**Set Up Sailpoint :**

1.JDK 1.8 (preferable)

2.database server (Mysql 5.7 preferable)

3. the application server(Tomcat 8 preferable)

4. then keep identity.war file in webapps of tomcat to deploy.

5.Execute database script "identityiq\WEB-INF\database 6.create\_identityiq\_tables-7.1.mysql".

7. If you use other database or change default war file name, update accordingly in identityiq\WEB-INF\classes\iiq.properties file

8. Go to identityiq>web-inf>bin and enter iiq console and import init.xml

**Sailpoint ( IdentityIQ Achitecture)**

User – firewall – load Balancer – UI Server/ Task Server – database server

**What is UI server?**

The UI interface is often talked about in conjunction with user experience (UX), which may include the aesthetic appearance of the device, response time and the content that is presented to the user within the context of the user interface

**What is Task Server?**

The Task server run just behind the UI server where all the functionality is done to present in UI server.

**IAM**-identity access mgmt.

**PAM**-Privilege access mgmt. (managers, shareholders, admins etc.)

**App server**: UI server and task server

**Beanshell:** Scripting language in Java

**Find Objects and Methods Help:** identityiq/doc/javadoc

**Authoritative sources:** create identity first then non authoritative are

used to correlate and generally not used to create an identity

**Connector:** to read and extract their data and to process provisioning requests.

**Custom Connector:** If there is no connector built-in, then custom connector should create. Writing a custom connector using the openconnector framework is a good option.

Connector design involves identifying the features the connector will support, the object types it will read or provision, and the object schemas.

**Object Type in Connector:** Account and group. Though new object classes can be defined if required.

**Rules:** Constructs through which logics are implemented

 Through UI

 Through XML Files

Same rule once made can be used multiple times

**Log4j.properties:** The ability to adjust logging for the rule by itself can be very helpful in the debugging process.

log4j.logger. [uniquename].[ruleName]=[loglevel] // info or debug

import org.apache.log4j.Logger; Logger custLog = Logger.getLogger(" [uniquename].[rulename]");

**BuildMap Rule:** A BuildMap rule applies only to applications of type DelimitedFile. It is run for each row of data as it is read in from a connector and buildmap rule is run before it is aggregation. It always return the value map. Map is like dictionary: Key Value pair

**JDBCBuildMap:** A JDBCBuildMap rule applies only to applications of type JDBC. It functions for JDBC applications just like the BuildMap rule does for Delimited File applications. It also return the value map.

**Object names in Sailpoint: (use in import for rules)**

Workflow -> Workflow

Entitlements -> managedAttribute

Application -> Application

Plan -> ProvisioningPlan

Role -> Bundle

AccountRequest -> AccountRequest

AttributeRequest -> AttributeRequest

Context -> SailPointContext

Identity -> Identity

**Creation Rule:** If the correlation rule cannot find an Identity that corresponds to the account, one must be created. It creates the new identity object and this rule is run during the aggregation task.

**Correlation Rule:** A Correlation Rule is used to select the existing Identity to which the aggregated account information should be connected. If the identity attribute doesn’t match with application attribute, then the new identity is formed. And it is run during the task aggregation.

**Customization Rule:** Itruns prior to any other aggregation rule to customize the resource object provided by the connector before aggregation begins.

**Active Directory**

1. The connector mainly uses the LDAP, ADSI and PowerShell interfaces to communicate with end system.
2. This connector manages multiple domains or forests for users, contacts, group memberships, foreign security principals (FSP), terminal services, and Dial-in Attributes.

DC (Domain Controller) -&gt; eg (dexpertrain.com)

OU (Organization Unit ): Indeed, TribleByte

Forest >Multiple Nodes eg: (Pratap.dexpertrain.com (domains and

subdomains); one root and other directories

There can be multiple forests with different names

Groups: Entitlements (Permissions); usually these are set by the LDAP Administrator like which group will have what kind of accesses

SAMA: Each user will have one unique SAMA

CN(Common Name)

DN ( Distiguished Name): CN+OU+DC

1. For JDBC you need to write your own provisioning policy

2. You can directly provision from AD

**For provisioning in AD**

IQservice.zip > used for provisioning (You need to run iqserver in the remote server)

iqservice -i (Install)

iqservice –s (Start iqservice)

port 5050 (Dynamic for different computer)

And, it is provision through manage user access where LCM provision workflow will run.

**Group Aggregation:** It is needed when it doesn’t have account but have the groups.

**Roles**

1. Organization Role: Container

2. Business Role: Job Function eg Payroll Admin (Printing, Files, Access); can be assigned

3. IT Role: set of entitlements ; Access of applications; cant be assigned(detected roles)

4. Entitlements

**RBAC:** Giving the access control based on Role provided to the user within the organization.

It can be implemented by using the role editing and modeling, role mining, entitlement analysis, certifications for role membership and role composition, and workflows for governing changes.

There is required and permitted access where required is automatically given where the permitted is optional.

**LCM Provisioning**

 Default workflow

 Provision and deprovision the Roles and Entitlement

 Unlock, Enable, Delete Accounts and update identity --Used to write something on the target system

**Workflow of LCM Provisioning**

1. Initialize: setup identity

2. Approve:

3. Provision

4. Notify

5. Finalize

**Policy Violation:** It is the set of rules. It can be run as detective also as preventive.

**Activity Policy:** It is based on specific events or for certain time frame.

Advance policy: We can customize policy according to the need of organization.

Policies:

 Evaluated per identity

 Detective way during identity refresh

(Check active policy)

 Preventive way during assigning access rights

If violation is found a PolicyViolation object is generated ; email can be sent and workitem object is created..

For Preventive policy violation we need to edit the Policy scheme in LCM workflow

 We can continue; or fail; or prevent???

 Use fail to stop

**Birthright Provisioning**

 Default accesses when someone joins the company

 Are given certain entitlements once the account is created

 Can be done by giving a Role(LCM) or Joiner Lifecycle Event(Custom Workflow)

**Birthright Provisioning**

**Eg.**

* Create IT Role and under role give the entitlement from the active directory and create business role.
* Under the business role, give the required and permitted role, then choose the assignment from any dropdown box.
* In base of attribute, choose any like matchlist and select the attribute in base eg. Jobcode.
* Run the application aggregation.
* Run the task containing like Refresh Identity task, detected roles and promote additional entitlements and Provision assignments.
* Then complete the form in workitem.

**Lifecycle Events:**

 Joiners

 Movers (Changes the department)

 Leavers

**Manual Work Item:** When there is read only connector then manual work item is assigned to make the changes in the delimited file and aggregate.

**Certification**

Advanced Analytics

 Can save as population and then certify

Population: dynamic set of identities; eg: all from AD (is diff everyime?)

Workgroup: set of identities with name; fixed; static eg: A, B,C

 Basic, lifecycle, notification, behavior, advanced

 Lifecycle

Revocation/: Auto in write connector

Remediation: in read only connector, need to delete manually; get work item; and then submit

Redelegate: first team lead will verify then manager

Reassignment—Assign to assistant and he can signoff

Delegation—assign to asst. he will approve/revoke but sign off comes back

Forwarding—Someone else and he will certify

**SSD**

* Unzip SSD.Zip file

SSD:

1. Extract the zip file of ssd and configure it
2. To configure: OPEN CMD
3. “deployssd”
4. Hit Enter
5. Is the detected path correct? [y|n] y
6. No SPTARGET variable detected.Would you like to create property files for additional targets? [y|n] y

SP Target: These are target define values.

Do Yes for all

1. Open build.properties and give your new version here
2. Sandbox.iiq.properties and copy your iiq.properties
3. Give your laptop name in server.properties
4. Go to base -> GA -> and copy the zip file for the version of sailpoint.
5. Go to CMD:

* Build clean
* Build war

1. Go to properties in My-PC, create environment variable SPTARGET = sandbox, if failure

Now deploy the war file inside the build folder.

1. // Copy ssd/ build/identityiq\_x.war to webapps/identity\_x.war
2. After deploying, make the db.
3. Then, open cmd for iiq, ,and run iiq console.

* Import init.xml
* Import sp.init-custom.xml