ISIQ Best Practices Deployment Guide

This document recommends a process for deploying IBM Security Information Queue (ISIQ). The process has been divided into three phases, with an estimated time for each. The phased approach recommended here is a best practice for achieving successful implementations.

The assumption in this document is that someone relatively new to Docker and ISIQ is doing the work. If it's an experienced practitioner, shrink the time estimates accordingly.

Phase 1: Familiarization and Testing — Estimate: 3-4 Days

Action	Reference/Comments
Read about Docker swarm, follow the tutorial	https://docs.docker.com/engine/swarm/ and https://docs.docker.com/engine/swarm/swarm-tutorial/
Review the ISIQ documentation set	Deployment Guide, User's Guide, and Troubleshooting Guide
Provision a Linux VM for a test ISIQ in a single-node Docker swarm configuration	See "System Requirements" in the Deployment Guide. Setting up a single-node configuration is intended to give you simple, hands-on experience with Docker and ISIQ.
Install latest Docker CE (v18.09 as of June 2019) on the provisioned Linux VM	See the Docker install info at https://docs.docker.com/install/ The instructions in this document assume VMs will be used to run ISIQ, but physical Linux systems would work just as well.
Go to the ISIQ starter kit web page and download the starter kit zip file to the Linux VM; extract the contents of the zip	https://www-01.ibm.com/support/docview.wss?uid=ibm10787861
Create secret key and SSL certificates for nginx using <starterKitDir>/cfg/setup.sh	This step and subsequent steps are described in "Summary of Installation & Configuration Steps" in the Deployment Guide.
Configure OIDC by updating <starterKitDir>/cfg/oidc/oidcSettings.json	If you don't have a preferred OpenID provider, you can specify IGI as the provider for ISIQ. Refer to the "OpenID" section of the Deployment Guide.
Review the default <starterKitDir>/cfg/connect/txdef.json file for possible modification	This is an important data analysis step and will vary in scope depending on whether, and to what extent, you have customized ISIM and IGI. The contents and format of txdef.json, and how to modify it, are discussed in the User's Guide sections, "Custom Transformations" and Appendix F.
Login to Docker Hub	To obtain a Docker account, see https://hub.docker.com/signup
Run <starterKitDir>/util/imagePullAll.sh to download ISIQ's Docker images	As your Docker stacks start up, any required ISIQ images missing from the local repo will be downloaded dynamically. These dynamic downloads can take many minutes to finish, which makes it difficult to monitor progress of the initial startup sequence. Running the "pull" script ahead of time ensures that ISIQ's images have been downloaded and are ready to be deployed.
Create the Docker swarm	See Step 5 in the Deployment Guide's "Summary of Installation & Configuration Steps".
Review the <starterKitDir>/yml/single_node files that deploy a single-node Docker swarm	The 3 required ISIQ .yml files are broker-stack, connect-stack, and app-stack, and should be started in that sequence. logs-stack.yml is optional (and can be implemented later) for ISIQ self-monitoring.
Start the ISIQ Docker stacks	See "Deploying the ISIQ stacks" in the Deployment Guide.
Verify the stacks started and the OIDC provider lets you login	See "Verifying the stacks are deployed" in the Deployment Guide. Also, if need be, refer to "Diagnosing Service Startup Failures" in the Troubleshooting Guide.

Action	Reference/Comments
(Optional) Configure the logs stack with Elasticsearch, Grafana, and Kibana	The ISIQ-supplied logging & monitoring components are discussed in the “logs-stack.yml” section of Appendix A in the Deployment Guide. If you have a logging/monitoring stack you’d prefer to use instead, you can do so. These particular components aren't required.
Configure your test ISIM and test IGI systems as ISIQ products	For instructions, refer to “Configuring ISIM” and “Configuring IGI” in the User's Guide.
Apply IGI customizations	Several IGI rule flow updates and other customizations must be made before you start integrating ISIM data. Refer to “Appendix A: IGI Customizations for ISIQ” in the User’s Guide.
Ensure at least one Service Center user is defined in the Access Governance Core -> Manage -> Users tab of IGI	Your IGI should already have at least one SC user since it’s the default user type. This is merely a caution that without an SC user, you will experience long delays when loading ISIM data. See the IMPORTANT note in <starterKitDir>/Readme.txt
Subscribe the test IGI to the test ISIM	In the ISIQ UI, navigate to the dashboard of your test IGI product and click the “Subscriptions” dropdown to define a subscription. This action will initiate ISIM-to-IGI data integration. For details, refer to the "Subscriptions" section of the User's Guide.
Login to your test IGI and navigate to the "Monitor" tab to observe how test ISIM entities are flowing across to IGI	At first, the Users, Accounts, Organizations, etc. from ISIM should be listed as "Unprocessed" events in IGI. After a while, they should move to a status of "Success". When there are no more unprocessed events, the initial ISIM-to-IGI data load is complete.
Make a few updates in the test ISIM and confirm the updates are integrated in IGI	This step validates that after the data load, ongoing ISIM updates get processed. For example, transfer an ISIM person to a different organization. You should see the change reflected in the IGI UI.
(Optional) Subscribe the test ISIM to the test IGI to create two-way integration	For a list of IGI user events that ISIQ will forward to ISIM, refer to Appendix C of the User's Guide. Deciding whether to set up bi-directional subscriptions depends on your requirements. For some customers, a one-way flow from ISIM-to-IGI will be sufficient.

Once you're familiar with installing/configuring ISIQ and with defining products and subscriptions, and you've validated ISIM-IGI data integration between your test systems, you're ready to move to Phase 2.

Phase 2: Preparation for Production — Estimate: 1-2 Days

Action	Reference/Comments
Provision three Linux VMs for your production ISIQ to run in a three-node Docker swarm cluster	These steps assume a multi-node cluster for fault tolerance, workload balancing, rolling updates, etc. But if assigning three VMs is not practical in your environment, it's a viable alternative to implement a single-node configuration.
Install the latest Docker CE on one of the Linux VMs; this will be the manager node in the swarm	https://docs.docker.com/install/
Download the starter kit zip file and extract its contents on the manager node	https://www-01.ibm.com/support/docview.wss?uid=ibm10787861
Create secret key and SSL certificates for nginx	
Update <starterKitDir>/cfg/oidc/oidcSettings.json	If you're using the same OIDC provider in test and prod, you can copy the oidcSettings.json file from Phase 1.
Apply any modifications to the default <starterKitDir>/cfg/connect/txdef.json	Again, you can copy the txdef.json file from your Phase 1 work, assuming the data transformations are the same.
Login to Docker Hub and run imagePullAll.sh	This ensures all images are downloaded ahead of time.
Create the Docker swarm	See Step 5 in the Deployment Guide's "Summary of Installation & Configuration Steps".
Review the <starterKitDir>/yml/cluster files that deploy a three-node Docker swarm	See "Appendix A: YAML Files" in the Deployment Guide.
Start the ISIQ Docker stacks; confirm that they start successfully and that OIDC login works	
(Optional) Configure the logs stack	
Configure your production ISIM as a product in ISIQ	Configuring an ISIM won't alter its contents. To receive updates, the ISIM must be subscribed to an ISIQ source.
If possible, set up a fresh IGI installation (not your production IGI) with an initialized database, and configure that IGI product in ISIQ	You can reuse your test IGI from Phase 1. The goal is to simulate an initialized IGI in order to observe and measure ISIQ's data load. To remove previously integrated ISIM data from IGI, see Appendix G in the User's Guide. Once you have an empty IGI DB, it's a good practice to take a snapshot of the IGI data tier for quick restores in case you do multiple ISIM-to-IGI test runs.
Apply IGI customizations	See Appendix A in the User's Guide.
Subscribe this IGI to your production ISIM	This will be a dry run to verify that production ISIM users, accounts, etc. are migrated across, that ISIM-to-IGI data transformations work correctly, and to note the elapsed time for the load. If you don't set up two-way integration with IGI, the production ISIM will not be altered in the dry run.
Login to IGI and navigate to the "Monitor" tab	Check "Target inbound – Account events" and the related tabs to ensure that "Unprocessed" events get processed.
Determine if any production ISIM entities were not inserted into IGI	There may be custom policies or other IGI items that need to be manually re-added.

After any integration issues have been resolved, and you've completed a set of basic test cases, you're ready to move to Phase 3.

Phase 3: Production Deployment of ISIQ **Estimate: 1 Day***

Action	Reference/Comments
Apply IGI customizations to the production IGI	See Appendix A in the User's Guide.
Subscribe your production IGI to your production ISIM	
Subscribe your production ISIM to your production IGI if you want two-way integration	
Use the "Monitor" tab in the IGI UI to observe the ISIM entities flowing across in the initial data load	
Re-add any missing items in the production IGI	
Implement a monitoring/alerting process to verify ongoing ISIM-IGI data integration	After the initial data load, you will want to establish a monitoring/alerting process that ensures ISIM updates continue to flow to IGI. ISIQ offers a logging/monitoring framework using Elasticsearch, Grafana, and Kibana. There is also an ISIQ System Health dashboard described in the "Metrics and Alerts" section of the User's Guide.

* The initial data load could take many hours depending on number of ISIM entities and the speed with which IGI processes the database inserts. As a result, the production deployment could require more than one day. Use the elapsed time from the dry run to help with this estimate.