REMOVED_Upgrading from the Previous Release

We recommend migrating directly to one of the latest stable releases of WSO2 Identity Server (i.e., version 5.6.0 or a later version). For instructions on migrating directly to a later version, see REMOVED_Upgrading From an Older Version of WSO2 IS.

The following instructions guide you through upgrading from WSO2 Identity Server 5.1.0 to WSO2 Identity Server 5.2.0.

Before you begin

This release is a WUM-only release. This means that there are no manual patches and any further fixes or latest updates for this release can be updated through the WSO2 Update Manager (WUM). Please note the following:

- **If you are upgrading to this version to use this version in your production environment**, use the WSO2 Update Manager and get the latest available updates for WSO2 IS 5.2.0. For more information on how to do this, see Updating WSO2 Products.
- **If you are upgrading to this version only to do an incremental upgrade to the next available version** (e.g., if you are upgrading from WSO2 IS 5.1.0 - 5.3.0), you can skip this step and migrate to 5.2.0 by following the steps given in this document. You do not need to use WUM in this instance because the WUM updates available for this version will be included in the WSO2 IS pack of the next version.

Migrating the embedded LDAP user store

It is not generally recommended to use the embedded LDAP user store that is shipped with WSO2 Identity Server in production setups. However, if migration of the embedded LDAP is required, follow the instructions below to migrate the existing IS 5.1.0 LDAP user store to IS 5.2.0.

1. Copy the `<IS-5.1-Home>/repository/data` folder to `<IS-5.2-Home/repository/data` folder.
2. Restart the server to save the changes.

To upgrade the version of WSO2 Identity Server, the user store database should be upgraded. Note that there are no registry schema changes between versions.

In this topic, `<OLD_IS_HOME>` is the directory that Identity Server 5.1.0 resides in and `<NEW_IS_HOME>` is the directory that Identity Server 5.2.0 resides in.

1. Download Identity Server 5.2.0 and unzip it in the `<NEW_IS_HOME>` directory.
2. Take a backup of the existing database used by Identity Server 5.1.0. This backup is necessary in case the migration causes issues in the existing database.
3. Make a copy of the `<OLD_IS_HOME>/repository/conf` folder.
4. Copy the following files from the `<NEW_IS_HOME>/repository/conf` folder and paste it in the copy of the `<OLD_IS_HOME>/repository/conf` directory in the relevant sub folder:
    - `repository/conf/event-processor.xml`
    - `repository/conf/security/Owasp.CsrfGuard.Carbon.properties`
    - `repository/conf/tomcat/carbon/WEB-INF/web.xml`
    - `repository/conf/identity/oidc-scope-config.xml`
5. Replace the `<NEW_IS_HOME>/repository/conf` folder with the modified copy of the `<OLD_IS_HOME>/repository/conf` folder.

6. Configure the <NEW_IS_HOME>/repository/conf/carbon.xml file with the same configurations made in the <OLD_IS_HOME>/repository/conf/carbon.xml file.
7. Open the `<NEW_IS_HOME>/repository/conf/identity/identity.xml` file and add the `<PoolSize>` tag under the `<SessionDataPersist>` tag with the default value as 200, if you have not already done so. If <SessionDataPersist> is commented out, be sure to uncomment it.

```
<SessionDataPersist>
            .....
            <PoolSize>200</PoolSize>
                        .....
</SessionDataPersist>
```

8. Open the `<NEW_IS_HOME>/repository/conf/claim-config.xml` file and add the following new claims.

```
<Claim>
    <ClaimURI>http://wso2.org/claims/identity/lastLoginTime</ClaimURI>
    <DisplayName>Last Login</DisplayName>
    <!-- Proper attribute Id in your user store must be configured for this -->
    <AttributeID>carLicense</AttributeID>
    <Description>Last Login Time</Description>
</Claim>
<Claim>
    <ClaimURI>http://wso2.org/claims/identity/lastPasswordUpdateTime</ClaimURI>
    <DisplayName>Last Password Update</DisplayName>
    <!-- Proper attribute Id in your user store must be configured for this -->
    <AttributeID>businessCategory</AttributeID>
    <Description>Last Password Update Time</Description>
</Claim>
<Claim>
    <ClaimURI>http://wso2.org/claims/identity/accountDisabled</ClaimURI>
    <DisplayName>Account Disabled</DisplayName>
    <!-- Proper attribute Id in your user store must be configured for this -->
    <AttributeID>ref</AttributeID>
    <Description>Account Disabled</Description>
</Claim>
```

9. Copy any custom OSGI bundles that were added manually from the `<OLD_IS_HOME>/repository/components/dropins` folder and paste it in the `<NEW_IS_HOME>/repository/components/dropins` folder.
10. Copy the `.jks` files from the `<OLD_IS_HOME>/repository/resources/security` folder and paste them in `<NEW_IS_HOME>/repository/resources/security`.
11. If you have created tenants in the previous Identity Server copy content in the `<OLD_IS_HOME>/repository/tenants` directory to `<NEW_IS_HOME>/repository/tenants/` directory.
12. If you have created secondary user stores in the previous Identity Server copy content in the `<OLD_IS_HOME>/repository/deployment/server/userstores` directory to `<NEW_IS_HOME>/repository/deployment/server/userstores/` directory
13. Download the migration resources and unzip it to a local directory. This folder is referred to as `<IS5.2.0_MIGRATION_TOOL_HOME>`.

    i. Copy the `<IS5.2.0_MIGRATION_TOOL_HOME>/dbscripts/identity/migration-5.1.0_to_5.2.0` folder and paste it in the `<NEW_IS_HOME>/dbscripts/identity` directory.

ii. Copy the `<IS5.2.0_MIGRATION_TOOL_HOME>/dbscripts/migration-5.1.0_to_5.2.0` folder and paste it in the `<NEW_IS_HOME>/dbscripts` directory.
iii. Copy the `org.wso2.carbon.is.migrate.client-5.2.0.jar` file in the `<IS5.2.0_MIGRATION_TOOL_HOME>/dropins` directory to the `<NEW_IS_HOME>/repository/components/dropins` directory.
iv. Alternatively, if you are using Oracle database, you can either provide the database owner credentials in the datasource configurations (identity and user management databases) or pass the identity database owner name with `-DidentityOracleUser` and user management database owner name with `-DumOracleUser`.

14. Run the respective migration script against your database.

Note: The db scripts add the following new claims and claim mappings to your database. If you have already mapped the **carLicense** and/or **businessCategory** attributes to a claim, follow the steps below to update the SQL with a different attribute value.

Claim	Mapped Attribute
http://wso2.org/claims/identity/lastLoginTime	carLicense
http://wso2.org/claims/identity/lastPasswordUpdateTime	businessCategory

a. Open the relevant db script in an editor.
b. Change the relevant SQL commands to add a suitable attribute.

Change the **'carLicense'** attribute value to a different attribute that is not mapped to a claim.

lastLoginTime claim for super tenant

```
INSERT INTO UM_CLAIM (
            UM_DIALECT_ID,
            UM_CLAIM_URI,
            UM_DISPLAY_TAG,
            UM_DESCRIPTION,
            UM_MAPPED_ATTRIBUTE,
            UM_TENANT_ID,
            UM_READ_ONLY)
VALUES ((SELECT UM_ID FROM UM_DIALECT WHERE
UM_DIALECT_URI='http://wso2.org/claims' AND UM_TENANT_ID=-1234),
'http://wso2.org/claims/identity/lastLoginTime','Last Login
Time','Last Login Time','carLicense',-1234,true);
```

lastLoginTime claim for migrating tenants

```
INSERT INTO um_claim
            (um_dialect_id,
             um_claim_uri,
             um_display_tag,
             um_description,
             um_mapped_attribute,
             um_tenant_id,
             um_read_only)

SELECT DIALECT.um_id,
```

```
       'http://wso2.org/claims/identity/lastLoginTime',
       'Last Login Time',
       'Last Login Time',
       'carLicense',
       DIALECT.um_tenant_id,
       true
FROM   um_dialect AS DIALECT
       JOIN um_tenant AS TENANT
         ON DIALECT.um_tenant_id = TENANT.um_id
WHERE  DIALECT.um_dialect_uri = 'http://wso2.org/claims';
```

Change the **'businessCategory'** attribute value to a different attribute that is not mapped to a claim.

lastPasswordUpdateTime claim for super tenant

```
INSERT INTO UM_CLAIM (
            UM_DIALECT_ID,
            UM_CLAIM_URI,
            UM_DISPLAY_TAG,
            UM_DESCRIPTION,
            UM_MAPPED_ATTRIBUTE,
            UM_TENANT_ID,
            UM_READ_ONLY)
VALUES ((SELECT UM_ID FROM UM_DIALECT WHERE
UM_DIALECT_URI='http://wso2.org/claims' AND UM_TENANT_ID=-1234),
'http://wso2.org/claims/identity/lastPasswordUpdateTime','Last
Password Update','Last Password Update','businessCategory',-1234,
true);
```

lastPasswordUpdateTime claim for migrating tenants

```
INSERT INTO um_claim
            (um_dialect_id,
             um_claim_uri,
             um_display_tag,
             um_description,
             um_mapped_attribute,
             um_tenant_id,
             um_read_only)

SELECT DIALECT.um_id,
       'http://wso2.org/claims/identity/lastPasswordUpdateTime',
       'Last Password Update',
       'Last Password Update',
       'businessCategory',
       DIALECT.um_tenant_id,
       true
FROM   um_dialect AS DIALECT
```

```
        JOIN um_tenant AS TENANT
          ON DIALECT.um_tenant_id = TENANT.um_id
WHERE   DIALECT.um_dialect_uri = 'http://wso2.org/claims';
```

15. To avoid a known issue related to OpenID Connect requested claims, update WSO2 IS using the WSO2 Update Manager (WUM). To do this, follow the instructions on the Updating WSO2 Products page and update the WSO2 Identity Server using WUM.
16. Start the Identity Server 5.2.0 with the following command to perform the data migration for all components.

    See the notes below to perform migration for individual components or for active tenants only.

    a. Linux/Unix:

    ```
    sh wso2server.sh -Dmigration=true -DmigrateActiveTenantsOnly=true -
    DmigrateIdentityDB=true -DmigrateUMDB=true
    ```

    b. Windows:

    ```
    wso2server.bat -Dmigration=true -DmigrateActiveTenantsOnly=true -
    DmigrateIdentityDB=true -DmigrateUMDB=true
    ```

Migrate individual components

Optional: To migrate certain components only, use the relevant commands in the table below.

Warning! Unless specifically required, it is recommended to perform the full data migration by executing the command given above. Component migration is intended for certain special cases only, and may cause errors due to incomplete migration, if done incorrectly.

Component	Linux/Unix	Windows
Identity Database	`sh wso2server.sh -Dmigration=true -DmigrateIdentityDB=true`	`wso2server.bat -Dmigration=true -DmigrateIdentityDB=true`
User Management Database	`sh wso2server.sh -Dmigration=true -DmigrateUMDB=true`	`wso2server.bat -Dmigration=true -DmigrateUMDB=true`

Migrate active tenants only

Optional: If you have any disabled/inactive tenants in your previous version of WSO2 IS that you do not want to bring forward to the next version, do a complete migration for all components with active tenants only. This also includes migration of the super tenant.

Start the server against the migration client jar located in the `<IS_HOME>/repository/components/dropins` directory using the `-DmigrateActiveTenantsOnly` flag, as shown below.

```
sh wso2server.sh -Dmigration=true -DmigrateActiveTenantsOnly=true
```

Configuration changes in IS 5.2.0

The table below lists out all the configuration changes from IS 5.1.0 to IS 5.2.0. You can scroll through the table and change the relevant configurations according to the features you are using.

Tip: Scroll left/right to view the entire table below.

Due to a fix done in this release, the effective default value of the system property `org.apache.xml.security.ignoreLineBreaks` has been changed from “true” to “false”. Due to this change, you will observe line breaks in SAML responses.

However, if the SAML response consuming client applications have used a standard library such as OpenSAML and use canonicalization when processing the response, this should not cause any problems. Therefore, our recommendation is to use a standard library to process SAML responses on consuming applications.

If you have any concerns about this behavioral change or if the SAML response consuming client applications does not use canonicalization when processing the response and the client cannot be updated to do so, add the following jvm parameter to the server startup script located in the `<IS_HOME>/bin/` folder to revert back to the previous behavior.

```
-Dorg.apache.xml.security.ignoreLineBreaks=true
```

Configuration File	Changes	
`oidc-scope-config.xml` file stored in the `<PRODUCT_HOME>/repository/conf/identity/` directory.	The following configuration file was added to enable grouping claims that are bound to a scope value in OpenID Connect (OIDC). When requesting for an OIDC token, you can specify a scope value that is bound to a set of claims in the `oidc-scope-config.xml` file. When sending that OIDC token to the userinfo endpoint, only the claims that are common to both the oidc-scope-config and the service provider claim configuration, will be returned.	

identity-mgt.properties file stored in the `<PRODUCT_HOME>/repository`	The following parameters were added:
	`# Whether to use hash of username when storing codes.`
	`# Enable this if Registry is used to store the codes and if username may contain non alphanumeric characters.`	

`/conf /identity/` directory.	`UserInfoRecovery.UseHashedUserNames=false`
	`UserInfoRecovery.UsernameHashAlg=SHA-1`
	
	If you have enabled the using email address as the username option, the confirmation codes are retained after they are used, due to the special character '@' contained in the email address. To resolve this, you can set the `UserInfoRecovery.UseHashedUserNames` parameter to true so that the registry resources will be saved by **hash of username** instead of the email address username which contains the '@' sign.	

	The following properties were added to support notification sending for account enabling and disabling:
	
	`Notification.Sending.Enable.Account.Disable=false`
	`Notification.Sending.Enable.Account.Enable=false`
	
	For more information, see User Account Locking and Account Disabling.	

	The following property was added to check if the account has been locked, at the point of authentication. `Authentication.Policy.Check.Account.Disable=false`	

`EndpointConfig.properties` file stored in the `<PRODUCT_HOME>/repository/conf/identity/` directory.	The following properties were replaced:
	
	Old configuration
	
	`identity.server.host=localhost`
	`identity.server.port=9443`
	`identity.server.serviceURL=/services/`
	
	The properties above were replaced with the following:
	
	New configuration
	
	`#identity.server.serviceURL=https://localhost:9443/services/`	

entitlement.properties file stored in the `<PRODUCT_HOME>/repository/conf/identity/` directory.	When policy sets are used with entitlements, the default policy set cache size is 100. This may cause frequent cache eviction if there are more than 100 policies in the set. To avoid this, configure the following property. It will cause the cache size to increase depending on the policy set size for better performance. `PDP.References.MaxPolicyEntries=3000`	
`identity.xml` file	Session data persistence is enabled by default from IS 5.2.0 onwards.	

stored in the <PRODUCT_HOME>/repository/conf/identity/ directory.

```
<SessionDataPersist>
    <Enable>true</Enable>
    <Temporary>true</Temporary>
    <PoolSize>0</PoolSize>
    <SessionDataCleanUp>
        <Enable>true</Enable>
        <CleanUpTimeout>20160</CleanUpTimeout>
        <CleanUpPeriod>1140</CleanUpPeriod>
    </SessionDataCleanUp>
    <OperationDataCleanUp>
        <Enable>true</Enable>
        <CleanUpPeriod>720</CleanUpPeriod>
    </OperationDataCleanUp>
</SessionDataPersist>
```

The following properties were removed:

```
<!--SessionContextCache>
         <Enable>true</Enable>
         <Capacity>100000</Capacity>
</SessionContextCache-->
```

The following property was added to the <SSOService> and <PassiveSTS> elements:

```
<SLOHostNameVerificationEnabled>true<
/SLOHostNameVerificationEnabled>
```

For more information on configuring hostname verification, see the info note at the bottom of the Configuring WS-Federation page.

Listeners and properties related to analytics in WSO2 Identity Server were added. For more information, see Prerequisites to Publish Statistics.

Listeners

```
<EventListener type="org.wso2.carbon.identity.core.
handler.AbstractIdentityMessageHandler" name="org.wso2.
carbon.identity.data.publisher.application.
authentication.impl.DASLoginDataPublisherImpl" orderId="
10" enable="false" />
<EventListener type="org.wso2.carbon.identity.core.
handler.AbstractIdentityMessageHandler" name="org.wso2.
carbon.identity.data.publisher.application.
authentication.impl.DASSessionDataPublisherImpl"
orderId="11" enable="false" />
<EventListener type="org.wso2.carbon.identity.core.
handler.AbstractIdentityMessageHandler" name="org.wso2.
```

```
carbon.identity.data.publisher.application.
authentication.AuthnDataPublisherProxy" orderId="11"
enable="true" />
```

Properties

```
<ISAnalytics>
        <DefaultValues>
            <userName>NOT_AVAILABLE</userName>
            <userStoreDomain>NOT_AVAILABLE<
/userStoreDomain>
            <rolesCommaSeperated>NOT_AVAILABLE<
/rolesCommaSeperated>
            <serviceprovider>NOT_AVAILABLE<
/serviceprovider>
            <identityProvider>NOT_AVAILABLE<
/identityProvider>
        </DefaultValues>
    </ISAnalytics>
```

The security element was updated:

```
<!-- Security configurations-->
<Security>
    <!-- The directory under which all other KeyStore
files will be stored-->
    <KeyStoresDir>${carbon.home}/conf/keystores<
/KeyStoresDir>
    <KeyManagerType>SunX509</KeyManagerType>
    <TrustManagerType>SunX509</TrustManagerType>
</Security>
```

The following elements were added under the <OAuth> element:

```
<OIDCCheckSessionEPUrl>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/oidc/checksession<
/OIDCCheckSessionEPUrl>
<OIDCLogoutEPUrl>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/oidc/logout<
/OIDCLogoutEPUrl>
<OIDCConsentPage>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/authenticationendpoint
/oauth2_consent.do</OIDCConsentPage>
<OIDCLogoutConsentPage>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/authenticationendpoint
/oauth2_logout_consent.do</OIDCLogoutConsentPage>
<OIDCLogoutPage>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/authenticationendpoint
/oauth2_logout.do</OIDCLogoutPage>
```

```
<EnableOAuthCache>false</EnableOAuthCache>
```

Caching Recommendation

It is recommended to keep the OAuth2 local cache and the distributed cache disabled as it may cause out-of-memory issues.
However, if you want to enable the OAuth2 local cache, you have to enable the distributed cache as well.

To enable the OAuth2 local cache and distributed cache, set the `<EnableOAuthCache>` property and `isDistributed` to true.

```
<EnableOAuthCache>true</EnableOAuthCache>
<Cache name="OAuthCache" enable="true" timeout="1"
capacity="5000" isDistributed="true"/>
```

The following elements were removed from the <OAuth><OpenIDConnect> element:

```
<IDTokenSubjectClaim>http://wso2.org/claims/givenname<
/IDTokenSubjectClaim>
<UserInfoEndpointClaimDialect>http://wso2.org/claims<
/UserInfoEndpointClaimDialect>
```

The following code was updated. To add audiences to the JWT token, use the code block below. For more information, see JWT Token Generation.

```
<OpenIDConnect>
    <IDTokenBuilder>org.wso2.carbon.identity.
openidconnect.DefaultIDTokenBuilder</IDTokenBuilder>
    <!-- Comment out to add Audience values to the
JWT token (id_token)-->
    <!--Audiences>
        <Audience>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/oauth2/token<
/Audience>
    </Audiences-->
    <!--Default value for IDTokenIssuerID, is
OAuth2TokenEPUrl.If that doesn't satisfy uncomment
the following config and explicitly configure the
value-->
    <IDTokenIssuerID>${carbon.protocol}://${carbon.
host}:${carbon.management.port}/oauth2/token<
/IDTokenIssuerID>
```

```
  ...

</OpenIDConnect>
```

The <CacheConfig> was replaced:

```
<CacheConfig>
    <CacheManager name="
IdentityApplicationManagementCacheManager">
        <Cache name="
AppAuthFrameworkSessionContextCache" enable="false"
timeout="1" capacity="5000" isDistributed="false" />
        <Cache name="AuthenticationContextCache"
enable="false" timeout="1" capacity="5000"
isDistributed="false" />
        <Cache name="AuthenticationRequestCache"
enable="false" timeout="1" capacity="5000"
isDistributed="false" />
        <Cache name="AuthenticationResultCache"
enable="false" timeout="1" capacity="5000"
isDistributed="false" />
        <Cache name="AppInfoCache" enable="true"
timeout="1" capacity="5000" isDistributed="false" />
        <Cache name="AuthorizationGrantCache" enable="
false" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="OAuthCache" enable="false"
timeout="1" capacity="5000" isDistributed="false" />
        <Cache name="OAuthSessionDataCache" enable="
false" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="SAMLSSOParticipantCache" enable="
false" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="SAMLSSOSessionIndexCache"
enable="false" timeout="1" capacity="5000"
isDistributed="false" />
        <Cache name="SAMLSSOSessionDataCache" enable="
false" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="ServiceProviderCache" enable="
true" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="ProvisioningConnectorCache"
enable="true" timeout="1" capacity="5000"
isDistributed="false" />
        <Cache name="ProvisioningEntityCache" enable="
false" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="
ServiceProviderProvisioningConnectorCache" enable="
true" timeout="1" capacity="5000" isDistributed="
```

```
false" />
        <Cache name="IdPCacheByAuthProperty" enable="
true" timeout="1" capacity="5000" isDistributed="
false" />
        <Cache name="IdPCacheByHRI" enable="true"
timeout="1" capacity="5000" isDistributed="false" />
        <Cache name="IdPCacheByName" enable="true"
timeout="1" capacity="5000" isDistributed="false" />
    </CacheManager>
</CacheConfig>
```

- `context.xml` file stored in the `<PRODUCT_HOME>/repository/conf/tomcat/carbon/META-INF/` directory.
- `context.xml` file stored in the `<PRODUCT_HOME>/repository/conf/tomcat/` directory.
- `web.xml file` stored in the `<PRODUCT_HOME>/repository/conf/tomcat/carbon/WEB-INF/` directory.

The entire file was replaced.

`carbon.xml` file stored in the `<PRODUCT_HOME>/repository/conf/` directory.

The following elements were added under the <Security> tag:

```
<STSCallBackHandlerName>org.wso2.carbon.identity.
provider.AttributeCallbackHandler<
/STSCallBackHandlerName>

<XSSPreventionConfig>
    <Enabled>true</Enabled>
    <Rule>allow</Rule>
    <Patterns>
        <!--Pattern></Pattern-->
    </Patterns>
</XSSPreventionConfig>
```

	The following elements were removed: `<!--Configurations to avoid Cross Site Request Forgery vulnerabilities--> <CSRFPreventionConfig> <!--CSRFPreventionFilter configurations that adopts Synchronizer Token Pattern--> <CSRFPreventionFilter> <!-- Set below to true to enable the CSRFPreventionFilter--> <Enabled>false</Enabled> <!--Url Pattern to skip application of CSRF protection--> <SkipUrlPattern > (.*)(/images|/css | /js|/docs)(.*) </SkipUrlPattern> </CSRFPreventionFilter> </CSRFPreventionConfig> <!-- Configuration to enable or disable CR and LF sanitization filter--> <CRLFPreventionConfig> <!--Set below to true to enable the CRLFPreventionFilter--> <Enabled>true</Enabled> </CRLFPreventionConfig`
`claim-config.xml` file stored in the `<PRODUCT_HOME>/repository/conf/` directory.	The following claims were added. For more information on configuring these, see Configuring Users or User Account Locking and Account Disabling depending on the claim you want to configure. `<Claim> <ClaimURI>http://wso2.org/claims/identity/lastLoginTime</ClaimURI> <DisplayName>Last Login</DisplayName> <!-- Proper attribute Id in your user store must be configured for this --> <AttributeID>carLicense</AttributeID> <Description>Last Login Time</Description> </Claim> <Claim> <ClaimURI>http://wso2.org/claims/identity/lastPasswordUpdateTime</ClaimURI> <DisplayName>Last Password Update</DisplayName> <!-- Proper attribute Id in your user store must be configured for this --> <AttributeID>businessCategory</AttributeID> <Description>Last Password Update Time</Description> </Claim> <Claim> <ClaimURI>http://wso2.org/claims/identity/accountDisabled</ClaimURI> <DisplayName>Account Disabled</DisplayName>`

```
    <!-- Proper attribute Id in your user store must be
configured for this -->
    <AttributeID>ref</AttributeID>
    <Description>Account Disabled</Description>
</Claim>
```

- `data-agent-config.xml` file stored in the `<PRODUCT_HOME> /repository /conf/data-bridge/` directory.
- `event-processor.xml` file stored in the `<PRODUCT_HOME> /repository /conf/` directory.

The file was newly added.

`metrics-datasources.xml` file stored in the `<PRODUCT_HOME> /repository /conf /datasources/` directory.

Set the `<defaultAutocommit>` property to true.

```
 <datasource>
            <name>WSO2_METRICS_DB</name>
            <description>The default datasource used
for WSO2 Carbon Metrics</description>
            <jndiConfig>
                <name>jdbc/WSO2MetricsDB</name>
            </jndiConfig>
            <definition type="RDBMS">
                <configuration>        <url>jdbc:h2:
repository/database/WSO2METRICS_DB;
DB_CLOSE_ON_EXIT=FALSE;AUTO_SERVER=TRUE</url>
                    <username>wso2carbon</username>
                    <password>wso2carbon</password>
                    <driverClassName>org.h2.Driver<
/driverClassName>
                    <maxActive>50</maxActive>
                    <maxWait>60000</maxWait>
                    <testOnBorrow>true</testOnBorrow>
                    <validationQuery>SELECT 1<
/validationQuery>
                    <validationInterval>30000<
/validationInterval>
                    <defaultAutoCommit>true<
/defaultAutoCommit>
                </configuration>
```

	<pre></definition>
	</datasource></pre>

application-authentication.xml file stored in the <PRODUCT_HOME>/repository/conf/identity/ directory.	<pre><AuthenticatorConfig name="EmailOTP" enabled="true">
	<Parameter name="GmailClientId"
	>gmailClientIdValue</Parameter>
	<Parameter name="GmailClientSecret"
	>gmailClientSecretValue</Parameter>
	<Parameter name="SendgridAPIKey"
	>sendgridAPIKeyValue</Parameter>
	<Parameter name="GmailRefreshToken"
	>gmailRefreshTokenValue</Parameter>
	<Parameter name="GmailEmailEndpoint">https://www.
	googleapis.com/gmail/v1/users/[userId]/messages/send<
	/Parameter>
	<Parameter name="SendgridEmailEndpoint"
	>https://api.sendgrid.com/api/mail.send.json<
	/Parameter>
	<Parameter name="accessTokenRequiredAPIs">Gmail<
	/Parameter>
	<Parameter name="apiKeyHeaderRequiredAPIs"
	>Sendgrid</Parameter>
	<Parameter name="SendgridFormData"
	>sendgridFormDataValue</Parameter>
	<Parameter name="SendgridURLParams"
	>sendgridURLParamsValue</Parameter>
	<Parameter name="GmailAuthTokenType">Bearer<
	/Parameter>
	<Parameter name="GmailTokenEndpoint">https://www.
	googleapis.com/oauth2/v3/token</Parameter>
	<Parameter name="SendgridAuthTokenType">Bearer<
	/Parameter>
	</AuthenticatorConfig>
	
	<AuthenticatorConfig name="
	x509CertificateAuthenticator" enabled="true">
	<Parameter name="AuthenticationEndpoint"
	>https://localhost:8443/x509-certificate-servlet<
	/Parameter>
	</AuthenticatorConfig>
	
	<AuthenticatorConfig name="totp" enabled="true">
	<Parameter name="encodingMethod">Base32<
	/Parameter>
	<Parameter name="timeStepSize">30</Parameter>
	<Parameter name="windowSize">3</Parameter>
	<Parameter name="enableTOTP">false</Parameter>
	</AuthenticatorConfig></pre>

metrics.xml file	The following elements were added:

stored in the <PRODUCT_HOME>/repository/conf/ directory.	<Metrics xmlns="http://wso2.org/projects/carbon/metrics.xml">
	  <Reporting>
	  <Console>
	  <Enabled>false</Enabled>
	  <!-- Polling Period in seconds.
	  This is the period for polling metrics from the metric registry and
	  printing in the console -->
	  <PollingPeriod>60</PollingPeriod>
	  </Console>
	
	  <DAS>
	  <Enabled>false</Enabled>
	  <!-- Source of Metrics, which will be used to
	  identify each metric sent in the streams -->
	  <!-- Commented to use the hostname
	  <Source>Carbon</Source>
	  -->
	  <!-- Polling Period in seconds.
	  This is the period for polling metrics from the metric registry and
	  sending events via the Data Publisher -->
	  <PollingPeriod>60</PollingPeriod>
	  <!-- The type used with Data Publisher -->
	  <Type>thrift</Type>
	  <!-- Data Receiver URL used by the Data Publisher -->
	  <ReceiverURL>tcp://localhost:7611</ReceiverURL>
	  <!-- Authentication URL for the Data Publisher -->
	  <!-- <AuthURL>ssl://localhost:7711</AuthURL> -->
	  <Username>admin</Username>
	  <Password>admin</Password>
	  <!-- Path for Data Agent Configuration -->
	  <DataAgentConfigPath>repository/conf/data-bridge/data-agent-config.xml</DataAgentConfigPath>
	  </DAS>

output-event-adapters.xml file stored in the <PRODUCT_HOME>/repository/conf/ directory.	The following adapter configurations were added: <adapterConfig type="http"> <!-- Thread Pool Related Properties --> <property key="minThread">8</property> <property key="maxThread">100</property> <property key="keepAliveTimeInMillis">20000<

```
/property>
    <property key="jobQueueSize">10000</property>
    <!-- HTTP Client Pool Related Properties -->
    <property key="defaultMaxConnectionsPerHost">50<
/property>
    <property key="maxTotalConnections">1000<
/property>
</adapterConfig>

<adapterConfig type="jms">
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
</adapterConfig>

<adapterConfig type="mqtt">
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
    <property key="connectionKeepAliveInterval">60<
/property>
</adapterConfig>

<adapterConfig type="kafka">
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
</adapterConfig>

<adapterConfig type="email">
    <!-- Comment mail.smtp.user and mail.smtp.
password properties to support connecting SMTP
servers which use trust
        based authentication rather username/password
authentication -->
    <property key="mail.smtp.from">abcd@gmail.com<
/property>
    <property key="mail.smtp.user">abcd</property>
    <property key="mail.smtp.password">xxxx</property>
    <property key="mail.smtp.host">smtp.gmail.com<
/property>
    <property key="mail.smtp.port">587</property>
    <property key="mail.smtp.starttls.enable">true<
/property>
```

```
    <property key="mail.smtp.auth">true</property>
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
</adapterConfig>

<adapterConfig type="ui">
    <property key="eventQueueSize">30</property>
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
</adapterConfig>

<adapterConfig type="websocket-local">
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
</adapterConfig>

<adapterConfig type="websocket">
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
</adapterConfig>

<adapterConfig type="soap">
    <!-- Thread Pool Related Properties -->
    <property key="minThread">8</property>
    <property key="maxThread">100</property>
    <property key="keepAliveTimeInMillis">20000<
/property>
    <property key="jobQueueSize">10000</property>
    <!-- Axis2 Client Connection Related Properties --
>
    <property key="axis2ClientConnectionTimeout"
>10000</property>
    <property key="reuseHTTPClient">true</property>
    <property key="autoReleaseConnection">true<
/property>
    <property key="maxConnectionsPerHost">50<
/property>
```

```
</adapterConfig>
```

`registry.xml` file stored in the `<PRODUCT_HOME>/repository/conf/` directory.

The following elements were added:

```
<indexingConfiguration>
    <startIndexing>false</startIndexing>
    <startingDelayInSeconds>35<
/startingDelayInSeconds>
    <indexingFrequencyInSeconds>5<
/indexingFrequencyInSeconds>
    <!--number of resources submit for given indexing
thread -->
    <batchSize>40</batchSize>
    <!--number of worker threads for indexing -->
    <indexerPoolSize>40</indexerPoolSize>
    <!-- location storing the time the indexing took
place-->
    <lastAccessTimeLocation>/_system/local/repository
/components/org.wso2.carbon.registry/indexing
/lastaccesstime</lastAccessTimeLocation>
    <!-- the indexers that implement the indexer
interface for a relevant media type/(s) -->
    <indexers>
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.MSExcelIndexer" mediaTypeRegEx="
application/vnd.ms-excel" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.MSPowerpointIndexer" mediaTypeRegEx="
application/vnd.ms-powerpoint" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.MSWordIndexer" mediaTypeRegEx="
application/msword" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.PDFIndexer" mediaTypeRegEx="
application/pdf" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.XMLIndexer" mediaTypeRegEx="
application/xml" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.XMLIndexer" mediaTypeRegEx="
application/(.)+\+xml" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.PlainTextIndexer" mediaTypeRegEx="
application/swagger\+json" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.PlainTextIndexer" mediaTypeRegEx="
application/(.)+\+json" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.PlainTextIndexer" mediaTypeRegEx="
text/(.)+" />
        <indexer class="org.wso2.carbon.registry.
indexing.indexer.PlainTextIndexer" mediaTypeRegEx="
```

```
application/x-javascript" />
    </indexers>
    <exclusions>
        <exclusion pathRegEx="/_system/config
/repository/dashboards/gadgets/swfobject1-5/.*[.]
html" />
        <exclusion pathRegEx="/_system/local
/repository/components/org[.]wso2[.]carbon[.]registry
/mount/.*" />
    </exclusions>
</indexingConfiguration>
```

user-mgt.xml file stored in the <PRODUCT_HOME>/repository/conf/ directory.	The following LDAP/AD property was added: `<Property name="AnonymousBind">false</Property>`