|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | Client Name | |  | | IdentityIQ  High Level Architecture | |  | | Manjunath Madiraju  September 2018 Version: 1 | |  | |

**Table of Contents**

[Document Control 3](#_Toc38881238)

[1 Solution Overview 4](#_Toc38881239)

[1.1 Executive Summary 4](#_Toc38881240)

[1.2 Project Information 4](#_Toc38881241)

[1.2.1 IdentityIQ Product Architecture 4](#_Toc38881242)

[1.2.2 Platform and Version Selection 5](#_Toc38881243)

[1.3 Scope Definition 5](#_Toc38881244)

[1.3.1 In Scope 6](#_Toc38881245)

[1.3.2 Out of Scope 6](#_Toc38881246)

[1.4 Design Constraints and Considerations 6](#_Toc38881247)

[1.4.1 Availability of Production Data 6](#_Toc38881248)

[2 Solution Architecture and Design 7](#_Toc38881249)

[2.1 Logical Architecture 7](#_Toc38881250)

[2.1.1 Authoritative Source 7](#_Toc38881251)

[2.1.2 Identity Seeding and Synchronization 7](#_Toc38881252)

[2.1.3 Self-Service and Administrator Interface 8](#_Toc38881253)

[2.2 Physical Architecture 9](#_Toc38881254)

[2.2.1 Development Environment 11](#_Toc38881255)

[2.2.2 Pre-Prod QA Environment 12](#_Toc38881256)

[2.2.3 Pre-Prod QA Warm Environment 12](#_Toc38881257)

[2.2.4 Production Environment 13](#_Toc38881258)

[2.2.5 Production Warm Environment 13](#_Toc38881259)

[2.3 High Availability/Disaster Recovery 14](#_Toc38881260)

[2.3.1 Separation of User Interface and Task Servers 14](#_Toc38881261)

[2.3.2 Load Balancing Configuration 14](#_Toc38881262)

[2.3.3 Disaster Recovery 15](#_Toc38881263)

[2.4 User Populations 15](#_Toc38881264)

[2.5 Identity Attributes 15](#_Toc38881265)

[2.6 Product Security 15](#_Toc38881266)

[2.6.1 Authentication 15](#_Toc38881267)

[2.6.2 Encryption 16](#_Toc38881268)

[2.6.3 User Rights 17](#_Toc38881269)

[2.7 Applications and Entitlements 19](#_Toc38881270)

[2.7.1 Data Warehouse 20](#_Toc38881271)

[2.7.2 Active Directory 2012 20](#_Toc38881272)

[2.8 Implementation Approach 20](#_Toc38881273)

[3 Workflows 21](#_Toc38881274)

[3.1 Joiner (new hire) 21](#_Toc38881275)

[4 Appendix: Abbreviations and Glossary 22](#_Toc38881276)

[4.1 Abbreviations 22](#_Toc38881277)

[4.2 Glossary 22](#_Toc38881278)

# Document Control

This is a version-controlled document. The control and release of this document is the sole responsibility of the document owner.

|  |  |  |  |
| --- | --- | --- | --- |
| **Version control** | | | |
| Version | 1.0 | Date | 04/25/2020 |
| Classification | Confidential | Author | Manjunath Madiraju |
| Document Title | IdentityIQ High Level Architecture | | |
| Approved by |  | | |
| Released by |  | | |
|  | | | |
| **Owner details** | | | |
| Name | Manjunath Madiraju | | |
| Office |  | | |
| Contact number | 925.699.9716 | | |
| Email Address | mmadiraju@kpmg.com | | |
|  | | | |
| **Revision history** | | | |
| **Issue** | **Date** | **Author** | |
| Draft 1.0 | 04/25/2020 | Manjunath Madiraju | |
|  |  |  | |

# Solution Overview

## Executive Summary

Client Name plans to acquire the SailPoint IdentityIQ to facilitate the access governance, compliance, auditing and provisioning for internal critical business applications. The initial implementation will comprise of the IdentityIQ governance manager and lifecycle manager. The primary goal for this implementation is to reduce the level of effort involved in making a new employee productive.

The Technical Architecture described herein is informed by the requirements identified within the Requirements Tractability Matrix, and is intended to provide a framework for deploying IdentityIQ in a manner that will support both the initial and subsequent phases as identified by Client Name.

## Project Information

### IdentityIQ Product Architecture

IdentityIQ is implemented as a standard 3-tier Java MVC application, comprising a presentation tier, application tier, and data tier. Presentation is handled by the Java Server Faces (JSF) framework. Extensive use is also made of other standard Java frameworks such as Hibernate, JAX-RPC, Jasper, and Log4j. IdentityIQ also exposes a web service interface, implemented as REST/JSON rather than SOAP/XML. Business rules are written in Javascript and processed by an embedded Beanshell engine.

Product configurations, customizations, and data structures, for example, identities, attributes, workflows, audit items, rules and certifications, are stored and persisted as serialized XML objects in a backend database repository. Since IdentityIQ is a highly transactional application, it is critical for the application server and database server to be located in close proximity to minimize latency.

|  |
| --- |
| *Figure 1: IdentityIQ product architecture* |

### Platform and Version Selection

At the time of writing, the current version of IdentityIQ is 7.2 Patch 2. This will be the version used for implementation.

IdentityIQ has demonstrated superior performance and reliability on the Tomcat platform, which is the recommended application server for this implementation. Client has expressed a preference to deploy IdentityIQ on Windows 2012 R2 Standard servers. This configuration favors the use of Microsoft SQL Server 2014 or 2016 for the IdentityIQ repository.

|  |  |
| --- | --- |
| Product Version | SailPoint IdentityIQ 7.2 (Patch 2) |
| Database Repository | Microsoft SQL Server 2014 Datacenter |
| Application Server | Tomcat 8.5 / JDK 8 |
| Operating System(s) | Microsoft Windows Server 2012 R2 Standard |
| Target Resource Platforms | * Microsoft Active Directory 2008 |

Table 1: Version

## Scope Definition

While the architecture specification described in this document is intended to provide optimal scalability for Client Name’s future IAM/IAG requirements, the initial items within the scope enumerated below are solely concerned with the initial phase.

These items are described in greater detail within the [Identity Lifecycle Processes](http://essdocs.ccf.org/display/ADT/Identity+Lifecycle+Processes), which is to be provided as part of this engagement.

### In Scope

* TBD

### Out of Scope

* TBD

## Design Constraints and Considerations

**Connectors:**

**Authoritative Source:**

* All the required user (identity) information is available from authoritative source names?

### Availability of Production Data

TBD.

# Solution Architecture and Design

## Logical Architecture



|  |
| --- |
| *Figure 2: Logical architecture* |

### Authoritative Source

TBD

### Identity Seeding and Synchronization

TBD

#### Ongoing Aggregation and Refresh

On an ongoing basis, aggregation will be performed against all application endpoints, followed by a full identity refresh. This sequence of tasks will be scheduled to run on a nightly basis during off-shift hours.

#### Naming Convention for Identity Cubes

Naming conventions and coding standards in the attached document will be followed.



### Self-Service and Administrator Interface

IdentityIQ runs exclusively within a web browser and does not require the installation of any client-side components.

#### Interface Branding

It is anticipated that the user interface will be customized with Client Name-specific stylesheets and Logo.

#### User Capabilities

IdentityIQ provides the ability to restrict functionality based on the defined user capabilities. For example, a System Administrator will have access to a much broader range of functionality than an ordinary manager, while a manager will have capabilities that an ordinary end user does not. In IdentityIQ, these capabilities are known as “user rights”.

It is anticipated that the Client Name deployment will make extensive use of user rights in delivering user functionality. However, it is not yet determined which specific user rights will be required and by whom.

#### Web Browser Security

IdentityIQ supports both SSL and non-SSL configurations. It is determined that SSL is a requirement for Client Name in Dev, Pre-Prod and Prod. SSL will be configured, and it will require installation of a trusted SSL certificate on the Tomcat application server.

* + - Get the .pfx file to be installed on your server and save it to the tomcat server location F:\SailPoint\tomcat\conf
    - Update the SSL section of the server.xml configuration file with the following information:

<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443 for CCF IIQ-->

<Connector port="8443" maxHttpHeaderSize="8192" maxThreads="150"

minSpareThreads="25" maxSpareThreads="75" enableLookups="false"

disableUploadTimeout="true" acceptCount="100" scheme="https"

secure="true" SSLEnabled="true" clientAuth="false"

sslProtocol="TLS" keystoreFile="conf/samplecert.pfx"

keystorePass="password" keystoreType="PKCS12"/>

* + - Restart Tomcat for your new certificate to take effect.

## Physical Architecture

The following figures illustrate the proposed physical architecture for Client Name’s implementation of IdentityIQ.

|  |
| --- |
| *Figure 3a: IdentityIQ physical architecture: production / prod-warm* |
| Figure b: IdentityIQ physical architecture: Preprod / Preprod-Warm |

|  |
| --- |
| *Figure 4: IdentityIQ physical architecture: Dev* |

### Development Environment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Server Name | Server Purpose | OS | Software/Versions | Hardware Specification |
|  | IdentityIQ Task Server | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ UI Server | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | I**QServic**e Server | Windows Server 2012 R2 Standard (64-bit) | IQService 7.2p2  (Version 4.5 of Microsoft .NET) | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Database Server | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 | 128Gb RAM, 1646Gb HDD |

*Table 3: Development environment*

### Pre-Prod QA Environment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Server Name | Server Purpose | OS | Software/Versions | Hardware Specification |
|  | IdentityIQ Task Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Task Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ UI Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IdentityIQ UI Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
| 51 | IQService Server | Windows Server 2012 R2 Standard (64-bit) | IQService 7.2p2  (Version 4.5 of Microsoft .NET) | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Database Server | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 | 128Gb RAM, 1646Gb HDD |

*Table 4: Test/QA environment*

### Pre-Prod QA Warm Environment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Server Name | Server Purpose | OS | Software/Versions | Hardware Specification |
|  | IdentityIQ Task Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Task Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ UI Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IdentityIQ UI Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IQService Server | Windows Server 2012 R2 Standard (64-bit) | IQService 7.2p2  (Version 4.5 of Microsoft .NET) | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Database Server | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 | 128Gb RAM, 1646Gb HDD |

### Production Environment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Server Name | Server Purpose | OS | Software/Versions | Hardware Specification |
|  | IdentityIQ Task Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Task Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ UI Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IdentityIQ UI Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IQService Server | Windows Server 2012 R2 Standard (64-bit) | IQService 7.2p2  (Version 4.5 of Microsoft .NET) | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Database Server#1 | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 or 2016 | 16 Core, 384Gb RAM, 3562Gb HDD |
|  | IdentityIQ Database Server#2 | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 or 2016 | 16 Core, 384Gb RAM, 20Gb HDD |

### Production Warm Environment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Server Name | Server Purpose | OS | Software/Versions | Hardware Specification |
|  | IdentityIQ Task Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Task Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ UI Server#1 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IdentityIQ UI Server#2 | Windows Server 2012 R2 Standard (64-bit) | IdentityIQ 7.2p2  Tomcat 8.5/ JDK 8 | 4 Core, 8Gb RAM, 40Gb HDD |
|  | IQService Server | Windows Server 2012 R2 Standard (64-bit) | IQService 7.2p2  (Version 4.5 of Microsoft .NET) | 4 Core, 16Gb RAM, 50Gb HDD |
|  | IdentityIQ Database Server#1 | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 or 2016 | 16 Core, 384Gb RAM, 3562Gb HDD |
|  | IdentityIQ Database Server#2 | Windows Server 2012 Standard (64-bit) | MS SQL Server 2014 or 2016 | 16 Core, 384Gb RAM, 20Gb HDD |

*Table 5: Production environment*

**Note:** Combined dedicated SQL cluster for both IdentityIQ and SecurityIQ. Requires MS SQL 2014 or 2016; SQL Enterprise for D/R. 1120 GB for IdentityIQ; 2102 GB for SecurityIQ, 200 GB TempDB formatted with NTFS to a 64K allocation unit, 100 GB logs; hyper-threading disabled.

## High Availability/Disaster Recovery

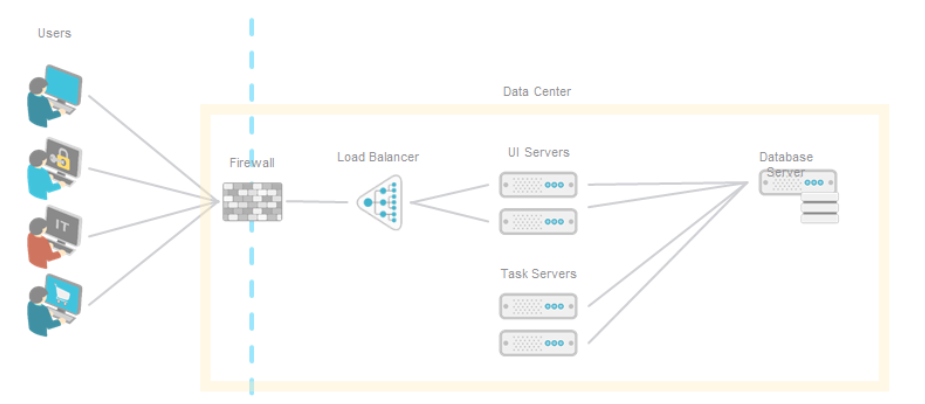
TBD.

### Separation of User Interface and Task Servers

A best practice for deployment of IdentityIQ is to use dedicated hardware for resource-intensive tasks such as aggregation and identity refresh, which frees up more resources for end user functionality. This configuration requires that only those application servers dedicated to UI functionality are exposed to users. However, task servers still contain a fully functional instance of IdentityIQ and may still be used as backup user interface servers in the event of an emergency.

### Load Balancing Configuration

In Dev, preprod and production environments, load balancer should be configured to load balance traffic between the two user interface servers. It is critical that in such a configuration, session affinity (or “sticky”) should be enabled on the load balancer. Client Name is responsible for this configuration.



### Disaster Recovery

A snapshot of each Production server is taken once a day between 12:00 am and 08:00 am. The last snapshot of the day is mirrored to the Production Warm server for DR purposes. In a DR scenario, this snapshot would be mounted on the secondary site infrastructure, and the DNS alias of the ?? device would be transferred to secondary site.

## User Populations

The following table illustrates the target population(s) that are in scope for this project. For example, users may be categorized by Employee type and Non-Employee

|  |  |  |
| --- | --- | --- |
| Population | Approx. Count | Description |
| Full Time Employees |  |  |
| Non-Employees |  |  |

*Table 6: User populations*

## Identity Attributes

IdentityIQ provides the ability to create up to 20 custom identity attributes to be stored on an identity cube. Identity attributes can be marked as searchable, and can also be used to auto-generate groups of users.

IdentityIQ ships with five preconfigured identity attributes. These are firstname**,** lastname**,** email**,** manager, and inactive.

## Product Security

### Authentication

SSO or AD or similar platform.

### Encryption

In a standard installation of IdentityIQ, passwords are all encrypted using the same encryption secret. In other words, encrypted passwords used in one installation can be reused in (i.e. decrypted by) any other installation of IdentityIQ. The keystore feature enables the use of a site specific key. With the keystore feature enabled, a password used on one site cannot be decrypted on another site without having the site specific encryption keys.

<https://community.sailpoint.com/docs/DOC-2031>

1. **Key Creation**

In order to create or manage the keystore, start the IdentityIQ command line tool with the 'keystore' command, from the WEB-INF/bin folder:

On Windows:

C:> iiq.bat keystore

To create a key, use the addKey command:

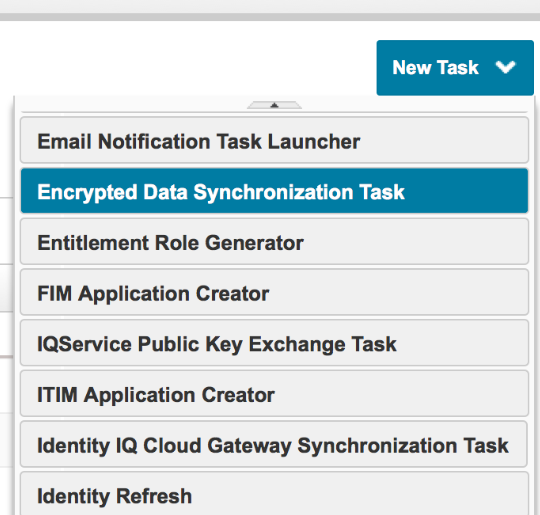
1. > addKey
2. Generate a new encryption key (y/n)?
3. y
4. Generating a new encryption key for keystore [/var/tomcat/webapps/identityiq/WEB-INF/classes/iiq.dat].
5. New encrpytion key successfully saved to keystore.
6. All application servers must be restarted for changes to take effect.
7. >

The tool asks you whether you'd like to generate a new encryption key. Enter 'y' and press enter to continue. If the keystore file does not exist, it will be created and a new, randomly generated key will be added.

Now restart your application server. After restarting the application server any newly set password will be encrypted using the new encryption key. Without the files iiq.dat and iiq.cfg, passwords cannot be decrypted by IdentityIQ. If you run more than one instance of IdentityIQ, these files need to be placed in the WEB-INF/classes folder of each instance, or in the location specified in iiq.properties.

1. **Re-Encrypt Passwords**

Now, the new encryption key will be used for newly encrypted passwords, but existing passwords can still be decrypted using the default method on any system. To prevent that, existing passwords need to be re-encrypted. To do so, you need to create a new task of type “Encrypted Data Synchronization Task”.



Give the task a name and optionally exclude types (applications, identities or integration configurations) from processing. Click “Save and Execute” to immediately run the task.

After the task has completed, all (selected) encrypted data will be changed. While a password encrypted with the default key is prefixed with “1:”, items encrypted with the new encryption key are prefixed with “2:” or another number if multiple encryption keys are stored.

When looking up The Administrator's password in the console, this will look like:

1. > search identity password where name spadmin
2. 2:WpTZ2hmNaInTAJzeK9Swcw==

### User Rights

Every identity cube may be assigned one or more capabilities, which govern what functions they can perform within IdentityIQ. Collectively, these capabilities are known as user rights.

User rights can be automatically assigned during an aggregation or refresh task by using a Customization Rule. They can also be assigned manually in the IdentityIQ administrator interface.

IdentityIQ 7.2 ships with 25 capabilities. It is expected that Client Name will make use of the following capabilities:

* Auditor
* Certification Administrator
* Compliance Officer
* Identity Administrator
* Policy Administrator
* System Administrator
* Help Desk Personnel

This will provide the user rights illustrated below:

|  |
| --- |
| *Figure 5: Customer confidential capabilities (1)* |

|  |
| --- |
| *Figure 6: Customer confidential capabilities (2)* |

## Applications and Entitlements

This section is used to describe the specific application connectors that will be implemented to support Client Name’s implementation of IdentityIQ.

The following table provides an overview of the connectors that are identified for deployment:

|  |  |  |
| --- | --- | --- |
| Application | Authoritative Source | Connectors |
| App1 | Y | * Connector type |
| App2 | N | * Active Directory |

*Table 7: Applications and entitlements*

1. Authoritative only for initialization of itentity cubes

### Data Warehouse

TBD

### Active Directory 2012

TBD

## Implementation Approach

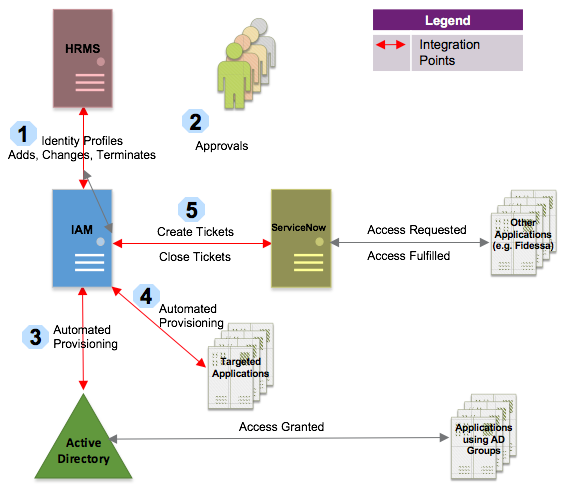
Detailed steps for the initial installation in a development environment would be as follows:

* Configure Tomcat application server for IdentityIQ:
* Configure sun.lang.ClassLoader.allowArraySyntax JVM setting.
* Set hibernate.bytecode.provider in the persistence.properties file.
* Install the base IdentityIQ product:
  + Expand the identityiq.war file into the Tomcat application container.
* Configure the number of allowable extended attributes:
  + Edit IdentityExtended.hbm.xml to allow 15 attributes.
  + Edit LinkExtended.hbm.xml to allow 10 attributes.
  + Edit CertificationItemExtended.hbm.xml to allow 10 attributes.
  + Edit ApplicationExtended.hbm.xml to allow 10 attributes.
  + Edit BundleExtended.hbm.xml to allow 10 attributes.
* Run the iiq schema script to generate the database DDL script.
* Create the IdentityIQ database.
  + Execute the generated DDL script.
  + Configure iiq.properties with the correct database connection settings.
* Start the application server and test connectivity by logging on to IdentityIQ as spadmin.
* Perform IdentityIQ 7.2 Patch 2 Upgrade.
  + Stop the application server.
  + Expand the upgraded identityiq.war file into the Tomcat application container.
  + Run the MS-SQL database upgrade script.
  + Execute the iiq patch command to complete the upgrade.
* Start the application server and test connectivity to the patched version by logging into IdentityIQ as spadmin.
* Configure Base Application.
  + Configure extended Identity, Application, and Account Attributes.
  + Email Integration
  + Create a Provisioning Integration Module (PIM) for HPSD integration.
* Configure extended identity attributes in IdentityIQ System setup.
* Configure Scope Hierarchy.
  + Create Scope Correlation Rule.
* Client Name-Specific Branding and Customization.
* Configure Authoritative Applications.
* Execute Identity Refresh task to assimilate all changes.
* Configure Target Applications.
* Completes Iteration 1 and 2.

# Workflows

## Joiner (new hire)

|  |
| --- |
| 1.) When a new hire in Data Warehouse is detected, IAM triggers the required changes by initiating the appropriate business processes including access requests, policy checking, notifications, approvals, and provisioning. |
| 2.) IAM triggers its workflow engine, carries out the required approval process. |
| 3.) For new hires, IAM automatically provisions the user into Active Directory and it adds the user into the desired AD group(s) based on the role(s) assigned. |
| 4.) IAM automatically executes the required scripts to create homeDrive and mailbox and updates AD with homeDrive and mailbox attributes. |
| 5.) IAM automatically provisions the user into the required target applications based on role(s) assigned. |
| 6.) IAM creates tickets in Service Desk to fulfill the rest of the onboarding tasks and to request for any additional access to applications. |
| \* All events during the new hires on-boarding process are logged. |



Assumptions:

1.) All Employee and Non-Employee identities can be identified.

2.) Caregiver DB will be used as the authoritative source for all Employee identities.

3.) Integrations between IAM and Caregiver DB, Active Directory have been completed.

4.) Targeted Applications have been on-boarded within IAM.

# Appendix: Abbreviations and Glossary

## Abbreviations

|  |  |
| --- | --- |
| Term | Expansion |
| IAG | Identity and Access Governance |
| IAM | Identity and Access Management |
| IDM | Identity Management |
| IIQ | SailPoint IdentityIQ |
| JDBC | Java Database Connectivity |
| JDK | Java Development Kit |
| JSON | JavaScript Object Notation data-interchange language |
| JVM | Java Virtual Machine |
| LDAP | Lightweight Directory Access Protocol |
| CUSTOMER CONFIDENTIAL | Customer Confidential |
| MVC | Model View Controller – architecture standard for 3-tier web applications |
| RACF | IBM Resource Access Control Facility |
| REST | Representational State Transfer |
| RPC | Remote Procedure Call |

## Glossary

|  |  |
| --- | --- |
| Term | Description |
| Access Request | Systems or processes used to request new access, make changes to existing access, or remove access to resources within an organization. |
| Activity | The normalized representation of the raw activity data collected from an activity data source such as a Windows Event Log or Syslog. Activity is represented as a java object (ApplicationActivity) and persisted in the database. |
| Activity monitoring | A means to monitor the user activity (raw system log data) for privileged (IT or business) users. IdentityIQ monitors and logs security activity at the operating system, application and database levels and identified security violations are reported to senior management. |
| Activity search | Use the Activity Search page to generate searches on activity on specific applications and by specific IdentityIQ identity. These searches can be used to isolate the risk areas and track activity on sensitive applications. |
| Activity Target Category | Groups of targets from one or more applications. For example, if you have inventory applications at three different locations and a procurement database on each, you can set each procurement database as a target, create a Procurement category, and then collect activity for all three procurement databases by using a single activity search. |
| Additional entitlement | Additional Entitlements are any entitlements to which the identity has access but do not comprise a complete role. For example, if a role is comprised of entitlements A, B, and C, but the identity only has access to entitlements A and B, A and B are included in the list of Additional Entitlements. Also, if the identity is assigned entitlements A, B, C, and D, and A, B, and C are grouped as the role, D is added to the Additional Entitlements list. |
| Aggregation | Aggregation refers to the discovery and collection of information from the applications configured to work with IdentityIQ. For example, IdentityIQ uses an Identity Aggregation task to pull the values associated with the identity attributes specified during the configuration process from user accounts on the designated applications. That information is then used to create the foundation of the IdentityIQ Identity Cubes. |
| Application | 1. The generic term used to refer to any data source with which IdentityIQ communicates to manage governance, risk management, and compliance for your enterprise. 2. The term used to refer to an instance of a configured IdentityIQ connector. Applications encapsulate the details of how a targeted system is accessed (Connector parameters), how the accounts and entitlement data on that system is classified (Schema) and how the accounts on that system are correlated to existing Identity Cubes. |
| Approval Workflow | Software that automates a business process for sending online requests to appropriate persons for approval. Approval workflow makes an approval business process more efficient by managing and tracking all of the human tasks with the process and by providing a record of the process after it is completed. |
| Audit Search | Use the Audit Search page to generate searches for audit records for specific time periods and for specific actions, sources, and targets. These searches can be used to locate and track events that occur within the IdentityIQ application. The information contained in the audit logs is different than the application activity, because the events in the audit log are not associated with an application or data source and might not be associated with a specific identity. |
| Authoritative Application | The identity authoritative application is the main repository for employee information for your enterprise, for example a human resources application. This might not be a risk application, but it is the data source from which the majority of the IdentityIQ Identity Cubes are built. |
| Business Process Modeler | Software that automates a business process for sending online requests to appropriate persons for approval. Approval workflow makes an approval business process more efficient by managing and tracking all of the human tasks invoCustomer Confidentialed with the process and by providing a record of the process after it is completed. |
| Capabilities | Capabilities control access within the IdentityIQ product. Access is controlled at the page, tab, and field level. |
| Certification | Certification enables you to automate the review and approval of identity access privileges, account group membership, and permissions, and role membership and composition. IdentityIQ collects fine-grained access (or entitlement) data and formats the information into reports, which are routed to the appropriate reviewers. Each report is annotated with descriptive business language - highlighting changes, flagging anomalies, and calling out violations where they appear.  Identity certifications enable reviewers to approve certifications for identities, or take corrective actions (such as removing entitlements that violate policy).  Role membership and composition certification enables reviewers to approve the composition of roles - the entitlements and roles that define the role being reviewed, and the identities to which the role is assigned, or take corrective actions.  Account group membership and permission certification enables reviewers to approve the permissions assigned to account groups and the members that make up the group, or take corrective actions. |
| Certification Periods | Certifications progress through phases as they move through their life-cycle; Active, Challenge, and Revocation. The phases associated with each certification are determined when the certification is scheduled.   * **Active:** The active phase is the review period during which all decision required within this certification should be made. During this phase changes can be made to decisions as frequently as required. You can sign off on a certification in the active stage only if no roles or entitlements were revoked or if the challenge period is not active. When you sign off on a certification it enters either the end phase or the revocation phase. To enter the revocation phase, the revocation period must be active and a revocation decision exist. * **Challenge**: The challenge phase is the period during which all revocation requests can be challenged by the user from which the role or entitlement is being removed. When the challenge phase begins, a work item and email is sent to each user in the certification affected by a revocation decision. The notifications contain the details of the revocation request and any comments added by the requestor. The affected user has the duration of the challenge period to accept the loss of access or challenge that decision.   Email notifications sent to non-IdentityIQ users contain a link to an end user portal which enables them to enter a revocation challenge as if they were logged into the product.  You can sign off on a certification in the challenge phase only if all challenges are completed and no open decision remains on the certification. When you sign off on a certification it enters either the end phase or the revocation phase. To enter the revocation phase, the revocation period must be active and a revocation decision exist.   * **Revocation:** The revocation phase is the period during which all revocation work should be completed. When the revocation phase is entered, revocation is be done either automatically, if your provisioning provider is configured for automatic revocation, or manually using a work request assigned to a IdentityIQ user with the proper authority on the specified application. The revocation phase is entered when a certification is signed off on or when the active and challenge phases have ended.   Revocation activity is monitored to ensure that inappropriate access to roles and entitlements is revoked in a timely manner. Revocation completion status is update at an interval specified during the deployment of IdentityIQ. By default this is performed daily. Click **Details** to see view detailed revocation information in the revocation report. |
| Certification Search | Use the Certification Search page to generate searches on certifications within your enterprise. These searches can be used to isolate specific certification risk areas and track the progress through their life-cycle. |
| Collector | Collectors provide the means by which IdentityIQ collects raw activity data for an application. A collector is a Java class that extends the AbstractActivityCollector class and implements the ActivityCollector interface. Collectors might have a one to many relationship with connectors. |
| Composite applications | Applications made up of multiple tiers, for example, platform account, database account and application account. Sometime referred to as an “n-tiered” application. |
| Connector | Connectors provide the means by which IdentityIQ communicates with targeted platforms, applications and systems. Connectors are Java classes that implement the IdentityIQ *Connector* interface.  There are two types of connector in IdentityIQ, application-type connectors that collect account information, and activity-type connectors that collect activity information. IdentityIQ uses the information from both types to maintain the identity cubes. |
| Correlation | Correlation refers to the process of correlating, or combining, all of the information discovered by IdentityIQ (identity attributes, entitlements, activity, policy violations, history, certification status, etc.) to create and maintain the IdentityIQ Identity Cubes. Correlation does not invoCustomer Confidentiale accessing external application to discover information. Correlation reviews the information contained within the IdentityIQ application and updates identity cubes as necessary. |
| Correlation Key | The attributes that IdentityIQ can use to correlate activity discovered in the activity logs for this application with information stored in identity cubes.  For example, activity logs might contain the full name of users instead of unique account ids. Therefore, correlation between the activity discovered by an activity scan and the identity cube of the user that performed the action must key off of the user’s full name. |
| Data Source | An instance of a configured IdentityIQ activity collector. Activity data sources encapsulate the details of how a given application activity source is accessed and how the raw activity data is parsed, normalized (fieldMap, Transformation Rule), and correlated to existing Identity Cubes. |
| Delegation | Passing a work item, such as the certification of an identity, role, or entitlement to someone else with certification authority. Delegation does not remove the item from your list of responsibilities, all delegated items must be acted upon before you can sign-off on the certification. |
| Entitlement | An entitlement is either a specific value for an account attribute, most commonly group membership, or a permission. |
| Entitlement Glossary | A business friendly dictionary of user access descriptions that can be associated with individual entitlements and permissions. |
| Forward | The Forward function is used to forward a certification request to a different IdentityIQ user with certification authority. When you forward a certification it is removed from your Certification page and does not show up on your risk score statistics. Owner history and all comments are maintained with forwarded work items on the View Work Item page. |
| Group | Groups are used to tracked accessibility, activity, and monitored risk by group membership. Risk scores are displayed on the Dashboard. Groups are defined automatically by values assigned to identity attributes or by account group membership. Account groups are based on common entitlement within an application, not common qualities as defined within IdentityIQ. |
| Group Factory | The Group Factory defines groups automatically by values assigned to identity attributes such as Department, Location, Manager and Organization. |
| Hierarchical role model | In role based access control, the role hierarchy defines an inheritance relationship among roles. For example, the role structure for a bank may treat all employees as members of the 'employee' role. Above this may be roles 'department manager' and 'accountant,' which inherit all permissions of the 'employee' role |
| Identity Cube | Multi-dimensional data models of identity information that offer a single, logical representation of each managed user. IdentityIQ automatically builds, manages and securely stores Identity Cubes, including both current and historical views. Each Cube contains information about entitlements, activity, and associated business context. |
| Identity Search | Use the Identity Search to generate searches on specific attributes of the IdentityIQ identities within your enterprise. These searches can be used to isolate specific risk areas or define interesting populations of people from multiple organizations, departments and locations. |
| Impact Analysis | Create a report that provides details on the impact changes will have on the rest of your product implementation. When you submit a change for analysis, no further changes can be made until the analysis process is completed or cancelled. |
| Lifecycle event | An identity-related event in which a user's relationship with the organization undergoes a chance, for example, new user is on boarded, existing user is promoted. |
| Lifecycle management | The end-to-end process of managing user access throughout a user's lifecycle within the organization. |
| Mitigation | Mitigation refers to any exceptions that are allowed on policy violations discovered during a certification process. |
| Password Management | Automation of the process for controlling setting, resetting and synchronizing passwords across systems. |
| Password Reset | The process of resetting a lost or forgotten password. Typically requires the user to answer a set of challenge questions to provide their identity. |
| Password Synchronization | The process of propagating changes to all passwords with the same value across multiple platforms and applications |
| Permitted (optional) Role | A role that is not automatically granted to a user, but may optionally be requested or assigned. Permitted roles are associated with higher-level business roles and allow the organization to enforce least privilege while controlling the total number of roles required to model access rights within the enterprise. |
| Phase | Certifications progress through phases as they move through their life-cycle; Active, Challenge, and Revocation. The phases associated with each certification are determined when the certification is scheduled. |
| Policy | Policies are comprised of rules used to enforce any policies, separation of duty, activity or risk, defined within your enterprise. For example, a rule might be defined that disallows a single IdentityIQ identity from having roles that enable them to both request and approve purchase orders. |
| Policy Type | The type of policy.   * **Activity:** Make snsure that users are not accessing sensitive application if they should not or when they should not. * **Advanced:** Custom policies created using match lists, filters, scripts, rules, or populations. * **Generic:** Any custom policies created in your enterprise. * **Risk:** Make ssure that users are not exceeding the maximum risk threshold set for your enterprise. * **SOD:** Separation of duties policies ensure that identities are not assigned conflicting roles or entitlements. |
| Population | Populations are query based groups created from the results of searches run from the Identity Search page. Searches that result in interesting populations of identities can, optionally, be saved as populations for reuse within IdentityIQ. Members of a population might not share any of the same identity attributes or account group membership. Population membership is based entirely on identity search parameters. |
| Profile | A profile is a set of entitlements on an application. An entitlement is either a specific value for an account attribute, most commonly group membership, or a permission. Profiles can be used in multiple roles. |
| Profile Class | An optional class used to associate an application with a larger set of applications for role modeling purposes. For example, you might set a profile class of XYZ on all of the applications on which any user that has read account privileges should be assigned the role XYZ Account Reader. You can then create a single profile for that role instead of a separate profile for each instance of the applications. During the correlation process any user with read account privileges on any of the applications with the profile class XYZ is assigned the role XYZ Account Reader. |
| Provisioning | The process of granting, changing, or removing user access to systems, applications and databases based on a unique user identity. |
| Reassign | Use the reassign feature to reassign certifications to the appropriate owner. Access reassignment is performed at the identity level. Identities that are reassigned are removed from the identities list and do not reflect as part of the completion status for this certification. All reassigned identities must be acted upon, however, before you can sign-off on the certification.  Bulk reassignment enables you to reduce cumbersome identity certification lists by reassigning identities to appropriate certification approvers. For example, if you are the owner of an application with thousands of accounts, you can use this feature to reassign identities for certification by department or manager. |
| Required Role | a role that is automatically provisioned to a user once the user is assigned to the higher-level role containing the required role. |
| Revocation | Use revocation to request the removal of an identities access to a specified role or entitlement. No action is taken on a revocation request until the certification containing the request is completed and sign off on. This is done to make sure that no entitlement is removed until final confirmation is received from the requestor.  Entitlements that are assigned to more than one role are not revoked with the role. For example, if role A is made up of entitlements X, Y and Z, and role B is made up of entitlements W and X, revoking role A only revokes entitlements Y and Z.  IdentityIQ can automatically revoke the specified access if automated revocation is configured for your provisioning provider.  Revoked entitlements continue to be listed with the identity until the next Account Aggregation type task is run on the application with which they are associated. Revoked roles are removed from the identity cube with the next Identity Refresh. |
| Risk | The IdentityIQ risk-management scoring system applies analytics to identity and activity data to pinpoint areas of risk and enable you to focus your compliance efforts where they are needed most. IdentityIQ uses configurable algorithms to assign a unique risk scores. Scores are based on multiple factors and updated regularly. Using this risk scoring system, you can configure IdentityIQ's automated controls to lower user risk scores and their overall corporate risk profile. |
| Role | A role is a collection of other roles or entitlements that enable an identity to perform certain operations within your enterprise. For example, one role might enable the request of purchase orders and another might enable the approval of purchase requests. IdentityIQ uses roles to monitor these entitlements, identify separation of duty policy violations, and compile identity risk scores to enable you to maintain compliance. |
| Role Assignment | The process of granting roles to users. Can be performed through self-service tools or via an automatic assignment rule. |
| Role Creation | The process of defining roles within a role model and mapping those roles to the appropriate set of access privileges based on business process and job function. |
| Role Certification | The periodic review of a role or roles in order to validate that the role contains the appropriate access privileges and that members of the role are correct. Role certifications are commonly used as an internal control and a way to prevent role proliferation. |
| Role Lifecycle Management | The process of automating role creation, modification, retirement; role approvals; role certifications; and role analytics. |
| Role Management | A new category of identity management software that focuses on the discovery, analysis, design, management, reporting, and distribution of roles and related policy. |
| Role Model | A schematic description of roles that defines roles and role hierarchies, subject role activation, subject-object mediation, as well as constraints on user/role membership and role set activation. |
| Rules | 1. Custom rules are created during the configuration process and are used by IdentityIQ to handle correlation, notification, escalation and IdentityIQ identity creation.    1. Correlation rules are used to define the identity attribute to use when correlating accounts discovered during an application aggregation with identities that exist in IdentityIQ. For example you might want to set the correlating attribute as email address or first and last name.    2. Notification rules are used to define the identity that is notified when policy violations are detected.    3. Escalation rules are used by the workitem expiration scanner to determine to whom to route workitems that have expired.    4. Identity creation rules are used to set attributes on new Identity objects when they are created. New identities may be created during the aggregation of application accounts, or optionally created after pass-through authentication. One common operation is to change the name property of the identity when the default application name is complex (such as a directory DN). Another common operation is to assign a set of initial capabilities based on the attributes pulled from the application account. 2. Rules are used to enforce your separation of duties policies by identifying IdentityIQ identities that are assigned conflicting roles. For example, a rule might be defined that disallows a single IdentityIQ identity from having roles that enable them to both request and approve purchase orders. 3. Violations on each of a policy’s rules, when detected, are stored in the offending identity cube. These violations also appear on identity score cards and enable you to identify high-risk employees and act accordingly. |
| Scope | A scope is a container within the product in which objects can be placed to restrict access.   * **Controlled Scope:** A scope over which an identity has access. This is combined with the identity's capabilities to determine to which objects a user has access. Every identity in the system can control zero or more scopes. * **Assigned Scope:** A scope in which an object lives and is used to control who can view and manage the object. Every object in the product is assigned zero or one scopes. By default, an object that does not have an assigned scope is available to everyone. The default behavior can be changed during configuration. |
| Self-service | Software that allows users to request access to resources using a self-service interface, which uses workflow to route the request to the appropriate manager(s) for approval. |
| Subordinate certification | Subordinate certifications are any certifications that must be completed before the top-level certification can be completed. Examples of subordinate certifications are any groups of identities that you reassign, or any lower-level, subordinate, manager certifications.  Subordinate certifications are not displayed as part of the identities list and do not reflect as part of the completion status for this certification. All subordinate certifications that require completion (manager/subordinate manager certifications) or reassigned certifications must be in a complete state before the certification can be signed off on. |
| Workgroups | Groups of users within IdentityIQ that can perform actions, for example, approvals, or own objects, for example, roles, policies, within the system. |
| Work Item | A work item is anything that requires action before it is completed. Work items can be entire processes, such as certifications, or any piece of a process, such as the approval of one entitlement for one identity on one application. |
| Work queues | Shared tasks from which Workgroup members can perform actions within the system. |

|  |  |
| --- | --- |
| Contact us |  |
| **Ken Dunbar**  Director - Cybersecurity  T 972 342 9772  E kbdunbar@kpmg.com |  |
|  |  |
| ©2018 KPMG LLP, a Deleware limited liability partnership and a member firm of the KPMG network of independent member firms affiliated with KPMG International Cooperative (“KPMG International”), a Swiss entity. All rights reserved.  The information contained herein is of a general nature and is not intended to address the circumstances of any particular individual or entity. Although we endeavour to provide accurate and timely information, there can be no guarantee that such information is accurate as of the date it is received or that it will continue to be accurate in the future. No one should act on such information without appropriate professional advice after a thorough examination of the particular situation.  The KPMG name, logo and “cutting through complexity” are registered trademarks or trademarks of KPMG International Cooperative (“KPMG International”). | |