Micro Focus Security
ArcSight Interset Standard Edition

Software Version: 6.1.0

Deployment Guide

Document Release Date: July 2020

Software Release Date: July 2020

Contents

Introduction

With the growing number of threats to monitor in the IT ecosystem, IT organizations have a demanding need to continuously think of better and effective ways to secure their enterprise network. In today’s world, employees can access their company’s data and applications from within the company, home, and even through personal mobile devices regardless of their geographical location. With IT organizations having to manage their assets both on-premises and in the cloud environment, it has become increasingly challenging for IT security teams to detect any malicious activities carried out by internal users intentionally or accidentally, such as data theft, data exfiltration, and account compromise.

While security information and event management (SIEM) solutions such as Micro Focus ArcSight Enterprise Security Manager (ESM) offer security and compliance monitoring solutions that focus mostly on external threats, IT organizations need solutions that also perform in-depth user behavior monitoring that can detect anomalies and potential threats happening within the organization. Most of the data loss and data breach activities are carried out by users with valid credentials.

ArcSight Interset is a user and entity behavioral analytics solution that uses data science and advanced analytics to identify the top risky entities and behaviors occurring in your organization. Using your organization's data, Interset first establishes the *normal* behavior for your organizational entities and then using advanced analytics, it identifies the *anomalous* behaviors that constitute potential risks such as compromised accounts, insider threats, or other cyber threats.

Interset detects potential threats by performing the following:

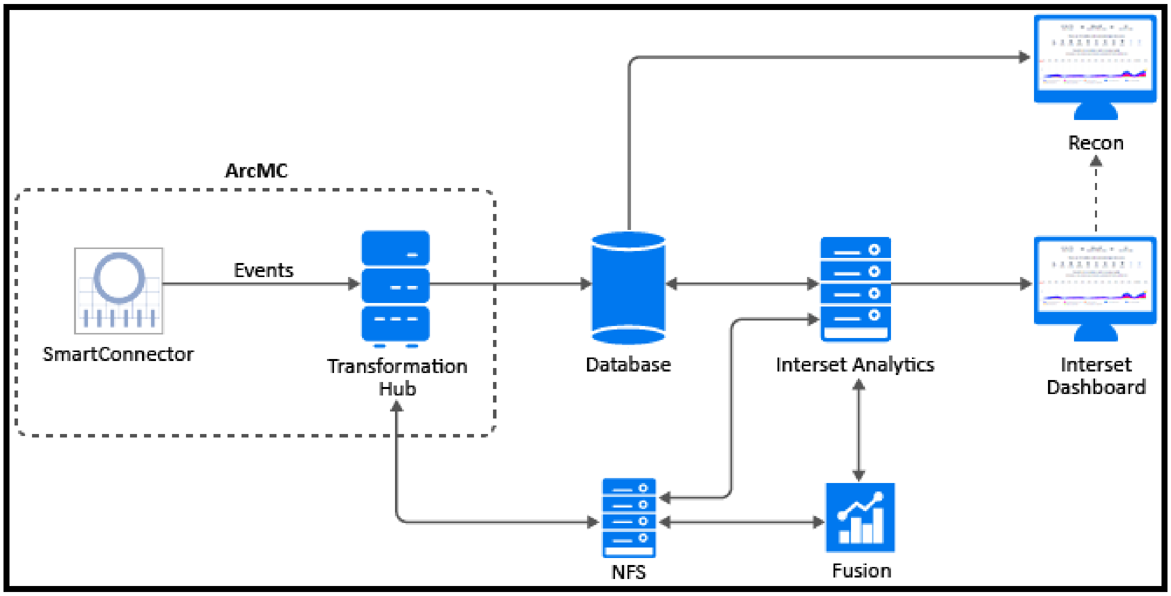
- uses unsupervised machine learning techniques to automatically define user profiles and baselines
- actively monitors account access patterns and actions on the associated entities against defined baselines to detect anomalies
- applies a risk score for each entity based on the anomalies detected
- displays anomalies prioritized by the user risk score in a user-centric, interactive dashboard that helps Security Analysts investigate the highest risks first and take necessary actions immediately

Therefore, Interset significantly decreases the number of threats that go undetected and increases a Security Analyst's ability to quickly investigate all detected anomalies.

Deployment Architecture

Interset is a containerized application that is based on Container Deployment Foundation (CDF). CDF is a container-based delivery and management model built on Kubernetes and Docker. After you install CDF, you can then use it to deploy and manage container-based products, such as Interset and Micro Focus Transformation Hub. You also need to install other software and components, such as the database for data storage and Micro Focus ArcSight SmartConnectors for data collection from various data sources.

The following diagram helps you understand the Interset deployment architecture:



The following table describes the components involved in your Interset deployment:

Component	Description
SmartConnectors	Collect events from the supported data sources and then send these events in Common Event Format (CEF) to the **th-cef** Kafka topic in Transformation Hub. Event data include entity information such as users, assets, and endpoints based on the event type.
Transformation Hub	Centralizes event processing and enables event routing. Using Transforming Stream Processors (C2AV), it transforms CEF events in the **th-cef** Kafka topic to Avro format and stores the transformed events in the **th-arcsight-avro** Kafka topic.
Database	The Kafka scheduler in the database reads events from the **th-arcsight-avro** Kafka topic and stores them in the database. In addition to events, the database also stores Interset Analytics data. Based on Vertica technology.
Interset Analytics	Performs the vital task of determining individual behavioral baselines, and then discovering and ranking deviations from those baselines.
Interset Dashboard	A browser-based rich user interface that allows you to visually explore

Component	Description
	the analytics results and raw data.
Recon	*(Optional)* A search and log management tool that helps you explore events to gain insight into specific alerts and hunt for hidden security threats.
Fusion	Required for the database, user management, and single sign-on configuration.
Network File System (NFS)	Stores some of the persistent data generated by Transformation Hub, Interset, and Fusion.
ArcSight Management Center (ArcMC)	*(Optional)* A centralized management interface that helps you to effectively administer and monitor Transformation Hub and SmartConnectors.

System Requirements

This section provides information about system requirements and tuning guidelines for ArcSight Interset.

⚠ **Important**: For information about system requirements for CDF, Transformation Hub, the database, and SmartConnectors, see the specific product documentation in the ArcSight Product Documentation website.

Software Requirements

Software	Version
CDF	2020.05
Transformation Hub	3.3.0
Fusion	1.1.0
Recon (Optional)	1.0.0
Database	Vertica 9.2.1
SmartConnectors	8.0.0

Interset 6.1.0 supports the following Web browsers:

- Microsoft Edge - Latest, Chromium only
- Google Chrome - 74 and above
- Mozilla Firefox - 67 and above

License Requirements

Interset comes with a trial license. Install a valid license before the trial license expires or if the trial license policy has been violated. Purchase the relevant license based on the number of entities you want Interset to analyze. The license policy is violated when the number of entities exceeds the maximum limit. Renew your license before its validity expires or if the license policy has been violated.

Transformation Hub and ArcMC come with a trial license each. For each of these components, install a valid license before the trial license expires.

The database comes with an inbuilt license. The storage capacity for the license is 976 PB. Renew your database license if the storage capacity exceeds 976 PB.

Contact Micro Focus Customer Support at https://softwaresupport.softwaregrp.com/ to procure the licenses.

Hardware Requirements

The guidelines in this section are for a deployment where you install all of the following software:

- Interset
- Database
- Transformation Hub
- Fusion
- Recon

Understanding the Workload for Interset

The total workload for Interset depends on the following factors:

- The number of events collected by the SmartConnectors from the data sources and sent to the different storage components, that is, Elasticsearch, Transformation Hub, and the database.
- The number of events and the number of entities processed by the Interset Analytics component to produce the Interset Analytics results that are sent to the different storage components, that is, Elasticsearch and the database.

The hardware requirements for Interset comprise the following:

- Processing requirements based on the Events per second (EPS) and the number of entities.
- Storage requirements based on the EPS, the number of entities, and the number of days' events.

The hardware specifications provided in this section were determined for the following metrics:

- EPS: 5000 to 10000
- Entities: 15000
- Interset Analytics run frequency: Once a day on 1 day's events
- Storage for Elasticsearch and the database: Storage capacity based on 1 day's events
- Storage for the system sizing: Storage capacity based on 30 days' events

You can use the given information to determine the processing and storage requirements for different values of the metrics.

System Sizing

The following system sizing was used to determine the processing and storage requirements for the specified metrics:

Type	Number of Nodes	CPU per Node	RAM per Node	Number of Disks	Storage per Node
Master	1	16 (32 vcpu)	64 GB	2 SSD	960 GB
Worker	4	24 (48 vcpu)	128 GB	6 SSD	5760 GB
Database	3	24 (48 vcpu)	128 GB	6 SSD	5760 GB

Processing Requirements for Interset

The following table provides the Interset processing requirements for the specified metrics:

Component	Number of Instances	CPU per Instance	RAM per Instance	Total CPU for Component	Total RAM for Component
Interset UI	1	1	0.2 GB	1	0.2 GB
Interset API	1	1	1 GB	1	1 GB
H2	1	1	1 GB	1	1 GB
Interset Exports	1	1	1 GB	1	1 GB
HDFS NameNode	1	1	0.5 GB	1	0.5 GB
HDFS DataNode	4	1	0.5 GB	4	2 GB
Logstash	12	2	2 GB	24	24 GB
Interset Analytics Driver	1	1	5 GB	1	5 GB
Interset Analytics Executor	40	1	8 GB	40	320 GB
Elasticsearch Master	1	2	2 GB	1	2 GB
Elasticsearch data	4	15	18 GB	60	72 GB

Total vcpu required: 135

Total RAM required: 428.7 GB

Interset Analytics Tuning Parameters

The following table provides the Interset Analytics tuning parameters information for the specified metrics:

Parameters	5000/10000 EPS
Parallelism	40
Number of Executors	40
Number of Cores per Executor	1
Memory per Executor (GB)	8
Driver Memory (GB)	5
esBatchEntries	0
esBatchBytes (MB)	15

Note:

- Increase the number of Logstash instances if the Kafka partitions are increased and there is sufficient CPU and RAM.
- Increase the number of Executors if there is sufficient CPU and RAM.

Processing Requirements for Transformation Hub

The following table provides the Transformation Hub processing requirements for the specified metrics:

Component	Number of Instances	CPU per Instance	RAM per Instance	Total CPU for Component	Total RAM for Component
TH Kafka	4	4	4 GB	16	16 GB
TH Zookeeper	4	0.5	2 GB	2	8 GB
TH Schema Registry	4	0.5	1 GB	2	4 GB
TH C2AV Processor	1	4	2 GB	4	2 GB
TH Kafka Manager	1	0.5	1 GB	0.5	1 GB
TH Web Service	1	0.5	1 GB	0.5	1 GB
TH Routing Processor	2	0.5	1 GB	1	2 GB
TH C2AV Processor ESM	1	1	1 GB	1	1 GB

Total vcpu required: 27

Total RAM required: 35 GB

Processing Requirements for Fusion

The following table provides the Fusion processing requirements for the specified metrics:

Component	Number of Instances	CPU per Instance	RAM per Instance	Total CPU for Component	Total RAM for Component
Hercules Common Services	1	0.5	1 GB	0.5	1 GB
Hercules Management	1	0.5	1 GB	0.5	1 GB
Hercules RethinkDB	1	0.5	2 GB	0.5	1 GB
Hercules Search Engine	1	0.5	1 GB	0.5	1 GB
Hercules OSP	1	0.5	1 GB	0.5	1 GB
Dashboard Metadata WebApp	1	0.5	1 GB	0.5	1 GB
Dashboard WebApp	1	0.5	1 GB	0.5	1 GB
Database Monitoring WebApp	1	0.5	1 GB	0.5	1 GB
Autopass Im	1	0.5	1 GB	0.5	1 GB
Common Doc WebApp	1	0.5	0.1 GB	0.5	0.1 GB

Total vcpu required: 5

Total RAM required: 9.1 GB

Processing Requirements for Recon

The following table provides the Recon processing requirements for the specified metrics:

Component	Number of Instances	CPU per Instance	RAM per Instance	Total CPU for Component	Total RAM for Component
Hercules Search	1	0.5	1 GB	0.5	1 GB
Hercules Analytics	1	0.5	1 GB	0.5	1 GB

Total vcpu required: 1

Total RAM required: 2 GB

Storage Requirements For Elasticsearch and the Database

The storage for Elasticsearch and the database is incremental. The following table provides the storage requirements for the specified metrics. It encompasses the storage capacity for both the raw events and the Interset Analytics data.

Component	Number of Instances	Number of Entities	EPS	Disk Size per Instance	Total Disk Size per Day
Elasticsearch	4	15000	5000	73 GB	292 GB
			10000	98 GB	392 GB
Database	3	15000	5000	81 GB	243 GB
			10000	161 GB	483 GB

Storage Requirements for Transformation Hub

The storage for Transformation Hub is non-incremental and it is a buffer for storing only the raw events. The following are applicable for storing events in Transformation Hub:

- Events are stored only for the Kafka retention period. Maximum is 2 days.
- Events beyond the maximum Kafka partition size are removed. Default is 60 GB.
- The storage capacity is independent of the number of entities.

The maximum storage for Transformation Hub is determined by the following formula:

Maximum storage = Number of Kafka Partitions * Maximum Partition Size * Number of Kafka Instances

The following table provides the storage requirements for the specified metrics:

Component	Number of Instances	Kafka Topics	Number of Kafka Partitions	EPS	Disk Size per Instance	Total Disk Size
Transformation Hub (with GZIP)	4	th-cef	12	5000	230 GB	920 GB
		th-arcsight-avro	12	10000	525 GB	2100 GB

Supported Data Sources and SmartConnectors

Interset supports the following Data Sources and SmartConnectors:

Data Sources	Supported Smart Connectors
Access	SmartConnector for Microsoft Windows Event Log - Native Application and System Event Support SmartConnector for Microsoft Windows Event Log - Unified Application and System Event Support
Active Directory	SmartConnector for Microsoft Active Directory Windows Event Log Native
VPN	SmartConnector for Microsoft Network Policy Server File SmartConnector for Pulse Secure Pulse Connect Secure Syslog SmartConnector for Nortel Contivity Switch Syslog
Web Proxy	SmartConnector for Microsoft Forefront Threat Management Gateway File SmartConnector for Squid Web Proxy Server File SmartConnector for Blue Coat Proxy SG Multiple Server File

In addition, a fuller set of SmartConnectors is supported for those sources which provide data of relevance to the Interset analytics models. Micro Focus may need to examine sample logs to optimize analysis of data from this broader set of sources.

To ingest packaged data from other containers such as IBM QRadar, McAfee ESM, and Splunk, please contact Micro Focus Customer Support at https://softwaresupport.softwaregrp.com/.

Ports Used

In addition to the ports used by CDF, Transformation Hub, and the database, Interset uses the following ports when firewall is enabled. Ensure that the following ports are available:

Ports	Direction	Description
TCP 30820	Inbound	Used for the database to connect to HDFS during Analytics processing
TCP 30070	Inbound	Used for Hadoop Monitoring Dashboard (Optional)
TCP 30010	Inbound	Used for communication between the HDFS NameNode and the HDFS DataNodes

Deployment Types

You can choose to deploy in a single-node or multi-node environment, depending on your anticipated workload and whether you need high-availability. For more information about deployment sizing and tuning, see the "Hardware Requirements" section in System Requirements.

If you already have an existing CDF cluster for Transformation Hub, you can deploy Interset to the same cluster. Reusing existing clusters would reduce costs and system management effort compared to deploying these software in a new cluster.

Single-node Deployment

In a single-node deployment, you deploy all of the Interset components and the necessary software on a single node. This method of deployment is suitable only for small workloads and where you do not need high availability.

Multi-node Deployment

For larger workloads, you must deploy Interset and the required software in a multi-node cluster setup. Multi-node deployment does load balancing across several worker nodes and is scalable to handle large workloads. You can add multiple master nodes and worker nodes to scale. While you can add worker nodes even after the installation, you can add master nodes only during the installation. Therefore, plan your deployment before you start the installation process.

High Availability Deployment

To avoid single point failures and reduce downtime, you should ensure that your deployment is highly available.

For high availability deployment, you must set up three master nodes and at least two or more worker nodes depending on the workload. Three master nodes are required to ensure that even in cases where two master nodes are unavailable, there is still another master node available. If only two master nodes are used and the primary master node is taken offline for maintenance or upgrade, there will only be a single master node available creating a single point of failure. If the available single master node fails, the cluster stops and cluster orchestration will not be possible until the master is back online.

For high availability of the database, you must set up three database nodes.

While you can add worker nodes even after the installation, you can add master nodes only during the installation. Therefore, plan your deployment before you start the installation process.

Deployment Methods

You can deploy Interset either by using the provided installation scripts or manually.

Deployment Using Scripts

To enable an easier deployment, Interset provides scripts that automatically take care of all the prerequisites, software installations, and post-installation configurations. The scripts are applicable for single-node, new deployments where high availability is not needed. However, if you prefer to manually set the configurations and the installations because of your organization’s security policies, you can deploy Interset manually in single-node deployments as well. The scripts configure the system to match the settings described for performing a manual deployment.

The installation scripts expect your environment to be in a specific state. Before deciding to use the installation scripts, review the considerations for installation.

For information about installing Interset by using scripts for a single-node deployment, see Deploy Interset Using Scripts.

Manual Deployment

In deployments with a larger workload where high availability is mandatory, you must manually perform all the necessary system configurations and software installations.

The following manual deployment methods are applicable for both single-node and multi-node deployments:

- Deploy Interset in a new cluster.
- Deploy Interset in an existing cluster.

For single-node and multi-node deployments of Interset in a new cluster, you can use some of the installation scripts to make your tasks easier, then complete the rest of the configurations and installations manually.

For information about deploying Interset manually, see Deploy Interset Manually.

Deciding to Use the Scripts or Manual Deployment Method

To determine whether to use the installation scripts or perform a manual deployment, review the following considerations:

- The scripts deploy Interset on the operating system with a default minimum installation. If you have any customizations on the operating system, we recommend you to perform the prerequisites manually and perform deployment and post deployment configuration using scripts.
- The scripts deploy Interset only on a singled-homed network (a single-homed stub system is one that is connected with a single network link). If you have a dual-homed network (dual or redundant

connections to a single Internet Service Provider), we recommend that you use the manual deployment process.

- The scripts automatically tune the system for a single-node deployment with a small workload.
- The script configures the database agent to use the port 5445 instead of the default port 5444, as the script installs both CDF and the database on the same node.
- The scripts register a service with the operating system to automatically start the database Kafka scheduler to collect event data.
- The scripts install the cluster with a single master node and single worker node running on the same system. You can add worker nodes after the installation to scale and enable worker high availability.
- If you use the scripts, you cannot configure high availability for the master node. If you want high availability for the master node, we recommend that you use the manual deployment process.
- The scripts disable the option to authorize Micro Focus to collect suite usage data.
- The scripts create NFS shares on the system used by the containers in the cluster. They configure the firewall to disable remote access to this NFS server. If you plan to add additional nodes to the cluster, you must enable remote access to the NFS server in the firewall.
- The scripts use the following paths by default:
  - To install Kubernetes: **/opt/arcsight/kubernetes**
  - To create NFS shared directories: **/opt/NFS_Volume**. You can edit the path.
  - To unzip the database installer file: **/opt/arcsight-database**
  - To install the database: **/opt/vertica**
- If you must use proxy in your environment, you must use the manual deployment process.
- If your network is already using the subnets defined for the default CDF subnets, we recommend using the manual deployment process. In this way, you can configure CDF to use a different subnet.

Prerequisites

Complete the following tasks before you proceed to deploy Interset:

- Review the deployment architecture to learn about the software components you need to install and configure.
- Ensure that the nodes on which you want to deploy Interset meet the specified software and hardware requirements.
- Download the required software components and the corresponding documents.
- (Conditional) To deploy Interset in an existing cluster, complete the following additional tasks:
  - Review the hardware requirements for Interset and add additional nodes if necessary.
  - Ensure that you have the supported version of the database. For more information about installing or upgrading the database, see the Database Deployment Guide.
  - Ensure that the Kafka scheduler is created for the database to receive data from Transformation Hub. For more information, see "Complete Database Setup" in the Database Deployment Guide.
  - Ensure that SSL connection between the database and Transformation Hub is configured. For more information, see "Configuring the Database with SSL" in the Database Deployment Guide.

Download Installation Files

Before you deploy Interset, download and untar the necessary product installation packages. The installation package also includes the respective signature file for validating that the downloaded software is authentic and not tampered by a third party.

Download the following installation files, corresponding MD5, and signature files:

Files	Description

interset-se-installer-6.1.0.13.tar.gz
• installers
  ◦ cdf-2020.05.00100-2.3.0.7
  ◦ db-installer_3.2.0-4.tar.gz
• suite_images
  ◦ transformationhub-3.3.0.29.tar
  ◦ arcsight-interset-se-6.1.0.29.tar
  ◦ fusion-1.1.0.29.tar
• metadata
  ◦ arcsight-installer-metadata-2.3.0.29.tar
• installer scripts
	Contains the files required for installing and deploying Interset:
	• Contains the following installer files:
	  ◦ CDF installer and CDF core image
	  ◦ Database installer
	• Contains the following image files:
	  ◦ Transformation Hub image
	  ◦ Interset image
	  ◦ Fusion image
	  ◦ Arcsight installer metadata
	• Installer scripts

Interset-6.1.0-license.txt	Legal information

Files	Description
recon-1.0.0.29.tar (Optional)	Recon image
ArcSight-ArcMC-2.9.5.2227.0.bin (Optional)	Installation file for ArcSight Management Center (ArcMC)
ArcSight-8.0.0.8322.0-Connector-Linux64.bin (Linux) ArcSight-8.0.0.8322.0-Connector-Win64.exe (Windows)	SmartConnector installer files

To download and verify the signature of the downloaded files:

1. Log in to the node where you want to install Interset.
2. Change to the directory where you want to download the installer files:

   ```
   cd <download_directory>
   ```

   Note: If you are planning to install Interset by using scripts, use /opt as the download location.

3. Download all the necessary installer files from the Micro Focus Entitlements Portal.
4. (Conditional) To verify the SHA-256 signature of the downloaded files, enter the following command:

   ```
   sha256sum <file_name>; cat <file_name>.sha256
   ```

   The output from each set of compressed installation packages must match their corresponding SHA-256 signatures. If they do not match, download the files again and verify the signature. If the checksum does not match even with the new files, contact Micro Focus Customer Support.

5. (Conditional) To verify the MD5 signature of the downloaded files, enter the following command:

   ```
   md5sum <file_name>; cat <file_name>.md5
   ```

   The output from each set of compressed installation packages must match their corresponding MD5 signatures. If they do not match, download the files again and verify the signature. If the checksum does not match even with the new files, contact Micro Focus Customer Support.

6. To extract the downloaded files, enter the following commands:

   For tar file: tar xvfz <file_name>.tar.gz

   For zip file: unzip <file_name>.zip

Download Product Documentation

Download the following product documentation:

- Database Deployment Guide
- CDF Planning Guide
- Transformation Hub Deployment Guide
- Transformation Hub Administrator's Guide

- Supported SmartConnectors Configuration Guide
- ArcMC Administrator's Guide

Deploy Interset by Using Scripts

You can use the installation scripts in single-node deployments for end-to-end installations starting from configuring prerequisites to completing post-installation configurations. In multi-node deployments, you can use the scripts only for some specific prerequisites and post-installation configurations.

Understanding the Installation Scripts

The installation scripts automatically take care of all the prerequisites, software installations, and post-installation configurations.

Script	Purpose
./prepare-install-single-node-host.sh	Installs all the necessary packages and configures the prerequisites such as adding ports to firewall, installing missing packages and modifying the limits.conf file.
./install-single-node.sh	Installs the database, CDF, Transformation Hub, Fusion, and Interset. To integrate Recon with Interset, install Recon manually as the script does not install it automatically.
./install-single-node-post.sh	Performs post-installation configurations, such as labeling the nodes and scheduling Kafka.

Using the Scripts in Single-node Deployments

Note: Applies only when your deployment does not need high availability.

The installation scripts automatically take care of all the prerequisites, software installations, and post-installation configurations. For deployments with a small workload, the script sets the appropriate configuration settings for the database.

To deploy Interset using scripts:

1. Launch a terminal session and then log in to the master node as the root or as a sudo user.
2. Change to the directory where you downloaded the Interset installer files:

   ```
   cd <download_directory>
   ```

3. Execute the scripts in the following order:
   a. ./prepare-install-single-node-host.sh
   b. ./install-single-node.sh

      When you execute this script, you are prompted to provide your input for the following:
      - License agreement
      - Database username - You can specify either the default username or a new name.

        - Database admin password
        - App admin password
        - Search user password
        - CDF admin password
        - NFS_FOLDER_ROOT - You can specify either the default path or a new path.
        - K8S_HOME - You must specify only the default path.
        - Interset System Admin Email ID

    c. ./install-single-node-post.sh

4. Do the following to ensure that the Kafka topic in Transformation Hub receives events from the SmartConnectors:

    a. Launch the CDF Management Portal on port 5443.

    b. Log in with the following credentials:
    User name: admin
    Password: *<the password you provided during CDF installation>*

    c. Click **Cluster > Dashboard**. The Kubernetes Dashboard is displayed.

    d. Select the **Namespace** and then in **Workloads**, click **Stateful Sets**.

    e. For the **th-kafka** stateful set, click ⋮ and then click **Edit**.

    f. In the **YAML** file, for the **TH_ALLOW_KAFKA_PLAINTEXT** name, ensure that the value is set to **'true'**.

    g. Click **Update**.

5. Secure NFS.
6. (Conditional) To scale out the cluster, add more worker nodes to it.
7. Create the System Admin user.
8. Configure SmartConnectors for data collection. For more information, see *SmartConnector User Guide* and *SmartConnector Configuration Guides.*

Using the Scripts in Multi-node Deployments

For multi-node deployments, you can use some of the scripts to make your tasks easier, and complete the rest of the configurations and installations manually. For more information, see the "Leveraging the Installation Scripts for a Manual Deployment" section in Deploy Interset Manually.

Deploy Interset Manually

In deployments with a larger workload where high availability is mandatory, you must manually perform all the necessary system configurations and software installations. This section gives information about deploying Interset manually. Topics include:

- Leveraging the Installation Scripts for a Manual Deployment
- Deploy Interset in a New Cluster for a Single-node Deployment
- Deploy Interset in a New Cluster for a Multi-node Deployment
- Deploy Interset in an Existing Cluster

Leveraging the Installation Scripts for a Manual Deployment

For single-node and multi-node deployments where you want to deploy Interset in a new cluster manually, you can use some of the installation scripts to make your tasks easier, then complete the rest of the configurations and installations manually.

The scripts use default paths for several components. If you plan to use non-default paths, some of the scripts listed in this section might not work as expected. You can use the following scripts:

prepare-install-single-node-host.sh

Applies only for single-node deployments

Applies only if the host meets the prerequisites.

prereq_* scripts

Applies for both single-node and multi-node deployments.

Designed as an alternative to the single-node deployments with small workloads. You can edit specific scripts based on your requirements, then use them individually to automate some of the prerequisites.

You can find these scripts in the scripts sub-folder.

Script	Purpose
prereq_sysctl_conf.sh prereq_rc_local.sh	Sets system parameters (Network Bridging).
prereq_1_required_packages.sh	Installs the required Operating System packages.
prereq_synchronize_time.sh	Synchronizes time of all the nodes in the cluster.
prereq_firewall.sh	Configures the firewall. The script does not enable the masquerade settings after configuring the firewall. You must enable the settings manually.
prereq_disable_ipv6.sh	Configures DNS.

install-single-node-post.sh

Applies only for single-node deployments.

You can use this script only in the following scenarios:

- You have completed the prerequisites.
- The database and CDF are installed on the same node.

postinstall_adjust_flannel_mem.sh

Applies for both single-node and multi-node deployments

Applies the flannel memory settings post-installation.

You can use this script in scenarios where you cannot use the **install-single-node-post.sh** script.

postinstall_label_master_node.sh

Applies only for single-node deployments.

Sets all labels.

/uninstall-single-node.sh

Applies only for single-node deployments.

Uninstalls the database, CDF, Transformation Hub, Fusion, and Interset.

Deploy Interset in a New Cluster for a Single-node Deployment

Perform the following tasks to deploy Interset and the necessary software in a new cluster for a single-node deployment:

1. Install the database. For more information, see "Installing the Database" in the Database Deployment Guide. *The prerequisites for the database installation do not apply if you used the /prepare-install-single-node-host.sh script.*
2. Prepare your environment for CDF. For more information, see *CDF Planning Guide*. *This task does not apply if you used the /prepare-install-single-node-host.sh script.*
3. Install CDF, Transformation Hub, Fusion, and Interset. For information about installing CDF and Transformation Hub, see *Transformation Hub Deployment Guide*.

   While performing the installation tasks for CDF and Transformation Hub, ensure that you also perform the following tasks at the relevant stages to install Fusion, Recon (Optional), and Interset along with other components:

   a. On the **Capabilities** page, when selecting the products to deploy, select **Fusion**, **ArcSight Recon** (Optional), and **ArcSight Interset Standard Edition** in addition to **Transformation Hub**.

   b. When uploading the Transformation Hub image to the local registry, upload the Fusion, Recon (Optional), and Interset images as well.

   c. On the **Configure/Deploy** page, ensure the following:

      - In the **Transformation Hub** tab, ensure that **# of CEF-to-Avro Stream Processor instances to start** is set to 1.
      - In the **Fusion** tab, specify the database connection details in the **Database Con-**

figuration section.

- For the **Interset** tab, verify that the values are auto-populated.

d. Configure the database with HDFS. *This task does not apply if you used the /install-single-node-post.sh script .*

e. In addition to labeling nodes for Transformation Hub, you must also label nodes for Fusion and Interset. For more information, see Label the Nodes. *This task does not apply if you used the /install-single-node-post.sh script or the postinstall_label_master_node.sh script.*

f. Secure NFS.

g. Create the System Admin user.

4. Configure the database with Transformation Hub for the database to read events from Transformation Hub. For more information, see "Complete the Database Setup" in the Database Deployment Guide. *This task does not apply if you used the /install-single-node-post.sh script.*
5. Configure SSL connection between the database and Transformation Hub. For more information, see "Configuring the Database with SSL" in the Database Deployment Guide. *This task does not apply if you used the /install-single-node-post.sh script*
6. Configure SmartConnectors for data collection. For more information, see *SmartConnector User Guide*, *SmartConnector Configuration Guides*, and *Transformation Hub Administration Guide*.
7. (Optional) Install and configure ArcMC to manage Transformation Hub and SmartConnectors. For more information, see *ArcMC Administrator's Guide*.
8. (Conditional) To scale out the cluster, add more worker nodes to it.

Deploy Interset in a New Cluster for a Multi-node Deployment

Perform the following tasks to deploy Interset and the necessary software in a new cluster for a multi-node deployment:

1. Install the database. For more information, see "Installing the Database" in the Database Deployment Guide.
2. Prepare your environment for CDF. For more information, see *CDF Planning Guide*. *You can also use the prereq_* scripts to automate some of the tasks involved in preparing your environment for CDF.*
3. Install CDF, Transformation Hub, Fusion, and Interset. For information about installing CDF and Transformation Hub, see *Transformation Hub Deployment Guide*.

   While performing the installation tasks for CDF and Transformation Hub, ensure that you also perform the following tasks at the relevant stages to install Fusion, Recon (Optional), and Interset along with other components:

   a. On the **Capabilities** page, when selecting the products to deploy, select **Fusion**, **ArcSight Recon** (Optional), and **ArcSight Interset Standard Edition** in addition to Transformation Hub.

   b. When uploading the Transformation Hub image to the local registry, upload the Fusion, Recon (Optional), and Interset images as well.

c. On the **Configure/Deploy** page, ensure the following:

- In the **Transformation Hub** tab, ensure that **# of CEF-to-Avro Stream Processor instances to start** is set to 1.
- In the **Fusion** tab, specify the database connection details in the **Database Configuration** section.
- For the **Interset** tab, read the tooltips carefully and set the desired values accordingly.

> **Important**: For **Analytics Configuration-Spark**, set the values based on the data load. For information about the values for Spark, see the "Hardware Requirements" section in System Requirements.

d. Configure the database with HDFS.

e. In addition to labeling nodes for Transformation Hub, you must also label nodes for Fusion and Interset. For more information, see Label the Nodes.

f. Secure NFS.

g. Create the System Admin user.

4. Configure the database with Transformation Hub for the database to read events from Transformation Hub. For more information, see "Complete Database Setup" in the Database Deployment Guide.
5. Configure SSL connection between the database and Transformation Hub. For more information, see "Configuring the Database with SSL" in the Database Deployment Guide.
6. Configure SmartConnectors for data collection. For more information, see *SmartConnector User Guide*, *SmartConnector Configuration Guides*, and *Transformation Hub Administration Guide*.
7. (Optional) Install and configure ArcMC to manage Transformation Hub and SmartConnectors. For more information, see *ArcMC Administrator's Guide*.
8. (Conditional) To scale out the cluster, add more worker nodes to it.

Deploy Interset in an Existing Cluster

If you already have an existing CDF cluster for Transformation Hub, you can deploy Interset to the same cluster. Reusing existing clusters would reduce costs and system management effort compared to deploying these software in a new cluster.

This section provides information about deploying Interset in an existing CDF deployment.

To deploy Interset in an existing CDF deployment:

1. Launch a terminal session and then log in to the master node as the root or as a sudo user.
2. Change to the following directory:

```
cd /[cdf_installer_directory]/kubernetes/scripts/
```

```
For example: cd /opt/arcsight/kubernetes/scripts/
```

3. (Conditional) If your deployment does not have Fusion, upload the Fusion image to the local registry:

   ```
   ./uploadimages.sh -d <download_directory>/fusion-x.x.x.x -u registry-admin -p '<cdf_password>'
   ```

4. (Optional) Upload the Recon image to the local registry:

   ```
   ./uploadimages.sh -d <download_directory>/recon-x.x.x.x -u registry-admin -p '<cdf_password>'
   ```

5. Upload the Interset image to the local registry:

   ```
   ./uploadimages.sh -d <download_directory>/arcsight-interset-se-x.x.x.x -u registry-admin -p '<cdf_password>'
   ```

6. Launch the CDF Management Portal on port 5443.
7. Log in with the following credentials:

   User name: admin

   Password: <*the password you provided during CDF installation*>

8. Click ⋮ and then click **Change**.
9. On the **Capabilities** page, select the following options:
   - **Fusion**
   - **ArcSight Recon** (Optional)
   - **ArcSight Interset Standard Edition**.
10. Click **Next** until you reach the **Configure/Deploy** page.
11. (Conditional) If your deployment did not have the database and you installed it, then you must configure the database connection details as follows:

    a. Click **Fusion**.

    b. Specify the database connection details in the **Database Configuration** section.

    c. Click **Save**.

12. In the **Interset** tab, read the tooltips carefully and set the desired values accordingly.

    > ⚠ **Important:**
    >
    > - For **Interset System Admin Email ID**, ensure that you specify the email ID of an existing System Admin user in Security, Risk & Governance. This user will be the default System Admin user of Interset.
    > - For **Analytics Configuration-Spark**, set the values based on the data load. For information about the values for Spark, see the "Hardware Requirements" section in System Requirements.

13. Click **Next**. On the **Configuration Complete** page, wait until the deployment is complete. The deployment process might take several minutes to complete.

> **Note**: Some of the pods in the **Configuration Complete** page might remain in a *Pending* state until the product labels are applied on worker nodes.

14. Go to the CDF Management portal page again.
15. Click ⋮ and then click **Reconfigure**.
16. In the **Transformation Hub** tab, ensure that **# of CEF-to-Avro Stream Processor instances to start** is set to 1.
17. Continue with configuring the Database with HDFS.

Configure the Database with HDFS

Note: Does not apply if you used the /install-single-node-post.sh script.

After deploying Interset, you must configure the database with HDFS for the database to receive the Interset Analytics results data from Spark through HDFS.

To configure the database with HDFS:

1. Launch a terminal session and log in to a Kubernetes node as a root user.
2. Execute the following command to retrieve the namespace:

   ```
   export NS=$(kubectl get namespaces |grep arcsight|cut -d ' ' -f1)
   ```

3. Execute the following command to retrieve the RPC port and the Web port:

   ```
   kubectl -n $NS get svc |grep hdfs-namenode
   ```

   An example of the output is:

   ```
   hdfs-namenode-svc    ClusterIP      None       <none>            30820/TCP,30070/TCP            4h32m
   ```

   The first TCP port number (30820) is of the RPC port and the second TCP port number (30070) is of the Web port.

4. Log in to a database node as a root user.
5. Create the **/etc/hadoop/conf/** directory if it does not already exist.
6. Create the **core-site.xml** file if it does not already exist, then update the **fs.defaultFS** and **dfs.namenode.http-address** properties along with the ports you retrieved in Step 3. Ensure that the NAMENODE_HOST value matches the hostname or IP address you provided in the **HDFS NameNode** field in the **CDF Management Portal > Configure/Deploy > Interset**.

   ```
   cat /etc/hadoop/conf/core-site.xml
   <configuration>
           <property>
                   <name>fs.defaultFS</name>
                   <value>hdfs://<NAMENODE_HOST>:<NAMENODE_RPC_PORT>/</value>
           </property>
           <property>
                   <name>dfs.namenode.http-address</name>
                   <value><NAMENODE_HOST>:<NAMENODE_WEB_PORT></value>
           </property>
                                   </configuration>
   ```

   Example:

   ```
   cat /etc/hadoop/conf/core-site.xml
   <configuration>
           <property>
                   <name>fs.defaultFS</name>
                   <value>hdfs://vlab012345.interset:30820/</value>
           </property>
           <property>
                   <name>dfs.namenode.http-address</name>
                   <value>vlab12345.interset:30070</value>
           </property>
   </configuration>
   ```

7. Create the **hdfs-site.xml** file as follows if it does not already exist:

   ```
   <configuration>
   </configuration>
   ```

8. Repeat Steps 4 to 7 on all database nodes.
9. Verify whether the database and HDFS configuration is successful:
   a. Change to the following directory:

      ```
      cd /opt/vertica/bin/
      ```

   b. Log in as a dbadmin:

      ```
      su dbadmin
      ```

   c. Log in to vsql and specify the password when prompted:

      ```
      vsql
      ```

      ```
      [password prompt]
      ```

   d. (Optional) Clear the cache after configuring the database with HDFS:

      ```
      SELECT CLEAR_HDFS_CACHES();
      ```

   e. Execute the following commands:

      ```
      SELECT VERIFY_HADOOP_CONF_DIR();
      ```

      ```
      SELECT node_name, node_address, export_address FROM nodes;
      ```

      The expected output is:

      ```
      Welcome to vsql, the Vertica Analytic Database interactive terminal.

      Type:  \h or \? for help with vsql commands
             \g or terminate with semicolon to execute query
             \q to quit

      dbadmin=> SELECT VERIFY_HADOOP_CONF_DIR();
                                      VERIFY_HADOOP_CONF_DIR
      -------------------------------------------------------------------------
       Validation Success
      v_investigate_node0001: HadoopConfDir [/etc/hadoop/conf] is valid

       (1 row)

      dbadmin=> SELECT node_name, node_address, export_address FROM nodes;
             node_name       |  node_address  |  export_address
      ----------------------------------------------------------
      v_investigate_node0001 |  <IP1>  |  <IP1>
      v_investigate_node0002 |  <IP2>  |  <IP2>
      v_investigate_node0003 |  <IP3>  |  <IP3>
      (3 rows)
      ```

10. Continue with labeling the nodes.

Label the Nodes

> **Note**: Does not apply if you used the /install-single-node-post.sh script.

Interset is a server-based product that is deployed in a clustered configuration. This means that the software is distributed across multiple machines, where each machine (which can be a physical machine or a virtual machine running on a VM server such as VMware ESX) is called a node. The distribution of load and responsibilities across multiple nodes is what makes the Interset solution a scalable system that can handle large amounts of data: the more nodes in your deployment, the more data Interset can handle.

Labeling a node tells Kubernetes what type of application can run on a specific node. It is a means for identifying application processing and qualifying the application as a candidate to run on a specific host system.

Because Interset is deployed in Kubernetes framework, labeling the nodes enables Kubernetes scheduler to physically deploy the component workloads on the labeled nodes. If a component workload increases, you can scale the component replication and then label more nodes to redistribute the workload.

Labeling is required only for worker nodes and not master nodes.

Understanding Interset Components

Interset installation includes the following components:

Interset Components

The Interset components include:

- **Interset Analytics**

  Performs the vital task of determining individual baselines, and then discovering and ranking deviations from those baselines.
- **Interset UI**

  Provides a rich user interface that allows you to visually explore the Interset Analytics results and raw data in the Interset dashboard.
- **Interset API**

  Provides the REST API from which **Interset UI** reads the Interset Analytics results.
- **Interset Exports**

  Renders PDF reports of organizational risks and the users involved in risky behaviors.
- **Interset-spark-config-server**

  Hosts a file server and provides configuration files for Spark2 to consume.

Third-Party Components

The Interset third-party components include:

- **Elasticsearch**

  Elasticsearch is an open source, broadly-distributable and easily-scalable enterprise-grade search engine. Elasticsearch houses all of the Interset Analytics raw events, and provides all of the data that drives the user interface.

- **Kibana**

  Kibana is an open source data visualization plug-in for Elasticsearch. Kibana serves as the user interface and data exploration mechanism for Elasticsearch.

- **Logstash**

  Logstash is an open source data collection tool that collects raw events from Transformation Hub and sends the raw events to Elasticsearch for indexing.

- **H2**

  H2 is an open source, in memory relational database. H2 is used to store user identities to authenticate and authorize users.

- **Apache HDFS**

  The Hadoop Distributed File System (HDFS) is a distributed file system that provides high throughput access to application data. HDFS stores all the Interset Analytics results initially before transferring them to the database.

- **Apache Spark2**

  Spark2 is a fast, general computing engine for Hadoop data. Spark2 executes the Analytics, providing a simple and expressive programming model to support a wide range of applications, including ETL, machine learning, stream processing, and graph computation. The Interset Analytics container launches Spark2 when Analytics is run.

Labeling Nodes

The following table shows the labels to use for each of the components on the worker nodes:

Components	Hardware Resource Usage	Label
Interset Analytics	Very low	interset:yes
Interset UI	Very low	interset:yes
Interset API	Low	interset:yes
Interset Exports	Low	interset:yes
Interset-spark-config-file-server	Very low	interset:yes
Elasticsearch	Low to Moderate for Elasticsearch Master Moderate to High for Elasticsearch Data	interset:yes
Kibana	Very low	interset:yes
Logstash	Very low	interset:yes
H2	Very low	interset:yes

Components	Hardware Resource Usage	Label
The HDFS NameNode stores the location of all the HDFS files distributed across the cluster.	Very low	interset-namenode:yes
The HDFS DataNodes contain blocks of HDFS files.	Moderate to High	interset-datanode:yes
Spark2	Moderate to High	interset-spark:yes
Fusion	Low	fusion:yes
Recon (Conditional)	Low	fusion:yes

Ensure the following when labeling the worker nodes:

- Label only the worker nodes that do not have labels related to Transformation Hub.
- For HDFS NameNode, label only one worker node . The worker node must match the hostname or IP address you provided in the **HDFS NameNode** field in the **CDF Management Portal > Configure/Deploy** page **> Interset**.
- For HDFS DataNodes, label the worker nodes in either of the following ways:
  - Label the worker nodes where the database is installed (recommended option).
  - Label the worker nodes where the network connection between the database nodes and the HDFS DataNodes is the fastest. This ensures that the latency is minimal and the system performance is optimal.
- For Spark2, label the worker nodes that have the interset-datanode:yes label.

The following steps label the nodes manually for both single-node and multi-node deployment. However, for a single-node deployment, you can alternatively use the *postinstall_label_master_node.sh* script to label the node automatically.

To label your worker nodes:

1. Launch the CDF Management Portal on port 5443.
2. Log in with the following credentials:

   User name: admin

   Password: <*the password you provided during CDF installation*>

3. Click **Cluster > Nodes**.
4. In **Predefined Labels**, specify the label **interset-namenode:yes** (case-sensitive) in the text box and click the + icon.
5. Repeat the previous step for each of the following labels to add them to the list of predefined labels. Labels are case-sensitive. Enter the text as shown below.
   interset-datanode:yes
   interset-spark:yes
   interset:yes

   If Fusion label not added already, add **fusion:yes**.

6. Drag and drop each of the labels you added to the corresponding worker nodes based on your workload sharing configuration. The corresponding components get deployed on the corresponding worker nodes.
7. Click **Refresh** to see the labels you applied to the nodes.
   After labeling the nodes, the pods in the **Configuration Complete** page change from the *Pending* state to the *Running* state.
8. Verify that all the pods are in the *Running* state:
   a. Launch a terminal session and log in to the master node as the root user.
   b. Execute the following commands:

      ```
      export NS=$(kubectl get namespaces |grep arcsight|cut -d ' ' -f1)
      kubectl -n $NS get pods --all-namespaces -o wide
      ```
9. (Conditional) To scale out the cluster, add more worker nodes to it.
10. Continue with securing NFS.

Secure NFS

You must secure the NFS shared directories from external access. This section provides one method for ensuring security while maintaining access to master and worker nodes in the cluster. However, you can use a different approach to adequately secure NFS.

1. Log in to the master node as root user.
2. Remove the firewall definition for all NFS ports:

   ```
   NFS_PORTS=('111/tcp' '111/udp' '2049/tcp' '20048/tcp')
   for port in "${NFS_PORTS[@]}"; do firewall-cmd --permanent --remove-port $port; done;
   ```

3. (Conditional) If you have installed Interset by using scripts, remove all rich rules:

   ```
   firewall-cmd --list-rich-rules |xargs -I '{}' firewall-cmd --permanent --remove-rich-rule '{}'
   ```

4. (Conditional) If you want to expose NFS shares to other hosts such as other master and worker node, execute the following:

   ```
   firewall-cmd --add-source="IP_address or CIDR expression of host or hosts" --zone="trusted" --permanent
   ```

5. Reload the new firewall configuration:

   ```
   firewall-cmd --reload
   ```

6. Restart the nginx pod to apply the new firewall configuration:

   ```
   SUITE_NAMESPACE=$(kubectl get namespaces |grep arcsight|cut -d ' ' -f1)
   kubectl delete pod --namespace=$SUITE_NAMESPACE -l app=nginx-ingress-lb
   ```

7. Continue with one of the following:
   - If you have deployed Interset in a new cluster, then continue with creating the System Admin user.
   - If you have deployed Interset in an existing cluster, then continue with adding users as the default System Admin user of Interset. For more information, see the "Manage Users" section in *Interset 6.1.0 Administrator's and User's Guide for CDF*.

Create the System Admin User

To create a user in the System Admin role:

1. Open a certified web browser.
2. Specify the following URL to log in to Security, Risk & Governance:

   https://<cdf_masternode_hostname>/mgmt
3. Specify the required information to create a System Admin user. For the **Email** field, add the Interset System Admin Email ID you specified during the installation in the **Configure/Deploy** page > **Interset** > **Interset System Admin Email ID** field or when you used the *./install-single-node.sh* script.

   After the account is created, you will be prompted to log in with the credentials you just created.
4. (Optional) Log in to Security, Risk & Governance with the Email ID and password you just created.
5. Specify the following URL in a new tab to launch Interset:

   https://<cdf_masternode_hostname>/interset
6. Use the same credentials to log in to Interset. You are now the default System Admin (root) user of Interset.
7. Continue with adding users. For more information, see the *Interset 6.1.0 Administrator's and User's Guide for CDF*.

Add Additional Worker Nodes to the Cluster

To scale out the cluster for increased events processing and analytics computing power, you can add worker nodes to it. You can add the worker nodes either before deploying Interset or after deploying it.

> **Note**: If you are yet to deploy Interset and need to add additional worker nodes, then consider the following:
>
> - When deploying Interset in a new cluster, add the worker nodes during the deployment. The following procedure is not applicable.
> - When deploying Interset in an existing cluster, add the worker nodes before starting the deployment.

To add worker nodes to a cluster:

1. Launch the CDF Management Portal on port 5443.
2. Log in with the following credentials:

   User name: admin

   Password: <*the password you provided during CDF installation*>
3. Click **Cluster > Nodes**.
4. Click + ADD.
5. In the **Add Worker Node** dialog box, specify the required configuration information and click **ADD**.
6. Repeat Steps 4 and 5 to add more worker nodes.
7. (Conditional) If you are yet to deploy Interset in an existing cluster, skip the remaining steps and proceed to the "Deploy Interset in an Existing Cluster" section in Deploy Interset Manually.
8. In **Predefined Labels**, specify a label in the text box and click the + icon. Repeat this step to add more labels.
   For more information on labeling nodes for Interset, see Label the Nodes.
   For more information on labeling nodes for Transformation Hub, see *Transformation Hub Deployment Guide*.
9. Drag and drop each of the labels you added to the corresponding worker nodes based on your workload sharing configuration. The corresponding components get deployed on the corresponding worker nodes.
10. Click **Refresh** to see the labels you applied to the nodes.
11. Click ⋮ and then click **Reconfigure**.
12. (Conditional) If you have assigned labels related to **Transformation Hub**, click **Transformation Hub** and reconfigure the properties. For more information on setting the values, see *Transformation Hub Deployment Guide*.
13. Click **Save**.
14. Click **Interset** and reconfigure the properties.

15. Click **Save**.
16. Verify that all the pods are in the *Running* state:
    a. Launch a terminal session and log in to the master node as the root user.
    b. Execute the following command:

    ```
    kubectl get pods --all-namespaces -o wide
    ```

Appendix A: Reinstall CDF

> **Note**: Does not apply if you used the /uninstall-single-node.sh script.

If you uninstalled CDF and plan to reinstall CDF and Interset in the same cluster, perform the following steps before reinstalling CDF and Interset:

1. Launch a terminal session and as a root user, log in to the node where NFS is present.
2. Delete the NFS directory:

   ```
   rm -rf /<nfs directory path>/nfs
   ```

3. Launch a terminal session and then log in to the node where the Kubernetes hostpath is present.
4. Navigate to the following directory:

   ```
   cd /opt/arcsight/
   ```

5. Delete the following directory:

   ```
   rm -rf k8s-hostpath-volume
   ```

6. Repeat Step 4 and Step 5 on all CDF worker nodes.
7. Launch a terminal session and then log in to a database node.
8. Navigate to the following directory:

   ```
   cd /[database_install_directory]/
   ```

9. Stop the Kafka Scheduler:

   ```
    ./kafka_scheduler stop
   ```

10. As a dbadmin user, do the following:

    a. Run the following command and enter your dbadmin password:

    ```
    /opt/vertica/bin/vsql
    Password:<password>
    ```

    b. Execute the following command to delete the data in the investigation.events table:

    ```
    DELETE FROM investigation.events;
    ```

    c. Execute the following command to delete the UEBA schema:

    ```
    drop schema UEBA cascade;
    ```

11. Continue with reinstalling CDF and deploying Interset. Do one of the following:

    a. Deploy Interset manually. For more information, see Deploy Interset Manually.

    b. Deploy Interset using scripts. For more information, see Deploy Interset by Using Scripts.

Appendix B: Restart Nodes in the Cluster

If you need to restart or shut down any node in the Interset cluster, you must stop the Kubernetes and the databases services running on the node. If you do not stop the services running on the node, the database on the node might get corrupted and the Kubernetes pods will not start after the restart.

You can restart the nodes in any of the following ways:

- Restart Nodes by Using Scripts
- Restart Nodes Manually

Restart Nodes by Using Scripts

> **Note**: Applies only if you have installed Interset by using scripts.

1. Log in to the node you need to restart.
2. Execute the following command:

   ```
   /opt/interset/bin/single-node-util.sh reboot_node
   ```

3. (Conditional) If restart fails, perform a hard reboot of the node.
4. After the node restarts, you must start the database services:

   ```
   /opt/interset/bin/single-node-util.sh start
   ```

Restart Nodes Manually

> **Note**: Applies only if you have installed Interset manually.

1. (Conditional) If the node contains CDF, do the following:
   a. Log in to the node you need to restart as the root user.
   b. Change to the following directory:

      ```
      cd <K8S_HOME>/bin/
      ```

      ```
      For example: /opt/arcsight/kubernetes/bin
      ```

   c. Execute the following command to stop the Kubernetes services:

      ```
      kube-stop.sh
      ```

   d. Execute the following command to unmount Kubernetes volumes:

      ```
      kubelet-umount-action.sh
      ```

2. (Conditional) If the node contains the database, do the following:

   a. Log in to the node as a database administrator.

   b. Execute the following command to stop the database services:

   ```
   /opt/vertica/bin/admintools -t stop_db -p <database_password> -d investigate --force
   ```

3. Restart the node:

   ```
   reboot
   ```

4. (Conditional) If restart fails, perform a hard reboot of the node.

5. (Conditional) After the node restarts, do the following if the node contains the database:

   a. Log in to the node as a database administrator.

   b. Execute the following command to start the database services:

   ```
   /opt/vertica/bin/admintools -t start_db -p <database_password> -d investigate --force
   ```

6. (Conditional) After the node restarts, do the following if the node contains CDF:

   a. Log in to the node as root.

   b. Change to the following directory:

   ```
   cd <K8S_HOME>/bin/
   ```

   ```
   For example: /opt/arcsight/kubernetes/bin
   ```

   c. Check whether all Kubernetes services are running:

   ```
   kube-status.sh
   ```

   d. (Conditional) If any of the services is not running, start the service:

   ```
   kube-start.sh
   ```

Appendix C: Configure Flannel Memory

> **Note**: Applies only if you installed Interset in a new cluster. Does not apply if you used the ./install-single-node-post.sh installation script, which automatically performed this configuration.

In some cases, after flannel pods have been running continuously for some time, the Kafka Manager pod (and others) might terminate abruptly. To prevent this issue, you must modify the flannel file.

You can either use the *postinstall_adjust_flannel_mem.sh* script automatically or you can configure the flannel memory manually:

1. Back up the existing **yaml** file. Execute the following commands:

   ```
   cp $
   {K8S_HOME}/objectdefs/flannel.yaml ${K8S_HOME}
   /objectdefs/flannel.yaml.orig
   ```

2. Modify the flannel **yaml** file. Do one of the following:

   - In the **vi ${K8S_HOME}/objectdefs/flannel.yaml** file, change both the request and limits memory to 250Mi.
   - Execute the command: sed -i s/50Mi/250Mi/g ${K8S_HOME}/objectdefs/flannel.yaml.

3. Delete the existing flannel file. Execute the following command:

   ```
   kubectl delete -f ${K8S_HOME}/objectdefs/flannel.yaml
   ```

4. Create a new flannel **yaml** file. Execute the following command:

   ```
   kubectl create -f ${K8S_HOME}/objectdefs/flannel.yaml
   ```

5. Verify the change. Execute the following command on each flannel pod:

   ```
   kubectl get pod $f_pod -n kube-system -o yaml | grep -A6 resources|grep memory
   ```

6. Ensure that the memory value is set to 250Mi.

Appendix D: Collect Diagnostic Logs

Diagnostic log files help in investigating and troubleshooting issues. You can collect diagnostic logs from Interset, Transformation Hub, and operating system.

To collect the logs:

1. Log in to the master node as root user.
2. Change to the directory where Interset is installed:

   ```
   cd /opt/interset
   ```

   Alternatively, if you have installed manually, you can find the support_utils script in the location where you extracted the Interset installer. For example, /opt/interset-se-installer-x.x.x.x.
3. Execute the script to generate logs:

   ```
   ./support_utils.sh
   ```

4. If you want to collect the operating system logs, specify Y to install the sos package. Otherwise, specify N.
   The sos package is required to generate the operating system logs. Installation of the sos package is a onetime activity.
5. Specify the password to encrypt the output file.
   The encrypted log file is stored in the location:

   ```
   /opt/support_util/<ddmmyyyyhhmmss>
   ```

   For example:

   ```
   /opt/support_util/20200707043015
   ```

6. To view the logs, you must decrypt the file as follows:
   1. Change to the directory where the log file is stored.

      For example:

      ```
      cd /opt/support_util/20200707043015
      ```

   2. Execute the following command:

      ```
      dd if=<log_file_name> | openssl aes-256-cbc -md sha1 -d -k <Encrypt-Password> | tar zxf -
      ```

      For example:

      ```
      dd if=interset-support-util-20200707043015.aes | openssl aes-256-cbc -md sha1 -d -k <Encrypt-Password> |
      tar zxf -
      ```