Forcepoint Behavioral Analytics Upgrade Guide

Upgrade Guide | Forcepoint Behavioral Analytics | v3.3 | 20-Dec-2019

These instructions describe how to upgrade from v3.1.2 or v3.1.3 of Forcepoint Behavioral Analytics to v3.3 of Forcepoint Behavioral Analytics.

Preparation for upgrade

1. Stop the nifi service on the nifi server.
   a. Validate nifi is stopped.
2. Copy nifi data to the backup directory.

   ```
   sudo mkdir -p /data/ro-nifi/backup
   sudo cp /data/ro-nifi/configuration_resources/flow.xml.gz /data/ro-nifi/backup/
   sudo cp /data/ro-nifi/nifi/conf/authorizers.xml /data/ro-nifi/backup/
   sudo cp -r /data/ro-nifi/database_repository/ /data/ro-nifi/backup/
   sudo cp -r /data/ro-nifi/content_repository/ /data/ro-nifi/backup/
   sudo cp -r /data/ro-nifi/content_repository/ /data/ro-nifi/backup/
   sudo cp -r /data/ro-nifi/flowfile_repository/ /data/ro-nifi/backup/
   sudo cp -r /data/ro-nifi/provenance_repository/ /data/ro-nifi/backup
   ```
3. Stop ro-conv service on conv server.

   ```
   sudo service ro-conv stop
   or stop
   ```
4. Wait for reveal.internal.event queue to drain.
5. Stop ro-qw service on qw server.

   ```
   sudo service ro-qw stop
   ```
6. Stop ro-ui service on ui server.

```
sudo service ro-ui stop
```

7. Check for elasticsearch repository on mds1.

```
curl -k -u elastic:changeme https://localhost:9200/
_snapshot
```

8. Create an elasticsearch snapshot from mds1 (replace $REPO with the repository from the previous step. Example: default_s3_repository):

```
REPO="default_s3_repository"
```

```
curl -XPUT -k -u elastic:changeme "https://
localhost:9200/_snapshot/$REPO/snapshot_$(date
+%Y%m%d%H%M%S)?wait_for_completion=false"
```

9. Verify the snapshot is complete from mds1.

```
curl -k -u elastic:changeme https://localhost:9200/
_snapshot/$REPO/_all | jq -r '.snapshots'
```

10. Verify green cluster health from mds1.

```
curl -k -u elastic:changeme https://localhost:9200/
_cluster/health | jq -r '.status'
```

11. Clear the analytics cache from mds1.

```
curl -XPOST -k https://localhost:8080/reference/
analytics/clear_cache -f
```

12. Backup the postgresql databases on the postgresql server.

```
pg_dump the_ui --username postgres --create --clean --
verbose --file the_ui_database_backup_file.sql
```

```
pg_dump mds --username postgres --create --clean --
verbose --file mds_database_backup_file.sql
```

```
pg_dump redowl_streaming --username postgres --create --
clean --verbose --file
redowl_streaming_database_backup_file.sql
```

13. Stop ro-content service on the cont server.

```
sudo service ro-content stop
```

14. Remove files from the cont server.

```
sudo rm -rf /var/qtd/*
```

15. Update /etc/ansible/group_vars/all

```
##Remove the '1' from mds_lb: mds1-${ro_unique_name_san}
```

```
mds_lb: mds-${ro_unique_name_san}
```

16. Upda##Update the es nodes

```
[es]
```

```
es1-upg-test1 ansible_host=es1-upg-test1
```

```
es2-upg-test1 ansible_host=es2-upg-test1
```

```
es3-upg-test1 ansible_host=es3-upg-test1
```

```
##Update the logstash nodes
```

```
[logstash]
logstash1-upg-test1 ansible_host=es1-upg-test1
logstash2-upg-test1 ansible_host=es2-upg-test1
logstash3-upg-test1 ansible_host=es3-upg-test1


##Updated mds nodes
[mds]
mds-STACKNAME consul_node_role=server
[mdslytics]
mdslytics-STACKNAME consul_node_role=server

##Update references to mds1,2,3 in filebeat group
filebeat-es1-upg-test1 ansible_host=es1-upg-test1
filebeat-es2-upg-test1 ansible_host=es2-upg-test1
filebeat-es3-upg-test1 ansible_host=es3-upg-test1
filebeat-logstash1-upg-test1 ansible_host=es1-upg-test1
filebeat-logstash2-upg-test1 ansible_host=es2-upg-test1
filebeat-logstash3-upg-test1 ansible_host=es3-upg-test1
filebeat-mds-STACKNAME ansible_host=mds-STACKNAME
filebeat-mdslytics-STACKNAME ansible_host=mdslytics-
STACKNAME


##Update/Add VM section
#Example
[vms]
api-STACK_NAME
cont-STACK_NAME
conv-STACK_NAME
conv2-STACK_NAME
jenkins-STACK_NAME
kafka-STACK_NAME
es1-STACK_NAME
es2-STACK_NAME
es3-STACK_NAME
mds-STACK_NAME
mdslytics-STACK_NAME
mon-STACK_NAME
nifi-STACK_NAME
postgres-STACK_NAME
```

```
qw-STACK_NAME
qw2-STACK_NAME
rabbit-STACK_NAME
redis-STACK_NAME
rose-STACK_NAME
ui-STACK_NAME
ups-STACK_NAME
vpn-STACK_NAMEte /etc/ansible/hosts
```

17. Clean up /etc/hosts file on jenkins server. Remove all references to mds1,2,3 and clean up any duplicates.

Offline Install

1. Remove ro-ansible package from the jenkins host.

   ```
   sudo yum remove ro-ansible
   ```

2. Backup the following files:

   ```
   sudo cp /etc/ansible/hosts /etc/ansible/hosts.bak
   sudo cp /etc/ansible/ansible.cfg /etc/ansible/ansible.bak
   ```

3. Run Forcepoint Behavioral Analytics binary.

   ```
   #copy the bin file to /{directory of choice} (ensure
   there is space available)
   sudo bash /{directory of choice}/Forcepoint-UEBA-3.3.0-
   CentOS-7.bin
   or
   sudo bash /{directory of choice}/Forcepoint-UEBA-3.3.0-
   RHEL-7.bin
   ```

4. Remove new files and restore files from step 2.

   ```
   sudo rm /etc/ansible/hosts
   sudo rm /etc/ansible/ansible.cfg
   sudo mv /etc/ansible/hosts.bak /etc/ansible/hosts
   sudo mv /etc/ansible/ansible.bak /etc/ansible/ansible.cfg
   ```

5. Add the new hosts

   a. In an AWS environment using the centos user and the associated private key run the following:

      ```
      ##MDS New Host
      ansible-playbook /usr/share/ro-ansible/ec2-launch-
      mds.yml -i /etc/ansible/hosts -f 5 --private-key {ssh
      key} -u centos -e inv_hostname=mds-{Stack-Name} -e
      jenkins_userid={Account_Name}
      ```

```
##MDSLYTICS New Host
ansible-playbook /usr/share/ro-ansible/ec2-launch-
mdslytics.yml -i /etc/ansible/hosts -f 5 --private-key
{ssh key} -u centos -e inv_hostname=mdslytics-{Stack-
Name} -e jenkins_userid={Account_Name}


##CONV2 New Host (Optional)
ansible-playbook /usr/share/ro-ansible/ec2-launch-
conversion.yml -i /etc/ansible/hosts -f 5 --private-
key {ssh key} -u centos -e inv_hostname=conv2-{Stack-
Name} -e jenkins_userid={Account_Name}


##QW2 New Host (Optional)
ansible-playbook /usr/share/ro-ansible/ec2-launch-
qw.yml -i /etc/ansible/hosts -f 5 --private-key {ssh
key} -u centos -e inv_hostname=qw2-{Stack-Name} -e
jenkins_userid={Account_Name}



##Update the ssh_known_hosts on the Jenkins host
sudo su
ssh-keyscan es1-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
ssh-keyscan es2-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
ssh-keyscan es3-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
#ensure all es nodes are added as above
ssh-keyscan mds-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
ssh-keyscan mdslytics-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
#Run for conv2 and qw2 if they were added
ssh-keyscan conv2-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
ssh-keyscan qw2-{Stack-Name} >> /etc/ssh/
ssh_known_hosts
```

b. In an on-prem environment, please see the sizing guidelines for the new hosts. The AWS instance types specified serve as a guideline for the expected hardware characteristics of the new nodes. Other concerns must then be addressed separately based on the on-prem environment:
   - DNS routing
   - Firewalling - make sure the service ports are available and the hosts can connect to the required dependent services

- Appropriate users/ssh/sudo access set up appropriately
- You will need to remove

6. Updates to Postgres Host
   a. In this release, it was determined the Postgres server was underpowered. As such in a new deployment, the new instance in an AWS environment uses an m5xlarge EC2 instance. There are memory and CPU requirement checks to ensure the stack is using the correct settings based on this new instance of Postgres.
   - In an AWS deployment, this is best accomplished by:
     a. From the aws console navigate to EC2 instances and select the postgres instance
     b. Stop the postgres host
     c. Under Actions select Instance Settings and Change Instance Type
     d. Select the new instance type of m5.xlarge and confirm
     e. Start the instance
   - In a deployment outside of AWS you will need to ensure the VM running postgres has adequate memory and CPU. The requirements are as follows:
     - 8192 mb memory
     - 2 vcpu
7. Run sudo yum clean all on the following hosts.

```
api-{var.stackname}.ro.internal
qw-{var.stackname}.ro.internal
conv-{var.stackname}.ro.internal
rose-{var.stackname}.ro.internal
cont-{var.stackname}.ro.internal
ups-{var.stackname}.ro.internal
ui-{var.stackname}.ro.internal
nifi-{var.stackname}.ro.internal
```

Upgrade Specific Services

1. Run the following playbooks in this order from /usr/share/ro-ansible:

```
ansible-playbook hostname.yml
ansible-playbook hosts_file.yml
ansible-playbook yum-mirror.yml
ansible-playbook ro-baseline.yml
ansible-playbook common.yml
ansible-playbook jenkins.yml
ansible-playbook redis.yml
```

```
ansible-playbook postgres.yml
ansible-playbook rabbit.yml
ansible-playbook ro-es.yml
```

2. Delete the analytics cache from es1.

```
curl -k -u elastic:changeme -XDELETE 'https://
localhost:9200/analytics_cache'
```

3. Run the following playbooks in this order from /usr/share/ro-ansible:

```
ansible-playbook kafka.yml
ansible-playbook ro-mon-es.yml
 (If the last task fails re run playbook TASK [ro-mon-es :
Create a disabled role mapping to initialize security
index (with auth)])
ansible-playbook ro-schema.yml
ansible-playbook ro-ui.yml
ansible-playbook ro-vault.yml   # this sets up certs on
any new hardware for an on prem set up. Not necessary if
you used our scripts to set up in aws
ansible-playbook ro-mds.yml
ansible-playbook ro-mdslytics.yml
ansible-playbook ro-monitoring.yml
ansible-playbook ro-kibana.yml
ansible-playbook ro-api.yml
ansible-playbook ro-qw.yml
ansible-playbook ro-conv.yml
ansible-playbook minigator.yml
ansible-playbook ro-logstash.yml
ansible-playbook ro-rose.yml
ansible-playbook ro-content.yml
ansible-playbook ro-ups.yml
ansible-playbook ro-ui.yml
ansible-playbook ro-nifi.yml
```

4. Compute the analytics cache from es1.

```
curl -XPOST -k https://localhost:8080/reference/
analytics/compute_dashboard | jq .
```

Final upgrade settings

1. Run the Prepare-UEBA Stack job.

   This will correct the necessary ntp conf settings for new mds server, mdslytics server, etc.

2. Run the Deploy-UEBA-Software job.

Cleanup

On the es servers after the upgrade, the ro-mds service may still be up and running redundantly. On each es server, the following should be run:

```
systemctl stop ro-mds
sudo rpm -e ro-mds
```