed systems.
- IT Resource/System: Application, system, platform for which access is required to perform a specific function. IAM integrates with these IT resources for user account management.

1.2 Need for IAM

IAM to provides better security and increases employee productivity because:

- Security: IAM helps in providing better security since IAM services narrow the points of failure and backstops them with tools to catch mistakes when they're made.
- Productivity: Once the user is log on to the main IAM portal, employee needs not to worry about having the right password or right access level to perform their duties. Also, IAM services make sure that right person is given access to the right resource. Hence, it helps in avoiding misuse of access. IAM reduces the burden on IT professions. They no longer need to maintain multiple accounts of user and also saves there time in writing the duplicate code for multiple user accounts.

1.3 Identity Lifecycle Management

Identity Lifecycle Management (ILM) is a process where a user creation, managing, coordinating and restricting the identification, access and governance of identities to different tools / applications is performed using the different technologies. ILM is the full life cycle of identity and access for any user in the network. It covers every aspect of IAM from the moment a person is on-boarded to the moment they leave the company.

Stages of ILM are:

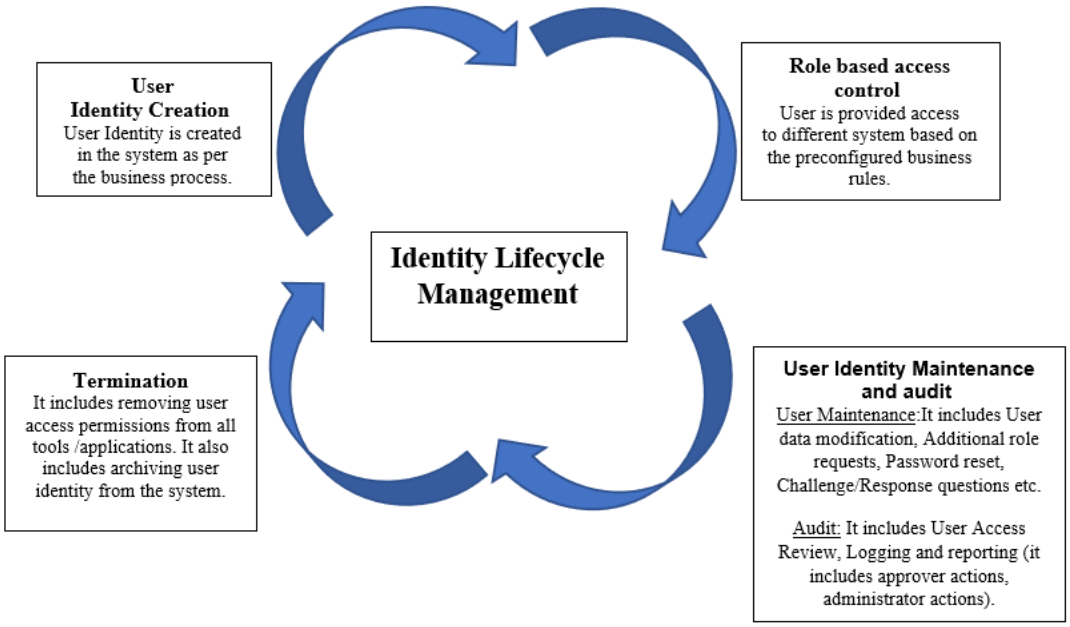


Figure 1: Identity Lifecycle Management Diagram

- User Identity Creation: Setting up of new identities should be governed by the principle of least privilege. That means once an identity is verified (single sign-on and multifactor authentication are typical methods of verification for human users) the user or machine is only given access at the level required to do their specific tasks.

- Role based access control: Role-based access controls dictated by stated policies help to maintain proper user access throughout the identity lifecycle. Revoking access when it's no longer needed should also be an integral part of the lifecycle process.

- User identity management and audit: It includes the maintenance of identities of user and making modifying time to time as per the requirement. Audit includes user access review to make sure that only required access is given to him. It also includes logging and reporting.

- Termination**:** Deprovisioning accounts on a timely basis is necessary to minimize risks from unauthorized access or malicious intent if the employee has been terminated for cause.

1.4 Web Access Management (WAM)

Web access management (WAM) is a type of access control that allows users to access web-based applications. WAM is a type of identity management that provides authentication management, policy-based authorizations, audit and reporting services, and single sign-on services for web resources.

1.5 Difference between IAM and WAM

IAM	WAM
○ IAM stands for Identity and Access Management.	○ WAM stands for Web Access Management.
○ IAM helps in ensuring that right person has the access to right application	○ WAM focuses on ensuring security to web-based tools and applications.
○ IAM provision user identities to the endpoint with the help of AD.	○ WAM does not maintain and provision user identities.
○ IAM focuses on Authentication, Authorization as well as Identity Federation.	○ WAM focuses only on 2 operations i.e., Authentication and Authorization.

Table 1: IAM vs WAM

NOTE: *Since WAM systems needs IdP(Identity Provider) to generate , provision and manage user identities, they need high maintenance and hence, they are more vulnerable to security breaches compared to IAM systems which itself can serve as IdP , identity manager (for provisioning and maintaining user identities) and access manager (for native, mobile or web applications).*

NOTE: *Identity Federation in Table 1 refers to ensuring users to use same identification data to access resources on related domains*.

CHAPTER-2: OKTA

2.1 What is Okta?

Okta connects any identity with any application on any device. Okta was the first vendor to provide services on-cloud applications. Okta can be deployed on various cloud platforms. Okta provides services such as authentication, MFA (Multi-Factor Authentication), LCM (Lifecycle Management), SSO (Single Sign-On) etc.

2.2 Leanings from Okta Bootcamp

2.2.1 Concept of SSO

- SSO stands for Single Sign On
- In SSO, user when signs into one account can get access to all other apps under that organization.

⇨ **Benefits of SSO:**

- No need to remember different passwords for different apps.
- Provides better security.

2.2.2 Concept of MFA

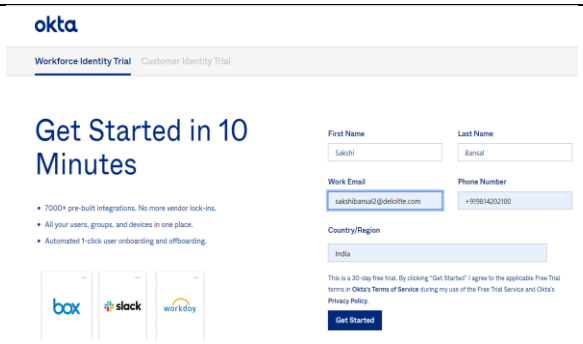
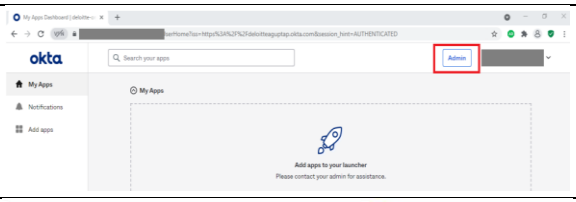
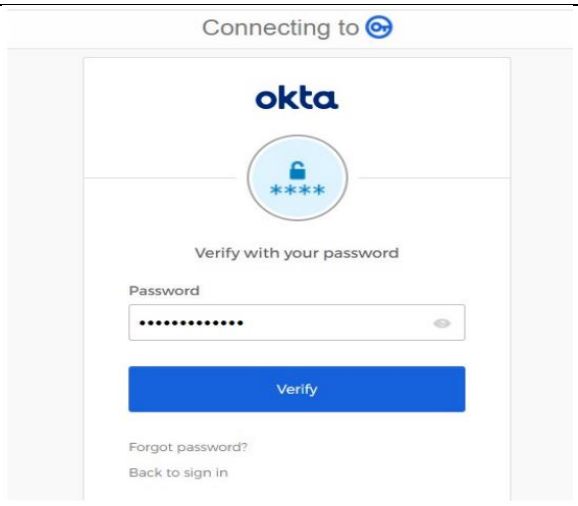
- MFA stands for Multi Factor Authentication.
- Need for MFA: In SSO, there was a problem that if the hacker can identify password for one application, he can get access to all other apps. To avoid this, MFA is used.
- In MFA, layers are added to the authentication process to maintain strong security.

⇨ Levels in MFA:

- LEVEL-1: This includes *Something you know* i.e., password. But anyone can get to know about password, so another level of authentication is added.
- LEVEL-2: This includes *Something you have* i.e., hardware tokens, authenticators, smart cards etc.
- LEVEL-3: This includes *Something you are* i.e., biometric, retina scan, face recognition etc.

2.3 Lab Activities

2.3.1 Create Okta free account and setup of MFA on Okta account

S.No.	STEPS	SCREENSHOTS
1.	Navigate to the link: https://www.okta.com/free-trial/. Enter First Name, Last Name, Work Email, Country and a subdomain as shown in the screenshot.	
2.	After creating the account, success message will appear as shown in the screenshot.	
3.	Login into the account and click on the admin button to land on admin console.	
4.	Enter the password to verify the account. Click on Verify to proceed.	

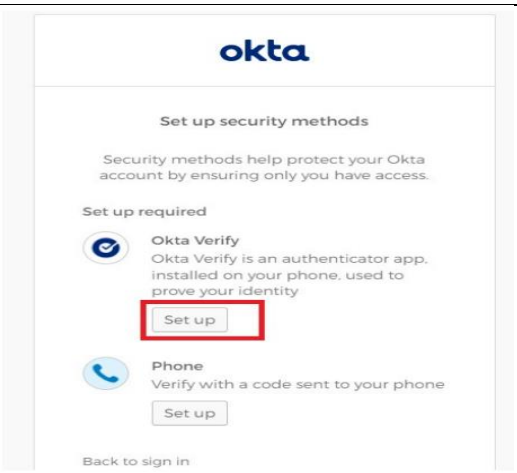
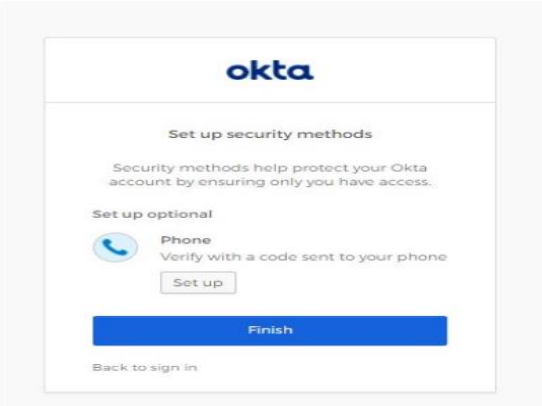
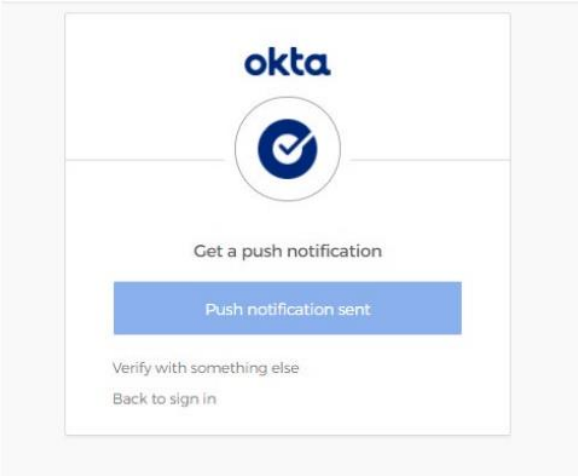
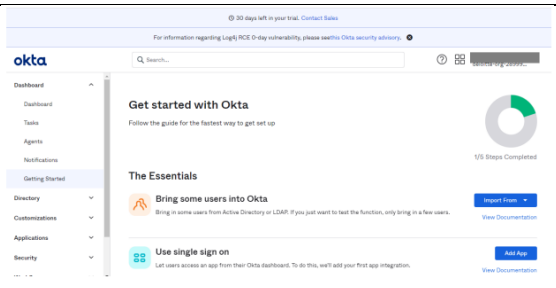
5.	On the next page, enroll MFA for the account. Choose one of the MFA options by clicking on Setup.	
6.	Go to the Okta Verify Account and scan the QR code to add the Okta account. Download the Okta Verify on the mobile. Click on Finish to proceed. ***Note:*** *We can optionally setup Phone as MFA options.*	
7.	Now, we will be redirected to choose the verify method for Okta verify. Enter a code or Get a Push Notification. Selected one and proceed.	
8.	Verify the account from the app Okta Verify from mobile. Now, we will land on Okta Admin console.	

Table 2: Okta setup

2.3.2 Create and Activate Okta sourced users

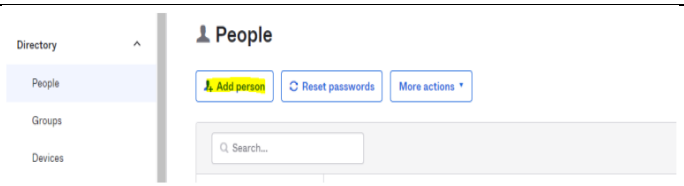
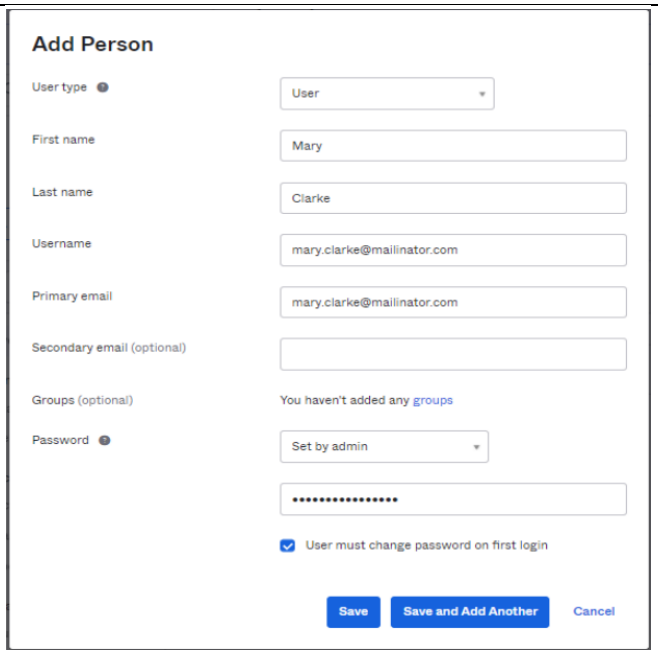
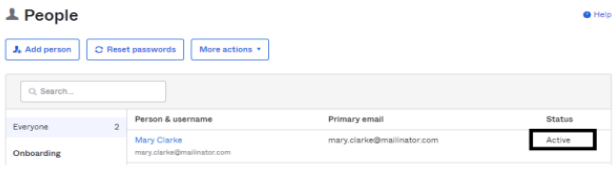
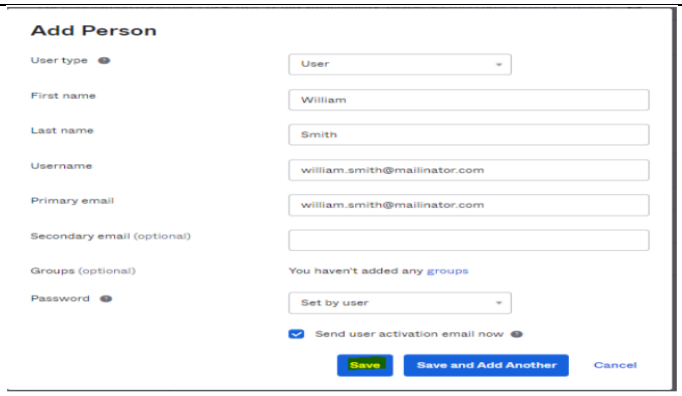
S.No.	STEPS	SCREENSHOTS
1.	From the left pane, click on Directory → People. Click on Add person button to add a user.	
2.	<u>Set password by admin:</u> Create a user user1 with a password. Click on Save.	
3.	We will see the user added. ***Note:*** *User status must be Active.*	
4.	Set password by user: Create another user user2 without a password.	

Table 3: Create and activate Okta sourced users

2.3.3 Managing Profiles

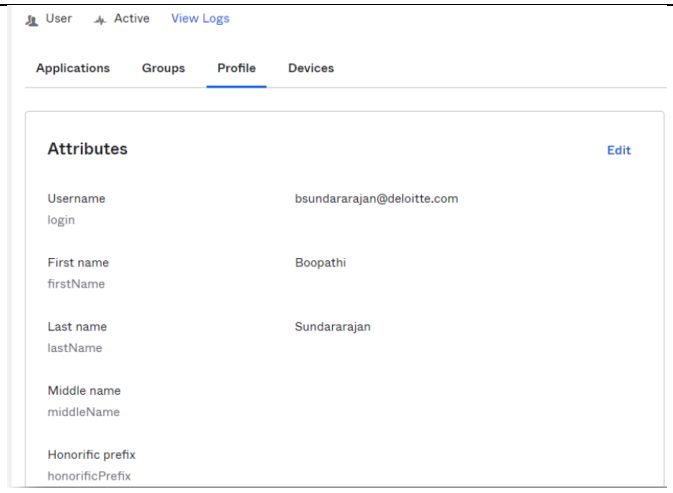
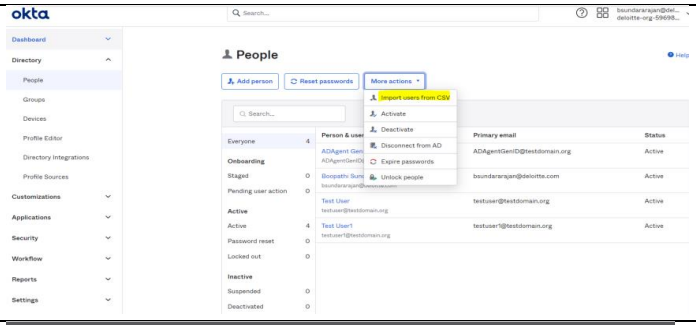
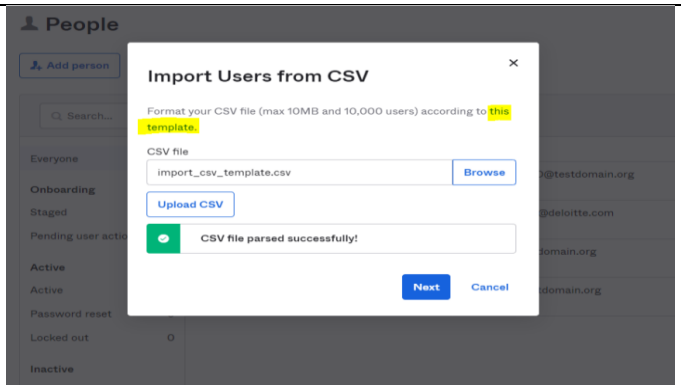
S.No.	STEPS	SCREENSHOTS
1.	Go to Directory → Profile →Search for your Okta mastered profile→Scroll down to bottom of the attribute →Update Interns attribute to "True" → Click Save.	

Table 4: Managing Profiles

2.3.4 Importing users from CSV

S.No.	STEPS	SCREENSHOTS
1.	Go to Directory → Profile →Import users from CSV	
2.	Fill the required fields, upload and Click next.	

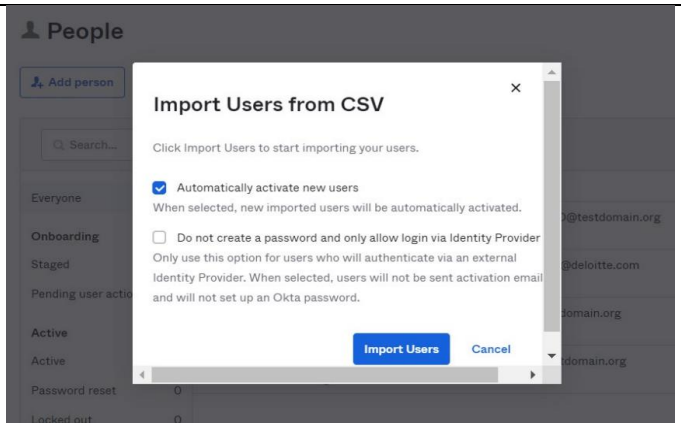
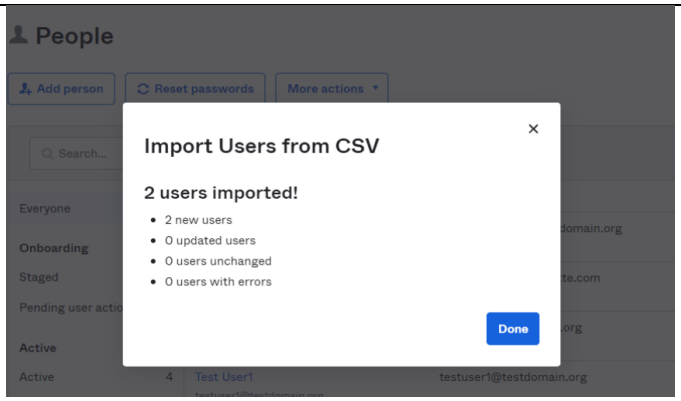
3.	Select “Automatically activate new users” and Click on import users.	
4.	Import successful message is seen as shown in the screenshot.	

Table 5: Importing users from CSV

2.3.5 Administrative Roles

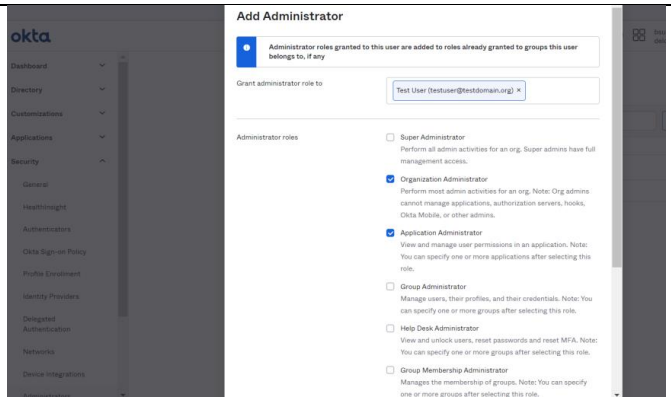
S.No.	STEPS	SCREENSHOTS
1.	Go to Security → Administrators→Click on Add administrators →Type user email address who required an admin role and Select respective admin roles depends on the requirement→Click Add administrators.	

Table 6: Administrative Roles

CHAPTER-3: JAVA TRAINING

3.1 Java Concepts Overview

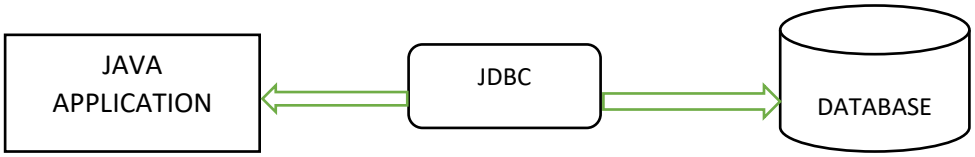
3.1.1 Java Basics

Java is a class-based, object-oriented programming language and is designed to have as few implementation dependencies as possible.

Some important terminologies in context with java are:

- Object − Objects have states and behaviors. Example: A dog has states - color, name, breed as well as behavior such as wagging their tail, barking, eating. An object is an instance of a class.
- Class − A class can be defined as a blueprint that describes the behavior/state that the object of its type supports.
- Methods − A method is basically a behavior. A class can contain many methods. It is in methods where the logics are written, data is manipulated, and all the actions are executed.
- Instance Variables − Each object has its unique set of instance variables. An object's state is created by the values assigned to these instance variables.

3.1.2 JDBC



JDBC allows Java Application to connect with Relational Database

JDBC stands for Java Database Connectivity, which is a standard Java API for database-independent connectivity between the Java programming language and relational databases. The JDBC library includes APIs for each of the tasks mentioned below:

- Making a connection to a database.
- Creating SQL queries.
- Executing SQL queries in the database.
- Viewing & Modifying the resulting records.

3.1.3 Servlet

Servlet can be described as:

- Servlet is a technology which is used to create a web application.
- Servlet is an API that provides many interfaces and classes including documentation.
- Servlet is a class that extends the capabilities of the servers and responds to the incoming requests. It can respond to any requests.

- Servlet is a web component that is deployed on the server to create a dynamic web page.

3.1.4 JSP

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.

A JSP page consists of HTML tags and JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tags, etc.

3.1.5 Hibernate

Hibernate is a high-performance Object/Relational persistence and query service, which is licensed under the open-source GNU Lesser General Public License (LGPL) and is free to download. Hibernate not only takes care of the mapping from Java classes to database tables (and from Java data types to SQL data types), but also provides data query and retrieval facilities.

3.1.6 Derby Database

Apache Derby is a relational database management system developed by the Apache Software Foundation that can be embedded in Java programs and used for online transaction processing.

3.1.7 Tomcat Server

It is an open-source Java servlet container that implements many Java Enterprise Specs such as the Websites API, Java-Server Pages and finally, the Java Servlet.

3.1.8 Spring and Spring Boot

- Spring: Spring Framework is the most popular application development framework of Java. The main feature of the Spring Framework is dependency Injection or Inversion of Control (IoC). With the help of Spring Framework, we can develop a loosely coupled application. It is better to use if application type or characteristics are purely defined.
- Spring Boot: Spring Boot is a module of Spring Framework. It allows us to build a stand-alone application with minimal or zero configurations. It is better to use if we want to develop a simple Spring-based application or RESTful services.

3.1.9 MVC Model

The **Model-View-Controller (MVC)** is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application. MVC is one of the most frequently used industry-standard web development frameworks to create scalable and extensible projects.

⇨ **Components of MVC Model**

Following are the components of MVC –

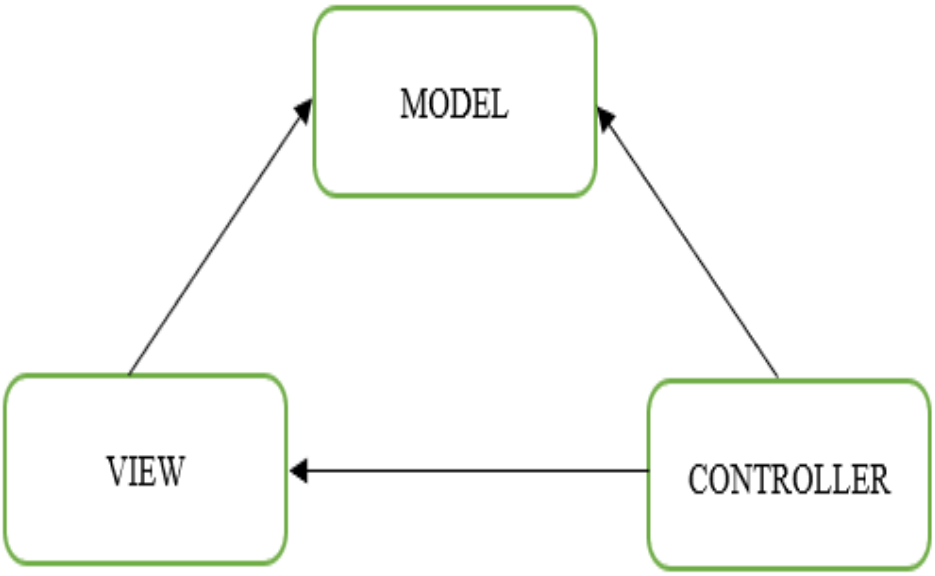


Figure 2: MVC model

- Model:

  The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data. For example, a Customer object will retrieve the customer information from the database, manipulate it and update it data back to the database or use it to render data.

- View:

  The View component is used for all the UI logic of the application. For example, the Customer view will include all the UI components such as text boxes, dropdowns, etc. that the final user interacts with.

- Controller:
  Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output. For example, the Customer controller will handle all the interactions and inputs from the Customer View and update the database using the Customer Model. The same controller will be used to view the Customer data.

3.1.10 React

React.js is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications. It has a single HTML page.

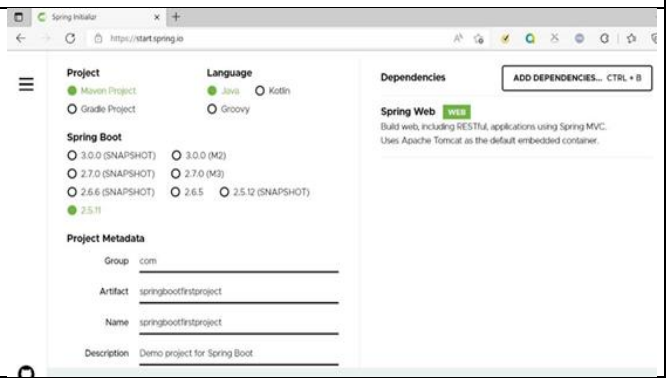
⇨ DOM stands for 'Document Object Model'. In simple terms, it is a structured representation of the HTML elements that are present in a webpage or web-app. DOM represents the entire UI of your application. The DOM is represented as a tree data structure. It contains a node for each UI element present in the web document. It is very useful as it allows web developers to modify content through JavaScript, also it being in structured format helps a lot as we can choose specific targets and all the code becomes much easier to work with.

3.1.11 Spring Boot Project: Employee Management System

In this project, Employee Management System is created in which Java Spring Boot was used for backend and react was used for frontend part. We used MVC model for writing java code. It contained majorly 4 files i.e., Controller, Entity, Service and DAO. DAO file included the database connection. Service file included the logic part. Entity file included the table entry details.

Stepwise execution of the project is explained below:

⇨ Java Code Execution

S.No.	Steps	Screenshots
1.	Navigate to the link: https://start.spring.io/ and make the following changes as shown in the screenshot and then download the metadata file.	

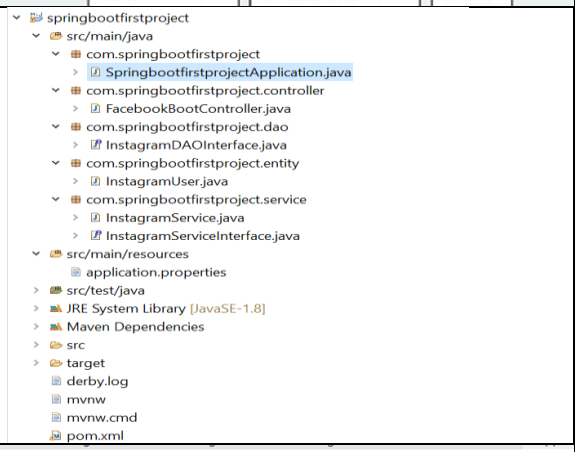
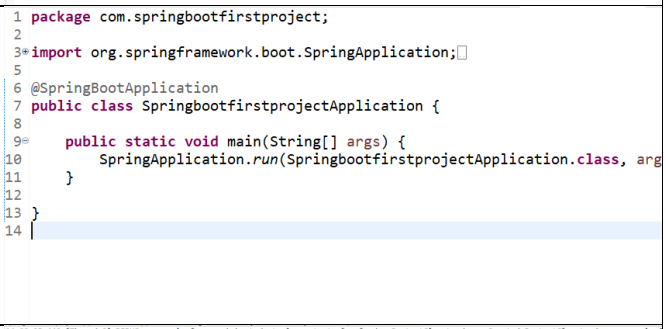
		
2.	Import the metadata file in eclipse and write the code.	
3.	In src/main/resources, navigate to application.properties and add the database properties as shown in the screenshot.	
4.	In src/main/java, navigate to Springbootfirstproject Application.java inside com.springbootfirstproject and run this file.	
5.	After the running the code, tomcat server will start.	

Table 7: Java Code execution

⇨ React Code Execution

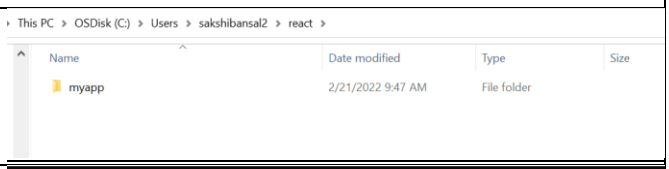
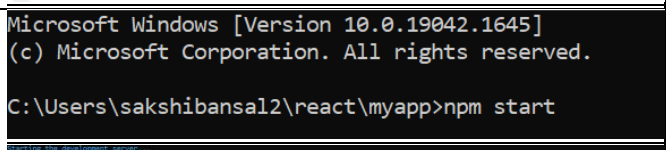
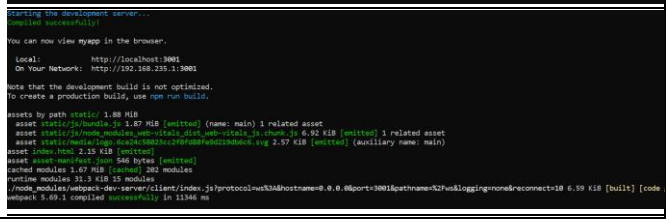
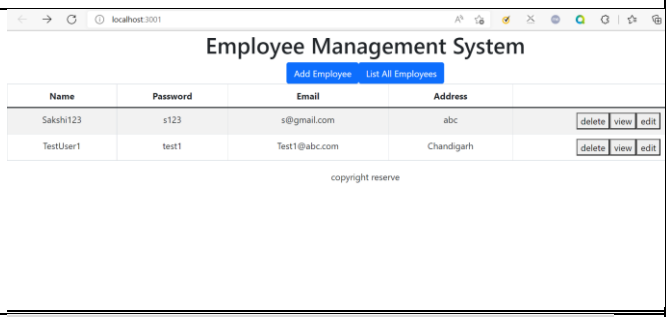
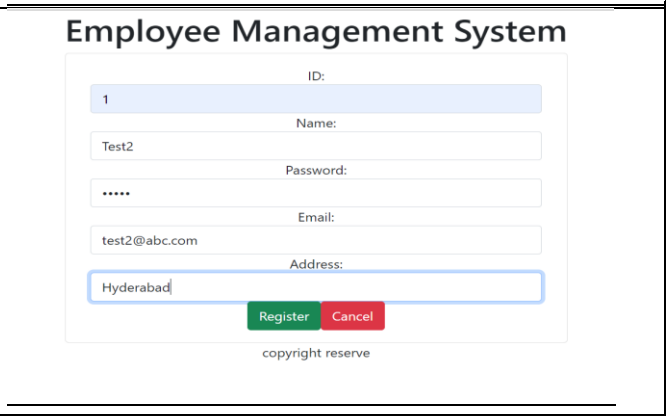
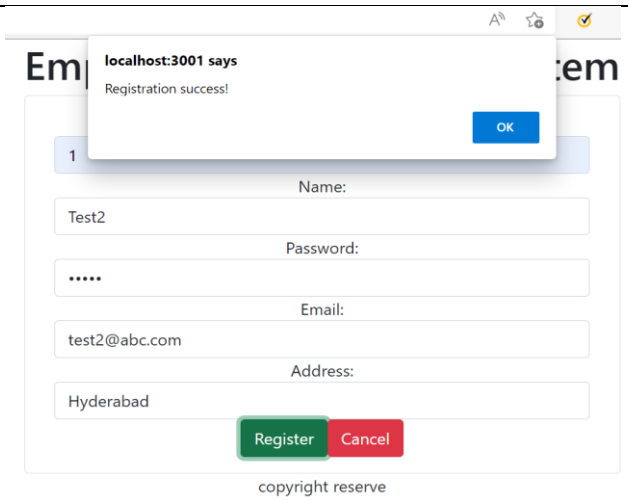
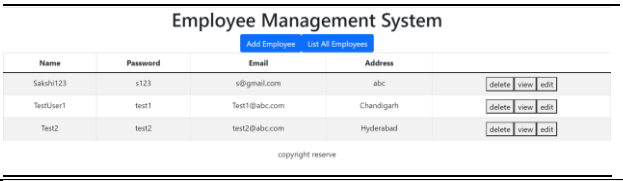
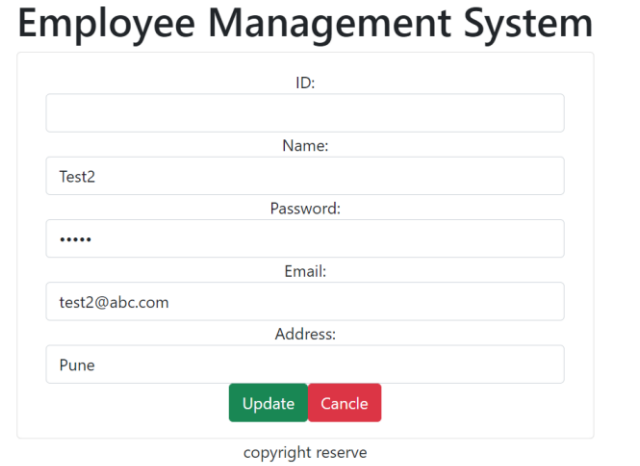
S.No.	Steps	Screenshots
1.	Go to the location where react application is created and open the command prompt at that location.	
2.	Run the react application by writing the command: npm start. Success message as shown in screenshot will be seen.	 

Table 8: React code execution

⇨ Employee Management System Application:

S.No.	Steps	Screenshots
1.	This is the first page of our Application: Employee Management System.	
2.	ADD NEW EMPLOYEE FEATURE: Click on Add Employee button, add new employee page will open. After entering the details and clicking on 'Register' button, we will an alert message and the new user will be created as shown in the screenshots.	

		 
3.	LIST ALL EMPLOYEES FEATURE: After clicking on this button, it will navigate us to the page as shown in the screenshot.	
4.	EDIT ENTRY FEATURE: After clicking on edit button, edit page will open. After making the required changes and clicking on Update button, we will receive an alert message and all the changes will be updated as shown in the screenshot. (In this screenshot, we changed the location of employee Test2 from Hyderabad to Pune. We can see the updated profile of the employee Test2 in the screenshot).	

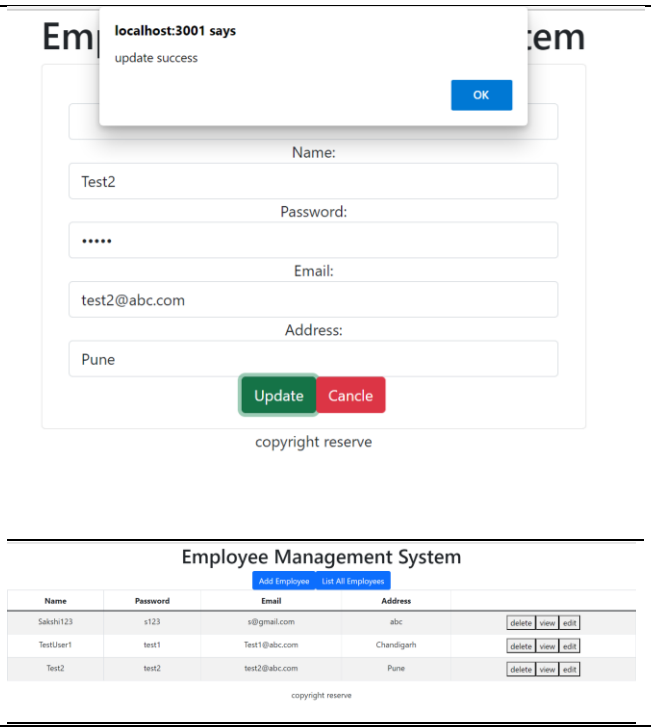
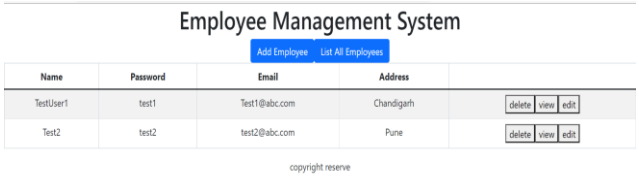
		
5.	VIEW ENTRY FEATURE: After clicking on view button, we will view the details of that particular employee as shown in the screenshot. (In this screenshot, we viewed the profile of Test2 employee.)	
6.	DELETE ENTRY FEATURE: After clicking on delete button, that particular entry will be deleted as shown in the screenshot. (In this screenshot, we deleted the employee profile of Sakshi123.)	

Table 9: Employee Management System

CHAPTER 4: INTRODUCTION TO PING IDENTITY

4.1 Overview

Ping Identity offers federated identity management and identity access management (IAM) solutions for web identities and single sign-on solutions. Ping Identity is identity and access management software that enables businesses to manage and secure access to mobile, cloud, and on-premise applications, external and internal networks, and APIs.

Ping Identity products include PingID, PingFederate, PingOne, PingAccess, PingDirectory, PingDataGovernance, and PingIntelligence. Brief description of each of these tools include:

- PingID: PingID is a cloud-based authentication service that links user identities with mobile devices. PingID service sends an authentication request to user's mobile device during the authentication process.
- PingFederate: PingFederate serves as a global authentication authority, allowing any kind of application to gain authentication and single-sign-on features to provide users with seamless resource access and eliminate insecure password proliferation.
- PingOne: PingOne is a tool that improves user experience by offering easy identity-as-a-service (IDaaS) single sign-on (SSO) services.
- PingAccess: PingAccess provides secure access to applications and APIs ensuring that only authorized users access the resources they need.
- PingDirectory: PingDirectory is an extensible data store that stores and maintains identity of users at a large scale. It provides better security and high performance.
- PingDataGovernance: PingDataGovernance provides solutions to manage and monitor sensitive user data and account resources that helps in avoiding data breaches and meet regulatory standards.
- PingIntelligence: PingIntelligence is a cloud-based service that adds an artificial intelligence (AI) security layer to the API gateways. This is done to improve the security and governance of your API infrastructure.

4.2 Open Standard Protocols

Service providers (SPs) and identity providers (IdPs) communicate identity information using open standard protocols for identity federation.

The protocols used in PingIdentity are:

- SAML 2.0
- OAuth 2.0
- OpenID Connect

4.2.1 SAML 2.0

SAML stands for Security Assertion Markup Language. It is a secure XML-based communication mechanism for communicating identities between organization. It

eliminates the need to maintain multiple authentication credentials such as passwords in multiple locations. It is used to provide SSO to web-based applications and can be used for both authentication and authorization.

4.2.2 OAuth 2.0

OAuth stands for Open Authorization. It is a standard protocol which allows an application or website to have access to resources which are hosted by some other application. It allows a client application to perform some actions on the resources on behalf on user without having user's credentials. OAuth is an authorization standard and not an authentication one.

4.2.3 OpenID Connect

OpenID connect is a simple identity layer on the top of OAuth 2.0 protocol. It enables client to verify the identities of End-user used based on authentication performed by authentication server as well to obtain basic profile information about the End-user.

4.2.4 SAML vs OAuth vs OIDC

PROTOCOL	SAML	OAuth	OIDC
Introduced in Year	2001	2010	2005
Current Version	2.0, released in 2005	2.0, released in 2012	OIDC 1.0, released in 2014
Purpose	Allows two web entities to exchange authentication and authorization data.	Provides delegated authorization to web applications and AP1s.	Provides a layer of authentication above OAuth 2.0
When to use	To allow a user to use SSO to access a web service.	To provide temporary resource access to a third-party application on user's behalf.	To authenticate users to use web or mobile applications without requiring them to create an account.
Application	SSO For enterprise	API Authorization	SSO for consumer application
Format	XML	JSON	JSON

Table 10: SAML vs OAuth vs OIDC

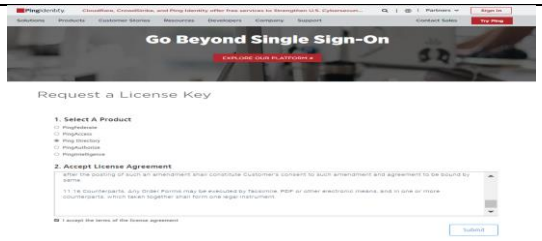
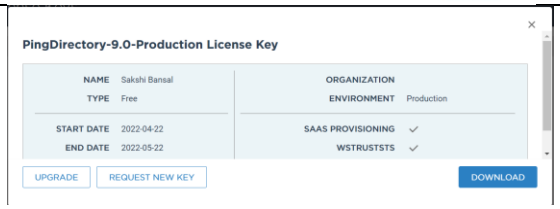
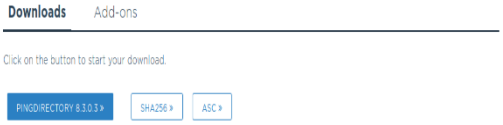
CHAPTER-5: PING DIRECTORY

5.1 What is Ping Directory?

PingDirectory is one of the products of Ping Identity which is an extensible data store that stores and maintains identity of users at a large scale. It provides better security and high performance. PingDirectory is a database that stores and manages sensitive information of customers, partners, and employees, such as credentials, profiles and privacy settings. PingDirectory allows you to handle a large number of identities as well as the attributes that go with them. By reducing redundant and inconsistent user data, PingDirectory allows businesses to save money. It protects against cyberattacks by storing identities and profile data in an encrypted format. It can be used on Ping's private, public, on-prem or cloud. The identities are accessed via the LDAP protocol.

5.2 Installation and Configuration

5.2.1 Installation Steps

S.No.	STEPS	SCREENSHOTS
1.	Navigate to the link: https://www.pingidentity.com/en/resources/downloads/pingdirectory-downloads.html and request for the Ping Directory license key.	
2.	Navigate to the link: https://support.pingidentity.com/s/ and then click on ***Manage License Keys → View → Download*** to download the license key.	
3.	Navigate to the link: https://www.pingidentity.com/en/resources/downloads/pingdirectory-downloads.html and download the Linux based Product Distribution (ZIP) file.	

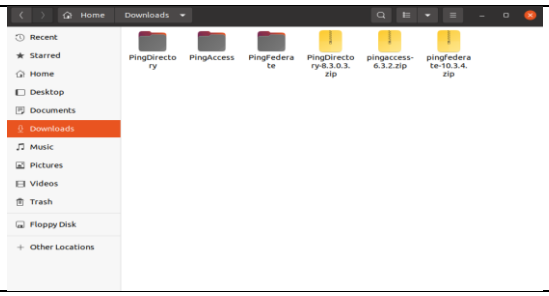
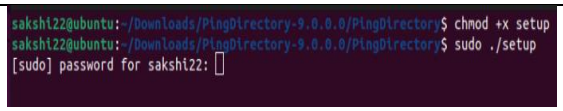
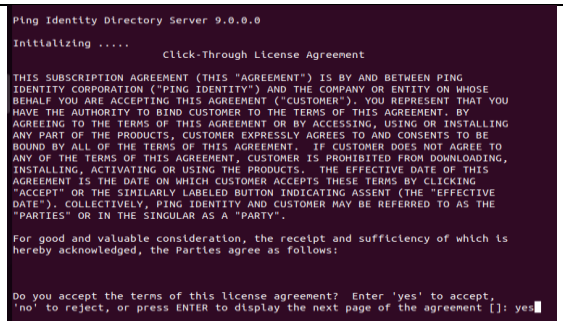
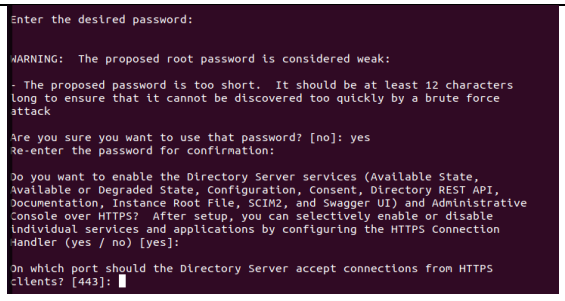
4.	Extract the downloaded ping directory zip file.	

Table 11: Installation of Ping Directory

5.2.2 Configuration Steps of Ping Directory

S.No.	STEPS	SCREENSHOTS
1.	Inside the PingDirectory folder, open a terminal and run the setup command as shown in the figure to begin installation.	
2.	Move the downloaded Ping Directory license file in the location specified in the image.	
3.	Enter localhost as IP address of localhost. Press enter to create DN as cn=Directory Manager.	
4.	Enter the desired password and re-enter it for confirmation. Press enter to select port no. 443 for Directory Server for HTTP clients.	

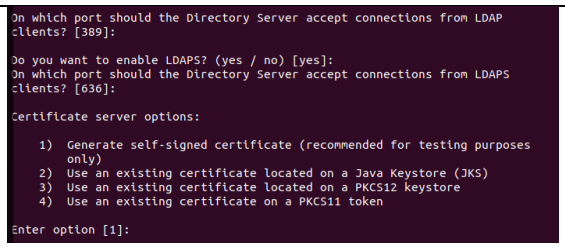
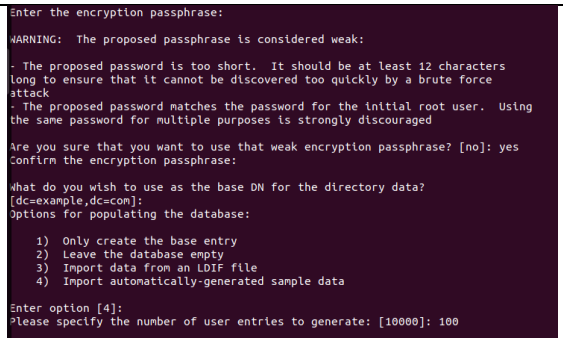
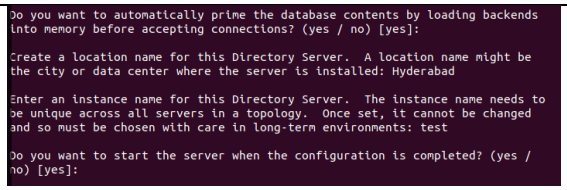
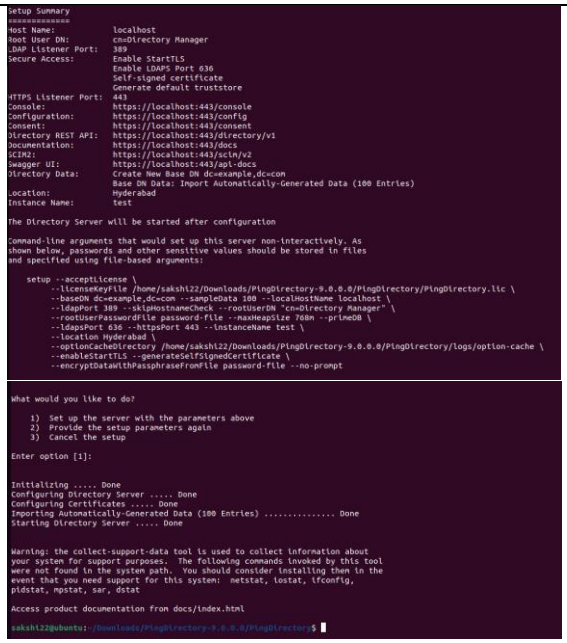
5.	Press enter to select port no. 389 for Directory Server for LDAPS clients.	
6.	Enter and confirm the encryption passphrase. Enter the desired number of user entries that we want to generate (dummy entries).	
7.	Enter yes to automatically prime the database contents by loading backends into memory before accepting connections.	
8.	View the summary as shown in image.	

Table 12: Configuration of Ping Directory

5.3 Folder Configuration

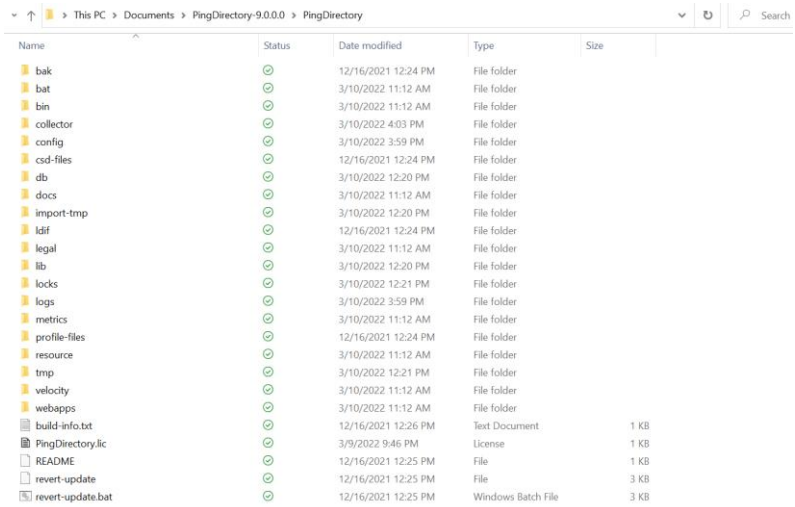


Figure 3: Installation path of Ping Directory

Some of the important ping directory files are:

- Config: It contains all the configurations of the directory.
- Logs: It contains audit, transaction, and startup logs which are important for troubleshooting.
- Bin/ Bat: Bin is essential for Linux and bat is used for Windows. There main purpose is to store all the important command line utilities.

5.3.1 Config Folder in Ping Directory:

- It is the schema folder containing all the user attributes, object class etc.
- We can also create custom attributes here.
- Config file has extension ".ldif"
- Config file is the most important file in the directory.
- When we restart our system, config.ldif will automatically take up backup before closing it.

5.3.2 Db Folder:

- In the backend, there Is a database that actually interacts using LDAP.
- All the database files are stored in directory.
- Cache is stored here.

5.3.3 Log Folder:

- Enables multiple logs.
- Mainly 3 types of logs:

➢ ACCESS: It is very important log. All the operations such as add, delete, modify, authentication is performed.

- ERROR: It is also very important log. Anything unexpected such as greater resource utilization, memory full, restart of instance etc faced by directory is handled here.

NOTE: Suppose we enter wrong credentials, this is not handled by access log. This is handled by access log.

- AUDIT: It handles modifications (add, delete) and changes are stored in audit log. It stores data in ".ldif" format.

NOTE: *In access log, only operations will be known and not the details of the attributes whereas audit log stores all the attributes detail as well.*

5.4 Protocol used in Ping Directory

5.4.1 What is LDAP?

LDAP stands for Lightweight Directory Access Protocol. It is a client server based standard protocol used to authenticate the identities stored in the directory. As the name suggests the operations are much faster as compared to Relational databases. Some important points about LDAP are:

- LDAP is a protocol used to fetch/manage data from Active Directory.
- LDAP directory provides authentication as well as authorization.
- All directory services vendors use LDAP protocol to interact with AD for search, authentication, manage identity purpose etc.
- Directory is a repository for user information.

5.4.2 LDAP vs Relational Database

LDAP	Relational Database
○ LDAP has hierarchical tree like structure.	○ It has Tabular structure.
○ In LDAP, it is difficult to represent complex relations.	○ In database, complex relationship can be represented efficiently.
○ LDAP is more optimized for read operation.	○ Relational Databases are more optimized for write operations.

Table 13: LDAP vs Relational Database

5.4.3 Important Terminologies in LDAP

- Entry**:** LDAP entry is a collection of information about an entity. Each entry consists of three components which includes distinguished name, collection of attributes and collection of object classes.
- Directory Information tree (DIT): It is a hierarchal structure which represents the entries of organization. It is also called as namespace or LDAP tree.
- Distinguished Name (DN): Distinguished name of an entry identifies the unique path of an entry in Directory information tree. It is comprised of attribute-value pairs.
- Object Class**:** Object class is used to group related information. It is used to define the structure of entries including the attributes. Each object class allows some attributes which can be seen in schema file.

5.4.4 Directory Information Tree (DIT)

- A directory is a specialized database that is designed to retrieve information quickly and securely. It is optimized for read access because the type of information in the directory is searched often but changes infrequently.
- Example of DIT is:

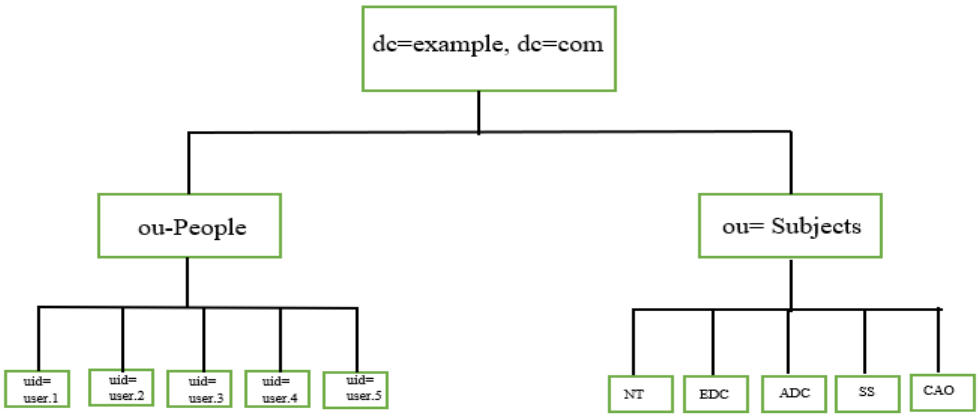


Figure 4: Example of DIT

Here,

- dc stands for domain component
- ou stands for organizational unit
- cn stands for common name

In this example, dc is the base/root of DIT

- The path to each entry in the tree is called its distinguished name (DN), and each DN in the tree is unique. DN is the virtual address of an entry in a tree.
- For example, DN for bob will be: uid=bob, ou=People, dc=example, dc=com

- DN is used to tell at what level of hierarchy that entry lies in DIT.
- DN always starts from that entry and move towards the root of DIT.

5.5 Operations of Ping Directory

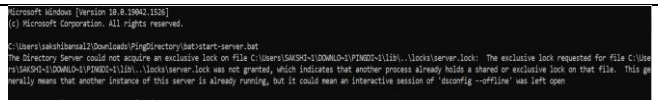
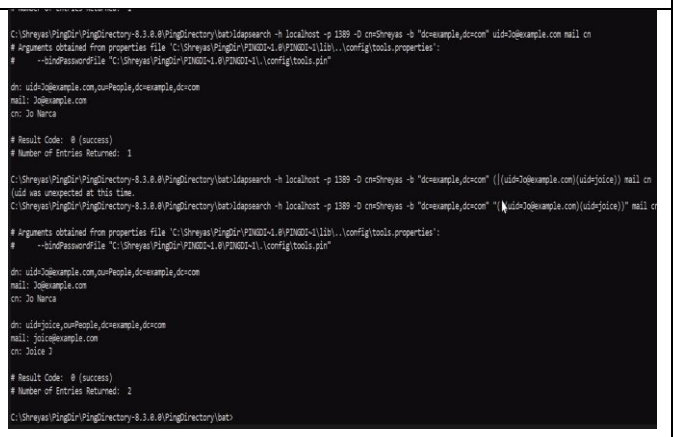
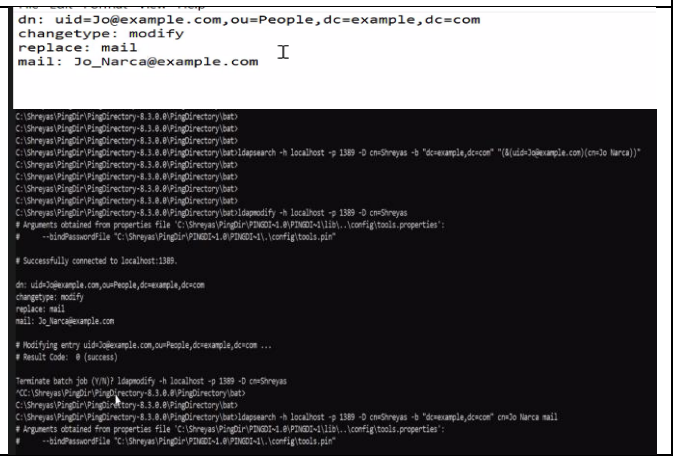
S.No.	Command Description	Command Screenshot
1.	start-server.bat: This command is used to start the server.	
2.	stop-server.bat: This command is used to stop server.	
3.	ldapsearch: This command is used to search for an account. Using this command, we can also define the information we want to fetch for the particular account. Also, we can perform logical AND and OR operations using "&" and "|" symbol to create conditions on search operations.	
4.	ldapmodify: To perform modify operation, ldapmodify command is used.	

Table 14:Ping Directory Operations

5.6 Use-case of Ping Directory

In this use-case, we aimed at implementing the concept of RBAC:

Concept of RBAC:

Figure 5: RBAC

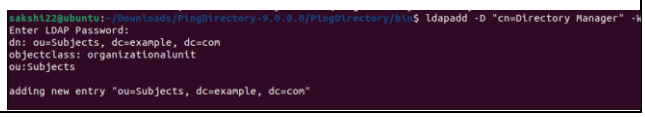
In RBAC, users are assigned some role and access and rights to the resources are mapped to the roles. So, in RBAC,

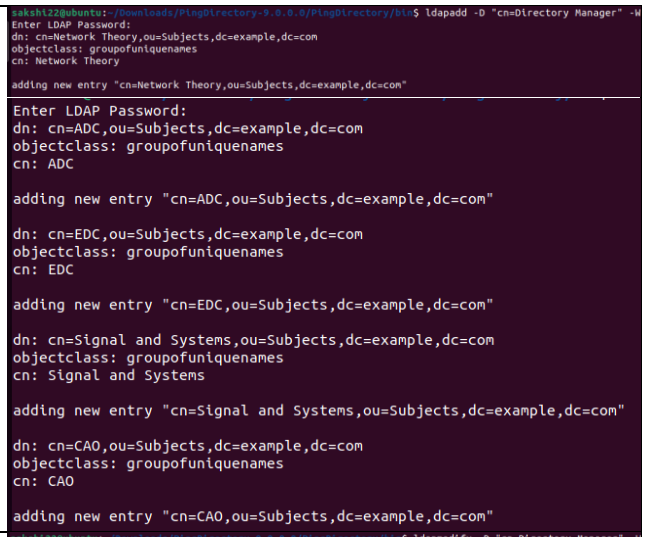
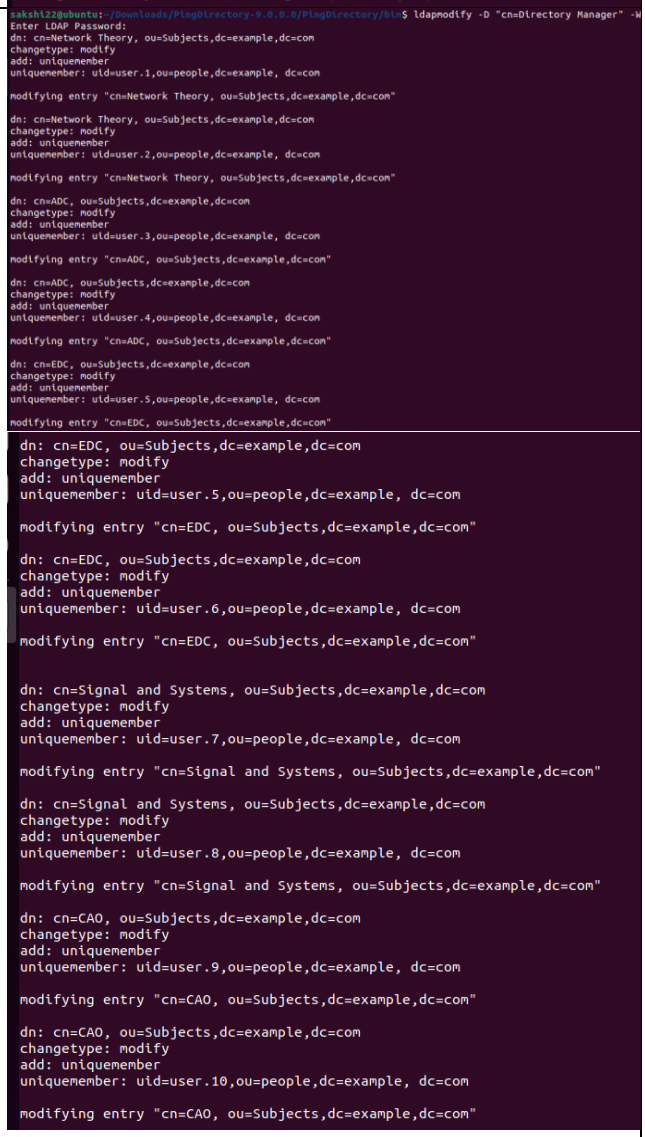
1. Firstly, user is authenticated.
2. After authentication, users then can activate one or more roles.
3. Based on the roles, users are given access and rights to the resources and application

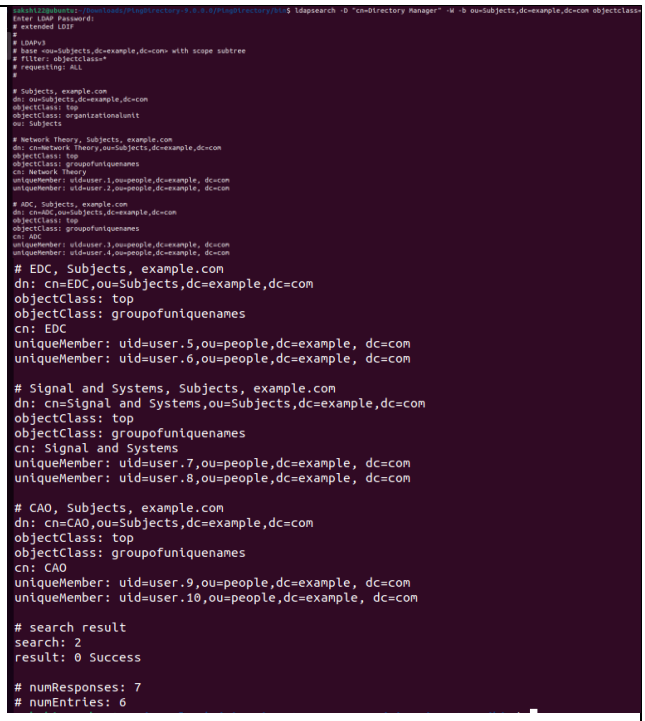
⇨ **Benefits of RBAC:**

- Policy needs not to be updated if the person leaves the organization since users are assigned to roles and not directly given access and rights to the resources.
- New employee can easily get access to the resources based on his role.
- Revisiting least privilege because user in one role has access only to the subset of the files and he has to switch his role in order to access other resources.

With this understanding of RBAC, in this use-case we created an organizational unit Subjects and created 5 groups within it. Then we included the existing users within those groups.

S.No.	Steps	Screenshots
1.	Start the Ping Directory server.	
2.	Create a new organizational unit (ou) as shown in the figure.	

3.	Create new groups inside ou as shown in the figure.	
4.	Now, map the existing users to this group by the following command as shown in the figure.	

5.	By using the search command, we can view all the groups and its entries as shown in the figure.	

```
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <ou=Subjects,dc=example,dc=com> with scope subtree
# filter: objectclass=*
# requesting: ALL
#

# Subjects, example.com
dn: ou=Subjects,dc=example,dc=com
objectClass: top
objectClass: organizationalunit
ou: Subjects

# Network Theory, Subjects, example.com
dn: cn=Network Theory,ou=Subjects,dc=example,dc=com
objectClass: top
objectClass: groupofuniquenames
cn: Network Theory
uniqueMember: uid=user.1,ou=people,dc=example, dc=com
uniqueMember: uid=user.2,ou=people,dc=example, dc=com

# ADC, Subjects, example.com
dn: cn=ADC,ou=Subjects,dc=example,dc=com
objectClass: top
objectClass: groupofuniquenames
cn: ADC
uniqueMember: uid=user.3,ou=people,dc=example, dc=com
uniqueMember: uid=user.4,ou=people,dc=example, dc=com

# EDC, Subjects, example.com
dn: cn=EDC,ou=Subjects,dc=example,dc=com
objectClass: top
objectClass: groupofuniquenames
cn: EDC
uniqueMember: uid=user.5,ou=people,dc=example, dc=com
uniqueMember: uid=user.6,ou=people,dc=example, dc=com

# Signal and Systems, Subjects, example.com
dn: cn=Signal and Systems,ou=Subjects,dc=example,dc=com
objectClass: top
objectClass: groupofuniquenames
cn: Signal and Systems
uniqueMember: uid=user.7,ou=people,dc=example, dc=com
uniqueMember: uid=user.8,ou=people,dc=example, dc=com

# CAO, Subjects, example.com
dn: cn=CAO,ou=Subjects,dc=example,dc=com
objectClass: top
objectClass: groupofuniquenames
cn: CAO
uniqueMember: uid=user.9,ou=people,dc=example, dc=com
uniqueMember: uid=user.10,ou=people,dc=example, dc=com

# search result
search: 2
result: 0 Success

# numResponses: 7
# numEntries: 6
```

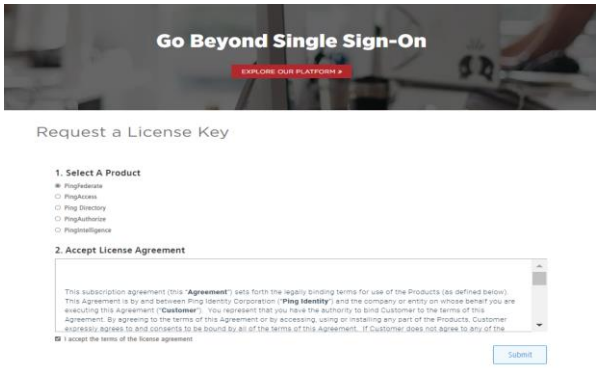
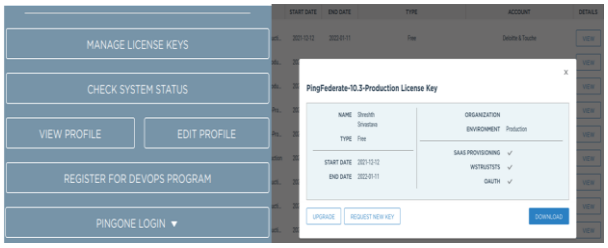
Table 15: Use-case of Ping Directory

CHAPTER-6: PING FEDERATE

6.1 What is PingFederate?

Ping Federate is one of the products of Ping Identity that serves as a global authentication authority, allowing any kind of application to gain authentication and single-sign-on features to provide users with seamless resource access and eliminate insecure password proliferation. PingFederate is a full-featured federation server that provides customers, partners, and employees with identity management, web single sign-on, and API security. Users can securely access the applications they need from any device using a single identity. Ping Federate supports the many open-standard protocols such as OAuth, OpenID Connect, SAML etc. Also, it can be deployed on both on-cloud and on-prem.

6.2 Installation and Configuration

S.No.	STEPS	SCREENSHOTS
1.	Navigate to the link: https://www.pingidentity.com/en/account/request-license-key.html and request for the Ping Federate license key.	
2.	Navigate to the link: https://support.pingidentity.com/s/ and then click on ***Manage License Keys → View → Download*** to download the license key.	
3.	Now, navigate to the link: https://www.pingidentity.com/en/resources/downloads/pingfederate.html and	

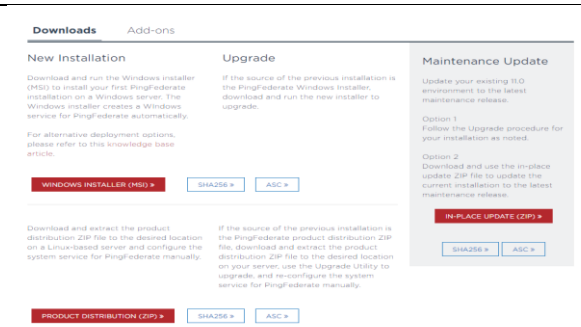
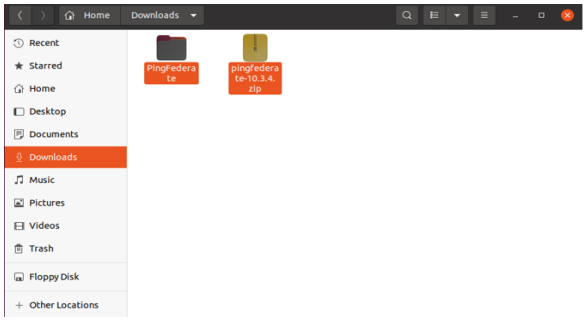
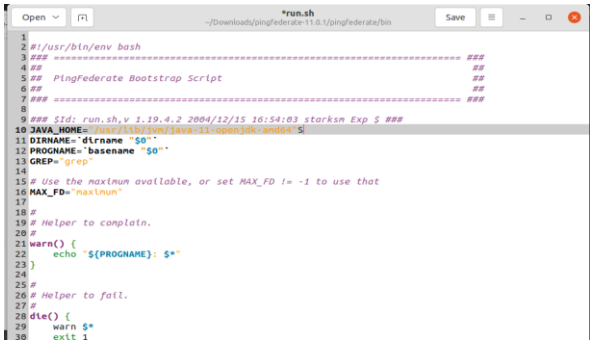
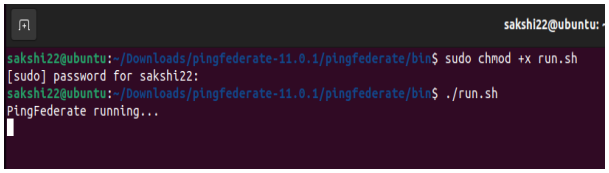
	download the Linux-based Product Distribution (ZIP) file.	
4.	Extract the downloaded Ping Federate zip file.	
5.	Go to the bin folder inside the extracted Ping Federate folder and add the following line at the beginning of run.sh file: JAVA_HOME="/usr/lib/jvm/java-11-openjdk-amd64"	
6.	Now, open the terminal inside the bin folder and type the following command as shown in the screenshot to run the Ping Federate.	
7.	Open the link: https://localhost:9999/ to access the PingFederate console application. Accept the agreement and then click no to Set Up Without PingOne for Enterprise and then select the downloaded license file.	

Table 16: Installation and Configuration of Ping Federate

6.3 Folder Configuration

Some of the important Ping Federate log files are:

S.No.	File	Description
1.	admin.log	Actions performed by administrative-console users are recorded here.
2.	transaction.log	Individual identity-federation runtime transactions at specified levels of detail are recorded here.
3.	audit.log	Transaction log information with additional details at runtime, intended to facilitate security auditing and regulatory compliance are recorded here.
4.	server.log	Runtime and administrative server activities are recorded here.

Table 17: Ping Federate logs

⇨ **Audit log explanation:**

Audit log contains the details of the transaction log information along with additional details at runtime, intended to facilitate security auditing and regulatory compliance. Activities from SSO, SLO, OAuth, WS-Trust STS, and SCIM inbound provisioning transactions are recorded in the audit log. Audit log contains the following details:

- Transaction Time**:** Gives the details of the transaction time of the session.
- Tracking id**:** It is the unique id for the session.
- Event**:** It tells about the type of session that took place. Example: AUTHN_ATTEMPT, SSO etc.
- Subject**:** Subject of the transaction.
- ID: Incoming IP address.
- App**:** Target SP application.
- Protocol: Protocol used to complete the transaction for that session.
- Host: Host IP address.

- Status: Status of the transaction.
- AdapterID: ID of Adapter instance.
- Responsetime: The time elapsed (in milliseconds) from when a final request for a transaction is received to when the audit message is written.

Example of Audit log:

2022-04-20 01:25:52,534| tid:r-mNh32hLEQ-7kz-9aj6ZP5gPJU| SSO| mailofsample5@deloitte.com| 127.0.0.1 | /| https://deSloitte-9a.my.salesforce.com| SAML20| ubuntu| IdP| success| LDAPadapter2| | 140

In this log,

- *2022-04-20 01:25:52,534*: Transaction time
- *tid:r-mNh32hLEQ-7kz-9aj6ZP5gPJU*: Tracking id
- *SSO*: Event
- *mailofsample5@deloitte.com*: Subject of transaction
- *127.0.0.1*: Incoming IP address
- *https://deSloitte-9a.my.salesforce.com*: Target SP application
- *SAML20*: Protocol used to complete the transaction
- *Ubuntu*: Host name
- *Success*: status of the transaction
- *IdP*: LDAPadapter2
- *140*: Response time

6.4 Logging Levels in Ping Federate

Logging level is a way of filtering the data in log file based on its urgency. Logging level helps in controlling the amount the information inside logs. In Ping Federate, logging level configurations are stored in log4j2.xml file. It is located in:

There are 6 types of logging levels in Ping Federate:

S.No.	Logging Level Type	Description
1.	Fatal	This log level tells about the crucial business functionality when it is not working in an application.
2.	Error	This log level tells about the issue in business functionality that is further preventing some functionalities from working properly.

3.	Warn	This log level tells if something unexpected happened in the application, but still the application business features are working properly.
4.	Info	This log level tells about the general information of the event that can be ignored.
5.	Debug	This log level gives granular information about the events that occurred during software debugging.
6.	Trace	This log level describes about the events showing step by step execution of the code.

Table 18: Logging level types

Logging level implementation:

S.No.	Description	Screenshots
1.	In logging level, when log level is changed to DEBUG from INFO, following changes as shown in the screenshots were seen when we start the Ping Federate server. It gave all the details of the software debugging when we run the Ping Federate server.	

Table 19: Logging level implementation

6.5 Protocols used in Ping Federate

Ping Federate uses SAML protocol to provide authentication (Only when it is not integrated with Ping Access. If used along with Ping Access, then Ping Federate uses OIDC protocol to provide authentication). Ping Federate uses SAML protocol to provide IdP and SP initiated SSO for web applications.

6.5.1 Working of SAML Protocol

SAML protocol has 3 entities:

- User Agent: It is the user's web-browser.
- Service Provider (SP): It is the application we try to access.
- Identity Provider (IdP): It is a trusted provider that lets you use single sign-on (SSO) to access other websites.

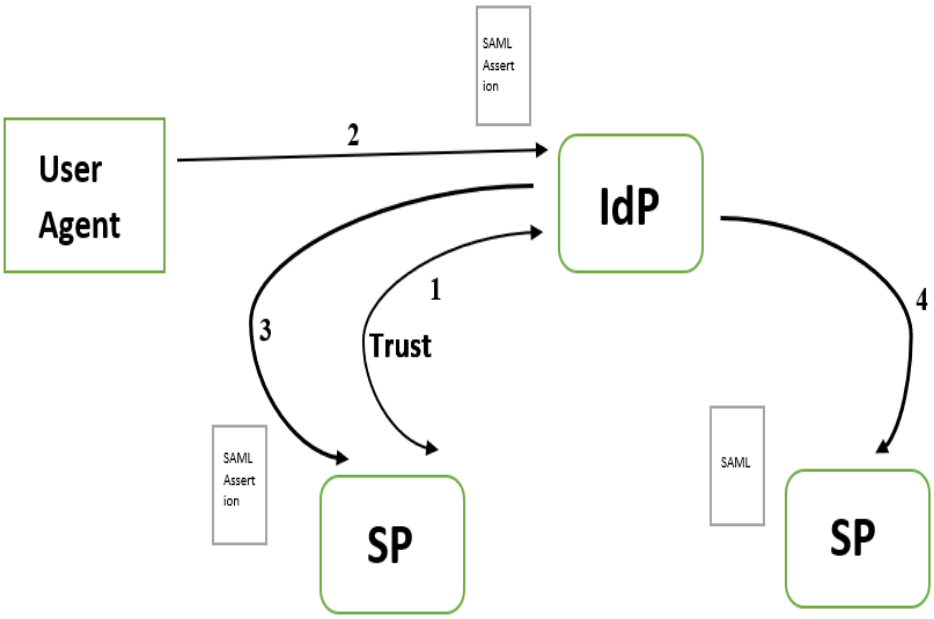


Figure 6: Working of SAML Protocol

Following steps explains the working of SAML protocol:

1. When configuring SAML Federation we establish a trust relationship between SP and IdP.
2. User who wants to access SP first needs to authenticate into IdP.
3. It the user can successfully authenticate and authorize, then IdP generates a SAML assertion. Assertion is then sent to the application via user agent and since the application trusts IdP, the user can access that application.
4. Since the user is already authenticated to the IdP, the user can single sign-on to other applications.

6.5.2 IdP initiated SSO

Identity Provider (IdP) initiated SSO entails the user clicking a button in the IdP and then being forwarded to an SP along with a SAML message containing an assertion. This flow is typically initiated by a page within the IdP that displays a list of all available SPs to which a user can log in.

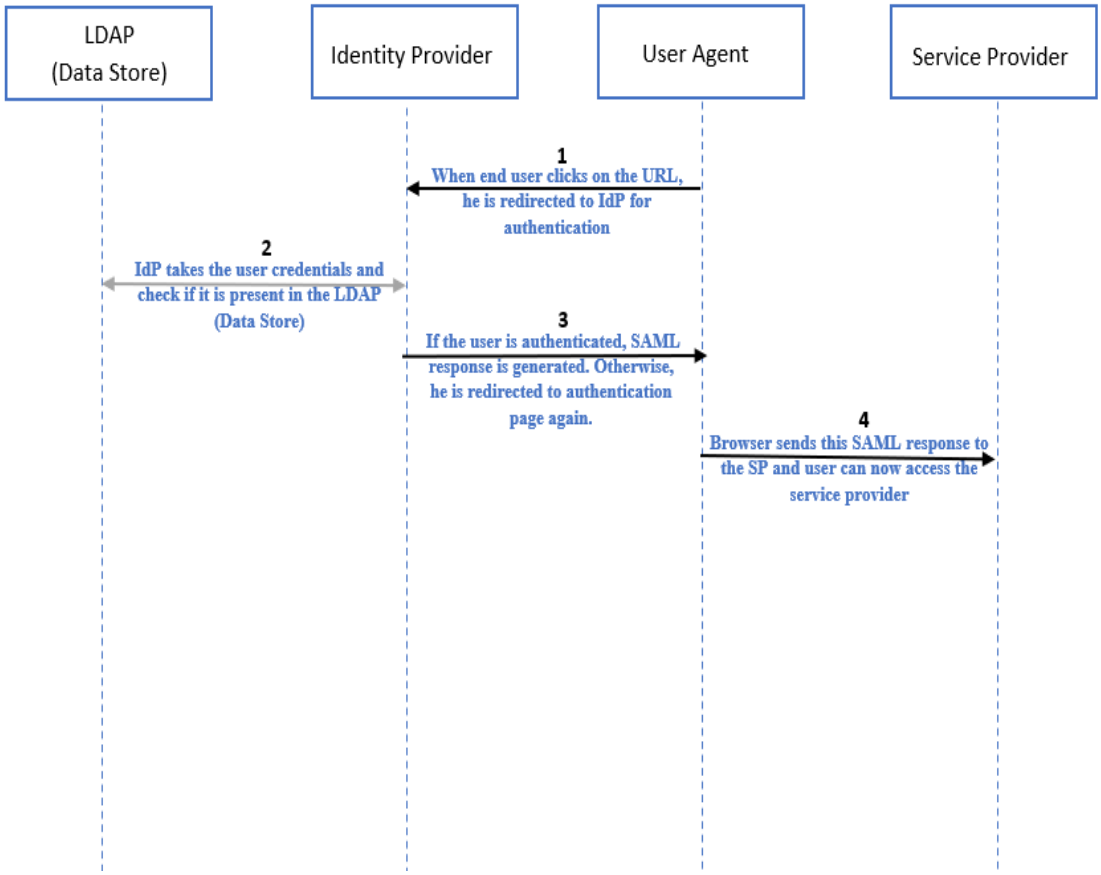


Figure 7: Sequence Diagram for IdP-Initiated SSO

STEP 1: User accesses application by hitting the identity provider-initiated URL.
STEP 2: It checks if sessions exists for the user. If session exists, user can directly access the SP without any authentication. If session does not exist, user is prompted for authentication to the IdP. IdP takes the user credentials and check if it is present in LDAP (data store).
STEP 3: If the authentication is done, IdP generates a SAML response and send it to SP.
STEP 4: SP verifies the SAML response and maps it to a local user and then the session can start.

6.5.3 SP initiated SSO

SP-initiated SSO creates a SAML request and redirects both the user and the request to the identity provider as part of SSO.

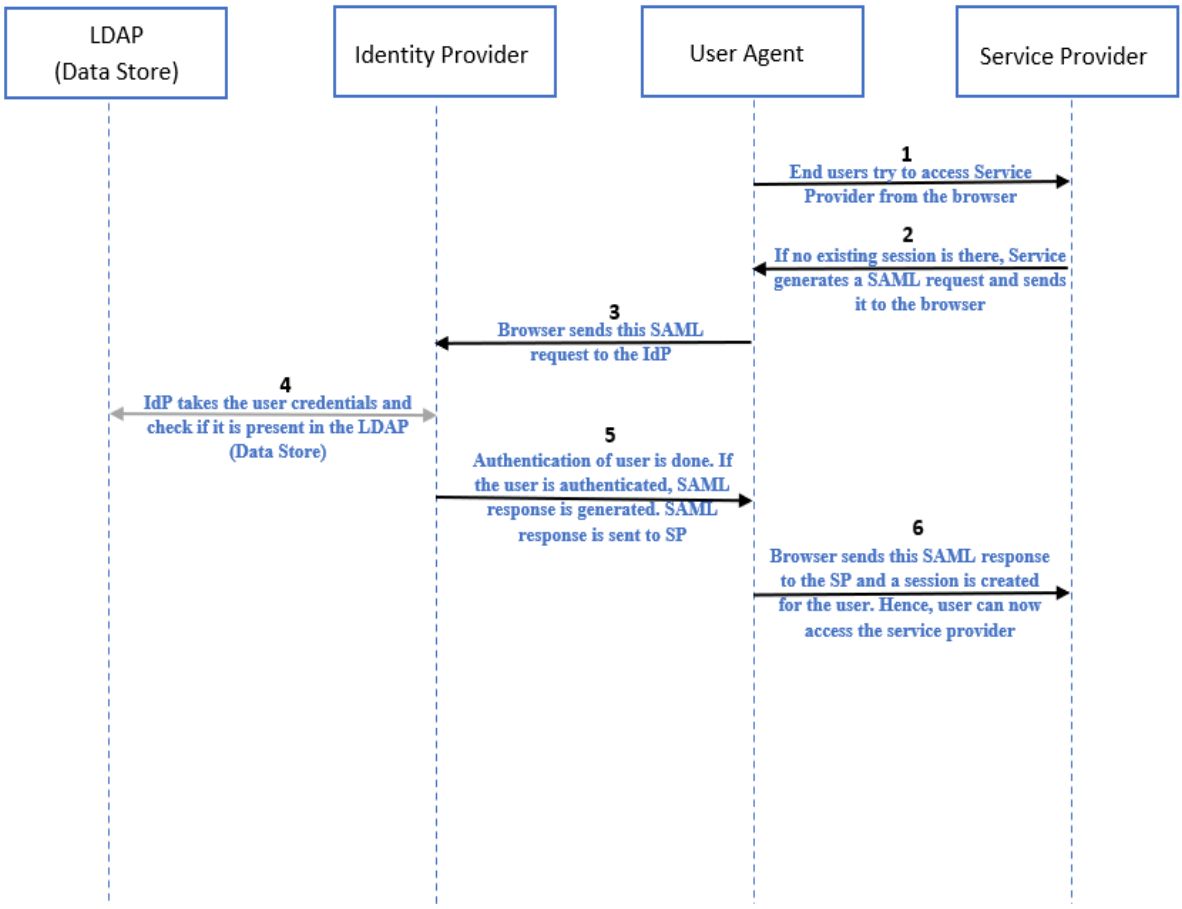


Figure 8: Sequence Diagram for SP-Initiated SSO

STEP 1: End user tries to access SP from the browser.
STEP 2,3: It checks if sessions exists for the user. If session exists, user can directly access the SP without any authentication. If session does not exist, SP generates a SAML request and user is prompted for authentication to the IdP via browser.
STEP 4: IdP further takes the credentials and validate it from Data Store(LDAP).
STEP 5: Once the user is validated the IdP generates a SAML response.
STEP 6: The assertion is sent to SP via browser and the session can start.

6.6 Use-cases of Ping Federate

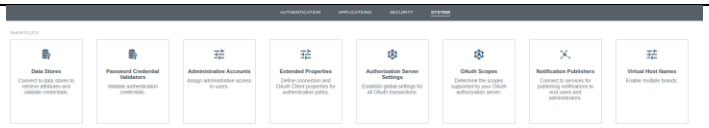
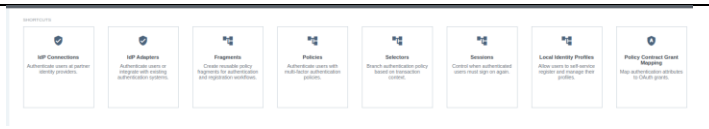
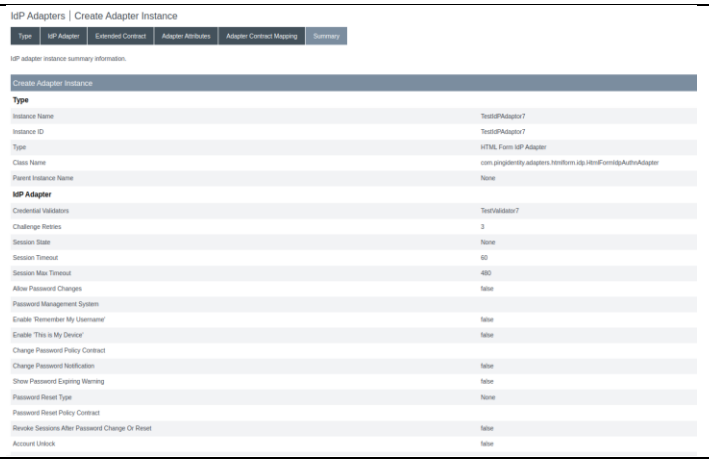
6.6.1 IdP and SP initiated SSO using IAMshowcase

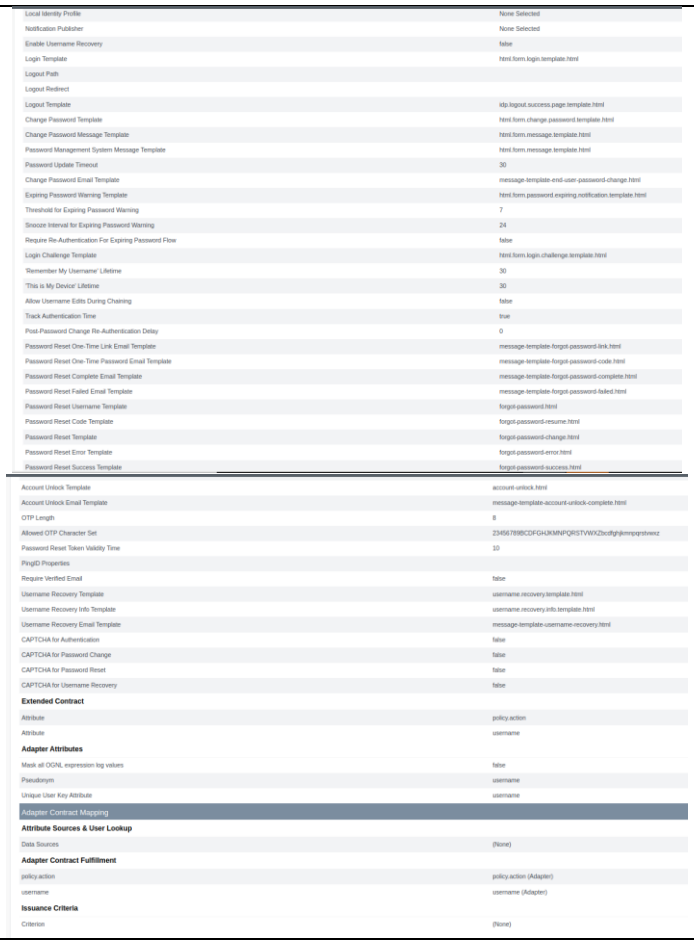
In this use-case, our aim is to understand the concept of SAML IdP and SP initiated SSO concept. We used a dummy application “IAMshowcase” as SP application for demonstrating the IdP and SP initiated connection. IAMshowcase is Test Service

Provider that can be used for demonstrating SAML IdP and SP initiated SSO connection. Ping Federate is used as IdP in this use-case. For user credentials, we directly created a used inside Password Credential Validator (PCV).

⇨ Adapter and PCV

Adapter acts as an intermediate between IdP and SP. It is Adapter that takes the user credentials during authentication and checks if it is present in PCV. PCV in simple terms, acts as an intermediate between Adapter and data store. In this case, we are not adding user from data store, instead we are creating a new user in PCV.

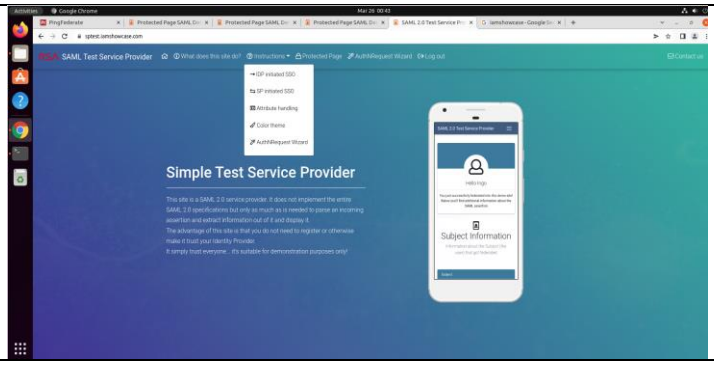
S.No.	Steps	Screenshots
1.	Navigate to PCV in System in Ping Federate console.	
2.	Make the following changes as shown in the screenshot. New PCV will be created.	
3.	Now, navigate to IdP Adapter inside authentication in Ping Federate console to create a new Adapter.	
4.	Make the following changes in the Create Adapter Instance as shown in the screenshot. Make sure to connect the adapter to the PCV just created.	

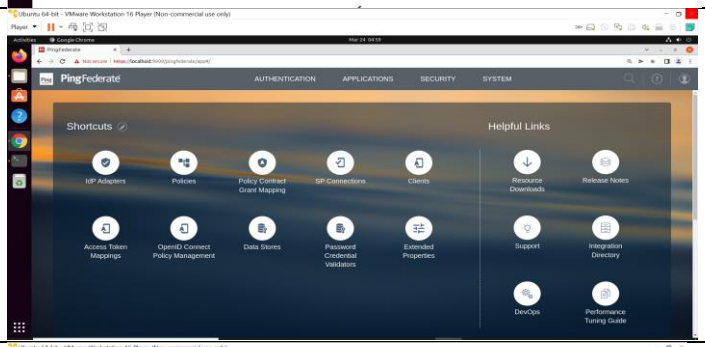
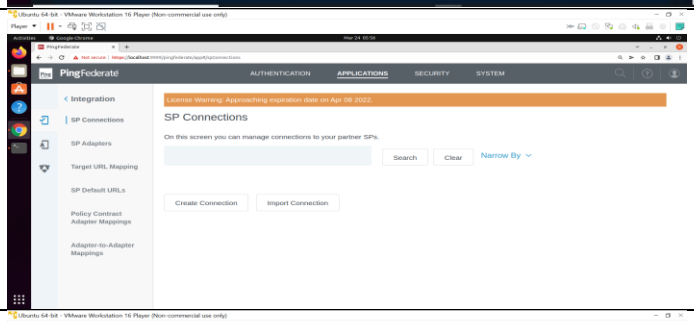
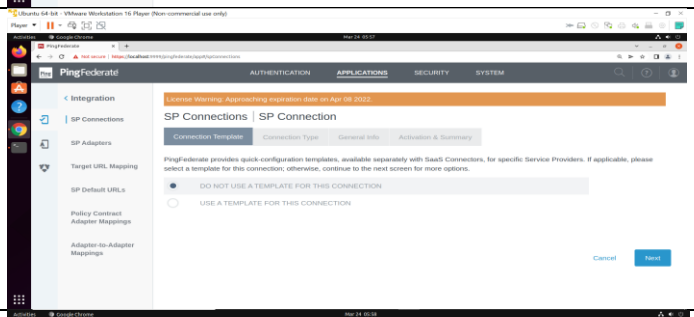
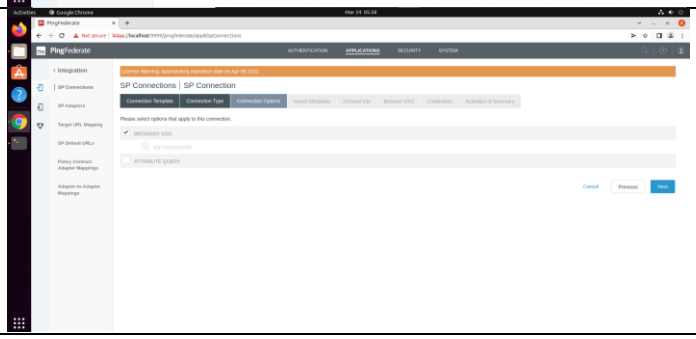
		

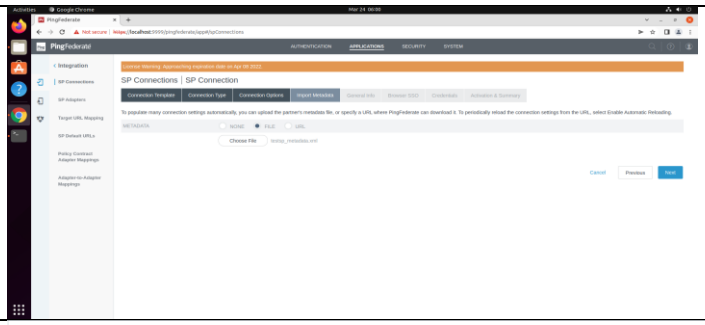
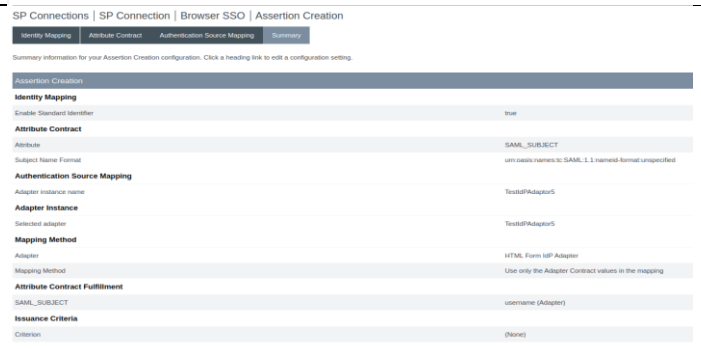
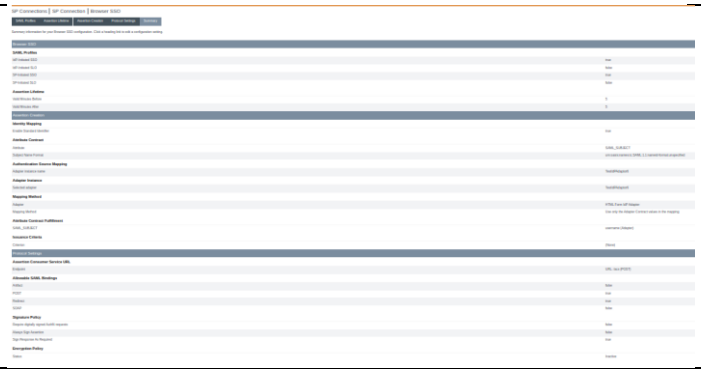
Local Identity Profile	None Selected
Notification Publisher	None Selected
Enable Username Recovery	false
Login Template	html.form.login.template.html
Logout Path	
Logout Redirect	
Logout Template	idp.logout.success.page.template.html
Change Password Template	html.form.change.password.template.html
Change Password Message Template	html.form.message.template.html
Password Management System Message Template	html.form.message.template.html
Password Update Timeout	30
Change Password Email Template	message-template-end-user-password-change.html
Expiring Password Warning Template	html.form.password.expiring.notification.template.html
Threshold for Expiring Password Warning	7
Snooze Interval for Expiring Password Warning	24
Require Re-Authentication For Expiring Password Flow	false
Login Challenge Template	html.form.login.challenge.template.html
'Remember My Username' Lifetime	30
'This is My Device' Lifetime	30
Allow Username Edits During Chaining	false
Track Authentication Time	true
Post-Password Change Re-Authentication Delay	0
Password Reset One-Time Link Email Template	message-template-forgot-password-link.html
Password Reset One-Time Password Email Template	message-template-forgot-password-code.html
Password Reset Complete Email Template	message-template-forgot-password-complete.html
Password Reset Failed Email Template	message-template-forgot-password-failed.html
Password Reset Username Template	forgot-password.html
Password Reset Code Template	forgot-password-resume.html
Password Reset Template	forgot-password-change.html
Password Reset Error Template	forgot-password-error.html
Password Reset Success Template	forgot-password-success.html
Account Unlock Template	account-unlock.html
Account Unlock Email Template	message-template-account-unlock-complete.html
OTP Length	8
Allowed OTP Character Set	23456789BCDFGHJKMNPQRSTVWXZbcdfghjkmnpqrstvwxz
Password Reset Token Validity Time	10
PingID Properties	
Require Verified Email	false
Username Recovery Template	username.recovery.template.html
Username Recovery Info Template	username.recovery.info.template.html
Username Recovery Email Template	message-template-username-recovery.html
CAPTCHA for Authentication	false
CAPTCHA for Password Change	false
CAPTCHA for Password Reset	false
CAPTCHA for Username Recovery	false
Extended Contract	
Attribute	policy.action
Attribute	username
Adapter Attributes	
Mask all OGNL expression log values	false
Pseudonym	username
Unique User Key Attribute	username
Adapter Contract Mapping	
Attribute Sources & User Lookup	
Data Sources	(None)
Adapter Contract Fulfillment	
policy.action	policy.action (Adapter)
username	username (Adapter)
Issuance Criteria	
Criterion	(None)

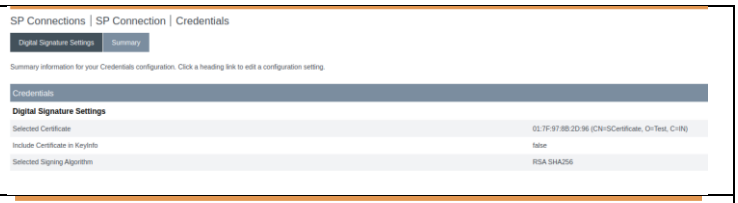
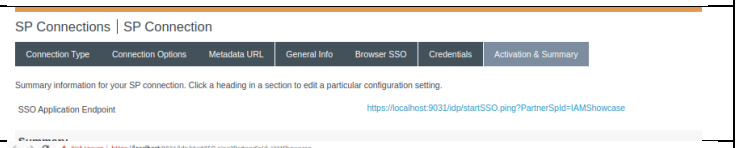
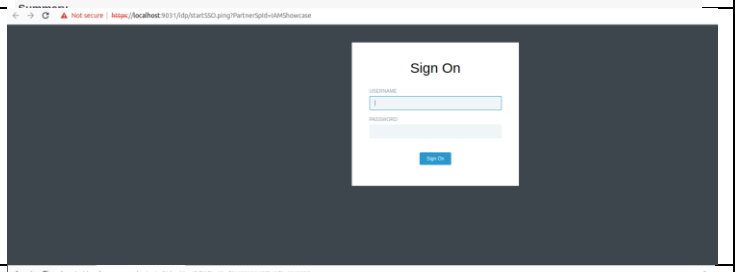
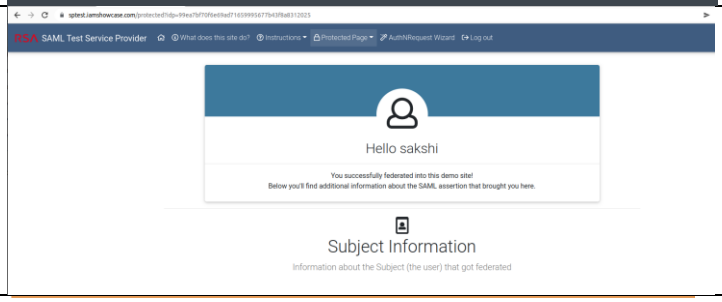
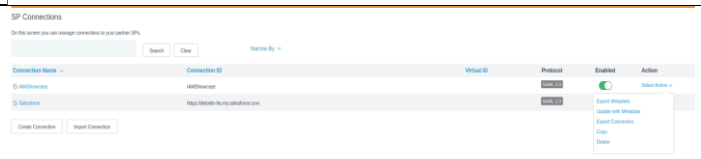
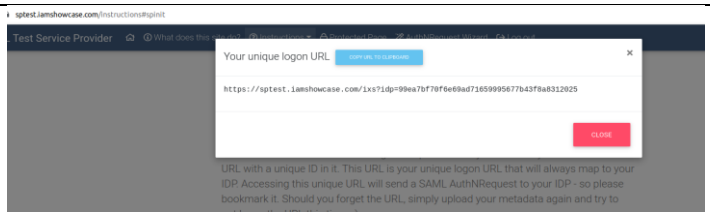
Table 20: Use-case: I (Creating Adapter and PCV)

⇨ Application Connection

S.No.	Steps	SCREENSHOTS
1.	Click on the link: https://sptest.iamshowcase.com/ and then click on IdP initiated SSO under Instructions option.	

2.	Download the metadata file by clicking on the button as shown in figure.	
3.	Open Ping Federate console and go to Applications.	
4.	Click on SP connections and then create a new connection.	
5.	Now select the option to not to use a template for this connection and click on next.	
6.	Select the browser SSO profiles and click on Next. Then select the Browser SSO option and click on Next.	

7.	Click on the file option and select the xml meta file that we downloaded from IAMshowcase website. Click on Next.	
8.	In Browser SSO, select the following settings in IdP Mapping inside Assertion creation.	
9.	Make the following changes in Assertion creation settings in Browser SSO.	
10.	Make the following changes in Protocol settings inside Browser SSO.	
11.	View the summary of the settings made in Browser SSO and click on Done.	

12.	Make the following changes in the Credentials. (If required, create a new certificate).	
13.	**IdP-initiated connection** is ready.	
14.	Navigate to this url. It will redirect directly to IdP for authentication.	
15.	After authentication, it will redirect us to SP.	
16.	Now, export the metadata file of the created IdP connection.	
17.	Navigate to the link: https://sptest.iamshowcase.com/instructions#spinit and upload the downloaded metadata file.	
18.	Copy and paste this url in new browser. Hence, **SP-initiated connection** is ready.	

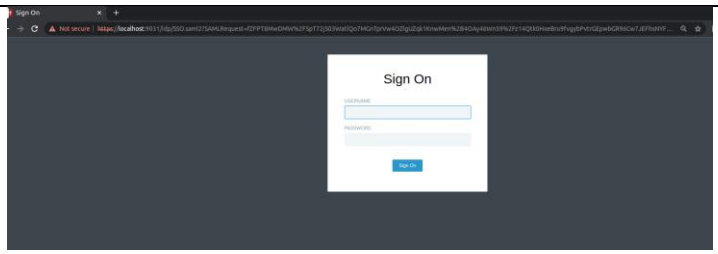
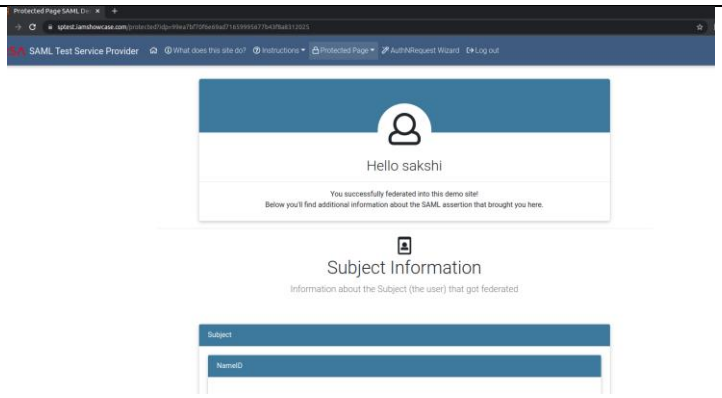
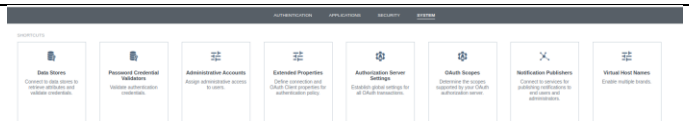
19.	Navigate to this url in a new browser. It will redirect us to SP first. Then for authentication (i.e., in order to get access to SP), it will take us to IdP.	
20.	After successful authentication, we will be redirected to SP and hence, we can access the application.	

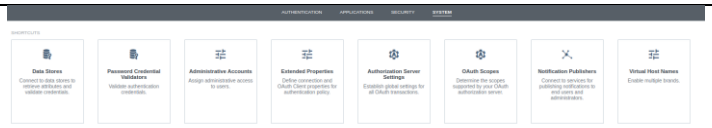
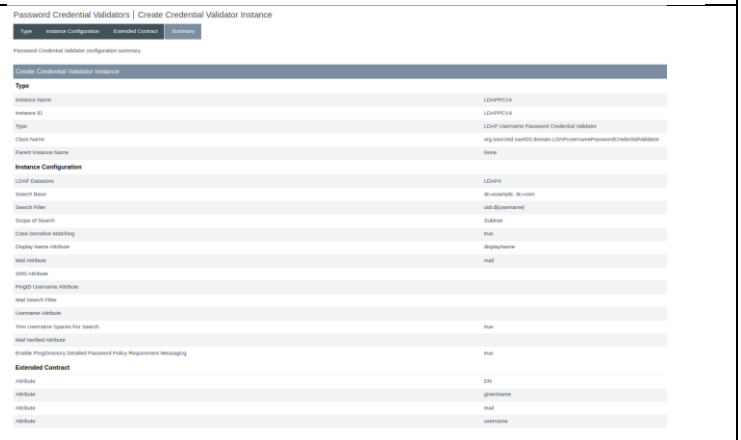
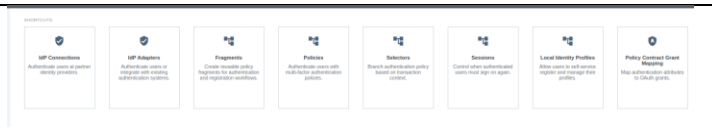
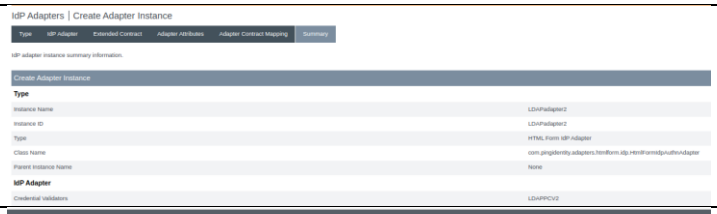
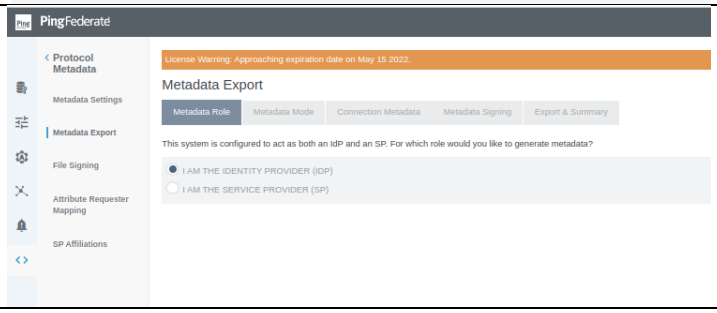
Table 21: USE-CASE: I (Creating connection)

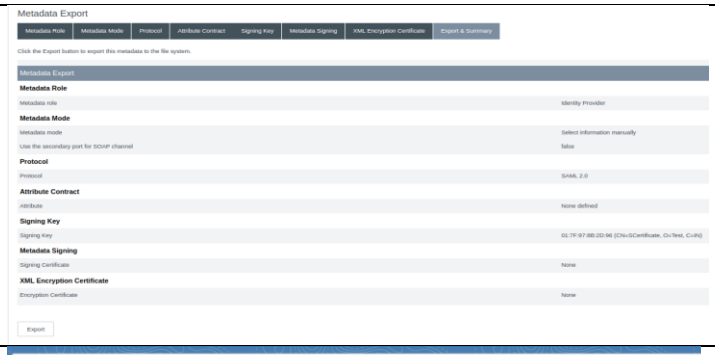
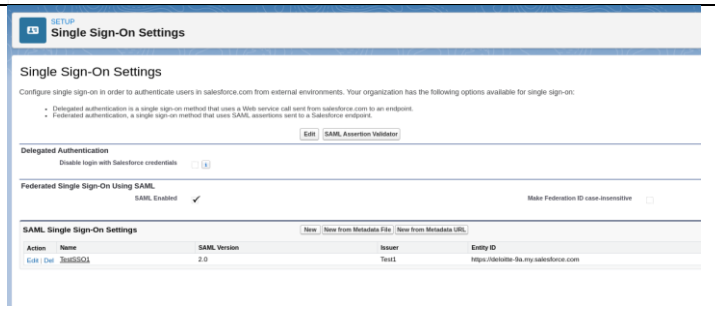
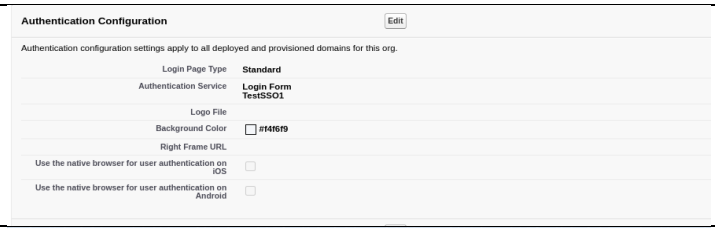
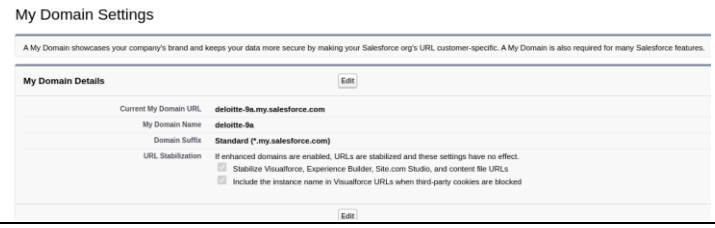
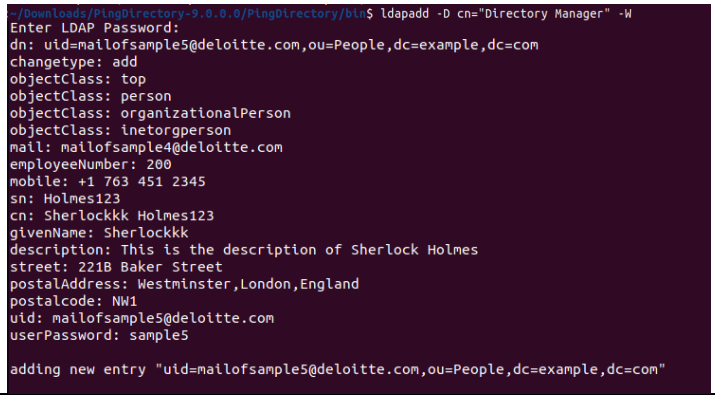
6.6.2 Protect Salesforce application using Ping Federate

In this use-case, main aim is to do the practical implementation of the understanding of IdP and SP initiated SSO concept using Salesforce Application. We built both IdP and SP initiated connection for Salesforce using Ping Federate as IdP. Also, in this use case, for user-credentials, we used Ping Directory. So, user created in Ping Directory had the access to SSO into the Salesforce Application.

Firstly, we integrated our Ping Directory with Ping Federate and after that, we created a new PCV that was linked to Ping Directory. A new adapter is created that and linked to the PCV. Then, we exchanged meta-data files between Salesforce and Ping Federate and established IdP and SP initiated connection for the application (Salesforce).

S.No.	Steps	Screenshots
1.	Login to Ping Federate console, navigate to Data Store in System to link Ping Directory configured with Ping Federate.	

2.	Create a new Data Store and connect the Ping Directory with Ping Federate as shown in the screenshot.	
3.	Go to PCV in System in Ping Federate console.	
4.	Make the following changes as shown in the figure. Make sure toc connect the right data store with PCV.	
5.	Now, go to IdP Adapter inside authentication in Ping Federate console to create a new Adapter.	
6.	Create a new adapter with PCV that we create for the data store (Ping Directory Data Store).	
7.	Now, go to System à Protocol Metadata à Metadata Export.	

8.	Make the following changes in the settings and export the meta data file of Ping Federate.	
9.	Login into Salesforce: Login | Salesforce and then go to setup à single-sign on. Select the SAML enabled checkbox and create a SAML SSO by inserting the PF metadata file and then download the Salesforce metadata file.	
10.	Then go to My domain in setup and make the following changes as shown in the figure.	
11.	Inside my domain, in my domain details, current my domain url is the SP-initiated SSO url.	
12.	Now, go to Ping Directory and create a new user.	

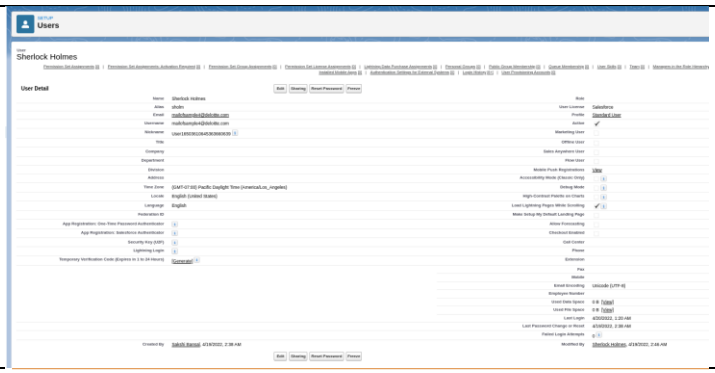
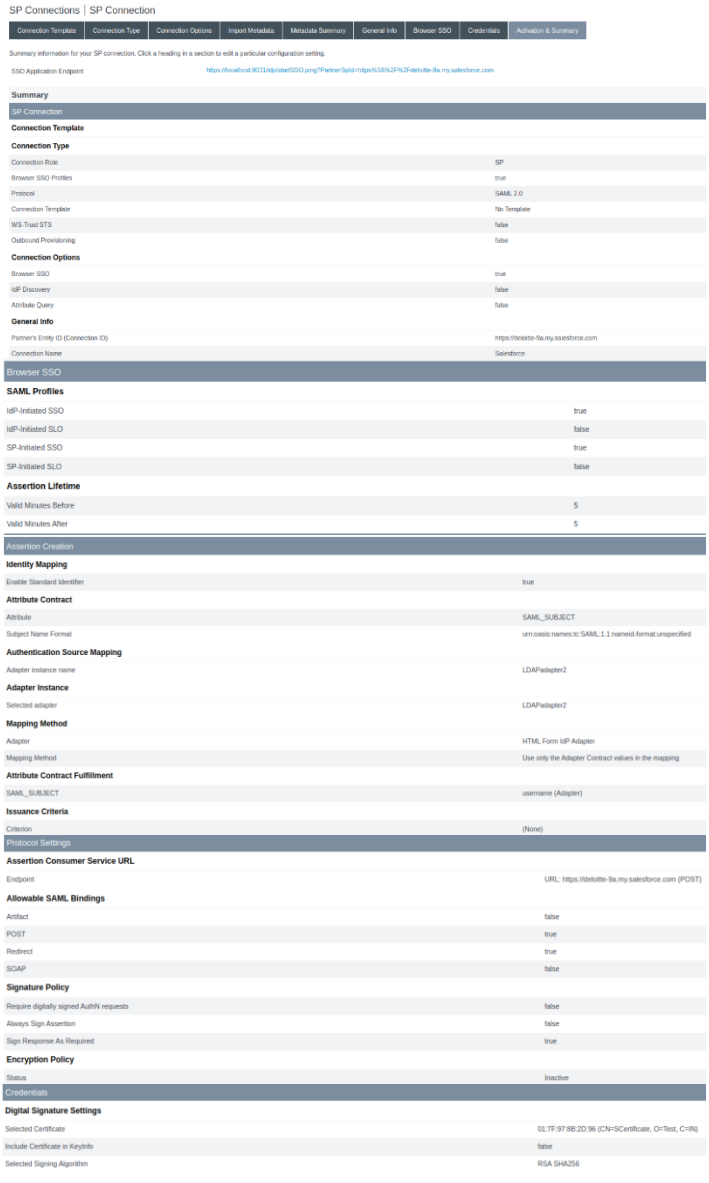
13.	Go to setup à users and create a new user. Enter the same details that have been entered for creating a new user in Ping Directory.	
14.	Create a new SP connection in Ping Federate as shown in figure. IdP-initiated connection with be built with Salesforce.	

Table 22: Use-case:II

⇨ Login into Salesforce with IdP-initiated url

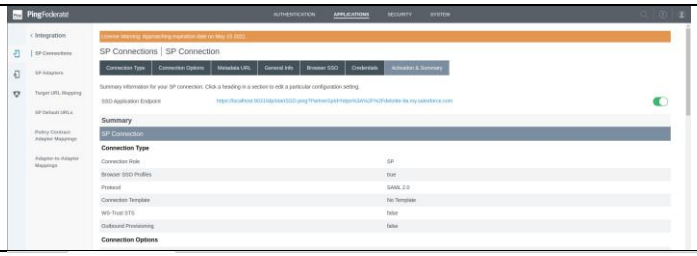
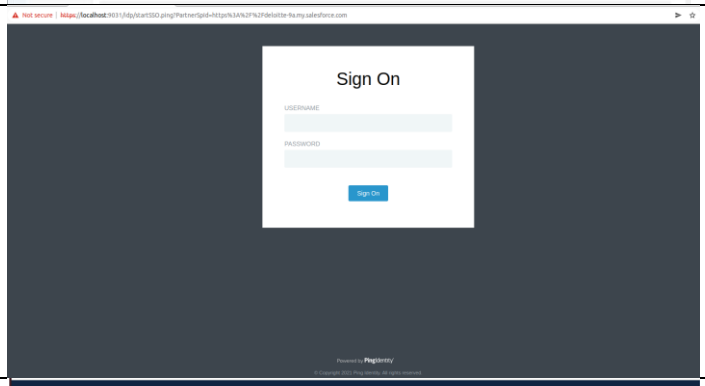
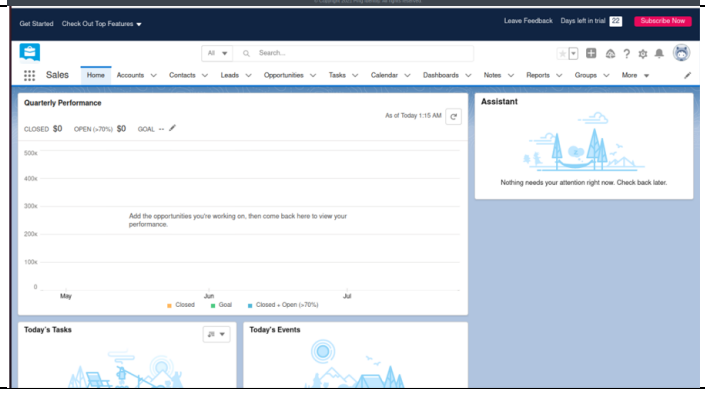
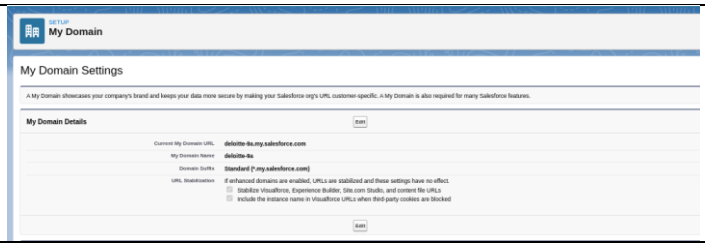
S.No.	Steps	Screenshots
1.	Click on the url shown in the screenshot.	
2.	Ping Federate authentication page as shown in screenshot will come. Enter the valid the credentials (i.e., the credentials stored in LDAP).	
3.	After successful authentication, salesforce application will open and we can access it.	

Table 23: Login into Salesforce with IdP initiated url

⇨ SSO login into Salesforce with SP-initiated url

S.No.	Steps	Screenshots
1.	Navigate to setup→ My Domain inside Salesforce account. Copy the current My Domain url.	

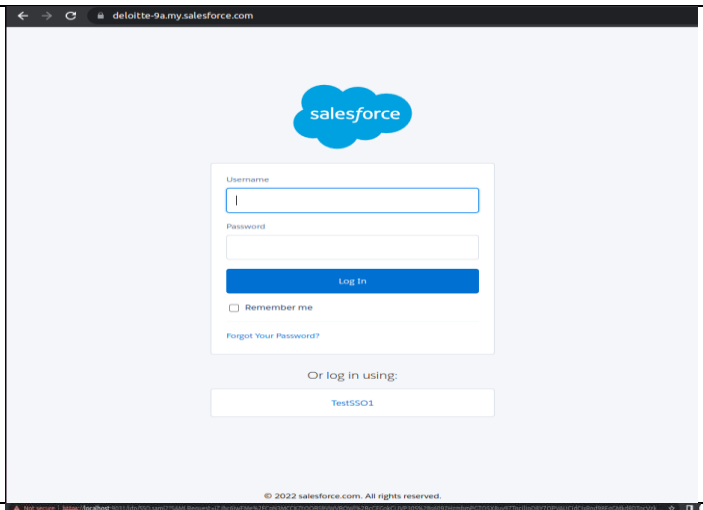
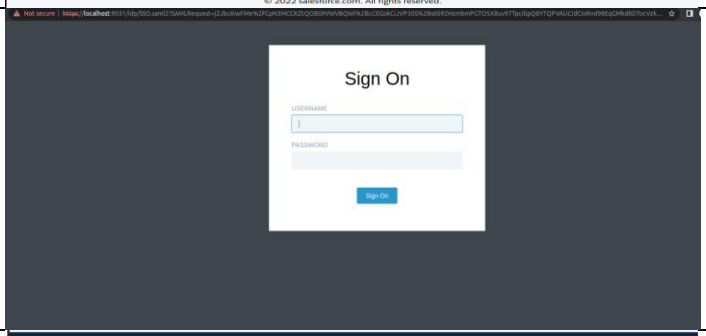
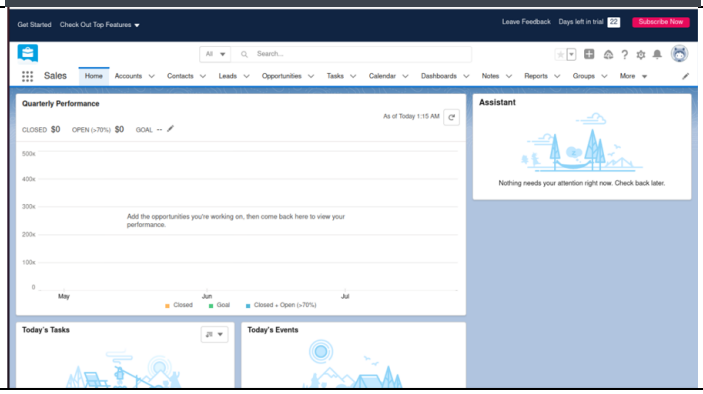
2.	Open that url in an incognito window. Salesforce login page along with SSO login option will view as shown in the screenshot.	
3.	After clicking on SSO login, it will land to Ping Federate authentication page, enter the valid user credentials (i.e., the credentials stored in LDAP).	
4.	After successful authentication into Ping Federate, we will land to Salesforce account of the LDAP user.	

Table 24: SSO login into Salesforce with SP initiated url

Hence, IdP and SP initiated connection will be created with Salesforce. It will authenticate user from Ping Directory.

In IdP initiated connection, after clicking on the url we will be redirected to Ping Federate for authentication. After entering the details of the user that we created in Ping Directory and Salesforce, we will be redirected to Salesforce application.

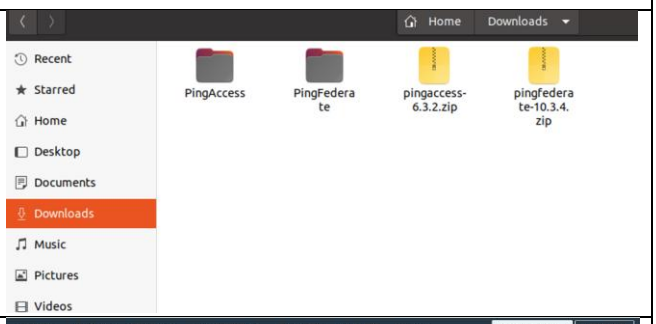
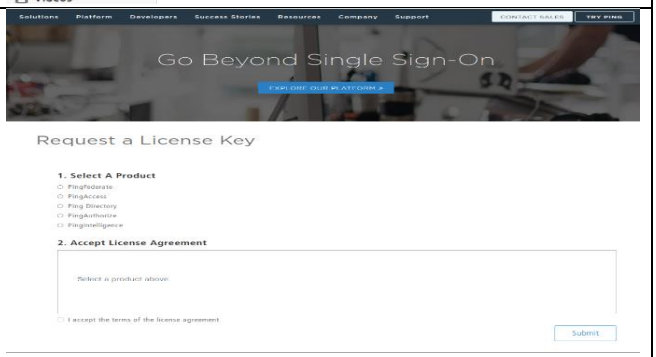
In SP initiated connection, after clicking on the url we will be first redirected to Salesforce login page along with an option for SSO. After clicking on SSO option, it will take us to Ping federate (integrated with Ping Directory) for authentication. After authentication, we will be redirected to Salesforce application account.

CHAPTER-7: PING ACCESS

7.1 What is PingAccess?

Ping Access is an alternative to the Web Access Management Solution. Ping Access is used to protect websites, APIs, and other web resources using rules and other authentication criteria. It allows both internal and external users to access web applications securely.

7.2 Installation and Configuration

S.No.	Steps	Screenshots
1.	Go to the link: https://www.pingidentity.com/en/resources/downloads/pingaccess.html and download the Linux-based Product Distribution (ZIP) file. Extract the downloaded zip file.	
2.	Go to the link: https://support.pingidentity.com/s/ and then download the license key under the tab ***Manage License Keys → View → Download***	
3.	Go to the bin folder inside the extracted ping access folder and add the following line at the beginning of run.sh file **JAVA_HOME="/usr/lib/jvm/java-11-openjdk-amd64"**	

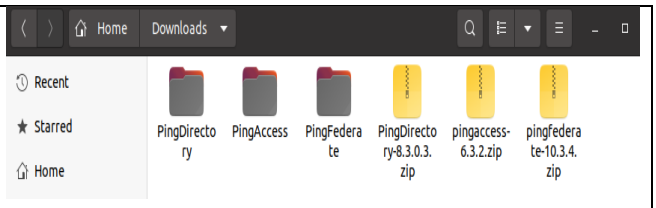
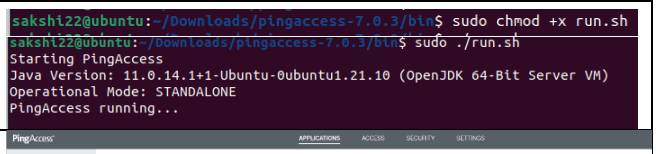
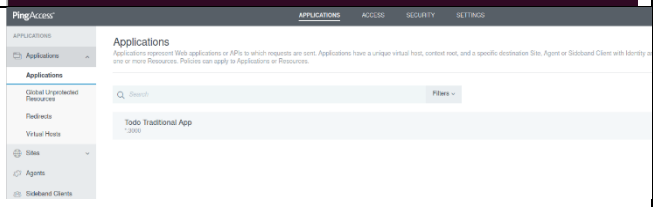
4.	open the terminal inside the bin folder and write the following command as shown in the image.	
5.	Write the following commands in terminal inside bin folder to start the ping access server.	
6.	Click on the link: https://localhost:9000/ to open the PingAccess console application. Import the downloaded license file and access the ping access.	

Table 25: Ping access installation and configuration

7.3 Protocols used in PingAccess

Ping Access uses OAuth 2.0 and OpenID connect. These protocols allow to have a granular access in Ping Access.

7.3.1 What is OAuth?

OAuth is an open-standard authorization protocol or framework that provides applications the ability for delegated access to any application or API. OAuth is a security mechanism that allows you to authorise one application to engage with another on your behalf without disclosing your password.

7.3.2 Terminologies used in OAuth 2.0

- Resource: It is something that is protected and needs to be accessed by some different service. It is also referred as protected resource.
- Resource Owner: It is an entity who has the access to the resource. An entity capable of granting access to a protected resource. Generally, user is the resource owner.
- Resource Server: It is the Server that is hosting the Resource.
- Client: It is an application making protected resource requests on behalf of the resource owner and with its authorization
- Authorization Server: Resource server is coupled with an Authorization Server. Authorization Server is responsible for making sure that whoever is accessing the resource server is authorized. Authorization server receives the request for access token from client and after successful authentication, it grants them the access token.
- Redirect URI: Authorization server redirects the user to a location once application has been successfully authorized and granted an authorization code using URL known as Redirect URI. This process is called as Callback.

- Access Tokens: Access token is a piece of some data which allows to perform authorization to gain access of resource.
- Scope: Scopes are list of permissions requested from client side. Scopes are sent from client to authorization server while sending the request to access the resource. They are used to specify the type of access (i.e., read only access or read and modify etc.) to be granted to client. Multiple scopes can also be sent.
- Consent: Consent is used by Authorization server to user asking whether client can do the actions mentioned in the scope.
- Back channel: It is a Highly secure channel and used to send request from user's server to other API server from backend. No one can decrypt the information.
- Front channel: It is Less secure channel and used to send request from browser. As browser might have loopholes like putting secret key in web app inside html then one can see it using view source with inspect element or with chrome developer tools etc.

7.3.3 How OAuth 2.0 Works?

Working of OAuth with the help of Sequence Diagram:

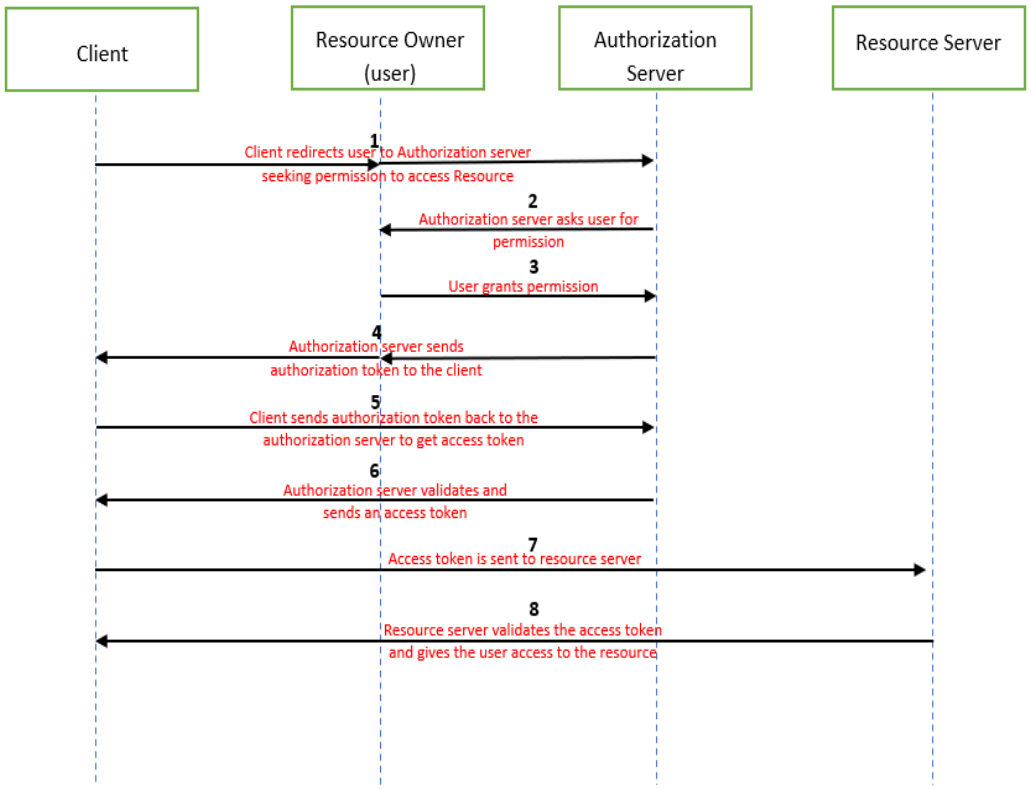


Figure 9: Sequence Diagram of OAuth

7.3.4 What is OIDC?

OIDC stands for OpenID Connect. OIDC is an open authentication protocol. It acts as an authentication layer above OAuth protocol. It acts as an extra identity layer added on top of the OAuth. It allows to check the identity of the user based on authorization done by authorization server. Using OIDC, we can fetch additional information of user as well. OpenID adds the following to OAuth 2.0:

- ID token
- If more information is required, then it connects with user information endpoint.
- Standardized implementation
- Standard set of scopes

CONCLUSION

In this internship, I learnt about the concepts of Identity and Access Management, Web Access Management, Multi Factor Authentication and Single sign-on. I also learned about the importance of cyber security and how these principles may aid in the prevention of cyber-attacks. Also, I learnt about various technologies such as SAML, OAuth and OIDC.

I learnt about Okta. Okta was the first vendor to provide services on-cloud applications. Okta is a service used for Identity Management. Okta provides various services such as MFA, SSO etc.

I also learnt about various PingIdentity products such as Ping Directory, Ping Federate and Ping Access. Ping Directory is basically a data store that can be used to maintains identity of users at a large scale. It aims at providing better security and high performance. Ping Federate is used to provide authentication and single sign-on services. Ping access aims at providing delegated access of applications and APIs to users. I performed numerous use-cases of Ping Directory and Ping Federate to gain a better understanding of the requirement for the Ping products.

Apart from that, my four-month internship at Deloitte provided me with valuable corporate experience. It assisted me in developing soft skills such as analytical abilities, time management, and teamwork. My mentors supported me through every step of my internship, and with their continual input, I was able to accomplish a great deal throughout this time.

REFERENCES

- https://developer.okta.com/docs/concepts/saml/
- https://en.wikipedia.org/wiki/Ping_Identity#:~:text=Ping%20Identity%20provides%20federated%20identity,for%20authenticating%20to%20web%20applications.
- https://www.pingidentity.com/en/resources/content-library/articles/openid-connect.html
- https://www.pingidentity.com/en/pingone/pingfederate.html
- https://www.pingidentity.com/en/pingone/pingdirectory.html#:~:text=PingDirectory%20is%20a%20high%2Dperformance,high%20performance%20during%20peak%20usage.
- https://www.pingidentity.com/en/pingone/pingaccess.html#:~:text=PingAccess%20is%20a%20centralized%20access,access%20the%20resources%20they%20need.
- https://oauth.net/
- https://openid.net/connect/
- https://www.microsoft.com/en-in/security/business/identity-access-management/mfa-multi-factor-authentication
- https://en.wikipedia.org/wiki/Single_sign-on