# Single-sign on (SSO)

* [Introduction](#Single-signon(SSO)-Introduction)
  + [Why Incorta needs to support SSO for user authentication?](#Single-signon(SSO)-WhyIncortaneedstosup)
  + [General workflow for SSO integration](#Single-signon(SSO)-GeneralworkflowforSS)
  + [Prerequisites for enabling SSO authentication](#Single-signon(SSO)-Prerequisitesforenab)
    - [Configure SSO provider](#Single-signon(SSO)-ConfigureSSOprovider)
    - [Configure Incorta](#Single-signon(SSO)-ConfigureIncorta)
    - [Update the tenant’s Authentication Type](#Single-signon(SSO)-Updatethetenant’sAut)

# Introduction

Single sign-on or (SSO) for short, is an authentication scheme that allows a user to log in with a single ID and password to any of several related, yet independent, software systems. check [this link](https://en.wikipedia.org/wiki/Single_sign-on) for more information.

Typically this scheme is accomplished using 2 popular standard protocols, [SAML](https://en.wikipedia.org/wiki/Security_Assertion_Markup_Language) & [OAuth2.0](https://en.wikipedia.org/wiki/OAuth#OAuth_2.0). Although OAuth is mainly used for access delegation i.e. allowing internet users to grant access to their information for websites and applications, it can be used as an SSO provider. We will see how that is accomplished but not in this document.

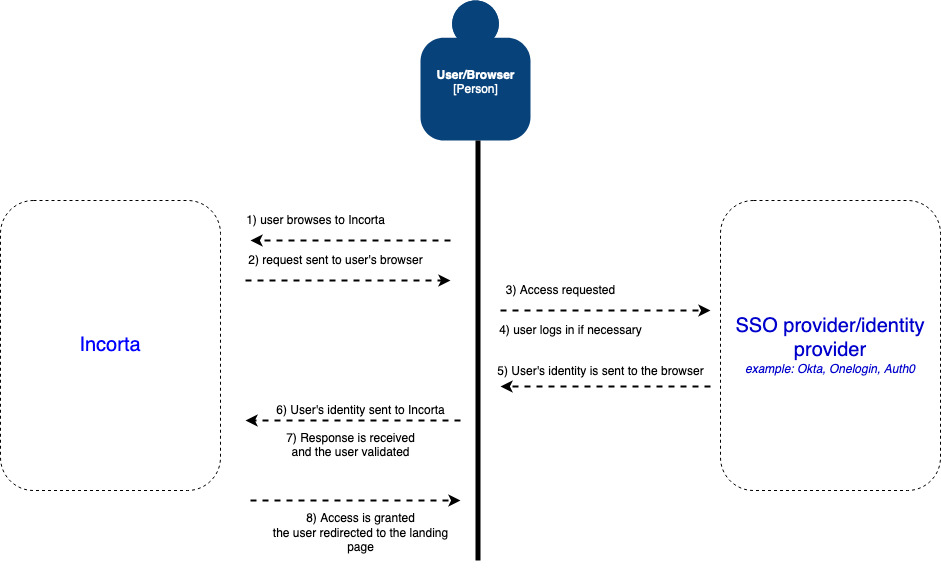
## Why Incorta needs to support SSO for user authentication?

Many of Incorta’s customers already use an identity management system that holds all enterprise employees' information in one place. having a central repository for all employees' data makes it easier to provision those employees, enable/disable application access to certain employees, etc.

Instead of having users typing their usernames/passwords to access Incorta, Incorta can integrate with the customer’s identity management system (SSO provider) to handle user authentication.

## General workflow for SSO integration

The following diagram explains the interaction between Incorta, the identity provider, and the user’s browser



1- The user browses to Incorta, typically from their browser.

2- Incorta checks if there is an existing user session, if not, the user is redirected to the identity provider to login.

3- The browser redirects the user to the identity provider and asks the user for credentials if necessary.

4- If the user is authenticated successfully, the identity provider redirects back to Incorta along with a token representing the user’s identity (username, email, etc.).

5- The user’s identity sent from the identity provider is verified by Incorta and if the user exists access to Incorta is granted and the user is redirected to Incorta’s landing page.

## Prerequisites for enabling SSO authentication

### Configure SSO provider

You need to refer to the SSO/identity provider on how to define Incorta as an application

### Configure Incorta

You will need to update the server.xml for the analytics service with the appropriate Valve configuration based on the identity provider you are integrating with, below is a sample for OktaValve

<Valve className="com.incorta.sso.valves.OktaValve"

confFilesMap="demo\_49=/path/to/okta/config/file/demo\_okta.xml"

LoggingEnabled="true" />

**For development environment,** the server.xml should be under /Servers/node/services/analytics\_service/conf

**For Incorta installation,** the server.xml should be under /path/to/incorta-node/services/analytics-service-uuid/conf. to know the exact path for the analytics service, run ./listServices.sh script which lists all available services along with their types under this node.

You will need to make sure that the appropriate jar file is present under /path/to/incorta-node/runtime/lib and other SSO-related jars are removed.

For Okta and Auth0, make sure that incorta-sso.jar is present and incorta-onelogin.valv.jar is removed

For Onelogin and other SAML-based SSO providers, make sure that incorta-onelogin.valv.jar is present and incorta-sso.jar is removed.

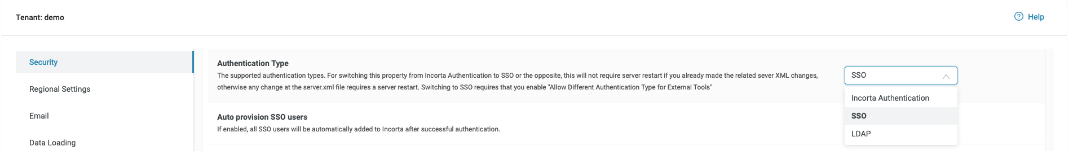
For Apple’s DS Auth, make sure that apple-dsauth.jar is present and both (incorta-sso.jar, incorta-onelogin.valv.jar) are removed.

**Note: Starting Incorta 5.0.x all three jar files can exist together so you do not need to manually remove any jar files.**

### Update the tenant’s Authentication Type

This can be performed by CMC or TMT

from CMC, navigate to the tenant’s security configurations and update the authentication type to SSO as the screenshot below:



and make sure to click on the “**Save**“ button below

From TMT (our beloved command-line tool (smile))

navigate to /path/to/cmc/tmt and execute the following command:

./tmt.sh -clnm <cluster\_name> -u <tenant\_name> sso-login-enable true