Forcepoint UEBA Installation Manual

Installation Manual | Forcepoint UEBA | v3.1.1 | 22-Feb-2019

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.

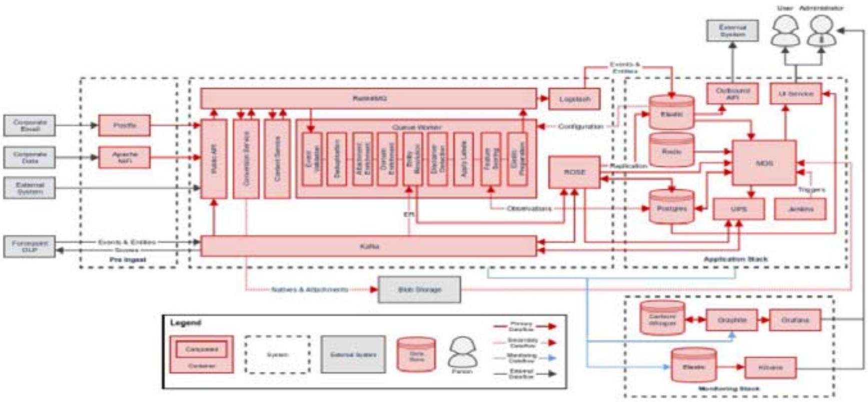
Although Jenkins is pre-configured at the time of install, we include Jenkins Setup information and important access and directory location information for a holistic understanding of this key installation facilitator.

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.

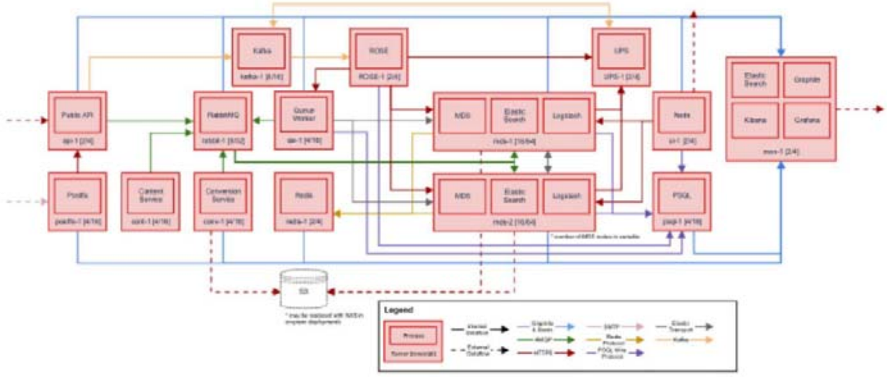
An addendum is included for additional components which can optionally be installed.

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 Forcepoint 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 it is not necessary or recommended. A headless install is required. Any install that is not headless will include dependencies that are not supported and will cause the install to fail. 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 disk encryption and network security practices to restrict access to the Forcepoint UEBA infrastructure. For instance, Forcepoint UEBA uses persistent queues in RabbitMQ by default to prevent the loss of events/data in the event that the RabbitMQ application goes down. Forcepoint recommends using full disk encryption for RabbitMQ for on premises installs. This ensures the data at rest is secure and GDPR compliant. Forcepoint strongly recommends 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
Vault	jenkins	8200/8201/8300/8301	All, Administrator Workstation
Kafka	kafka	9092-9095	API, Rose
Kafka Manager	kafka	9000	Administrator Workstation
Postgres	postgres	5432	Conversion, Rose, 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
UEBA Content	cont	9700	RabbitMQ, ES
RabbitMQ	rabbit	4369	RabbitMQ
RabbitMQ	rabbit	5672	API, Conversion, Logstash, Queue Worker
RabbitMQ	rabbit	15672	Administrator Workstation

Service	Host	Port	Consumers
UEBA Queue Worker	qw	9090	RabbitMQ
UEBA Queue Worker	qw	9091	Administrator Workstation
Redis	redis	6379	UI
UI	ui	80/443	Users
Elasticsearch	mds	9200	UI, Jenkins, ES, MDS, API, Conv, QW
Elasticsearch	mds	9201	Administrator Workstation
Elasticsearch	mds	9300-9400	Elastic Search
UEBA Master Data Service	mds	8080	UI, Jenkins, MDS
UEBA Master Data Service	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

Installation Requirements

The Forcepoint UEBA installation is Ansible based and requires Ansible version 2.5.8.0. No action is required as the installer has prerequisites packaged.

Our current internal version is stable/3.5 in the Ansible git repository. The most recent stable version of this must be available to properly deploy the Forcepoint UEBA platform. This Ansible code is distributed via installer. 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 Forcepoint UEBA software.

Installation Procedures

Things to consider

- Infrastructure must be provisioned beforehand.
- Python version 2 or 3 is required on all hosts.
- All hosts have SSH enabled and reachable from the provisioning ansible host via /etc/hosts or DNS.
- 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.
- All commands are assumed to be run on a fully updated CentOS 7 host.
- Root or sudo is requires for installation.
- Being that SSH root login is allowed by default with RedHat and CentOS, it is easiest to login as root instead or sudo.
- Identify a deployment machine which will host Forcepoint Continuous Deployment Jenkins server.
- The Jenkins host will be where you perform all actions from here on.
- In advance of install, the installer must request IP whitelisting from the Forcepoint UEBA team for the customer to access our public yum repository.
- The Jenkins server is initialized and after, the Forcepoint UEBA platform is deployed via the Continuous Delivery server.

Installing miscellaneous tools

Install:

```
sudo yum install wget
```

Download the UEBA installer

1. Retrieve Forcepoint UEBA installer from support.

   Go to [https:// support.forcepoint.com](https://support.forcepoint.com)
2. Set Forcepoint UEBA installer to be executable.
   ```
   sudo chmod +x Forcepoint-UEBA-3.1.1-CentOS-7.bin
   ```
3. Extract Forcepoint UEBA installer.
   ```
   sudo bash Forcepoint-UEBA-3.1.1-CentOS-7.bin
   ```

Create and Configure Client's Inventory Directory

You will also need to create a hosts file in order for the playbook to run successfully. In the hosts file, xxxxx should be changed to something short and intuitive. It will be visible to users, so we recommend an abbreviation of the customer's name.

An example file to get you going, can be found on the machine you ran the UEBA installer at

```
/usr/share/ro-ansible/sysconfdir/
```

Below is a quick search and replace command. Change intuitive in the example below, to the recommendation above then run the sed command.

```
sed -i -e 's/xxxxx/intuitive/g' /etc/ansible/hosts
```

```
Example: /etc/ansible/hosts
    [api]
    api-xxxxx
    [ca]
    ro-root-ca ansible_host=jenkins-xxxxx
    [content]
    cont-xxxxx
    [conversion]
    conv-xxxxx
    [curator]
    curator-xxxxx ansible_host=jenkins-xxxxx
    [es]
    es1-xxxxx ansible_host=mds1-xxxxx
    es2-xxxxx ansible_host=mds2-xxxxx
    es3-xxxxx ansible_host=mds3-xxxxx
    [grafana]
    grafana-xxxxx ansible_host=mon-xxxxx
    [jenkins]
    jenkins-xxxxx
    [kafka]
    kafka-xxxxx
    [logstash]
    logstash1-xxxxx ansible_host=mds1-xxxxx
    logstash2-xxxxx ansible_host=mds2-xxxxx
    logstash3-xxxxx ansible_host=mds3-xxxxx
    [mds]
    mds3-xxxxx
    mds2-xxxxx
    mds1-xxxxx
```

```
[minigator]
minigator-xxxxx ansible_host=jenkins-xxxxx
[monitoring]
Mon-xxxxx
[nfs]
jenkins-xxxxx
[oapi]
#oapi-xxxxx #uncomment if used
[vpn]
#vpn-xxxxx #uncomment if used
[postgres]
postgres-xxxxx
[qw]
qw-xxxxx
[rabbit]
rabbit-xxxxx
[redis]
redis-xxxxx
[rose]
rose-xxxxx
[schema]
schema-xxxxx ansible_host=jenkins-xxxxx
[ui]
ui-xxxxx
[ups]
ups-xxxxx
[nifi]
nifi-xxxxx
[dlp-entity]
dlp-entity-xxxxx ansible_host=nifi-xxxxx
[vault]
vault-${ro_unique_name_san}
ansible_host=jenkins-${ro_unique_name_san}
[filebeat]
filebeat-api-xxxxx ansible_host=api-xxxxx
filebeat-cont-xxxxx ansible_host=cont-xxxxx
filebeat-conv-xxxxx ansible_host=conv-xxxxx
filebeat-curator-xxxxx ansible_host=jenkins-xxxxx
filebeat-filebeat-curator-xxxxx ansible_host=jenkins-
xxxxx
filebeat-es1-xxxxx ansible_host=mds1-xxxxx
```

```
filebeat-es2-xxxxx ansible_host=mds2-xxxxx
filebeat-es3-xxxxx ansible_host=mds3-xxxxx
filebeat-grafana-xxxxx ansible_host=mon-xxxxx
filebeat-jenkins-xxxxx ansible_host=jenkins-xxxxx
filebeat-kafka-xxxxx ansible_host=kafka-xxxxx
filebeat-logstash1-xxxxx ansible_host=mds1-xxxxx
filebeat-logstash2-xxxxx ansible_host=mds2-xxxxx
filebeat-logstash3-xxxxx ansible_host=mds3-xxxxx
filebeat-mds3-xxxxx ansible_host=mds3-xxxxx
filebeat-mds2-xxxxx ansible_host=mds2-xxxxx
filebeat-mds1-xxxxx ansible_host=mds1-xxxxx
filebeat-minigator-xxxxx ansible_host=jenkins-xxxxx
filebeat-mon-xxxxx ansible_host=mon-xxxxx
filebeat-postgres-xxxxx ansible_host=postgres-xxxxx
#filebeat-oapi-xxxxx ansible_host=oapi-xxxxx #uncomment
if
used
#filebeat-vpn-xxxxx ansible_host=vpn-xxxxx #uncomment if
used
filebeat-qw-xxxxx ansible_host=qw-xxxxx
filebeat-rabbit-xxxxx ansible_host=rabbit-xxxxx
filebeat-redis-xxxxx ansible_host=redis-xxxxx
filebeat-rose-xxxxx ansible_host=rose-xxxxx
filebeat-schema-xxxxx ansible_host=jenkins-xxxxx
filebeat-ui-xxxxx ansible_host=ui-xxxxx
filebeat-ups-xxxxx ansible_host=ups-xxxxx
filebeat-nifi-xxxxx ansible_host=nifi-xxxxx
filebeat-vault-xxxxx ansible_host=jenkins-xxxxx
```

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. It can be tailored to work for AWS or on premises installs. In that file, xxxxx should be changed to something short and intuitive. It will be visible to users, so we recomment an abbreviation of the customer’s name. Please note, this example assumes an AWS cloud deployment model; modify as needed. An example file can be found: /usr/share/ro-ansible/sysconfdir/group_vars.

```
    touch /etc/ansible/group_vars/all
Example: /etc/ansible/group_vars/all
```

```
##environment name (domain)
ro_env: xxxxx
domain: "{{ domain_name }}"
tld: internal
domain_name: "ro.{{ tld }}"
## Networking specific vars
network_subnet: xxxxx
network_subnetmask: xxxxx
##Monitoring related vars
yum_branch: "{{ release }}"
#AWS specific vars
cloudprovider: aws
aws_ami_image_default: "{{ aws_ami_image_el7 }}"
aws_ami_image_el6: ami-xxxxx
aws_ami_image_el7: ami-xxxxx
aws_ro_region: us-east-1
aws_ansible_keyname: xxxxx
aws_ansible_key: xxxxx
aws_ansible_secret: xxxxx
aws_route53_ansible_key: xxxxx
aws_route53_ansible_secret: xxxxx/64e0+p4cew/GLSZvYcS7a
aws_vpc_id: vpc-xxxxx
aws_vpc_public_subnet_id: subnet-xxxxx
aws_vpc_private_subnet_id: subnet-xxxxx
aws_vpc_route53_ip: xxxxx
#MDS specific vars
mds_lb: mds1-xxxxx
#ES specific
es_plugins:
- plugin: repository-s3
##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-xxxxx.{{ domain_name }}
ui_lb_cert: "xxxxx"
api_lb_cert: "xxxxx"
upload_thread_pool: 40
thread_pool: 40
ueba_jk_id: ueba-ansible
```

```
ueba_ansible:
user: xxxxx
private_key: |
-----BEGIN RSA PRIVATE KEY-----
xxxxx
-----END RSA PRIVATE KEY-----
public_key: "ssh-rsa xxxxx"
## Use Externally Provided Cert Chain
## We store files locally on jenkins-ueba
## Example:
## /etc/pki/external-certs/root/root-ca.key
## /etc/pki/external-certs/root/root-ca.crt
## /etc/pki/external-certs/intermediate/int-ca.key
## /etc/pki/external-certs/intermediate/int-ca.crt
#external_ca_certs: true
#vault_pki_ca_external_dir: "/etc/pki/external-certs"
#vault_pki_root_ca_external_dir: "{{
vault_pki_ca_external_dir }}/root"
#vault_pki_int_ca_external_dir: "{{
vault_pki_ca_external_dir }}/intermediate"
#vault_pki_root_ca_external_bundle: "{{
vault_pki_ca_external_dir
}}/root_ca_bundle.pem"
#vault_pki_int_ca_external_bundle: "{{
vault_pki_ca_external_dir }}/int_ca_bundle.pem"
```

✓ **Note**

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

Setup /etc/hosts, static table lookup for hostnames.

```
Example: /etc/hosts
   127.0.0.1 localhost localhost.localdomain localhost4
   localhost4.localdomain4
   ::1 localhost localhost.localdomain localhost6
   localhost6.localdomain6
   10.55.10.110 api-ueba
   10.55.10.106 conversion-ueba
   10.55.10.105 jenkins-ueba
   10.55.10.120 kafka-ueba
```

```
10.55.10.122 mds1-ueba
10.55.10.124 mds2-ueba
10.55.10.136 mds3-ueba
10.55.10.113 monitoring-ueba
10.55.10.126 nifi-ueba
10.55.10.119 postgres-ueba
10.55.10.130 qw-ueba
10.55.10.131 rabbit-ueba
10.55.10.132 redis-ueba
10.55.10.129 rose-ueba
10.55.10.133 ui-ueba
10.55.10.134 ups-ueba
10.55.10.127 vpn-ueba
```

Generating an SSH key pair:

```
ssh-keygen -t ed25519
```

Copy SSH public key to all hosts defined in /etc/hosts.

```
#!/bin/bash
read -s -p "Enter SSH Password: " PASSWORD
for ip in `getent hosts | grep -v '127.0.0.1' | cut -d' ' -
f5`; do
echo "Adding SSH keys to: $ip"
sshpass -p "$PASSWORD" ssh-copy-id -o
PubkeyAuthentication=no -o
StrictHostKeyChecking=no "$ip"
done
```

1. Save the bash script above to:

   ```
   /root/SSH_key_copy.sh
   ```

2. chmod +x /root/SSH_key_copy.sh
3. yum install sshpass
4. bash /root/SSH_key_copy.sh hosts_password (assumes all hosts have the same password).

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. Deploy Jenkins host (Before running playbook, all hosts must have SSH enabled and reachable from the provisioning ansible host via /etc/hosts or DNS).

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

2. Deploy NFS client and server for shared storage.

   ✓ **Note**
   This step is only for on premise installs that need shared storage

   ```
   ansible-playbook /usr/share/ro-ansible/nfs-server.yml
   ansible-playbook /usr/share/ro-ansible/nfs-client.yml
   ```

3. Deploy Forcepoint UEBA by navigating to Jenkins web-based service in a browser.

   The hostname can be reached by hostname, FQDN, or IP.

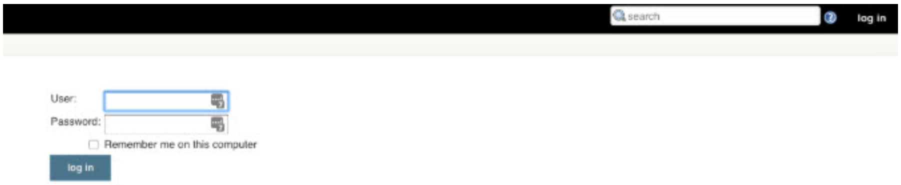
   ```
   e.g.
       - http://jenkins-customer.domain.com:8080
       - http://jenkins-customer:8080
       - http://10.0.0.100:8080
   ```

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

4. Login to Forcepoint Continuous Delivery Server - Jenkins

   Default credentials are:

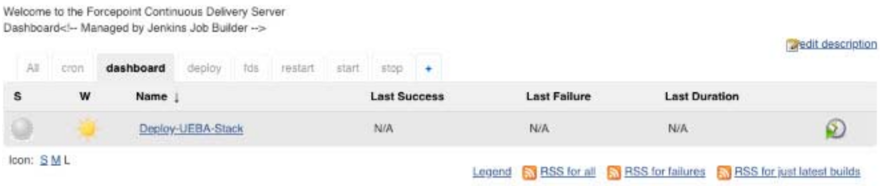
   ```
   Username: forcepoint
   Password: forcepoint
   ```



5. Deploy the Forcepoint 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.

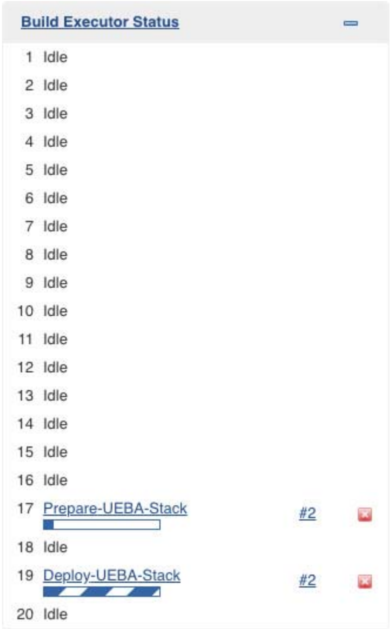


Welcome to the Forcepoint Continuous Delivery Server
Dashboard<!-- Managed by Jenkins Job Builder -->

S	W	Name ↓	Last Success	Last Failure	Last Duration
●	☀	Deploy-UEBA-Stack	N/A	N/A	N/A

6. 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.



Build Executor Status

```
1  Idle
2  Idle
3  Idle
4  Idle
5  Idle
6  Idle
7  Idle
8  Idle
9  Idle
10 Idle
11 Idle
12 Idle
13 Idle
14 Idle
15 Idle
16 Idle
17 Prepare-UEBA-Stack   #2
18 Idle
19 Deploy-UEBA-Stack    #2
20 Idle
```

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 install 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 2.5.8.0 installed following the Ansible installation guide.
- In advance of install, the installer must request IP whitelisting from the Forcepoint 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 Forcepoint UEBA Stack

1. Forcepoint UEBA hostnames.

   ```
   ansible-playbook hostname.yml
   ansible-playbook hosts_file.yml
   ```

2. Forcepoint UEBA baseline.

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

3. 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 ansible-openssh.yml
   ```

4. Deploy Forcepoint UEBA secrets:.

   ```
   ansible-playbook vault.yml
   ```

5. To deploy Forcepoint UEBA Middleware, deploy Jenkins host.

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

6. Deploy Redis.

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

7. Deploy Postgresql.

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

8. Deploy RabbitMQ.

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

9. Deploy Kafka.

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

10. Deploy Elastic Search.

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

11. Deploy Monitoring Elastic Search.

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

12. Initialize Forcepoint UEBA Schema.

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

13. Deploy Forcepoint UEBA Monitoring Software.

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

14. Deploy Forcepoint UEBA Master Data Service.

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

15. Deploy Logstash.

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

16. Deploy Forcepoint UEBA API Service.

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

17. Deploy Forcepoint UEBA Queue Worker Service.

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

18. Deploy Forcepoint UEBA Conversion Service.

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

19. Deploy Forcepoint UEBA Content Service.

    ```
    ansible-playbook ro-cont.yml
    ```

20. Deploy Forcepoint UEBA UPS Service.

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

21. Deploy Rose Service.

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

22. Deploy Apache Nifi Service.

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

23. Deploy Forcepoint UI Service.

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

24. Deploy Forcepoint Integration Service (optional).

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