Forcepoint UEBA Installation Manual

Installation Manual | Forcepoint UEBA | v3.0.x | 10-Jul-2018

Installation Overview

This Forcepoint UEBA Installation manual guides technical Forcepoint UEBA users through a complete installation of a Forcepoint UEBA deployment. This guide includes step-by-step instructions for installing Forcepoint UEBA via Ansible and Jenkins. This document covers system architecture, required software installation tools, and finally a step-by-step guide for a complete install.

The System Architecture section shows how data moves throughout software components, as well as how 3rd party software is used for key front- and back-end functionalities.

The Installation Components section elaborates on important pre-installation topics. In preparation for initial installation setup, we discuss high level topics regarding Jenkins and Ansible - the tools Forcepoint UEBA utilizes to facilitate installation commands.

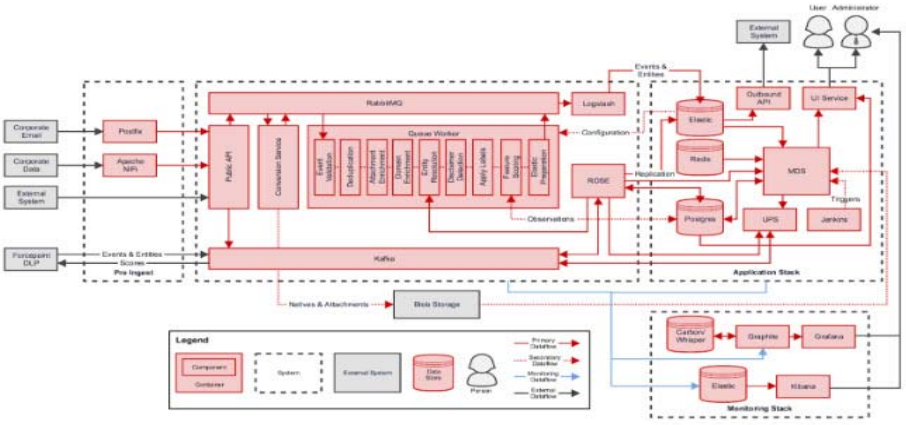
Although Jenkins is pre-configured at the time of install, we include Jenkins Setup information and important access and directory location information for a holistic understanding of this key installation facilitator.

To conclude this document, we include step-by-step instructions for using Ansible to initialize the Jenkins CI/CD server to install each required software component.

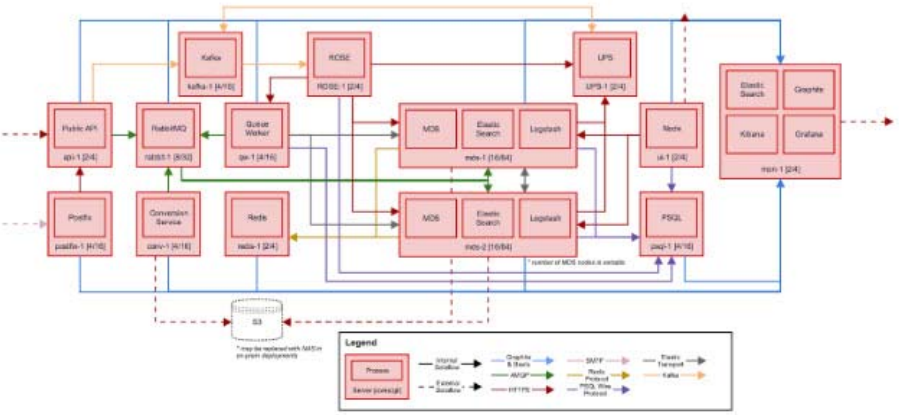
An addendum is included for additional components which can optionally be installed.

Go to the Downloads page and navigate to User and Entity Behavior Analytics to find the downloads for Forcepoint UEBA.

Platform Overview - Component



Platform Overview - Physical



Installation Components

Host OS

Forcepoint requires a RedHat 7 host based Operating System for the UEBA platform to be installed. CentOS 7 (minimal) is the recommended OS to be used. Please note, other heavier install media can be used, but not necessary or recommended. At the time of publication the latest version is CentOS 7.5. CentOS 6 is not supported, as it has known incompatibilities with our installation process and may introduce bugs into the Forcepoint UEBA product due to OS differences.

Security

Forcepoint recommends using commonly accepted network security practices to restrict access to the Forcepoint UEBA infrastructure. For instance, creating rules in IPTables, or implementing a network firewall that only allows the access defined below.

Port List

Service	Host	Port	Consumers
Graphite	mon	2003	all
Grafana	mon	443	Administrator Workstation
Jenkins	jenkins	8080/8443	All, Administrator Workstation
Kafka	kafka	9092-9095	API, Rose
Kafka Manager	kafka	9000	Administrator Workstation
Postgres	postgres	5432	Conversion, Master Data Service, Queue Worker, UI
UEBA API	api	9000	External Data Sources, RabbitMQ
UEBA API	api	9001	Administrator Workstation
UEBA Conversion	conv	9080	RabbitMQ
UEBA Conversion	conv	9081	Administrator Workstation
RabbitMQ	rabbit	4369	RabbitMQ
RabbitMQ	rabbit	5672	API, Conversion, Logstash, Queue Worker
RabbitMQ	rabbit	15672	Administrator Workstation
UEBA Queue Worker	qw	9090	RabbitMQ
UEBA Queue Worker	qw	9091	Administrator Workstation
Redis	redis	6379	UI
UI	ui	80/443	Users

Service	Host	Port	Consumers
Elasticsearch	mds	9200	UI, Jenkins, ES, MDS, API, Conv, QW
Elasticsearch	mds	9201	Administrator Workstation
Elasticsearch	mds	9300-9400	Elastic Search
UEBA Master Data	mds	8080	UI, Jenkins, MDS
UEBA Master Data	mds	8081	Administrator Workstation
UEBA Rose	rose	9500	API, Postgresql, Nifi, QW, UPS
UEBA Rose	Rose	9501	Administrator Workstation
UEBA UPS	ups	9600	MDS
UEBA UPS	ups	9601	Administrator Workstation
OpenVPN	vpn	1194	External Clients
Postfix	postfix	25, 587	External Mail Source

Installation Requirements

The Forcepoint UEBA installation is Ansible based and requires Ansible version 2.3.2.0. The recommended host OS for running Ansible is Linux, although any officially supported OS will suffice. Please refer to the official Ansible Installation Guide for further instructions.

Our current internal version is stable/3.3 in the Forcepoint UEBA Ansible git repository. The most recent stable version of this must be available to properly deploy the UEBA platform. This Ansible code is distributed via RPM. Please contact Forcepoint Support for further information.

Installation Facilitators

Ansible

Ansible is an IT automation tool that can configure systems, deploy software, and orchestrate more advanced IT tasks such as continuous deployments or zero downtime rolling updates. Ansible playbooks are used to incrementally install the separate components of a Forcepoint UEBA instance.

File Format: YAML

- Ansible uses YAML because it is easier for humans to read and write than other common data formats, like XML or JSON. Further, there are libraries available in most programming languages for working with YAML.
- Example YAML Usage

Playbooks

- Playbooks are the basis for a really simple configuration management and multi-machine deployment system that is well suited to deploy complex applications.
- Playbooks can declare configurations, and they can also orchestrate steps of any manual ordered process, even as different steps must bounce back and forth between sets of machines in particular orders. They can launch tasks synchronously or asynchronously.

Individual “Tasks” Make Up a role or playbook. A “Playbook” is comprised of tasks and roles.

```
- hosts: webservers
  remote_user: root

  tasks:
  - name: ensure apache is at the latest version
    yum: name=httpd state=latest
  - name: write the apache config file
    template: src=/srv/httpd.j2 dest=/etc/httpd.conf

- hosts: databases
  remote_user: root

  tasks:
  - name: ensure postgresql is at the latest version
    yum: name=postgresql state=latest
  - name: ensure that postgresql is started
    service: name=postgresql state=started
```

Jenkins

Jenkins an open source automation server. Jenkins helps to automate the non-human part of continuous delivery. This is the primary way in which Forcepoint installs the UEBA software.

Installation Procedures

Things to consider

- Infrastructure must be provisioned before hand
- All hosts have SSH enabled and reachable from the provisioning ansible host.
- Every major component in the Forcepoint UEBA tech stack runs on its own host.
- Our install and configuration is Ansible based.
  - /etc/ansible/hosts file must be accurate.
  - /etc/ansible/group_vars/all must be accurate and tailored to any site specific overrides if necessary.
  - Host machine running ansible playbooks must have ssh access to all hosts in the /etc/ansible/hosts inventory file.
  - Example files can be found after installing the ro-ansible rpm: /usr/share/ro-ansible/sysconfdir
- All commands are assumed to be run on a fully patched CentOS 7 host with Ansible 2.3.2.0 installed following the Ansible installation guide.
- Identify a deployment machine which will host Forcepoint Continuous Deployment Jenkins server.
- In advance of install, the installer must request IP whitelisting from the UEBA team for the customer to access our public yum repository.
- The Jenkins server is initialized and after, the UEBA platform is deployed via the Continuous Delivery server.

Create and Configure Client's Inventory Directory

Create an Ansible configuration directory:

```
mkdir -p /etc/ansible/
```

Create a hosts file for the playbook to run successfully. in the hosts file, CHANGEME should be changed to something short and intuitive. It will be visible to users, an abbreviation of the customer's name is recommended. An example file can be found: /usr/share/ro-ansible/sysconfig/host.example

```
touch /etc/ansible/hosts
```

Example:

```
/etc/ansible/hosts
[api]
api-CHANGEME
[ca]
ro-root-ca ansible_host=jenkins-CHANGEME
[conversion]
conv-CHANGEME
```

```
[curator]
curator-CHANGEME ansible_host=jenkins-CHANGEME
[es]
es1-CHANGEME ansible_host=mds1-CHANGEME
es2-CHANGEME ansible_host=mds2-CHANGEME
es3-CHANGEME ansible_host=mds3-CHANGEME
[jenkins]
jenkins-CHANGEME
[logstash]
logstash1-CHANGEME ansible_host=mds1-CHANGEME
logstash2-CHANGEME ansible_host=mds2-CHANGEME
logstash3-CHANGEME ansible_host=mds3-CHANGEME
[mds]
mds3-CHANGEME
mds2-CHANGEME
mds1-CHANGEME
[minigator]
minigator-CHANGEME ansible_host=jenkins-CHANGEME
[monitoring]
mon-CHANGEME
[oapi]
oapi-CHANGEME
[postgres]
postgres-CHANGEME
[qw]
qw-CHANGEME
[rabbit]
rabbit-CHANGEME
[redis]
redis-CHANGEME
[schema]
schema-CHANGEME ansible_host=jenkins-CHANGEME
[ui]
ui-CHANGEME
[nifi]
nifi-CHANGEME
[filebeat]
filebeat-api-CHANGEME ansible_host=api-CHANGEME
filebeat-conv-CHANGEME ansible_host=conv-CHANGEME
filebeat-curator-CHANGEME ansible_host=jenkins-CHANGEME
```

```
filebeat-es1-CHANGEME ansible_host=mds1-CHANGEME
filebeat-es2-CHANGEME ansible_host=mds2-CHANGEME
filebeat-es3-CHANGEME ansible_host=mds3-CHANGEME
filebeat-jenkins-CHANGEME ansible_host=jenkins-CHANGEME
filebeat-logstash1-CHANGEME ansible_host=mds1-CHANGEME
filebeat-logstash2-CHANGEME ansible_host=mds2-CHANGEME
filebeat-logstash3-CHANGEME ansible_host=mds3-CHANGEME
filebeat-mds3-CHANGEME ansible_host=mds3-CHANGEME
filebeat-mds2-CHANGEME ansible_host=mds2-CHANGEME
filebeat-mds1-CHANGEME ansible_host=mds1-CHANGEME
filebeat-minigator-CHANGEME ansible_host=jenkins-CHANGEME
filebeat-mon-CHANGEME ansible_host=mon-CHANGEME
filebeat-postgres-CHANGEME ansible_host=postgres-CHANGEME
filebeat-oapi-CHANGEME ansible_host=oapi-CHANGEME
filebeat-qw-CHANGEME ansible_host=qw-CHANGEME
filebeat-rabbit-CHANGEME ansible_host=rabbit-CHANGEME
filebeat-redis-CHANGEME ansible_host=redis-CHANGEME
filebeat-rose-CHANGEME ansible_host=rose-CHANGEME
filebeat-schema-CHANGEME ansible_host=jenkins-CHANGEME
filebeat-ui-CHANGEME ansible_host=ui-CHANGEME
filebeat-nifi-CHANGEME ansible_host=nifi-CHANGEME
```

Create and configure client's group_vars

Create a group_vars directory relative to the /etc/ansible/ directory.

```
mkdir -p /etc/ansible/group_vars
```

Create an "all" file for any environmental variables specific to the client. In that file, the first CHANGEME should match what you used for the hosts file, while all other CHANGEME instances should be replaced with their relevant values. Please note, this example assumes an AWS cloud deployment model; modify as needed. An example file can be found: /usr/share/ro-ansible/sysconfig/group_vars/all.example.

```
touch /etc/ansible/group_vars/all
```

Example:

```
/etc/ansible/group_vars/all
##environment name (domain)
release: 2610
ro_env: CHANGEME
domain: "{{ domain_name }}"
tld: internal
domain_name: "ro.{{ tld }}"
```

```
##Monitoring related vars
yum_branch: "{{ release }}"
#AWS specific vars
cloudprovider: aws
aws_ami_image_default: "{{ aws_ami_image_el7 }}"
aws_ami_image_el6: ami-1c221e76
aws_ami_image_el7: ami-CHANGEME
aws_ro_region: us-east-1
aws_ansible_keyname: CHANGEME
aws_ansible_key: CHANGEME
aws_ansible_secret: CHANGEME
aws_route53_ansible_key: CHANGEME
aws_route53_ansible_secret: CHANGEME/64e0+p4cew/GLSZvYcS7a
aws_vpc_id: vpc-CHANGEME
aws_vpc_public_subnet_id: subnet-CHANGEME
aws_vpc_private_subnet_id: subnet-CHANGEME
aws_vpc_subnet: CHANGEME
aws_vpc_subnetmask: CHANGEME
aws_vpc_route53_ip: CHANGEME
#MDS specific vars
mds_lb: mds1-CHANGEME
##conversion service vars
natives_bucket: es.natives.{{ ro_env }}
attachments_bucket: es.attachments.{{ ro_env }}
#UI specific vars
ro_ui_env_host_url: https://ui-CHANGEME.{{ domain_name }}
ui_lb_cert: "CHANGEME"
api_lb_cert: "CHANGEME"
upload_thread_pool: 40
thread_pool: 40
ueba_jk_id: ueba-ansible
ueba_ansible:
  user: CHANGEME
  private_key: |
    -----BEGIN RSA PRIVATE KEY-----
    CHANGEME
    -----END RSA PRIVATE KEY-----
```

```
public_key: "ssh-rsa CHANGEME"
```

Note
Do not forget to include the following section at the bottom of the file if this applies to you.

```
## Use DLP Provided Cert Chain
## We store files locally on jenkins-ueba
#external_ca_certs: true
##external_ca_scheme: https
#external_root_ca_cert_url: "file:///etc/ansible/dlp-certs/
ca.cer"
#external_int_ca_key_url: "file:///etc/ansible/dlp-certs/
ueba.key"
#external_int_ca_cert_url: "file:///etc/ansible/dlp-certs/
ueba.cer"
```

Note
Ensure machine running ansible playbooks have ssh access to every host in the /etc/ansible/hosts inventory file. You may use private key or password based ssh connections. However, passwordless ssh key authentication is preferred.

Example (do not run this at this time) ansible command with ssh key:

```
ansible-playbook ro-baseline.yml -u centos --private-key=~/
.ssh/client.pem
```

Example (do not run this at this time) ansible command with ssh username and password:

```
ansible-playbook ro-baseline.yml -u centos -k
```

Running the Commands

Based on the client-dictated ssh authentication method, adjust the following commands as necessary (remembering to include the private key or credentials, according to the previous section).

Initialize Forcepoint Continuous Delivery Server

1. Retrieve ro-ansible rpm

> **Note**
> Do not copy and paste the text below directly. The line wrapping does not allow the commands to be executed correctly.

```
wget https://yum.redowl.com/redowl/3001/7/x86_64/ro-ansible-3.3.2-1.e17.noarch.rpm
```

2. Install ro-ansible

```
yum install ro-ansible-3.3.2-*
```

3. Deploy Jenkins host

```
ansible-playbook /usr/share/ro-ansible/jenkins-init.yml
```

4. Navigate to jenkins web-based service in a browser. The url can be reached by hostname, FQDN or ip.

```
e.g.
- http://jenkins-customer.domain.com:8080
- http://jenkins-customer:8080
- http://10.0.0.100:8080
```

> **Note**
> The Deploy-UEBA-Stack Jenkins job should be running.

5. Login to Forcepoint Continuous Delivery Server - Jenkins

```
Default credentials are:
Username: forcepoint
Password: forcepoint
```

6. Deploy UEBA Stack from Forcepoint Continuous Delivery Server (optional)

> **Important**
> This step is not needed if the job is already running - as indicated by a flashing icon in Jenkins - as Ansible should start it automatically for you. If it is not running, then you can start the job yourself from the dashboard tab (in Jenkins), click “Schedule a Build for Deploy-UEBA-Stack” button located on the right hand side.

7. Check the deployment status from Forcepoint Continuous Delivery Server (optional)

   The status and currently running deployment jobs can be found in the Build Executor Status window.

Create Default UI Admin User

1. Create first admin user.

- By default Forcepoint UEBA does not ship with an initial user configured. You must manually create this user to successfully log into the UI.
- These commands must be executed on the postgres host from the command line.
- This will create the following default user:
  - Username: redowl@redowl.com
  - Password: redowl

Note

Do not copy and paste the text below directly. The line wrapping does not allow the commands to be executed correctly.

```
psql -U redowlpostgres -d the_ui -c "INSERT INTO USERS
(email, encrypted_password, name, created_at, updated_at,
password_updated_at) VALUES
('redowl@redowl.com','\$2a\$06\$mMhM9IWYk1J3Q15tGgP5rOryw7Mo
1m3JL0eydVOtJ20gmm4twDKMW','Red Owl', CURRENT_DATE,
CURRENT_DATE, CURRENT_DATE);"


psql -U redowlpostgres -d the_ui -c "INSERT INTO roles_users
(role_id, user_id) (SELECT r.id, u.id FROM roles r INNER
JOIN users u ON (u.email LIKE 'redowl@redowl.com'));"
```

2. Manually installing an Elasticsearch license.

   This step is no longer required, as the license is now installed by default. If you still need to perform this step, the steps are in the Appendix.

Appendix

Notes on OpenVPN

This process is currently operations intensive due to the evolving customer deployment models. These operations should be performed by Professional Services.

Things to consider

- The VPN host must be provisioned before hand.
- The VPN host must have SSH enabled and reachable from the provisioning ansible host.
- The install and configuration is Ansible based.
  - /etc/ansible/hosts file must be accurate.
  - /etc/ansible/group_vars/all must be accurate and tailored to any site specific overrides if necessary.

- Host machine running ansible playbooks must have ssh access to all hosts in the /etc/ansible/hosts inventory file.

- All commands are assumed to be run on a host with Ansible 2.3.2.0 installed following the Ansible installation guide.
- In advance of install, the installer must request IP whitelisting from the UEBA team for the customer to access our public yum repository.
  - Step 3 provides guidance to retrieve VPN IP public address.

1. Baseline Forcepoint UEBA VPN host.

   ```
   ansible-playbook ro-baseline.yml --limit openvpn
   ```

2. Ensure SSH Key is Copied to Forcepoint UEBA VPN host.

   ```
   cp user.pem ~/.ssh/user.pem
   chmod 600 ~/.ssh/user.pem
   ```

3. Retrieve Forcepoint UEBA VPN host Public IP.

   ```
   curl ipecho.net/plain
   ```

4. Install common Forcepoint UEBA packages

   ```
   Option 1 - Run everything:
      ansible-playbook ro-common.yml --limit openvpn
   Option 2 - Run select playbooks, based on customer needs:
      Always run (do NOT confuse this with ro-common.yml):
         ansible-playbook common.yml
      Optionally run:
         ansible-playbook ro-users.yml --limit openvpn
         ansible-playbook sudoers.yml --limit openvpn
         ansible-playbook selinux.yml --limit openvpn
         ansible-playbook ntp.yml --limit openvpn
         ansible-playbook hostname.yml --limit openvpn
         ansible-playbook ro-ssh.yml --limit openvpn
         ansible-playbook hosts_file.yml --limit openvpn
   ```

5. Deploy OpenVPN Service

   ```
   ansible-playbook openvpn.yml
   ```

6. Start OpenVPN Service

   Run from Forcepoint UEBA VPN host:

   ```
   sudo systemctl restart openvpn@server.service
   ```

7. Create OpenVPN Users

   Note

   Substitute {{user}} with correct username.

   Run from Forcepoint UEBA VPN host:

   ```
   sudo /etc/openvpn/addvpnuser.sh fp-ueba-ops-{{user}}
   ```

```
sudo su - {{user}}
passwd - enter password twice when prompted
cp /etc/openvpn/keys/{{user}}-vpn-*.tar.gz /home/{{user}}
```

Copy /home/{{user}}-vpn-*.tar.gz to remote machine for Professional Services Engineer use.

8. Configure 2FA - Google Authenticator

> **Note**
> Substitute {{user}} with correct username.

Run from Forcepoint UEBA VPN host logged in as newly created user:

```
google-authenticator
```

- Correct question answers are: YYYNY
- Copy the barcode and/or the url to add to the authenticator app.

9. Test Forcepoint UEBA VPN connection

> **Note**
> Substitute {{user}} with correct username.

Run from Professional Services OSX host:

```
tar {{user}}-vpn-*.tar.gz -C {{user}}-vpn.tblk
```

Drag and drop {{user}}-vpn.tlbk into tunnelblick configuration windows.

Connect using username,password+googleauth

Troubleshooting OpenVPN

If authentication fails ensure password is set correctly. Reset password as necessary.

Google-authenicator may need to be rerun.

If name lookups are failing there is a bug in the tunnelblik software to where the client does not push the AWS DNS server and search domains to the local machine. In this case go to your primary network interface and manually add the route53 address x.x.x.2 for the DNS server and appropriate search domain.

Deployment - AWS Encryption Options for Native and Attachment Storage

Version 2.60 adds SSE-KMS support to the AWS S3 document handlers in the Conversion Service for native and attachment storage. More generally, all three available encryption methods are now configured through an "encryption" field in the Conversion Service yml (see below for examples of each). The default configuration (in absence of any "encryption" field) will continue to be SSE-S3 as it was in previous

releases. No UI configuration changes are necessary for SSE-KMS to work; the AWS IAM credentials used by the UI must be on the KMS key policy.

Example:

SSE-KMS configuration

```
handlers:
nativesHandler:
    type: aws
    bucket: my.natives
    encryption:
      type: sse-kms
      keyArn: arn:aws:kms:us-east-1:902691740976:key/
f2dca481-b989-4bb4-8aad-51c9da0358ee
  attachmentsHandler:
    type: aws
    bucket: my.attachments
    encryption:
      type: sse-kms
      keyArn: arn:aws:kms:us-east-1:902691740976:alias/aws-
develop-test-kms
```

Example:

Sample SSE-S3 configuration

```
handlers:
  nativesHandler:
    type: aws
    bucket: my.natives
    encryption:
      type: sse-s3
  attachmentsHandler:
    type: aws
    bucket: my.attachments
    encryption:
      type: sse-s3
```

Example:

Sample SSE-C configuration

```
handlers:
  nativesHandler:
    type: aws
bucket: my.natives
    encryption:
```

```
      type: sse-c
      sseKeyFile: /path/to/my.key
  attachmentsHandler:
    type: aws
    bucket: my.attachments
    encryption:
      type: sse-c
      sseKeyFile: /path/to/my.key
```

Deployment - Manually Run Ansible Playbooks

Prepare RedOwl Stack

1. Baseline Forcepoint UEBA hosts

   ```
   ansible-playbook ro-baseline.yml
   ```

2. Install common Forcepoint UEBA packages

   ```
   Option 1 - Run everything:
      ansible-playbook ro-common.yml
   Option 2 - Run select playbooks, based on customer needs:
      Always run (do NOT confuse this with ro-common.yml):
         ansible-playbook common.yml
      Optionally run:
         ansible-playbook ro-users.yml
         ansible-playbook sudoers.yml
         ansible-playbook selinux.yml
         ansible-playbook ntp.yml
         ansible-playbook hostname.yml
         ansible-playbook ro-ssh.yml
         ansible-playbook hosts_file.yml
   ```

3. To deploy RedOwl PKI, generate Certificate Chain.

   ```
   ansible-playbook generate-crts.yml
   ```

4. To deploy RedOwl Middleware, deploy Jenkins host.

   ```
   ansible-playbook jenkins.yml
   ```

5. Deploy Redis.

   ```
   ansible-playbook redis.yml
   ```

6. Deploy Postgresql.

   ```
   ansible-playbook postgres.yml
   ```

7. Deploy RabbitMQ.

   ```
   ansible-playbook rabbit.yml
   ```

8. Deploy Kafka.

   ```
   ansible-playbook kafka.yml
   ```

9. Deploy Elastic Search.

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

10. Deploy Monitoring Elastic Search.

    ```
    ansible-playbook ro-mon-es.yml
    ```

11. Initialize Forcepoint UEBA Schema.

    ```
    ansible-playbook ro-schema.yml
    ```

12. To deploy RedOwl software, deploy Forcepoint UEBA Monitoring Software

    ```
    ansible-playbook ro-monitoring.yml
    ansible-playbook ro-kibana.yml
    ```

13. Deploy UEBA Master Data Service.

    ```
    ansible-playbook ro-mds.yml
    ```

14. Deploy Logstash.

    ```
    ansible-playbook ro-logstash.yml
    ```

15. Deploy UEBA API Service.

    ```
    ansible-playbook ro-api.yml
    ```

16. Deploy UEBA Queue Worker Service.

    ```
    ansible-playbook ro-qw.yml
    ```

17. Deploy UEBA Conversion Service.

    ```
    ansible-playbook ro-conv.yml
    ```

18. Deploy UEBA UPS Service.

    ```
    ansible-playbook ro-ups.yml
    ```

19. Deploy Rose Service.

    ```
    ansible-playbook ro-rose.yml
    ```

20. Deploy Apache Nifi Service.

    ```
    ansible-playbook ro-nifi.yml
    ```

21. Deploy Forcepoint UI Service.

    ```
    ansible-playbook ro-ui.yml
    ```

22. Deploy Forcepoint Integration Service (optional).

    ```
    ansible-playbook ro-oapi.yml
    ```

Post-Installation Steps

1. Configure the Elasticsearch license (optional- as this should now be part of the default installation playbooks)

   ```
   Acquire the license JSON file from our Yum repo (only
   accessible to whitelisted IP addresses) http://
   yum.redowl.com/redowl/ro-es-prod.json
   ```

   ```
   curl -XPUT -u elastic 'http://<host>:<port>/_xpack/license'
   -d @license.json
   ```