Micro Focus Security
ArcSight Vertica

Software Version: 9.2.1

Deployment Guide

Document Release Date: April, 2020
Software Release Date: April, 2020

Legal Notices

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK

https://www.microfocus.com

Copyright Notice

Trademark Notices

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number
- Document Release Date, which changes each time the document is updated
- Software Release Date, which indicates the release date of this version of the software

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

ArcSight Product Documentation on the Micro Focus Security Community

Support

Contact Information

Phone	A list of phone numbers is available on the Technical Support Page: https://softwaresupport.softwaregrp.com/support-contact-information
Support Web Site	https://softwaresupport.softwaregrp.com/
ArcSight Product Documentation	https://community.softwaregrp.com/t5/ArcSight-Product-Documentation/ct-p/productdocs

Contents

Chapter 1: Configuring the Vertica Server and Installing the Database

This chapter provides information about configuring the Vertica server and installing the database.

Note: Before you install Vertica, make sure to estimate the storage needed for the incoming EPS (event per second) and event size, and also to evaluate the retention policy accordingly.

Configuring the Vertica Server

To configure the Vertica server details, please see the Vertica Hardware Guide, and the Vertica System Configuration Task Overview.

The procedure described in this section is a guideline for reference only.

The server configuration is based on an HPE ProLiant DL380 Gen9 server with 48 cores and 128 GB memory.

To avoid performance issues, the Vertica server should be a dedicated server.

Note: Vertica data should be backed-up routinely. For more information, please see "Backing Up and Restoring the Vertica Database" on page 21.

Note: To manage disk usage old Vertica data can be cleaned up, for more information, please see "Enabling the Data Retention Policy on the Vertica Cluster" on page 18.

Note: Vertica cluster status should be monitored constantly, for more information, please see "To monitor the Vertica status./vertica_installer status " on page 6

To configure the Vertica server:

1. Provision the server with at least 2 GB of swap space, running on CentOS 7.6 and 7.7 or RHEL 7.6 and 7.7.

   Note: Vertica 9.2.1 supports ext3, ext4, NFS, and XFS file system. In case pre-check on swap space fails after provisioned 2 GB on swap, provision swap with 2.2 GB should solve the problem.

2. Add the following parameters to `/etc/sysctl.conf`. You must reboot the server for the changes to take effect.

Parameter	Description
`net.core.somaxconn = 1024`	Increases the number of incoming connections
`net.core.wmem_max = 16777216`	Sets the send socket buffer maximum size in bytes
`net.core.rmem_max = 16777216`	Sets the receive socket buffer maximum size in bytes
`net.core.wmem_default = 262144`	Sets the receive socket buffer default size in bytes
`net.core.rmem_default = 262144`	Controls the default size of receive buffers used by sockets
`net.core.netdev_max_backlog = 100000`	Increase the length of the processor input queue
`net.ipv4.tcp_mem = 16777216 16777216 16777216`	
`net.ipv4.tcp_wmem = 8192 262144 8388608`	
`net.ipv4.tcp_rmem = 8192 262144 8388608`	
`net.ipv4.udp_mem = 16777216 16777216 16777216`	
`net.ipv4.udp_rmem_min = 16384`	
`net.ipv4.udp_wmem_min = 16384`	
`vm.swappiness = 1`	Defines the amount and frequency at which the kernel copies RAM contents to a swap space For more information, see Check for Swappiness.

3. Add the following parameters to `/etc/rc.local`. You must reboot the server for the changes to take effect.

Note: The following commands assume that sdb is the data drive(i.e. /opt), and sda is the operating system/catalog drive.

Parameter	Description
`echo deadline > /sys/block/sdb/queue/scheduler`	Resolve FAIL (S0150)
`/sbin/blockdev --setra 8192 /dev/sdb`	Resolve FAIL (S0020) Vertica resides on `/dev/sdb`
`echo always > /sys/kernel/mm/transparent_hugepage/enabled`	
`cpupower frequency-set --governor performance`	Resolve WARN (S0140/S0141) (**CentOS only**)

4. To increase the process limit, add the following to `/etc/security/limits.d/20-nproc.con`:

   ```
   * soft nproc 10240
   * hard nproc 10240
   * soft nofile 65536
   * hard nofile 65536
   * soft core unlimited
   * hard core unlimited
   ```

5. In `/etc/default/grub`, append line `GRUB_CMDLINE_LINUX` with `intel_idle.max_cstate=0 processor.max_cstate=1`. For example:

   ```
   GRUB_CMDLINE_LINUX="vconsole.keymap=us crashkernel=auto
   vconsole.font=latarcyrheb-sun16 rhgb quiet intel_idle.max_cstate=0
   processor.max_cstate=1"
   grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

6. Use `iptables` to disable the firewall **WARN (N0010):**

   ```
   iptables -F
   iptables -t nat -F
   iptables -t mangle -F
   iptables -X
   systemctl mask firewalld
   systemctl disable firewalld
   systemctl stop firewalld
   ```

   For more information, see Firewall Considerations.

Firewall Requirements

Vertica requires several ports to be open on the local network. It is not recommended to place a firewall between nodes (all nodes should be behind a firewall), but if you must use a firewall between nodes, ensure the following ports are available:

Port	Protocol	Service	Notes
22	TCP	sshd	Required by Administration Tools and the Management Console Cluster Installation wizard.
5433	TCP	Vertica	Vertica client (vsql, ODBC, JDBC, etc) port.
5434	TCP	Vertica	Intra- and inter-cluster communication.
5433	UDP	Vertica	Vertica spread monitoring.
5444	TCP	Vertica Management Console	MC-to-node and node-to-node (agent) communications port. See Changing MC or Agent Ports.

Port	Protocol	Service	Notes
5450	TCP	Vertica Management Console	Port used to connect to MC from a web browser and allows communication from nodes to the MC application/web server.
4803	TCP	Spread	Client connections.
4803	UDP	Spread	Daemon to daemon connections.
4804	UDP	Spread	Daemon to daemon connections.
6543	UDP	Spread	Monitor to daemon connection.

7. Set SELinux to permissive mode:

   ```
   In /etc/selinux/config
   SELINUX=permissive
   ```

   For more information, see SELinux Configuration.

8. Configure the BIOS for maximum performance:

   System Configuration > **BIOS/Platform Configuration (RBSU)** > **Power Management** > **HPE Power Profile** > **Maximum Performance**

9. Reboot the system, and then use the `ulimit -a` command to verify that the limits were increased.

Configuring Password-less Communication

This section describes how to configure password-less communication from the node 1 server to all of the node servers in the cluster.

Note: You must repeat the authentication process for all nodes in the cluster.

To configure password-less communication:

1. On the node 1 server, run the `ssh-keygen` command:

   ```
   ssh-keygen -q -t rsa
   ```

2. Copy the key from node 1 to all of the nodes, including node 1, using the node IP address:

   ```
   ssh-copy-id -i ~/.ssh/id_rsa.pub root@11.111.111.111
   ```

   The system displays the key fingerprint and requests to authenticate with the node server.

3. Enter the required credentials for the node.

   The operation is successful when the system displays the following message:

   ```
   Number of key(s) added: 1
   ```

4. To verify successful key installation, run the following command from node 1 to the target node to

verify that node 1 can successfully log in:

```
ssh root@11.111.111.111
```

To Install Vertica

After you configured the Vertica server and enabled password-less SSH access, install the Vertica database.

1. On the Vertica cluster node 1 server, create a folder for the Vertica Vertica database installer script:

   ```
   mkdir $vertica-install-DIR
   ```

   Note: `$vertica-install-DIR` should not be under `/root`.

2. Copy `arcsight-vertica-installer_3.1.0-3.tar.gz` to `$vertica-install-DIR.`
3. Extract the `.tar` file:

   ```
   cd $vertica-install-DIR
   tar xvfz arcsight-vertica-installer_3.1.0-3.tar.gz
   ```

4. Edit the `config/vertica_user.properties` file. The `hosts` and `license` properties are required.

Property	Description
`hosts`	A comma separated list of the Vertica Vertica database servers in IPv4 format (for example, 1.1.1.1, 1.1.1.2, 1.1.1.3) If it is necessary to construct the cluster, avoid using local loopback (localhost, 127.0.0.1, etc.).
`license`	`$path/$license-file` Download the license file from the Software Licenses and Downloads portal, and then edit this parameter to point to the license file. **Note:** Without a valid license, an instant-on license will be applied to build a 3 node Vertica cluster only.
`db_retention_day`	Used for the data retention policy.

5. Install Vertica:

   ```
   ./vertica_installer install
   ```

   When prompted, create the database administrator user and the Investigate search user.

   Vertica now supports multiple users:

   • **Database administrator:** Credentials required to access the Vertica database host to perform database related operations, i.e. setup, configuration, and debugging.

   • **Search user:** Credentials required when configuring Vertica from the ArcSight Installer for Investigate search engine.

• **Ingest user:** Should not be used or changed, this user is internally used for Vertica-scheduler, i.e. ingestion.

For a list of options that you can specify when installing Vertica, see Vertica Installer Options.

6. To monitor the Vertica status

```
./vertica_installer status
```

- **Vertica nodes status:** Ensures all nodes are up
- **Vertica nodes storage status:** Ensures storage is sufficient

Chapter 2: Complete Vertica Setup

Follow the steps below to complete the Vertica Setup.

1. Create the schema:

```
./vertica_installer create-schema
```

2. In order to create the Kafka scheduler, run the below commands:

   - If SSL is disabled:

     ```
     ./sched_ssl_setup --disable-ssl
     ```

   - If SSL is enabled, see "Configuring Vertica SSL " on page 11.

3. Create the Kafka scheduler:

```
./kafka_scheduler create <Transformation_Hub_Node_1_IP>:9092
```

Note: Scheduler will obtain the Transformation Hub node information from kafka manager.

For a list of options that you can specify when installing the scheduler, see Kafka Scheduler Options.

4. Check the Vertica status:

```
./vertica_installer status
```

5. Check the scheduler status, event-copy progress, and messages:

```
./kafka_scheduler status
./kafka_scheduler events
./kafka_scheduler messages
```

Vertica Installer Options

You can specify the following options when installing Vertica. To specify an option, type `./vertica_installer <Option_Name>`.

Option	Description
install	Installs the Vertica database
uninstall	Uninstalls the Vertica database and deletes data and users
create-schema	Creates the database schema for Investigate
delete-schema	Deletes the Investigate database schema

Option	Description
start-db	Starts the Vertica database with the `dba_password` specified in `vertica_credentials.properties`
stop-db	Stops the Vertica database
status	Prints the Vertica cluster status

Kafka Scheduler Options

You can specify the following options when installing the Kafka scheduler. To specify an option, type `./kafka_scheduler <Option_Name>`.

Option	Description
update	Updates the scheduler
start	Starts the scheduler and begins copying data from all registered Kafka brokers
stop	Stops the scheduler and ends copying data from all registered Kafka brokers
delete	Deletes all registered Kafka instances from the scheduler
status	Prints the following information and log status for a running or stopped scheduler: • Current Kafka cluster assigned to the scheduler • Name and Vertica host where the active scheduler is running • Name, Vertica host, and process ID of every running scheduler (active or backup)
events	Prints event copy progress for the scheduler
messages	Prints scheduler messages

Chapter 3: Setting FIPS on Vertica

In order to enable FIPS mode in Investigate we have to set the OS in FIPS mode.

To enable FIPS in the OS

1. Run the below commands:

```
yum install dracut-fips
```

```
yum install dracut-fips-aesni
```

```
rpm -q prelink && sed -i '/^PRELINKING/s,yes,no,' /etc/sysconfig/prelink
```

Ignore the error if prelink was not installed.

```
mv -v /boot/initramfs-$(uname -r).img{,.bak}
```

```
dracut
```

```
grubby --update-kernel=$(grubby --default-kernel) --args=fips=1
```

```
uuid=$(findmnt -no uuid /boot)
```

```
[[ -n $uuid ]] && grubby --update-kernel=$(grubby --default-kernel) \
--args=boot=UUID=${uuid}
```

```
reboot
```

2. To verify if FIPS has been enabled, run the following command:

```
sysctl crypto.fips_enabled
```

Expected Result: `crypto.fips_enabled = 1`

To disable FIPS

1. Run the below commands:

```
yum remove dracut-fips
```

```
dracut --force
```

```
grubby --update-kernel=$(grubby --default-kernel) --remove-args=fips=1
```

```
reboot
```

2. To verify if FIPS has been disabled, run the following command:

```
sysctl crypto.fips_enabled
```

Expected Result: crypto.fips_enabled = 0

Enabling FIPS in Nginx

No user action is required to enable FIPS for Nginx. The Nginx docker container is FIPS enabled by default. The FIPS enabled Nginx server will accept TLS 1.2 connections using FIPS compliant Cipher Suites.

Chapter 4: Configuring Vertica SSL

Certificate Creation:

Create a self-signed CA:

```
openssl req -newkey rsa:4096 -sha256 -keyform PEM -keyout ca.key -x509 \
-days 3650 -outform PEM -out ca.crt \
-subj "/C=US/ST=California/L=Santa Clara/O=Micro Focus/OU=Arcsight/\
CN=RootCA/emailAddress=admin@microfocus.com" -nodes
```

Generate the Certificate for Vertica

1. Create the server key:

```
openssl genrsa -out vertica.key 4096 -nodes -sha256
```

```
Generating RSA private key, 4096 bit long modulus
.........................................................................................................................................++
.........................................................................................................................................++
e is 65537 (0x10001)
```

2. Create Server certificate signing request:

   ```
   openssl req -new -key vertica.key -out vertica.csr \
   -subj "/C=US/ST=California/L=Santa Clara/O=Micro Focus/OU=Arcsight/\
   CN=Vertica/emailAddress=admin@microfocus.com" -nodes -sha256
   ```

3. Sign the Certificate Signing Request with self-signed CA:

   ```
   openssl x509 -req -in vertica.csr -CA ca.crt -CAkey ca.key \
   -CAcreateserial -extensions server -days 3650 -outform PEM -sha256 \
   -out vertica.crt
   ```

   ```
   Signature ok
   subject=/C=US/ST=California/L=Santa Clara/O=Micro
   Focus/OU=Arcsight/CN=FQDN/emailAddress=admin@microfocus.com
   Getting CA Private Key
   ```

Create the Vertica Scheduler Client Certificate

1. Create the certificate key for the Vertica scheduler:

   ```
   openssl genrsa -out scheduler.key 4096
   ```

   Generating RSA private key, 4096 bit long modulus

   ....++

   ....++

   e is 65537 (0x10001)

2. Create the Vertica scheduler client certificate signing request:

   ```
   openssl req -new -key scheduler.key -out scheduler.csr \
   -subj "/C=US/ST=California/L=Santa Clara/O=Micro Focus/OU=Arcsight/\
   CN=Scheduler/emailAddress=admin@microfocus.com" -nodes -sha256
   ```

3. Sign the certificate signing request:

```
openssl x509 -req -in scheduler.csr -CA ca.crt -CAkey ca.key \
-CAcreateserial -extensions client -days 3650 -outform PEM -sha256 \
-out scheduler.crt
```

Signature ok

subject=/C=US/ST=California/L=Santa Clara/O=Micro Focus/OU=Arcsight/CN=scheduler/emailAddress=admin@arcsight.com

Getting CA Private Key

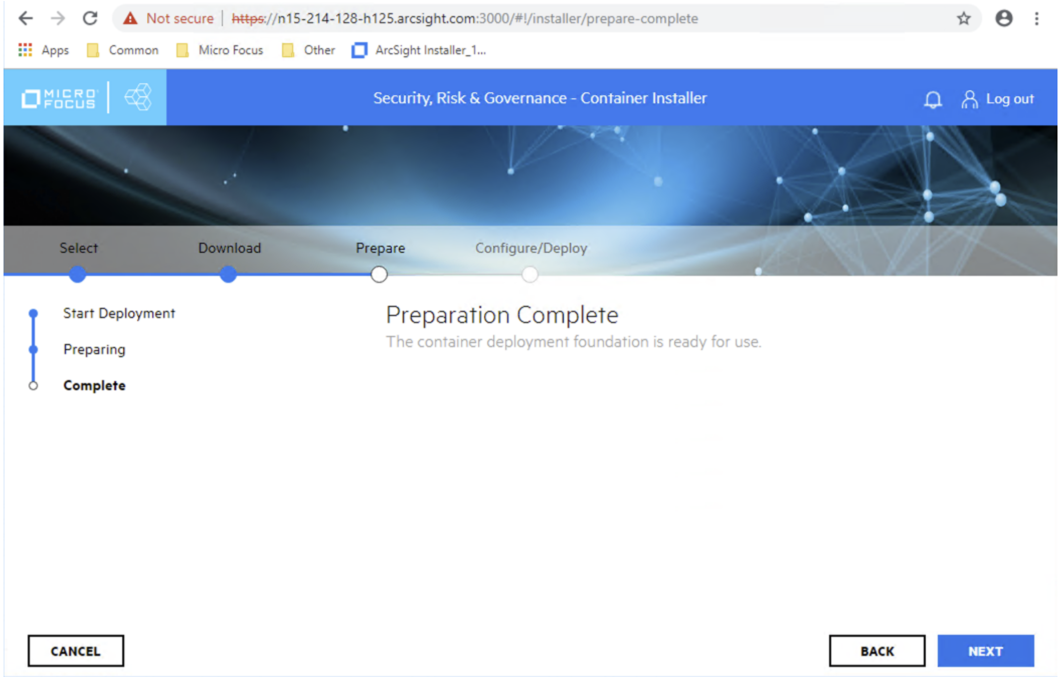
Change the key files permissions

Run the following command:

```
chmod 600 ca.key vertica.key scheduler.key
```

Installing Self-Signed CA during the Transformation Hub Installation

1. Install the Transformation Hub. For more information see the Transformation Hub Deployment guide available from the Micro Focus Community.
2. Access the CDF UI



Preparation Complete

The container deployment foundation is ready for use.

3. After infrastructure services have been deployed, copy the generated ca.crt and ca.key to the Transformation Hub server /tmp directory and Install the self-signed CA

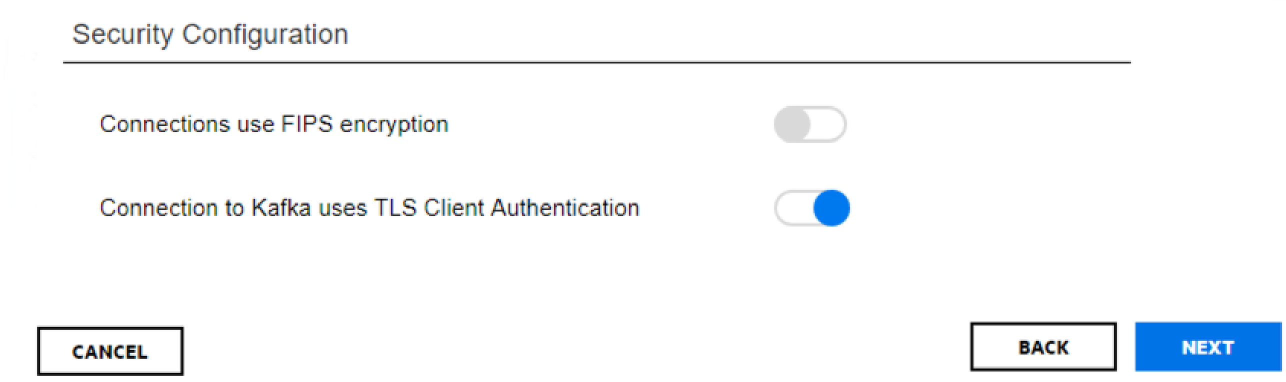
```
/opt/arcsight/kubernetes/scripts/cdf-updateRE.sh write \
--re-key=/tmp/ca.key --re-crt=/tmp/ca.crt
```

```
--------------------------------------------------------------------

Dry run to check the certificate/key files.

Success! Enabled the pki secrets engine at: RE_dryrun/

Success! Data written to: RE_dryrun/config/ca

Success! Disabled the secrets engine (if it existed) at: RE_dryrun/

Dry run succeeded.

Submitting the certificate/key files to platform. CA for external communication will be replaced.

Success! Disabled the secrets engine (if it existed) at: RE/

Success! Enabled the pki secrets engine at: RE/

Success! Data written to: RE/config/ca

Success! Data written to: RE/roles/coretech

Success! Data written to: RE/config/urls

Warning: kubectl apply should be used on resource created by either kubectl create --save-config or kubectl apply

secret/nginx-default-secret configured

configmap/public-ca-certificates patched
```

```
configmap/public-ca-certificates patched
```

4. Proceed with the Transformation Hub installation and into the configuration page

> **Note:** TLS Client Authentication and FIPS need to be enabled at this time. Client Authentication and FIPS cannot be enabled or disabled in the Transformation Hub **Reconfigure** page.



Security Configuration

Connections use FIPS encryption ○

Connection to Kafka uses TLS Client Authentication ●

Enabling Vertica SSL

1. Copy the following files to the Vertica server /tmp directory:
   - vertica.crt
   - vertica.key
   - schedule.crt
   - schedule.key
   - ca.crt
2. Change the certificate key file ownership:

```
chown <dbadmin user> vertica.key scheduler.key
```

3. Enable the Vertica server SSL

```
./vertica_ssl_setup --enable-ssl --vertica-cert-path /tmp/vertica.crt \
--vertica-key-path /tmp/vertica.key --client-ca-path /tmp/ca.crt
```

Verification:

4. Login to vertica server as dbadmin user

```
mkdir ~/.vsql
cp /tmp/scheduler.crt ~/.vsql/client.crt
cp /tmp/scheduler.key ~/.vsql/client.key
cp /tmp/ca.crt ~/.vsql/root.crt
chmod 600 ~/.vsql/client.key
```

5. Login to vertica cluster node 1 as root user:

```
rm -rf /tmp/vertica.crt /tmp/vertica.key /tmp/issue_ca.crt /tmp/ca.crt
```

6. Check the Vertica connection:

```
vsql -m require
```

Password:

Expected result:

```
 SSL connection (cipher: DHE-RSA-AES256-GCM-SHA384, bits: 256, protocol: TLSv1.2)
```

Run the following command:

```
dbadmin=> select user,authentication_method, ssl_state from sessions where session_id = current_session();
```

Expected result:

```
current_user | authentication_method | ssl_state
--------------+-----------------------+-----------
dbadmin | Password | Mutual
(1 row)
```

Enabling SSL in Scheduler

To enable SSL in scheduler, run the following command:

```
./sched_ssl_setup --enable-ssl --sched-cert-path /tmp/scheduler.crt \
--sched-key-path /tmp/scheduler.key --vertica-ca-path /tmp/ca.crt \
--kafka-ca-path /tmp/ca.crt
```

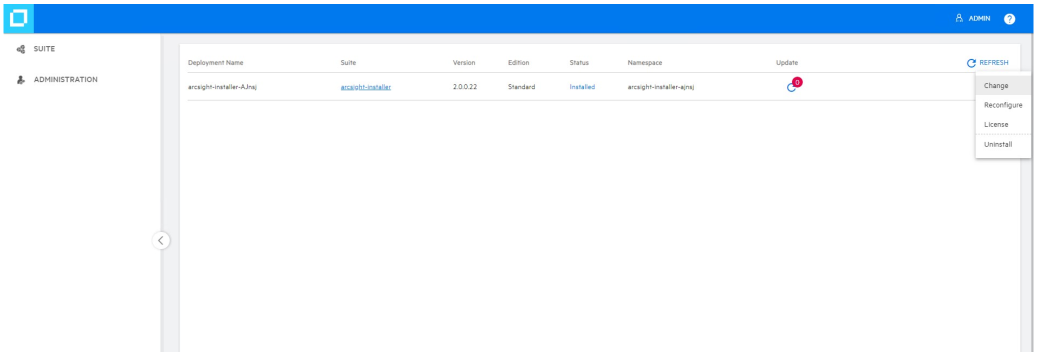
Creating Scheduler with SSL Enabled

To create Scheduler with SSL enabled, run the following command:

```
$vertica-install-DIR/kafka_scheduler create <WorkerNode1>:9093
```

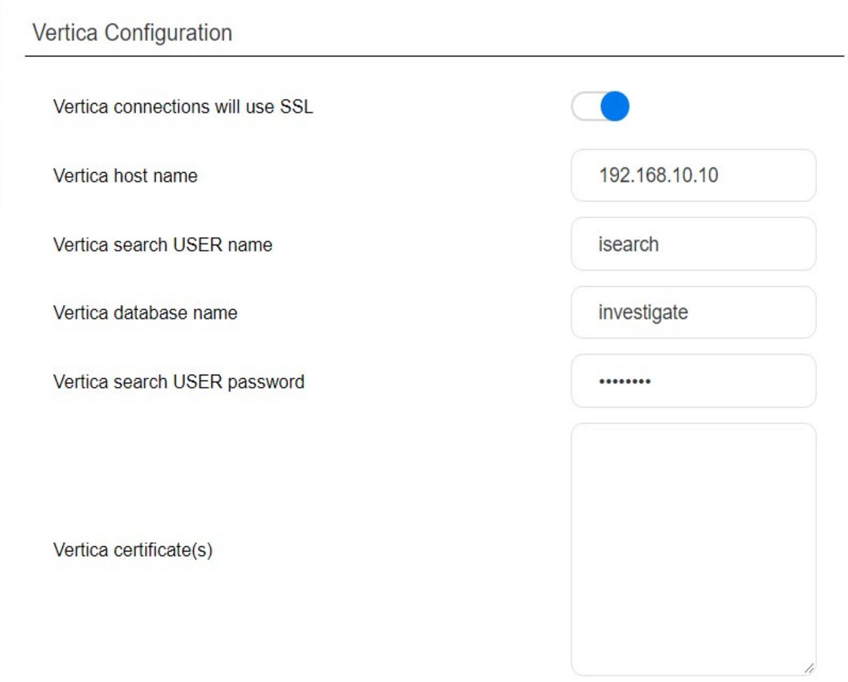
Setting up Investigate with SSL Enabled

1. Browse to https://<virtual-server-FQDN>:5443, if it is a multiple master, or https://<master-FQDN>:5443, if it is a single master.
2. Navigate to suite options: **Suite > Management**
3. Click the **...** icon under **REFRESH** and Select **Reconfigure.** A new tab will be opened.



Deployment Name	Suite	Version	Edition	Status	Namespace	Update
arcsight-installer-AJnsj	arcsight-installer	2.0.0.22	Standard	Installed	arcsight-installer-ajnsj	0

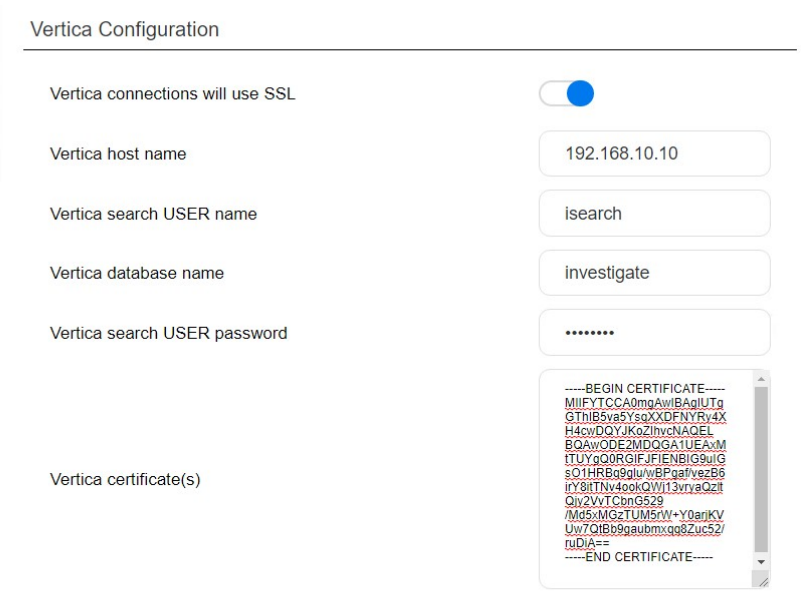
4. Select **ANALYTICS**, and scroll down to **Vertica Configuration**
5. Under **Vertica Configuration**, enable **Vertica connections will use SSL**



Vertica Configuration

Vertica connections will use SSL	●
Vertica host name	192.168.10.10
Vertica search USER name	isearch
Vertica database name	investigate
Vertica search USER password	••••••••
Vertica certificate(s)	

6. Copy the Vertica ca certificate into the **Vertica Certificate(s)** field, make sure not to include any blank spaces or missing line breaks to prevent a handshake authentication failure.



Vertica Configuration

Vertica connections will use SSL	●
Vertica host name	192.168.10.10
Vertica search USER name	isearch
Vertica database name	investigate
Vertica search USER password	••••••••
Vertica certificate(s)	

```
-----BEGIN CERTIFICATE-----
MIIFYTCCA0mgAwIBAgIUTg
GThIB5va5YsqXXDFNYRy4X
H4cwDQYJKoZIhvcNAQEL
BQAwODE2MDQGA1UEAxM
tTUYgQ0RGIFJFIENBIG9uIG
sO1HRBq9glu/wBPgaf/vezB6
irY8itTNv4ookQWj13vryaQzIt
Qjy2VvTCbnG529
/Md5xMGzTUM5rW+Y0arjKV
Uw7QtBb9gaubmxqq8Zuc52/
ruDiA==
-----END CERTIFICATE-----
```

7. Click **SAVE**. This will restart the search engine pod for the SSL changes to take effect

Chapter 5: Enabling the Data Retention Policy on the Vertica Cluster

When Vertica storage approaches usage limits, storage needs to be cleaned up for new events. Data retention script purges old data to reclaim storage.

> **Note:** Storage usage limits are defined by the User.

The retention period can range from 1 to 366 days. The data retention policy is based on calendar days. Calendar day is based on event's Normalized Event Time (NET).

The default data retention period is 90 days. If you run the data retention script on 6/30/2019 and the `db_retention_days property` is set to 90, then data older than 04/01/2019 will be deleted. You can purge data in real time or by using a scheduled cron job. Confirmation is needed when retention period is set to less than 30 days.

> **Note:** Vertica data needs to be backed-up routinely. The backup policy is defined by the user. Always evaluate (-e option) retention policy before purging data.

To enable data retention:

1. Run the following command to check disk usage:

   ```
   cd $vertica-install-DIR
   ./vertica_installer status
   Check the disk_space_free_percent
   ```
2. Back up Vertica data.

   For more information, see "Backing Up the Vertica Database" on page 26.
3. Run the following commands:

   ```
   cd $vertica-install-DIR/config
   vi vertica_user.properties
   ```

   Uncomment `#db_retention_days=90`

4. Verify the number of days of data in the Vertica database:

```
cd $vertica-install-DIR/script
./retention_policy_util.sh -t
```

The result should be similar to the following:

```
--------------------------------------------------------------------------
Investigate has 100 day(s) with time-range: [2017-10-26 - 2018-02-06].
--------------------------------------------------------------------------
```

Note: There are more than 100 calendar days between 2017-10-26 and 2018-02-06. The results above show that there are only 100 event days, meaning that 100 days have incoming events. Certain calendar days did not have incoming events.

5. To change the default retention period, enter the following command:

```
./retention_policy_util.sh -u <Number_of_Days>
```

To purge Vertica data:

1. To create the purge process, enter the following command:

```
./retention_policy_util.sh -s
```

Note: A cron job is scheduled to purge data daily.

2. To verify the created cron job, enter the following command:

```
./retention_policy_util.sh -l
```

Expected results:

```
------------------------------------------------------------------------------------------
Current retention value is set to: 90 day(s)
------------------------------------------------------------------------------------------
Current cronjob is running:
(59 23 * * * /opt/installer/scripts/retention_policy_util.sh -p &>>
/opt/installer/vertica-installer.log)
-------------------------------------------------------------------------------------------
```

3. To preview the purge results, enter the following command:

```
./retention_policy_util.sh -e
```

The results should be similar to the following:

```
***********************************************************************
No data will be purged. This is only evaluation for your retention policy
************************************************************************
```

```
Will purge time range : [ 2017-10-26 - 2017-10-31 ].
Will purge day 1, (2017-10-26)
Will purge day 2, (2017-10-27)
Will purge day 3, (2017-10-28)
Will purge day 4, (2017-10-29)
Will purge day 5, (2017-10-31)
***** done *****
```

4. To purge data in real time, enter the following command:

   ```
   ./retention_policy_util.sh -p
   ```

5. To disable the purge cron job, enter the following command:

   ```
   ./retention_policy_util.sh -d
   ```

6. To verify the disabled cron job, enter the following command:

   ```
   ./retention_policy_util.sh -l
   ```

   Expected results:

   ```
   ------------------------------------------------------------------------------------------
   Current retention value is set to: 90 day(s)
   ------------------------------------------------------------------------------------------
   ```

Chapter 6: Backing Up and Restoring the Vertica Database

You should back up and restore the Vertica database before you upgrade Vertica or before you add or remove a Vertica node.

Consider the following when backing up and restoring the database:

- The backup process can consume additional storage. The amount of space that the backup consumes depends on the size of your catalog and any objects that you drop during the backup. The backup process releases this storage after the backup is complete.
- You can only restore backups to the same version of Vertica. For example, you cannot back up Vertica 9.1.0 and restore it to Vertica 9.2.1.
- Ingesting events into the database during backup might exclude the most recently ingested events from the backup. To ensure that all events are backed up, stop ingestion before you start the backup.
- For optimal network performance, each Vertica node should have its own backup host.
- Use one directory on each Vertica node to store successive backups.
- You can save backups to the local folder on the Vertica node or to a remote server.
- You can perform backups on ext3, ext4, NFS and XFS file systems.

Preparing the Backup Host

Micro Focus recommends that each backup host have space for at least twice the database node footprint size. Consider your long-term backup storage needs.

If you are using a single backup location, you can use the following Vertica operation to estimate the required storage space for the Vertica cluster:

```
dbadmin=> select sum(used_bytes) as total_used_bytes from v_monitor.storage_
containers;

total_used_bytes

------------------

 5717700329

(1 row)
```

If you are using multiple backup locations, one per node, use the following Vertica operation to estimate the required storage space:

```
dbadmin=> select node_name, sum(used_bytes) as total_used_bytes from v_
monitor.storage_containers group by node_name;

 node_name | total_used_bytes

------------------------+---------------------

 v_investigate_node0002 | 1906279083

 v_investigate_node0003 | 1905384292

 v_investigate_node0001 | 1906036954

(3 rows)
```

Remote backup hosts must have SSH access.

The database administrator must have password-less SSH access from Vertica node 1 to the backup hosts, as well as from the restored Vertica node 1.

To set up password-less SSH:

1. Log in to the backup server.
2. Create user `$db_admin`.

   `$db_admin` is the administrator for the Vertica cluster.
3. Ensure that `$db_admin` has write permission to the dedicated directory where you will store the backup.
4. Log in to Vertica node 1 as `root`.
5. Become the Vertica database administrator:

   ```
   # su -l $db_admin
   ```
6. Setup password-less SSH for all backup servers:

   ```
   # ssh-copy-id -i ~/.ssh/id_rsa.pub $db_admin@$back_up_server_ip
   ```

Preparing Backup Configuration File

Vertica includes sample configuration files that you can copy, edit, and deploy for your various ***vbr*** tasks. Vertica automatically installs these files at `/opt/vertica/share/vbr/example_configs.`

For more information, please see: Sample VBR .ini Files.

The default number of restore points (`restorePointLimit`) is 52, assuming a weekly backup for one year. Using multiple restore points gives you the option to recover from one of several backups. For example, if you specify 3, you have 1 current backup and 3 backup archives.

We use `backup_restore_full_external.ini` as an example.

```
# su - idbadmin

# cp /opt/vertica/share/vbr/example_configs/backup_restore_full_external.ini
vertica_backup.ini

# vi vertica_backup.ini
```

> **Note:** You must save a copy of `vertica_backup.ini` for future tasks.

> **Note**: The following is an example for reference only .`v_investigate_node000*` is hard coded. `dbName = investigate` is hard coded.

```
# cat vertica_backup.ini

; This sample vbr configuration file shows full or object backup and restore
to a separate remote backup-host for each respective database host.

; Section headings are enclosed by square brackets.

; Comments have leading semicolons (;) or pound signs (#).

; An equal sign separates options and values.

; Specify arguments marked '!!Mandatory!!' explicitly.

; All commented parameters are set to their default value.

; ------------------------------------------- ;

;;; BASIC PARAMETERS ;;;

; ------------------------------------------- ;

[Mapping]

; !!Mandatory!! This section defines what host and directory will store the
backup for each node.

; node_name = backup_host:backup_dir

; In this "parallel backup" configuration, each node backs up to a distinct
external host.

; To backup all database nodes to a single external host, use that single
hostname/IP address in each entry below.

v_investigate_node0001 = 192.168.1.1:/opt/dbadmin/backups

v_investigate_node0002 = 192.168.1.2:/opt/dbadmin/backups

v_investigate_node0003 = 192.168.1.3:/opt/dbadmin/backups

[Misc]

; !!Recommended!! Snapshot name. Object and full backups should always have
different snapshot names.
```

```
; Backups with the same snapshotName form a time sequence limited by restorePointLimit.
; SnapshotName is used for naming archives in the backup directory, and for monitoring and troubleshooting.
; Valid characters: a-z A-Z 0-9 - _
snapshotName = Vertica_backup_09_09_2019
[Database]
; !!Recommended!! If you have more than one database defined on this Vertica cluster, use this parameter to specify which database to backup/restore.
dbName = investigate
; If this parameter is True, vbr prompts the user for the database password every time.
; If False, specify the location of password config file in 'passwordFile' parameter in [Misc] section.
dbPromptForPassword = True
; ------------------------------------------- ;
;;; ADVANCED PARAMETERS ;;;
; ------------------------------------------- ;
[Misc]
; The temp directory location on all database hosts.
; The directory must be readable and writeable by the dbadmin, and must implement POSIX style fcntl lockf locking.
tempDir = /tmp
; How many times to retry operations if some error occurs.
retryCount = 2
; Specifies the number of seconds to wait between backup retry attempts, if a failure occurs.
retryDelay = 1
; Specifies the number of historical backups to retain in addition to the most recent backup.
; 1 current + n historical backups
restorePointLimit = 52
; Full path to the password configuration file
; Store this file in directory readable only by the dbadmin
; (no default)
; passwordFile = /path/to/vbr/pw.txt
; When enabled, Vertica confirms that the specified backup locations contain
; sufficient free space and inodes to allow a successful backup. If a backup
; location has insufficient resources, Vertica displays an error message
explaining the shortage and
; cancels the backup. If Vertica cannot determine the amount of available
space
; or number of inodes in the backupDir, it displays a warning and continues
; with the backup.
enableFreeSpaceCheck = True
; When performing a backup, replication, or copycluster, specifies the
maximum
; acceptable difference, in seconds, between the current epoch and the backup
epoch.
; If the time between the current epoch and the backup epoch exceeds the
value
; specified in this parameter, Vertica displays an error message.
SnapshotEpochLagFailureThreshold = 3600
[Transmission]
; Specifies the default port number for the rsync protocol.
port_rsync = 50000
; Total bandwidth limit for all backup connections in KBPS, 0 for unlimited.
Vertica distributes
; this bandwidth evenly among the number of connections set in concurrency_
backup.
total_bwlimit_backup = 0
; The maximum number of backup TCP rsync connection threads per node.
; Optimum settings depend on your particular environment.
; For best performance, experiment with values between 2 and 16.
concurrency_backup = 2
; The total bandwidth limit for all restore connections in KBPS, 0 for
unlimited
total_bwlimit_restore = 0
```

```
; The maximum number of restore TCP rsync connection threads per node.

; Optimum settings depend on your particular environment.

; For best performance, experiment with values between 2 and 16.

concurrency_restore = 2

[Database]

; Vertica user name for vbr to connect to the database.

; This setting is rarely needed since dbUser is normally identical to the database administrator

dbUser = $your_db_admin
```

Backing Up the Vertica Database

The `$db_admin` user must perform the backup from the Vertica node 1 of the cluster.

> **Note:** vbr Command Reference.

To back up the database:

1. Stop Vertica scheduler

   Login Vertica node 1 as `root`

   ```
   # cd $vertica-install-DIR
   # ./kafka_scheduler stop
   ```
2. Initialize backup location

   ```
   # su - $db_admin
   # vbr -t init --config-file vertica_backup.ini
   ```
   Initializing backup locations.

   Backup locations initialized.
3. Back up Vertica data:

   ```
   # vbr -t backup -c vertica_backup.ini
   Enter vertica password:
   Starting backup of database investigate.
   Participating nodes: v_investigate_node0001,v_investigate_node0002,v_
   investigate_node0003.
   Snapshotting database.
   Snapshot complete.
   Approximate bytes to copy: 270383427 of 270383427 total.
   [==================================================] 100%
   ```

```
Copying backup metadata.
Finalizing backup.
Backup complete!
```

4. Verify that the backup files were written to the backup locations:

```
# ssh 192.161.1.1 ls /opt/dbadmin/backups
backup_manifest
Objects
Snapshots
# ssh 192.161.1.2 ls /opt/dbadmin/backups
backup_manifest
Objects
Snapshots
# ssh 192.161.1.3 ls /opt/dbadmin/backups
backup_manifest
Objects
Snapshots
```

Backing Up Vertica Incrementally

Incremental backups use the same setup as a full backup and only back up what changed from the previous full backup. When you perform a full backup using the same configuration file, subsequent backups are incremental. When you start an incremental backup, the `vbr` tool displays a backup size that is a portion of the total backup size. This portion represents the delta changes that will be backed up during the incremental backup.

Run the following command to perform an incremental backup:

```
# vbr --task backup --config-file vertica_backup.ini
```

Verifying the Integrity of the Backup

Use the `full-check` option to verify the integrity of the Vertica database backup. The option reports the following:

- Incomplete restore points
- Damaged restore points
- Missing backup files
- Unreferenced files

To verify the backup integrity, run the following command:

```
# vbr --task full-check --config-file vertica_backup.ini

Enter vertica password:

Checking backup consistency.

List all snapshots in backup location:

Snapshot name and restore point: Vertica_backup_09_09_2019_20190909_010826,
nodes:['v_investigate_node0001', 'v_investigate_node0002', 'v_investigate_
node0003'].

Regenerating backup manifest for location rsync://
[192.168.10.11]:50000/opt/dbadmin/backups

Regenerating backup manifest for location rsync://
[192.168.10.12]:50000/opt/dbadmin/backups

Regenerating backup manifest for location rsync://
[192.168.10.13]:50000/opt/dbadmin/backups

Snapshots that have missing objects(hint: use 'vbr --task remove' to delete
these snapshots):

Backup locations have 0 unreferenced objects

Backup locations have 0 missing objects

Backup consistency check complete.
```

Managing Backups

This section describes how to view and delete backups.

To view available backups, run the following command:

```
# vbr --task listbackup --config-file vertica_backup.ini
```

Enter vertica password:

```
backup backup_type epoch objects include_patterns exclude_patterns nodes
(hosts) version file_system_type

Vertica_backup_09_09_2019_20190909_010826 full 6058

 v_investigate_node0001(192.168.10.11), v_investigate_node0002
(192.168.10.12), v_investigate_node0003(192.168.10.13) v9.2.1-6 [Linux]
```

The backup name includes the backup time-stamp.

Backup times-tamp can be found by using listbackup option, i.e. `20190909_010826` from `Vertica_backup_09_09_2019_20190909_010826.`

To delete a backup, run the following command:

```
# vbr --task remove --config-file vertica_backup.ini --archive 20190909_
010826

Enter vertica password:

Removing restore points: 20190909_010826

Remove complete!
```

Restoring Vertica Data

Before you restore Vertica data, ensure that your environment meets the following requirements:

- You can only restore backups to the same version of Vertica from which you made the backup. For example, you cannot backup Vertica 9.1.0 and restore it to Vertica 9.2.1.
- You can restore backup to the original cluster where the backup was generated. However, all data ingested to the Vertica after backup will be lost. If backup is restored to a new cluster, you must restore to a cluster that is identical to the cluster from which you made the backup (same or larger disk size). Ensure that the cluster meets the following requirements:
  - The target database is created and empty.
  - The target database name matches the backup database name.
  - The target database is stopped.

- All Vertica nodes in the target cluster are running.
- All Vertica node names in the target cluster match the names from the backup.

Restoring the Vertica Database

The `$db_admin` user must restore from the Vertica node 1 of the cluster.

To set up password-less SSH:

1. Log in to the target Vertica node 1 as root.

2. Become the Vertica database administrator:

```
# su -l $db_admin
```

3. Setup password-less SSH for all backup servers:

```
# ssh-copy-id -i ~/.ssh/id_rsa.pub $db_admin@$back_up_server_ip
```

To restore the database:

1. Build a target Vertica cluster that is identical to the original cluster.
2. Log in to the target Vertica node 1 and stop the database:

   ```
   # cd $vertica-install-DIR
   # ./vertica_installer stop-db
   ```
3. Become the `$db_admin` user:

   ```
   # su -l $db_admin
   ```
4. Copy `vertica_backup.ini` to `/home/$db_admin`.
5. Restore the backup data:

   ```
   # vbr --task restore --config-file vertica_backup.ini
   ```

   The output should be similar to the following:

   ```
   Enter vertica password:
   Starting full restore of database investigate.
   Participating nodes: v_investigate_node0001, v_investigate_node0002, v_
   investigate_node0003.
   Restoring from restore point: investigate_backup_20190909_010826
   Determining what data to restore from backup.
   [==================================================] 100%
   Approximate bytes to copy: 270383427 of 270383427 total.
   Syncing data from backup to cluster nodes.
   [==================================================] 100%
   ```

```
Restoring catalog.
Restore complete!
```

6. Start the database:

   ```
   # exit
   # ./vertica_installer start-db
   ```

   The output should be similar to the following:

   ```
   Starting nodes:
   v_investigate_node0001 (127.0.0.1)
   Starting Vertica on all nodes. Please wait, databases with a large catalog
   may take a while to initialize.
   Node Status: v_investigate_node0001: (DOWN)
   Node Status: v_investigate_node0001: (DOWN)
   Node Status: v_investigate_node0001: (DOWN)
   Node Status: v_investigate_node0001: (DOWN)
   Node Status: v_investigate_node0001: (UP)
   Database investigate started successfully
   ```

7. Start the Kafka scheduler:

   ```
   # ./kafka_scheduler start
   ```

Chapter 7: Vertica upgrade

Before performing the upgrade

- Stop all investigate operations
- Stop scheduler
- Pause outliers scoring
- Backup the database

Note: The upgrade process is irreversible, make sure to backup the database.

Vertica upgrade steps

- On the Vertica cluster node 1 server, create a folder for the new Investigate Vertica database installer script:

```
mkdir $new-vertica-install-DIR
```

Note: `$new-vertica-install-DIR` should not be under /root.

- Copy arcsight-vertica-installer_3.1.0-3.tar.gz to `$new-vertica-install-DIR.`
- Untar arcsight-vertica-installer_3.1.0-3.tar.gz.

```
tar xvfz arcsight-vertica-installer_3.1.0-3.tar.gz
```

- Run the upgrade command in order

Note: The command execution can't be re-ran.

```
./investigate_upgrade
```

Usage:

Execute the following commands in this order

1. `./investigate_upgrade -c upgrade-investigate`
2. `./investigate_upgrade -c update-configuration`

Options:

-h, --help show this help message and exit

```
-c COMMAND, --command=COMMAND
```

[REQUIRED] specify upgrade command:

['upgrade-investigate', 'update-configuration',

'upgrade-vertica-rpm']

Run as an example: `./investigate_upgrade -c upgrade-investigate`

```
Upgrade related changes cannot be rolled back, do you want to continue with the upgrade (Y/N): y

Starting upgrade...

******************** Start of Investigate Upgrade *****************

Enter previous installed location (/opt/install-vertica):/opt/installer

Running Pre-Upgrade checks

Checking all Vertica nodes are UP

All Vertica nodes are UP

Replacing files in installed location

Upgrading script and config files.

Creating backup directory: /opt/installer/oldVersion

Backing up: /opt/installer/vertica_installer

Backing up: /opt/installer/resources

Backing up: /opt/installer/scripts

Backing up: /opt/installer/data

Backing up: /opt/installer/upgrade

Backing up: /opt/installer/lib

Backing up: /opt/installer/vertica.properties

Backing up: /opt/installer/kafka_scheduler

Backing up: /opt/installer/sched_ssl_setup

Backing up: /opt/installer/vertica_ssl_setup

Backing up: /opt/installer/vertica_upgrade.py

Backing up: /opt/installer/investigate_upgrade

Backing up: /opt/installer/copyright.txt

Upgrading: /opt/installer/vertica_installer

Upgrading: /opt/installer/resources
```

```
Upgrading: /opt/installer/scripts

Upgrading: /opt/installer/data

Upgrading: /opt/installer/upgrade

Upgrading: /opt/installer/lib

Upgrading: /opt/installer/vertica.properties

Upgrading: /opt/installer/kafka_scheduler

Upgrading: /opt/installer/sched_ssl_setup

Upgrading: /opt/installer/vertica_ssl_setup

Upgrading: /opt/installer/vertica_upgrade.py

Upgrading: /opt/installer/investigate_upgrade

Upgrading: /opt/installer/copyright.txt

Upgrading: /opt/installer/vertica-upgrade.log

********* Start of Investigate Upgrade to 3.10.0 *********

Pre Upgrade check for 3.10.0

Current Investigate version is: 3.00.0

Investigate will be upgraded to 3.10.0

Create data quality table and create data quality crontab ...

data quality table has been created successfully.

********************* Investigate Upgraded Complete. Version is 3.10.0 ******************
```

Run as an example: `./investigate_upgrade -c update-configuration`

```
Upgrade related changes cannot be rolled back, do you want to continue with the upgrade (Y/N): y

Starting upgrade...

********************* Start of Configuration Updade ******************

Enter previous installed location (/opt/install-vertica):/opt/installer

Running Pre-Upgrade checks

Checking all Vertica nodes are UP

All Vertica nodes are UP

Grant general resource pool to search user
```

Restart Kafka scheduler,

```
cd $vertica-install-DIR

./kafka_scheduler start

SSL/TLS mode is disabled
Terminating all running scheduler processes for schema: [investigation_scheduler]

scheduler instance(s) deleted for 192.168.100.100

scheduler instance(s) added for 192.168.100.100
```

Note: If Investigate has not been upgraded, continue to upgrade Investigate. If Investigate has been upgraded, resume normal operations.

Appendix G: Fields Indexed by Default in Vertica

Investigate indexes a subset of event fields for use in free form text search. Free form text search can only be done for values in event fields that are indexed. Following is the list of event fields that are indexed by default in Vertica:

- agentDnsDomain
- agentHostName
- agentTranslatedZoneURI
- agentZoneURI
- applicationProtocol
- cryptoSignature
- destinationDnsDomain
- destinationGeoLocationInfo
- destinationHostName
- destinationNtDomain
- destinationProcessName
- destinationServiceName
- destinationTranslatedZoneURI
- destinationUserId
- destinationUserName
- destinationUserPrivileges
- destinationZoneURI
- deviceAction
- deviceAssetId
- deviceCustomDate1Label
- deviceCustomDate2Label
- deviceCustomFloatingPoint1Label
- deviceCustomFloatingPoint2Label
- deviceCustomFloatingPoint3Label
- deviceCustomFloatingPoint4Label
- deviceCustomIPv6Address1Label
- deviceCustomString2Label
- deviceCustomString3
- deviceCustomString3Label
- deviceCustomString4
- deviceCustomString4Label
- deviceCustomString5
- deviceCustomString5Label
- deviceCustomString6
- deviceCustomString6Label
- deviceDnsDomain
- deviceDomain
- deviceEventCategory
- deviceExternalId
- deviceFacility
- deviceHostName
- deviceNtDomain
- devicePayloadId
- deviceProcessName
- deviceProduct
- deviceSeverity
- deviceTranslatedZoneURI
- deviceVendor
- deviceZoneURI
- eventOutcome
- externalId
- fileId
- flexNumber2Label
- flexString1
- flexString1Label
- flexString2
- flexString2Label
- message
- name
- oldFileId
- oldFileName
- oldFilePath
- oldFileType
- rawEvent
- reason
- requestClientApplication
- requestContext
- requestCookies
- requestUrl
- requestUrlFileName
- requestUrlQuery
- sourceDnsDomain
- sourceGeoLocationInfo
- sourceHostName
- sourceNtDomain
- sourceProcessName
- sourceServiceName
- sourceTranslatedZoneURI

deviceCustomIPv6Address2Label	fileName	sourceUserId
deviceCustomIPv6Address3Label	filePath	sourceGeoCountryCode
deviceCustomIPv6Address4Label	fileType	sourceUserName
deviceCustomNumber1Label	flexDate1Label	sourceUserPrivileges
deviceCustomNumber2Label	categoryBehavior	sourceGeoPostalCode
deviceCustomNumber3Label	destinationGeoCountryCode	sourceGeoRegionCode
deviceCustomString1	flexNumber1Label	sourceZoneURI
deviceCustomString1Label	destinationGeoPostalCode	
deviceCustomString2	destinationGeoRegionCode	

If users need to index certain event fields that are not in the list above, they can work with support in editing the **`superschema_vertica.sql`** file in the Vertica installer before installing Vertica.

If users want to modify the event fields indexed after Vertica has been installed, and there are already events in the database, they will need to drop the text index and recreate it. This may take a while depending on how many events are in the system.

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this computer, click the link above and an email window opens with the following information in the subject line:

Feedback on Deployment Guide (Vertica 9.2.1)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to arcsight_doc@microfocus.com.

We appreciate your feedback!