[Abstract] *HSOP Platform Service Operations and Maintenance Runbook*

## Document Approval

| | Function | Name | Signature/ Date |
|---|---|---|---|
| **Prepared by:** | Developer | G K, Raghavendra<br>Parvatikar, Shrinivas | |
| **Reviewed by:** | Product Owner | G K, Raghavendra | |
| | Architect (if applicable) | G K, Raghavendra | |
| **Approved by:** | Platform Service Product Owner | Vikram Rao | |
| | Release Train Engineer (RTE) | Nandini Raj | |

## 1.0 Purpose

The purpose of this Operating and Maintenance Runbook is to provide operator with instructions on deploying & maintaining the Health**Suite** OnPremise MicroCloud platform on AWS Outposts Full Form Factor. [Abstract]

## 2.0 Scope

This guide is a step-by-step process for the Operations Team completed by the HealthSuite On Premise MicroCloud platform <[Abstract]> deploying services on AWS Outposts Full Form Factor infrastructure.

This document specifically serves to guide those who will be maintaining, supporting, and using the services in day-to-day operational basis.

The following skill sets are required by the service engineers for deploying and supporting day-to-day activities.

| Skills | Skill Level |
|---|---|
| **Linux fundamentals** | Basic |
| **Terraform** | Basic |
| **Shell scripts** | Basic |
| **Git** | Basic |
| **Docker, Packer** | Basic |
| **kubernetes** | Basic |
| **CloudFoundry** | Basic |
| **AWS** | Basic |

## Table of Contents

## 3.0 Terms and Abbreviations

See the QMS Glossary [REF-1] for Terms and Abbreviations, commonly used in our QMS.

| Terms & Abbreviations | Description/Definition |
|---|---|
| **API** | Application Program Interface |
| **AWS** | Amazon Web Services |
| **D/R** | Disaster / Recovery |
| **GFS** | Grandfather, Father, Son relationship |
| **HSDP** | HealthSuite Digital Platform Services |
| **HSP** | HealthSuite Platforms |
| **HSOP** | HealthSuite On Premise |
| **MC** | MicroCloud |
| **IAM** | Identity Access Management |
| **IaaS** | Infrastructure as a service (IaaS) is a form of cloud computing that provides virtualized computing resources over the internet. |
| **I&S** | Innovation and Strategy Platforms |
| **IP&S** | Intellectual Property and Standards |
| **IQ/OQ** | Installation Qualification and Operational Qualification |
| **O&M** | Operations and Maintenance |
| **PaaS** | Platform as a Service |
| **P&R** | Privacy & Regulatory |
| **RTE** | Release Train Engineer |
| **RTO** | Recovery Time Objective (RTO) is the length of time beginning when the service is first down after a failure or disaster occurs and ending when the Platform is restored to service in accordance with its specifications and all then current configuration documents. Please document your application RTO. |

| Terms & Abbreviations | Description/Definition |
|---|---|
| **RPO** | Recovery Point Objective (RPO) is the length of time between data backup intervals for the purpose of maintaining data that might be lost from the service due to a major incident. |
| **LDAP** | Lightweight Directory Access Protocol |
| **CF** | Cloud Foundry |

# 4.0 Overview of Solution

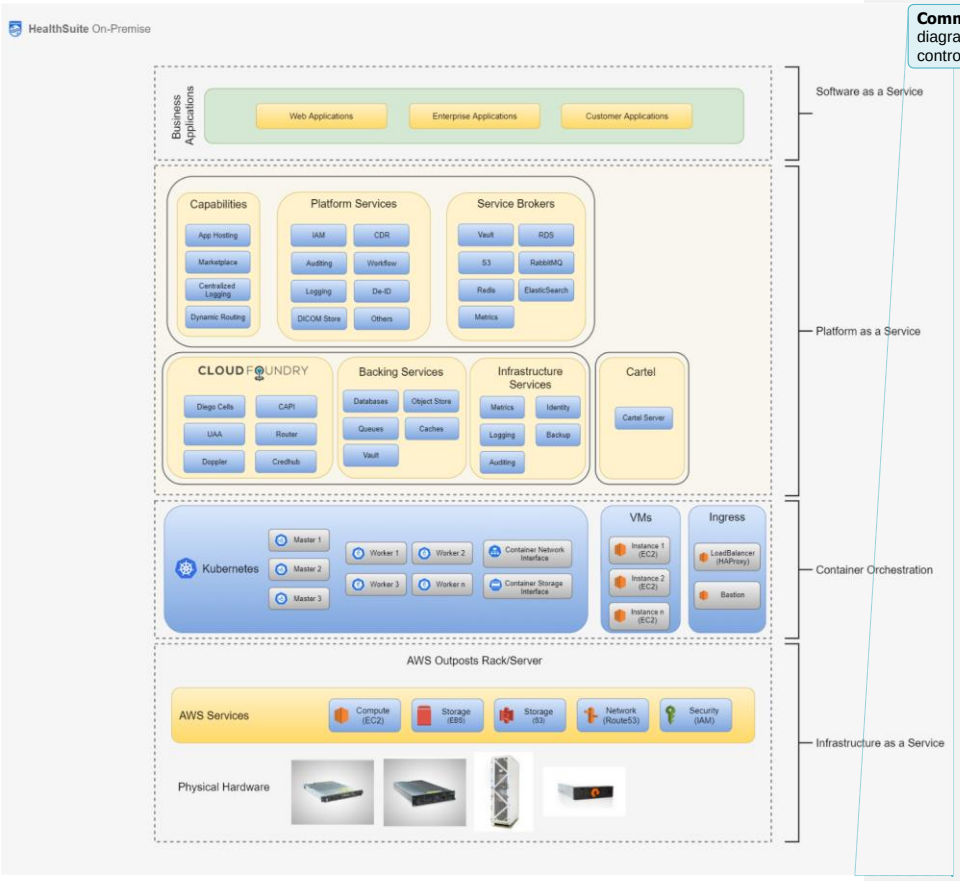
## 4.1. Overview Solution Architecture

HealthSuite OnPremise MicroCloud platform is a cloud foundry environment primarily meant to enable a harmonized development and continuous integration/continuous deployment (CI/CD) experience between the cloud and premises platforms.

## 4.2. Functional Solution Overview

Below diagram depicts the high-level architecture view of MicroCloud. Each layer provides key capabilities/functionality to MicroCloud.

**Note:** HSOP MicroCloud v2.0 release is intended for non-commercial use.




**Commented [UT1]:** Please replace with new HSOP diagram. There's a current one Vikram created for the export controls submission.

**Physical layer**: This layer is AWS Outposts Full Form Factor.

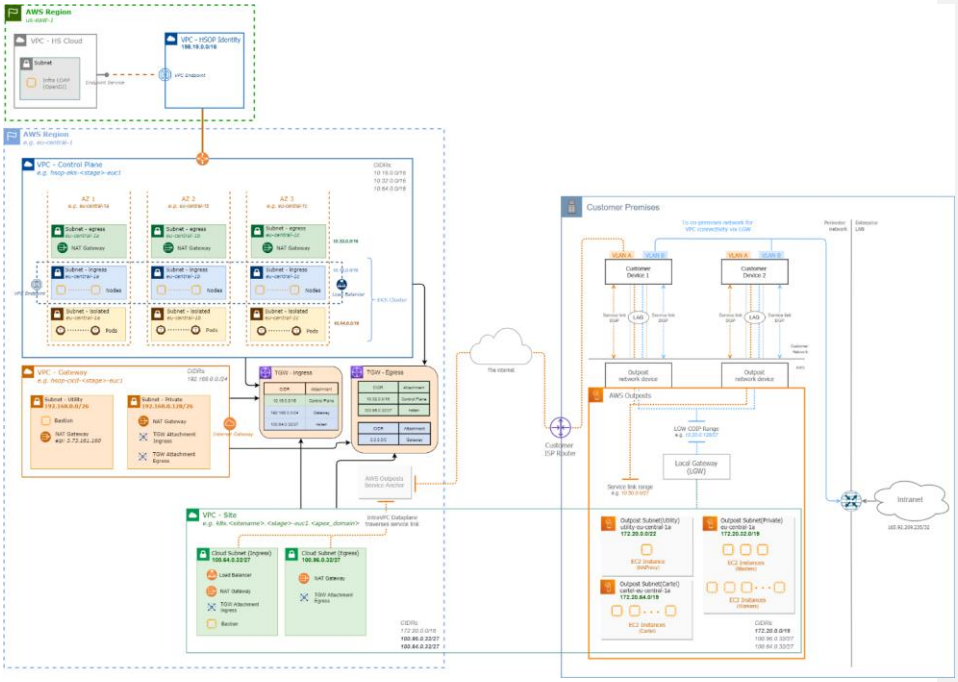
**Infrastructure as a Service (IaaS)**: This layer provides services to provision required compute, storage, and resource requirements to host/deploy other layers.

IaaS layer abstracts underlying hardware, HSOP uses Terraform tooling to provision the infrastructure and configure required infrastructure services.

**Container Orchestration Layer**: This layer consists of a kubernetes distribution deployed on nodes provisioned by IaaS layer.

**Containerized Cloud Foundry Layer**: This layer leverages open-source CloudFoundry (CF) distribution for kubernetes called Kubecf

### 4.2.1. Physical Architecture/Deployment Model




MicroCloud deployment consist of following high level components

| Components | Description |
|---|---|
| **MicroCloud VPC** | This VPC consist of all the assets required for provisioning of MicroCloud |
| **Public Subnet** | This subnet consists of Bastion entry point and Load balancer configured for MC. All the external interactions happen via this subnet |
| **Private Subnet** | This subnet consists of K8S and all MC deployment assets |
| **kubernetes cluster** | This is the primary backbone for the platform |

| | |
|---|---|
| **HAProxy host** | This is the publicly accessible load balancer node used as API entry point for all access |
| **Bastion host** | This the publicly accessible SSH entry point for HSOP deployment |
| **Identity provider host** | This host consist of LDAP and KeyCloak |

**HSOP Regional deployment model**

*TODO*

### 4.2.2. Data Flows

NA

### 4.2.3. Data and Database Administration

N/A

### 4.2.4. Capacity Management - Region Deployment

| System | Configuration | EC2 Instance Count |
|---|---|---|
| **Region eks cluster configuration** | | |
| worker node | m5a.xlarge | **5** |
| bastion | t3.micro | **1** |

### 4.2.5. Kubernetes namespaces

| Namespace | Description |
|---|---|
| **kube-system** | Standard kubernetes namespace. |
| **minio** | Namespace to hold pods for infrastructure minio(blobstore). |
| **cf-mysql** | Namespace to hold the mysql instance that contains cloudfoundry databases (cc db, uaa db etc.). |
| **cert-manager** | Namespace to hold cert-manager pods required for certificate generation & renewal. |
| **certificate** | Namespace to hold certificate (custom CRD) resource. Deployed on if use_external_certificate is false in the deployment profile. |
| **cf-operator** | Namespace to hold quarks resources required to bootstrap kubecf. |
| **kubecf** | Namespace to hold Cloudfoundry (cf-depoyment) resources. |
| **monitoring** | Namespace to hold resources required for monitoring the cluster. Include kube-promethues, alert-manager, grafana pods & services. |
| **vault** | Namespace to hold vault instance. |
| **identity** | Namespace to hold keycloak, openldap pods & services. |
| **nginx-reverse-proxy** | Namespace to hold nginx reverse proxy required node exporters on EC2 instances not managed by kubernetes. |
| **dynamodb** | Namespace to hold dynamodb local instance used to store broker state. |
| **minio-s3** | Namespace to hold minio instance used by the S3 service broker. |
| **logging** | Namespace to hold log drainer pods & services. |
| **managed-redis** | Namespace to hold redis instances provisioned by the hsdp-redis service broker. |
| **managed-rds** | Namespace to hold postgres & mysql instances provisioned by the hsdp-rds service broker. |
| **managed-rmq** | Namespace to hold RabbitMQ instances provisioned by the hsdp-rmq service broker. |
| **managed-es** | Namespace to hold Elasticsearch instances provisioned by the hsdp-elasticsearch service broker. |
| **velero** | Namespace to hold pods & services required for automatic backup of cluster resources. |

### 4.2.6. HSDP Platform Service APIs consumed

N/A

### 4.2.7. Configuration Management of HSOP MicroCloud Platform

The source repository for the deployment of HSOP MicroCloud Platform can be accessed at https://bitbucket.hsdp.io/scm/dep/hsonprem.git.

Region specific configurations are specified in the terragrunt.hcl file under regions folder. Each regions where HSOP is deployed should have a sub directory under regions with the AWS region name.

### 4.2.8. Platform Components

| Components | Version |
|---|---|
| **terraform** | 1.0.6 |
| **kubecf** | 2.7.12* |
| **kubernetes** | 1.21.5 |
| **kops** | 1.21.1 |
| **cf cli** | 6.53.0 |
| **docker** | 1.7.4 |
| **git** | 20.10.8 |
| **packer** | 2.17.1 |
| **credhub cli** | 2.9.1 |
| **mysql** | Supported versions:<br>• engine version 5<br>• engine version 5.6<br>• engine version 5.7<br>• engine version 8 |
| **postgres** | engine Supported version<br>• engine version 9<br>• engine version 9.6<br>• engine version 10<br>• engine version 10.10<br>• engine version 11<br>• engine version 11.5<br>• engine version 12<br>• engine version 12.2 |
| **elasticsearch** | Versions:<br>• v6.8<br>• v7.8 |
| **rabbitmq** | Version:<br>• v3.7.17 |
| **velero client** | Version:<br>• v1.6.3 |
| **vault** | Version:<br>• v1.8.1 |

* kubecf version for MVP is v2.7.12, to enable testing CF updates.

# 5.0 Operations Procedures

The following sections cover aspects of installation, maintenance, monitoring and backup/recovery of the environment.

## 5.1. Event Based Access Management

Product team needs to be onboarded on the HSOP MicroCloud Platform & added to the level4-support group in LDAP. Refer Add user to Group (Only for Level 2, Level 3 Level 4 Users)

### 5.1.1. Deployment Role

level-3 support users as defined in Add user to Group (Only for Level 2, Level 3 Level 4 Users)

### 5.1.2. Monitoring Role

level-2 & level-3 support users as defined in Add user to Group (Only for Level 2, Level 3 Level 4 Users)

## 5.2. HSOP Region Installation Instructions

### 5.2.1. HSOP Prerequisites

- A valid domain name & the corresponding public Route53 Hosted Zone. Please note that the only one apex domain is required for HSOP & the Route53 Hosted Zone needs to be created only once across all regional deployments. An example of such a domain would be *hsop.io* or *hsop.hsdp.io*.
- The AWS account used to deploy a HSOP region needs to have read/write access to the aforementioned apex zone.
- A valid email domain & the corresponding AWS SES configuration including the MX records in the aforementioned Route53 Hosted Zone of the apex domain. An example of such an email domain would be *mail.hsop.io* or *mail.hsop.hsdp.io*.
- A git repository to manage site manifests files. Note that this repository can be created in either HSDP bitbucket (*bitbucket.hsdp.io*) or Philips inner source (*github.com/philips-internal*).

### 5.2.2. Installation Prerequisites

- One of the following POSIX environments on the system used for installation.
  - Mac OS (darwin)
  - Linux (Ubuntu 20.04 LTS)
  - Windows 10 64bit + WSL2 (Ubuntu 20.04 LTS)
- The following applications/tools are required to be installed on the system used for installation.

Commented [UT2]: @Rao, Vikram will need this to be more specific. Please see MC-74

Commented [UT3R2]: For example, do we have a minimum BW requirement?

| Application/Tool | URL | Version | Environment |
|---|---|---|---|
| **terraform** | link | v1.0.6 | All (darwin/linux/wsl2) |
| **terragrunt** | link | v0.37.1 | All (darwin/linux/wsl2) |
| **git** | link | 2.34.1 | All (darwin/linux/wsl2) |
| **kubectl** | link | v1.22.3 | All (darwin/linux/wsl2) |
| **aws-cli** | link | 2.2.4 | All (darwin/linux/wsl2) |
| **kubectx** | link | v0.9.4 | All (darwin/linux/wsl2) |
| **sshuttle** | link | 0.78.3 | darwin/linux/wsl2 |
| **OpenSSH** | link | latest | Windows 10 |
| **jq** | link | 1.5 | darwin/linux/wsl2 |
| **yq** | link | 4.9.0 | darwin/linux/wsl2 |
| **packer** | link | 1.8.2 | darwin/linux/wsl2 |
| **docker** | link | 20.10.14 | darwin/linux/wsl2 |
| **eksctl** | link | 0.98.0 | All (darwin/linux/wsl2) |
| **vault** | link | v1.8.2 | All (darwin/linux/wsl2) |
| **nmap** | link | 7.60 | All (darwin/linux/wsl2) |

- Internet connectivity with at least 100Mbps bandwidth.
- Ensure the linux user used for installation has password less sudo privilege enabled.
- Ensure that sufficient resources & quota are configured for the AWS account used to deploy HSOP. The required resources & quotas are described in section Appendix – AE AWS Resource requirement & quotas.
- The following environment variables, with appropriate values, must be exported in the user shell. **As a security best practice, please do not add the environment variables to the bash profile or any persistent file.**

```
export AWS_ACCESS_KEY_ID=<your aws access key id>
export AWS_SECRET_ACCESS_KEY=<your aws secret access key>
export AWS_REGION=<the aws region>
export AWS_DEFAULT_REGION=<the aws region>
export PIP_EXTRA_INDEX_URL=<full url with secret to the HSDP pypi repository>
export PAGERDUTY_KEY=<pager duty key for alerts>
export SMTP_FROM="<from email address>"
export SMTP_HOST="<email host smtp endpoint>:<port>"
export SMTP_LOGIN="<email login username>"
export SMTP_PASSWORD="<email login password>"
export TWILIO_AUTH_TOKEN="<twilio auth token>"
export TWILIO_SID="<twilio sid>"
```

```
export TWILIO_FRIENDLY_NAME="<twilio friendly name, e.g. HSOP>"
export GIT_REPO_URL="<url to the git repository where site manifests are created>"
export GIT_USERNAME="<username to login to the git repository>"
export GIT_PASSWORD="<password/token to login to git repository>"
export HS_CLOUD_LDAP_PWD="<HSDP infra ldap password for readonly user>"
export APP_REGISTRY_HOSTNAME=<Docker registry endpoint where images are stored>
export APP_REGISTRY_REPOSITORY_PREFIX=<Prefix name/namespace name in the docker
registry>
export APP_REGISTRY_USERNAME=<Docker registry username>
export APP_REGISTRY_PASSWORD=<Docker registry password/token>
```

- HSOP MicroCloud deployment requires an AWS account with access to the tethered region with admin permissions. The permissions required are outlined in Appendix-A.

### 5.2.3. Regional Deployment Instructions

1. Clone & checkout the version of HSOP MicroCloud platform source code that needs to be deployed.

```
mkdir -p hsop
cd hsop

git clone <repository_url> .
git checkout tags/<HSOP_version>

# For OPS dry run, use the following values for
# repository URL & the version tag.
git clone https://bitbucket.hsdp.io/scm/dep/hsonprem.git .
git checkout hsop_mc2.1_dryrun

# For production, use the following values for
# repository URL & the version tag.
git clone https://bitbucket.hsdp.io/scm/dep/hsonprem.git .
git checkout tags/v2.1.0 -b <new_branch_name>
```

2. In the hsop directory where the repository has been cloned, change directory to *source/deployment_region/regions*.

```
cd source/deployment_region/regions/
```

3. Copy the example deployment template provided as part of the *example-region-name* directory, in the *regions* directory, to a destination directory whose name matches the name of the AWS region where HSOP needs to be deployed.

```
cp -r example-region-name <aws_region_name>
# For example, to deploy in eu-central-1 region,
# run the following command.
# cp -r example-region-name eu-central-1
```

4. Change directory to the newly copied directory, with intended region name & edit the *terragrunt.hcl* file with your favorite text editor.

```
cd <aws_region_name>/
vi terragrunt.hcl
```

5. Provide appropriate values for the following variables in the *terragrunt.hcl* file.

```
apex_domain_name = "<the top level domain name>"
stage = "<provide a unique RFC 1738 compatible name>"
email = "<your email id>"
hs_cloud_ldap_service_name = "<HS Cloud LDAP Endpoint Service Name>"
hs_cloud_ldap_service_azs = ["Availability zones for the HS Cloud LDAP Endpoint
Service"]
hs_cloud_ldap_service_fqdn = "<The fully qualified name of the HS Cloud LDAP
instance>"
hscloud_ldap_username =  "<Username to authenticate with HS Cloud LDAP instance>"
hscloud_ldap_password = "${get_env("HS_CLOUD_LDAP_PWD", "")}"
```

   a. Please note that the *hscloud_ldap_password* variable is read from the environment variable *HS_CLOUD_LDAP_PWD* & should not be specified in the *terragrunt.hcl* file. Export the aforesaid environment variable before proceeding with deployment.

```
export HS_CLOUD_LDAP_PWD="<the HS Cloud ldap password>"
```

6. Change directory back to *source/deployment_region* folder.

```
cd ../..
```

7. The deployment script for a new HSOP region is present in the current directory & is called *hsop*.

   a. The region deployment script accepts the following arguments.

```
hsop v2.1.0
Usage: hsop [OPTIONS]
OPTION includes:
   -U | --update - Create or update a HSOP region or site
   -T | --teardown - Teardown existing HSOP region or site
   -R | --region - Name of the HSOP region
   -M | --modules - Comma separated list of modules to create or teardown.
Specify '*' to affect all modules
   -v | --verbose - Make the operation more talkative
   -V | --version - prints out version information of HSOP
   -H | --help - displays this message
```

   b. The deployment script can be used to:

      i. deploy a new HSOP region
      ii. update an existing HSOP region
      iii. teardown an existing HSOP region

   c. The deployment process consists of the following discreet steps or modules that are executed in order shown below.

```
1. network
2. iam
3. bastion
4. transit-gateway
5. eks
6. loadbalancers
```

```
7. services
8. finalize
```

8. Trigger deployment of a new region by running the following command.

```
./hsop --update --region <aws_region_name> --modules "*" --verbose
```

9. The deployment of a new HSOP region takes approximately 2 to 3 hours depending on the network bandwidth available & other environment parameters. Upon successful completion, a new HSOP region is available for deployment of individual sites. See appendix for troubleshooting any errors you might encounter during deployment.

#### 5.2.4. Post Installation Instructions

1. Verify access to bastion host using your HS Cloud ldap credentials.

```
ssh -i /path/to/your/id_rsa \
  -o "StrictHostKeyChecking no" \
  <ldap_username>@bastion.<stage>-<aws_region_short_name>.<apex_domain>
```

2. Enable other users access to the EKS cluster by following the instructions described in section Appendix – V Provide access to other AWS IAM users after deployment of HSOP Control Plane (or the regional EKS cluster).

#### 5.2.5 Teardown Instructions

### 5.3. HSOP Site Installation Instructions

#### 5.3.1. HSOP Site Installation Prerequisites

- One of the following POSIX environments on the system used for installation.
  - Mac OS (darwin)
  - Linux (Ubuntu 20.04 LTS)
  - Windows 10 64bit + WSL2 (Ubuntu 20.04 LTS)
- The following applications/tools are required to be installed on the system used for installation.

| Application/Tool | URL | Version | Environment |
|---|---|---|---|
| **git** | link | 2.34.1 | All (darwin/linux/wsl2) |
| **sshuttle** | link | 0.78.3 | darwin/linux/wsl2 |
| **OpenSSH** | link | latest | Windows 10 |
| **cf-cli** | link | v6 | All (darwin/linux/wsl2) |
| **kubectl** | link | v1.22.3 | All (darwin/linux/wsl2) |

- Ensure that sufficient resources & quota are configured for the AWS account used to deploy HSOP. The required resources & quotas are described in section Appendix – AE AWS Resource requirement & quotas.

- A text editor.
- Internet connectivity with at least 10Mbps bandwidth.
- A web browser

### 5.3.2. Capacity Management - Site

| System | Configuration | EC2 Instance Count |
|---|---|---|
| **Small Profile** | | |
| Bastion | t3.micro | **1** |
| Master | m5.large | **1** |
| Nodes | m5.2xlarge | **6** |
| HA Proxy | m5.xlarge | **1** |
| **Medium Profile** | | |
| Bastion | t3.micro | **1** |
| Master | m5.large | **3** |
| Worker Nodes -1 | M5.xlarge | **3** |
| Worker Nodes -2 | M5.2xlarge | **4** |
| HA Proxy | m5.xlarge | **1** |
| **Large Profile** | | |
| Bastion | t3.micro | **1** |
| Master | m5.large | **3** |
| Worker Nodes -1 | M5.2xlarge | **4** |
| Worker Nodes -2 | M5.4xlarge | **1** |
| HA Proxy | m5.xlarge | **1** |
| **XLarge Profile** | | |
| Bastion | t3.micro | **1** |
| Master | m5.large | **3** |
| Worker Nodes -1 | M5.2xlarge | **2** |
| Worker Nodes -2 | M5.4xlarge | **2** |
| HA Proxy | m5.xlarge | **1** |

### 5.3.3. Site Deployment Instructions

1. Clone the git repository where the site manifests are maintained. See section HSOP Prerequisites.

```
# for example
git clone https://github.com/philips-internal/hsop-deployments.git
```

2. Checkout a new branch to add the manifest file for the site to be deployed.

```
git checkout -b feature_new_site_<sitename>
```

3. Under the *regions* directory in the cloned git repo, create a new directory with the name of the region where the site is needed to be deployed if it does not exist.

```
cd regions
mkdir -p <aws_region_name>
# for example: mkdir -p eu-central-1
```

4. Create a new site manifest yaml file under the region-specific directory (*e.g. eu-central-1*) that was created in the previous step & specify the site specific details as per the schema shown below. The specific information for a site should be filled in based on the details mentioned in the site on-boarding JIRA ticket.

```
apiVersion: cd.cicd.hsop.io/v1
kind: Site
metadata:
  name: <name of the site as per RFC 1123/RFC 1035>                    # Required
spec:
  version: "<microcloud version to deploy, currently v2.1.0>"          # Required
  stage: <stage name, e.g. dev,test,prod etc.>                         # Required
  outpost_id: <the outpost id to deploy microcloud stack>              # Optional
  availabilityZones:                                                   # Required
    - <the availability zone of the outpost/deployment>                # Required
  email: <email address of the person triggering the deployment>       # Required
  plan: <the plan type - small,medium,large,xlarge>                    # Required
  type: <type of deployment - outpost,cloud>                           # Required
  operation:                                                           # Optional
    type: <type of operation to perform - update,teardown>             # Optional
    modules:                                                           # Optional
      - <module name>                                                  # Optional
  network: <COIP configuration for LAN setup>                          # Optional
    subnet_routes:                                                     # Optional
      - <the subnet cidrs for LGW>                                     # Optional
    ha_proxy: <the ipv4 pool name for HAProxy>                         # Optional
    pool: <the COIP address pool provided by the Hospital>             # Optional
  demographics: <demographics of the customer location>                # Optional
    fullName: <name of the customer facility>                          # Optional
    address: <address of the customer facility>                        # Optional
    contact:                                                           # Optional
      - name: <name of the contact person>                             # Optional
        email: <email id of the contact person>                        # Optional
        telephone: <telephone number of the contact>                   # Optional
```

5. Add & commit the newly created site manifest file to the git repository.

```
git add <new_site_manifest_yaml>
git commit -m "Deploy site <sitename>"
```

6. Raise a pull request to merge changes from your feature branch created earlier to the master branch of the git repository.

Deployment of the new site will automatically begin upon successful merge of the new site manifest file to the master branch of the git repository. Note that the deployment of a new site takes approximately 3 to 4 hours.

#### 5.3.4. Post Installation Instructions

1. Check the status of the newly deployed site as outlined in section Appendix – AA Check status of a HSOP site & verify that the value of the state attribute is `Up`. If the state attribute value is `Pending`, then check back again after some time.
2. Once the site deployment is complete & state attribute for the site is Up in ArgoCD UI, add users to the appropriate support groups in LDAP for the site as describe in section Onboarding Users to support groups in LDAP for a site.

## 5.4. Tenant Lifecycle

### 5.4.1. Onboarding Users to support groups in LDAP for the HSOP control plane

1. Access the central LDAP service for the region as described in section Appendix – Z Access central LDAP instance in region.
2. Under the base DN, expand and select the entry named `ou=Group`.
3. Select the appropriate group for a site by clicking on the relevant child element under `ou=Group`.
    a. Note that the names of the support groups for the control are not prefixed and are called `level1-support`, `level2-support`, `level3-support`, `level4-support` & `platform-services-support`.


4. Under `uniqueMember` field, click the `(add value)` link
5. In the popup window, expand the base DN and expand `ou=People` entry. Select the user you wish to add from the list.
6. Click `Update object` in the bottom of the screen and click `Update object` in the subsequent dialog.

### 5.4.2. Onboarding Users to support groups in LDAP for a HSOP site

1. Access the central LDAP service for the region as described in section Appendix – Z Access central LDAP instance in region.
2. Under the base DN, expand and select the entry named `ou=Group`.
3. Select the appropriate group for a site by clicking on the relevant child element under `ou=Group`.
    a. Note that the names of the support groups for a site begin with the site name followed by a hyphen. For example, for a site called `apollo`, the following support groups exists under `ou=Group`.


4. Under `uniqueMember` field, click the `(add value)` link
5. In the popup window, expand the base DN and expand `ou=People` entry. Select the user you wish to add from the list.
6. Click `Update object` in the bottom of the screen and click `Update object` in the subsequent dialog.
7. Note that it can take up to an hour for the permissions to reflect in the chosen site.

### 5.4.3. Onboarding Client Users to HSOP Platform

1. Access the central LDAP service for the region as described in section Appendix – Z Access central LDAP instance in region.
2. Under the base DN, expand and select the entry named `ou=Group`.
3. Select the `platform-services-support` group for a site by clicking on the relevant child element under `ou=Group`.
    a. Note that the names of the support groups for a site begin with the site name followed by a hyphen. For example, for a site called `apollo`, you'll find a group called `cn=apollo-platform-services-support`.
4. Under `uniqueMember` field, click the `(add value)` link
5. In the popup window, expand the base DN and expand `ou=People` entry. Select the user you wish to add from the list.
6. Click `Update object` in the bottom of the screen and click `Update object` in the subsequent dialog.
7. Note that it can take up to an hour for the user to reflect in Cloud Foundry at the chosen site.
8. Onboard the newly created user to Cloud Foundry.
    a. Create a new Cloud Foundry Org & Space for the client if it does not exist as described in section Appendix – AC Create new Cloud Foundry Org & Space in a HSOP site.
    b. Assign Cloud Foundry Org & Space roles to the user as described in section Appendix – AD Assign Cloud Foundry Org & Space roles to a User in a HSOP site.

**Accessing kubernetes Cluster**

**Prerequisites**

- kubectl v1.21.1
- kubelogin plugin
- Kubernetes client id and secret from Appendix-E Fetching kubernetes client id and Secret from Keycloak

**Procedure**

1. Run the command:
   **kubectl oidc-login setup --oidc-issuer-url https://identity.cf.<domain>/auth/realms/hsop --oidc-client-id kubernetes --oidc-client-secret <oidc-client-secret>**

```
## Set up the kubeconfig
Run the following command:
kubectl config set-credentials oidc \
  --exec-api-version=client.authentication.k8s.io/v1beta1 \
  --exec-command=kubectl \
  --exec-arg=oidc-login \
  --exec-arg=get-token \
  --exec-arg=--oidc-issuer-url=https://identity.cf.<DOMAIN_NAME>/auth/realms/hsop \
  --exec-arg=--oidc-client-id=xxx \
  --exec-arg=--oidc-client-secret=xxx
## Verify cluster access
## Make sure you can access the Kubernetes cluster.
kubectl --user=oidc get nodes
## You can switch the default context to oidc.
kubectl config set-context --current --user=oidc
## You can share the kubeconfig to your team members for on-boarding.
```

### 5.4.4. Offboarding a User

Offboarding of users is handled in the HS Cloud Infra LDAP.

### 5.4.5. Enabling / Disabling

*N/A*

## 5.5. Monitoring

### 5.5.1. Accessing Grafana Dashboard

Grafana dashboard is integrated with ldap. So, the ldap user will be able to login to grafana dashboard with their username and password.

**Prerequisites**

- . Tunnel to the bastion host. Instructions are here in Appendix

**Procedure**

1. Navigate to the URL: **https://metrics.cf.< domain_name>**
2. Login with the following credentials:
    a. Username: <ldap username>
    b. Password: <ldap password>
3. After successful login, Grafana homepage appears.
4. Navigate to **Manage** tab and the list of available dashboards would be listed
5. Click on any dashboard of interest to view the metrics. A sample dashboard metrics for alert manager is shown in Appendix- I List of Dashboards

### 5.5.2. Alerting

Pager duty alerts are automatically triggered based on the configured rules.

A sample dashboard metrics for alert manager is specified in Appendix-C Alert Resolution

## 5.6. Service Recovery Plan

The Service Recovery plan provides detailed information to be considered in backup and disaster recovery of the HSOP Service.

### 5.6.1. Backup Procedures

**Backups**

HSOP platform performs periodic backup to the cloud. Velero is used to schedule automated backups. The default backup frequency is once every 72 hours. The following table outlines the resources backed up, periodicity and retention.

| Resource | Schedule Name | Periodicity/Frequency | Retention |
|---|---|---|---|
| **Vault** | vault-backup | 72 hours | 360 hours (15 days) |
| **RDS** | managed-rds-backup | 72 hours | 360 hours (15 days) |

| Resource | Schedule Name | Periodicity/Frequency | Retention |
|---|---|---|---|
| **Elasticsearch** | managed-es-backup | 72 hours | 360 hours (15 days) |
| **Redis** | managed-redis-backup | 72 hours | 360 hours (15 days) |
| **RabbitMQ** | managed-rmq-backup | 72 hours | 360 hours (15 days) |
| **Minio (Client)** | minio-s3-backup | 72 hours | 360 hours (15 days) |
| **MySQL** | cf-mysql-backup | 72 hours | 72 hours (3 days) |
| **Minio (Infrastructure)** | minio-backup | 72 hours | 72 hours (3 days) |
| **Cert Manager** | cert-manager-backup | 72 hours | 72 hours (3 days) |
| **DynamoDB** | dynamodb-backup | 72 hours | 72 hours (3 days) |
| **Identity** | identity-backup | 72 hours | 72 hours (3 days) |
| **Logging** | logging-backup | 72 hours | 72 hours (3 days) |
| **Monitoring** | monitoring-backup | 72 hours | 72 hours (3 days) |
| **Nginx** | nginx-reverse-proxy-backup | 72 hours | 72 hours (3 days) |
| **CF Operator** | cf-operator-backup | 72 hours | 72 hours (3 days) |
| **KubeCF** | kubecf-backup | 72 hours | 72 hours (3 days) |

TODO: Info on CCDB encryption key – how it is stored and used..

### 5.6.2. Service Continuity

HSOP platform is comprised of a kubernetes cluster with CF running on it. Client provisioned, services and applications run on CF. It is necessary to be able to recover both the platform components and services as well as client applications and their data.

The Recovery Time Objective of the platform itself occurs after the restoration of underlying dependencies including Infrastructure as a Service. After the underlying infrastructure services have been restored, HSP Operations will work to restore the impacted HSOP Platform.

| Service Recovery Plan Categories | *Response* |
|---|---|
| **Responsibility** | *HSP Operations, HSOP engineering team* |
| **External Service Dependencies** | *AWS* |
| **Service Dependencies** | *AWS Outposts* |
| **Service Impact** | *Partial or Full Client Service outage* |
| **Recovery Scenarios in Scope** | • *Failure of platform, components and services*<br>• *Failure of applications and services hosted on CF.*<br>• *Outposts hardware failure.* |
| **Recovery Strategy and Location** | *In-place recovery* |
| **Assumptions** | • *Connectivity to AWS region.* |
| **Recovery Time Objective (RTO) and Recovery Point Objective (RPO)** | • *RTO – 8 hours (subject to availability of high-speed network connectivity)*<br>• *RPO – 72 hours* |
| **Recovery Procedure** | *See Appendix-G Procedure to Restore Failed Services from Components* |
| **Test Procedure** | *See Appendix-G Procedure to Restore Failed Services from Components* |
| **Resume Procedure** | *N/A* |

### 5.7. Upgrade and update

When we want to upgrade the kOps cluster, we need to mention that Kubernetes version in */source/deployment_stable/k8s_version.txt* file.

E.g., if the current cluster has Kubernetes version of *v1.21.5* and we want to upgrade it to *v1.22.0*, we only need to update our /source/deployment_stable/k8s_version.txt file with the content ***v1.22.0*** in it.

Once we update this file, we should just run *kubernetes* module again where our *deploy.sh* script and *create_cluster.sh* script will compare the version mentioned in *k8s_version.txt* file and the current Kubernetes version of existing cluster. If there's mismatch found, kOps will update the cluster with the Kubernetes version mentioned in the *k8s_version.txt* file. Hence, we need to ensure that we mention the correct Kubernetes version in

## 6.0 Communications / Escalations

Each Service is required to maintain a list of engineers that are available to support the service. The individuals must be able to answer and resolve system level technical questions. HSOP utilizes an automated paging tool to meet this need. Please refer to team ENG-Microcloud and eng-microcloud@hsdp.pagerduty.com in hsdp.pagerduty.com for engineering contact details to allow the HSOP Operations and Support team access to resolve issues that may occur.

## 7.0 Service Key Performance Indicators (KPI)

| Item | Acceptable Objectives | Ultimate Objectives | Special Consideration |
|---|---|---|---|
| **Recovery Point Object** | 72 hours | 72 hours | |
| **Recovery Time Objective** | 8 hours | 8 hours | |
| **Data Retention** | Logs – 2160 hours<br>Infrastructure backup – 72 hours<br>Managed services backup – 360 hours | Logs – 2160 hours<br>Infrastructure backup – 72 hours<br>Managed services backup – 360 hours | |
| **System Availability** | 99.7% | | |

## 8.0 Billing

N/A

## 9.0 Certificate of Destruction Request

N/A

## 10.0 References

| Reference Number | Document Title | Document ID |
|---|---|---|
| **REF-1** | QMS Glossary | SNIP-R-000007 |
| **REF-2** | Operations Lifecycle Procedure | SNIP-P-060007 |
| **REF-3** | Service Agreement | SNIP-T-060007.05 |
| **REF-4** | IQ-OQ Work Instruction | SNIP-W-060007.01 |

## 11.0 Appendix-A Permission Table

Level I Support - T1 Operations User LDAP group

| **Environment: Prod** | Regional | Sites |
|---|---|---|
| **Resource** | **Permission** | **Permission** |
| AWS | None | None |
| HSOP Infrastructure LDAP* | Permission to change group membership | Permission to change group membership |
| Kubernetes | No Access | No Access |
| Cloudfoundry | N/A | N/A |
| Grafana | Not Available | Read |
| HSDP_Metrics | N/A | Read |
| Vault | No access | No access |

(*)T1 will have access to cloud LDAP not HSOP

Level 2 Support - T2 Operations User LDAP group

| **Environment: Prod** | Regional | Sites |
|---|---|---|
| **Resource** | **Permission** | **Permission** |

| | | |
|---|---|---|
| AWS | As defined in Appendix-B AWS IAM Permissions | As defined in Appendix-B AWS IAM Permissions |
| HSOP Infrastructure LDAP | On board/ Off board Ops Users | On board/ Off board Ops Users |
| Kubernetes | Admin access (read/write) | Admin access (read/write) |
| Cloudfoundry | N/A | Admin access |
| Grafana | Not Available | Read |
| HSDP_Metrics | N/A | Read |
| Vault | No access | No access |

Level 3 Support - T3 Operations User LDAP group

| **Environment: Prod** | **Regional** | **Sites** |
|---|---|---|
| **Resource** | **Permission** | **Permission** |
| AWS | As defined in Appendix-B AWS IAM Permissions | As defined in Appendix-B AWS IAM Permissions |
| HSOP Infrastructure LDAP | Admin access | Admin access |
| Kubernetes | Admin access (read/write) | Admin access (read/write) |
| Cloudfoundry | N/A | Admin access |
| Grafana | Not Available | Admin |
| HSDP_Metrics | N/A | Read |
| Vault | Read/Write | Read/Write |

Level 4 Support -T4 Operations User ( LDAP group – level4-support )

| **Environment: Prod** | **Regional** | **Sites** |
|---|---|---|
| **Resource** | **Permission** | **Permission** |
| AWS | None | None |
| HSOP Infrastructure LDAP | No Access | No Access |
| Kubernetes | Read-only access across all namespaces* | Read-only access across all namespaces* |
| Cloudfoundry | N/A | Read* |
| Grafana | Not Available | Read* |

| | | |
|---|---|---|
| HSDP_Metrics | N/A | Read* |
| Vault | No Access | No Access |

(*) Access provided by L3 based on need

## 12.0 Appendix-B AWS IAM Permissions

The following AWS permissions are required.

| Type | Value |
|---|---|
| AWSManagedPolicy | AdministratorAccess |
| Policy ARN | arn:aws:iam::aws:policy/AdministratorAccess |
| Policy Json |  |



## 13.0 Appendix-C Alert Resolution

| Alert Name | Alert Wait Duration | Severity | Description | Resolution |
|---|---|---|---|---|
| **CF APP RULES** | | | | |
| CFAppCrashed | 8 h | Critical | CF Application has not had any instance running during the last 8h | Restage the app: "cf restage <app-name" |
| ApplicationHighCpuUsage | 10 m | Critical | - | Restart the app: "cf restart <app-name" |
| ApplicationHighMemoryUsage | 10 m | Critical | CF App has exceeded Memory usage threshold of 85%. | Restart the app: "cf restart <app-name" |

| Alert Name | Alert Wait Duration | Severity | Description | Resolution |
|---|---|---|---|---|
| ApplicationHighDiskUsage | 10 m | Critical | CF App has exceeded disk usage threshold of 85%. | Restage the app: "cf restage <app-name" |
| **NODE EXPORTER RULES** | | | | |
| HostOutOfMemory | 2 m | Warning | Node memory is filling up (< 10% left) | 1. Find the top memory consuming pod in the host. Refer section Command to find top memory consuming pods in a node.<br>2. Delete the top memory consuming pod. |
| HostMemoryUnderMemoryPressure | 2 m | Warning | The node is under heavy memory pressure. High rate of major page faults | 1. Find the top memory consuming pod in the host. Refer section Command to find top memory consuming pods in a node.<br>2. Delete the top memory consuming pod. |
| HostHighCpuLoad | 0m | Warning | CPU load is > 80% | 1. Find the top cpu consuming pod in the host. Refer section Command to find top cpu consuming pods in a node .<br>2. Delete the top cpu consuming pod. |
| HostPhysicalComponentTooHot | 5m | Warning | Physical hardware component too hot | Raise ticket with AWS. |
| HostNodeOvertemperatureAlarm | 0m | Critical | Physical node temperature alarm triggered | Raise ticket with AWS. |
| **HAPROXY RULES** | | | | |
| HAProxyFrontendDown | 10m | Critical | Frontend HAProxy is not up (Status and their values; 0=STOP, 1=UP, 2=FULL) | SSH into HAProxy node & restart haproxy service.<br>systemctl restart haproxy |

| Alert Name | Alert Wait Duration | Severity | Description | Resolution |
|---|---|---|---|---|
| HAProxyServerNotUp | 10m | Critical | HAProxy Server is not up (Status and their values; 0=STOP, 1=UP, 2=MAINT, 3=DRAIN, 4=NOLB) | SSH into HAProxy node & start haproxy service. systemctl start haproxy |
| CFExporterApplicationsScrapeError | 10m | Critical | cf_exporter was unable to scrape Applications metrics during the last 10m. | Restart CF Exporter.<br>kubectl -n monitoring rollout restart deployment stratos-metrics-cf-exporter |
| **PROMETHEUS SELF MONITORING RULES** | | | | |
| PrometheusTargetMissing | 0m | Critical | A Prometheus target has disappeared. | Raise a ticket with HSOP Engineering team. |
| PrometheusAllTargetsMissing | 0m | Critical | A Prometheus job does not have living target anymore. | Restart Prometheus.<br>kubectl -n monitoring rollout restart deployment hsop-prometheus-kube-prome-operator |
| PrometheusNotConnectedToAlertmanager | 0m | Critical | Prometheus cannot connect the alertmanager. | Restart Prometheus.<br>kubectl -n monitoring rollout restart deployment hsop-prometheus-kube-prome-operator |
| PrometheusRuleEvaluationFailures | 0m | Critical | Prometheus rule has syntax errors. | Raise a ticket with HSOP Engineering team. |
| **Redis** | | | | |
| RedisDown | 0m | critical | Redis instance is down. | Restart the affected pod in managed-redis namespace.<br>kubectl -n managed-redis delete pod <pod_name> |
| **Postgres** | | | | |
| PostgresqlDown | 0m | critical | Postgresql instance is down. | Restart the affected pod in managed-rds namespace. |

| Alert Name | Alert Wait Duration | Severity | Description | Resolution |
|---|---|---|---|---|
| | | | | kubectl -n managed-rds delete pod <pod_name> |
| **MySql** | | | | |
| MysqlDown | 0m | critical | MySql instance is down. | Restart the affected pod in managed-rds namespace.<br>kubectl -n managed-rds delete pod <pod_name> |
| **Vault** | | | | |
| VaultUp | 15m | critical | Vault instance has been down for the last 15m. | Restart Vault.<br>kubectl -n vault rollout restart statefulset vault |
| VaultUninitialized | 30m | critical | Vault instance has been uninitialized for the last 30m. | Restart Vault.<br>kubectl -n vault rollout restart statefulset vault |
| VaultSealed | 0m | critical | Vault instance is sealed. | Restart Vault.<br>kubectl -n vault rollout restart statefulset vault |
| **Velero** | | | | |
| VeleroBackupFailures | 15m | warning | Velero backup has a certain percentage of failed backups. | Raise a ticket with HSOP Engineering team. |
| **Other Alerts** | | | | |
| Miscellaneous | | | | Raise a ticket with HSOP Engineering team. |



## 14.0 Appendix-E Fetching kubernetes client id and Secret from Keycloak

Instructions for fetching secrets from keycloak.

1. Navigate to **Error! Hyperlink reference not valid.**
2. Login to keycloak with **keycloakadminuser** and **keycloakadminpassword** credentials obtained from Appendix-D Fetching Secrets from Credhub
3. On the left panel, navigate to **Clients** tab.

4. Click **kubernetes** client on the table from the right panel.
5. Go to **Credentials** tab.
6. Copy the client secret from the field **Secret**.
7. The client id is **kubernetes** and client secret obtained from step 6 are the OIDC credentials required for authentication.

# 15.0 Appendix-H Procedure to Restore Failed Services from Components

**Prerequisites**

- Velero Client 1.6.3
- Access to kubernetes cluster as mentioned in Accessing kubernetes Cluster

HSOP platform comprises of several components and services. Recovery can be effected, either for an individual service or for full platform. The specific service(s) selected for recovery is incumbent upon the diagnosis of the failure.
For example, a full cluster restore is needed only in the rare occurrence of a complete rack failure.
**Note:** Restore procedure requires admin access to the kubernetes cluster and can only be performed by Level-3 support engineer.
**Note:** For HSOP MicroCloud v2.0, restoration will be performed by Tier4(R&D). This will be facilitated by Tier3(Ops) by enabling temporary Tier3 access for the assigned Tier4 engineer.

## 15.1. Procedure for Individual Resource Restore

1. For a service selected for restore, look up the schedule name from Backups.
2. Run the following command:
   **velero restore create <JOB_NAME> --from-schedule <SCHEDULE_NAME from Step 1>**
3. Wait for the completion of the restore job. Use the following command to view the status of the job:
   **velero restore get <JOB_NAME>**
4. Repeat steps 1 - 3 for other failed services.

## 15.2. Procedure for Full Cluster Restore

1. Deploy HSOP platform as described in Deployment Instructions.

2. Ensure the same baseline configuration is used as the previous deployment that needs to be restored.
   For example, use the same .tfvars file from the **hsonprem/source/deployment_stable/profiles** in the bitbucket repository as the previous deployment.
3. Execute the Procedure for Individual Resource Restore for the services ensuring the following order for restore.
   1. MySQL
   2. Minio(Infrastructure)
   3. Cert Manager
   4. CF Operator
   5. KubeCF
   6. Monitoring
   7. Vault
   8. Identity
   9. DynamoDB
   10. Logging
   11. Nginx
   12. Minio (Client)
   13. Redis
   14. RDS
   15. RabbitMQ
   16. Elasticsearch

## 15.3. Additional Instructions Post Recovery

Some of the services need additional steps to be performed post recovery or restoration of the service from backup. The following sections outline the additional steps required for these services. **The instructions are required to be followed only if the service mentioned below are restored.**

### 15.3.1. Restore Monitoring

When the Monitoring service from the Backup services list is restored, the following additional steps need to be performed.

**Prerequisites**

- Terraform 1.0.6

- cf cli v6
- git (v2.17.1)
- Clone of HSOP source repository
- Export the following environment variables with appropriate values:
    - AWS_ACCESS_KEY_ID, AWS_SECRET_ACCESS_KEY, AWS_REGION and AWS_DEFAULT_REGION.

**Procedure**
The following steps needs to be executed:

1. Navigate to the HSOP source repository folder.
2. Alert rules are not backed up need to be restored. Run the following commands to restore the alert rules.

```
cd scripts
./setup_env_vars.sh <PROFILE_FILE> # where PROFILE_FILE refers to the full path of the profile file used for deployment.
cd ../modules/alerting/
terraform init -backend-config="bucket=$TF_VAR_bucket_name" -backend-config="key=tfstate"
terraform select alerting-$TF_VAR_environment.$TF_VAR_domain_name
terraform taint null_resource.apply_alerting_rules
echo alerting > ../../configuration/mod.txt
export HSOP_CONF=mod
export HSOP_PROFILE=<PROFILE_NAME> # where PROFILE_NAME is the PROFILE_FILE name without extension
cd ../../
./deploy.sh
```

3. Restart the service proxy using the following commands.

```
cf l -a api.cf.$TF_VAR_domain_name -u <USERNAME> -p <PASSWORD> -o hsop -s services
cf restart service-proxy
```

### 15.3.2. Restore Managed Service

When the services listed below from the Backup services list is restored, the following additional steps need to be performed.

| Resource/Service | Impacted kubernetes namespace | Service Broker name |
|---|---|---|
| Vault | vault | vault-sb-app |

| | | |
|---|---|---|
| RDS | managed-rds | hsdp-rds |
| Elasticsearch | managed-es | hsdp-elasticsearch |
| Redis | managed-redis | hsdp-redis |
| RabbitMQ | managed-rmq | hsdp-rabbitmq |
| Minio (Client) | minio-s3 | hsdp-s3 |

**Prerequisite**

- cf cli v6
- Access to kubernetes cluster as mentioned in Accessing kubernetes Cluster
- jq v1.5
- yq v4.9.0

**Procedure**

Pick the kubernetes namespace and broker name from the table above for the impacted service & perform the following steps.

1. Export the kubernetes namespace name, service broker name and the domain name of the deployment as an environment variables.

```
export k8s_namespace=<kubernetes namespace from table>
export broker_name=<service broker name from table>
export domain_name=<domain name of the deployment>
```

2. Login to cloudfoundry with level3-support user credentials using the following command.

```
cf l -a api.cf.$domain_name -u <username> -p <password> -o hsop -s brokers
```

3. Update the service account token in the service broker and restage the service broker using the following commands.

```
cf set-env $broker_name K8S_BROKER_TOKEN \
  $(kubectl -n $k8s_namespace get secret \
    $(kubectl -n $k8s_namespace get secrets -o json | \
      jq -r '.items[] |
select(.metadata.annotations."kubernetes.io/service-
account.name"=="'${k8s_namespace}'-deploy") | .metadata.name') \
    -o yaml | yq e '.data.token' -)
cf restage $broker_name
```

## 16.0 Appendix- I List of Dashboards

| Dashboard | Description |
|---|---|
| **Alertmanager/Overview** | To get quick data about the number of alerts fired and other relevant parameters like alerts received rate, emails sent, etc., |
| **Prometheus/Overview** | To get quick data about the targets of Prometheus and other relevant parameters like scrape failures, query rate, etc., |
| **cf-app-overview** | To get metrics about cloud foundry apps and metadata |
| **node-monitoring** | Monitor all nodes - both kubernetes nodes and external nodes |
| **credhub dashboard** | credhub service specific metrics |
| **haproxy-monitoring** | haproxy service specific metrics |
| **LDAP** | ldap service specific metrics |
| **etcd** | kuberenetes etcd specific metrics |
| **core-dns** | kubernetes core-dns specific metrics |
| **kubernetes / API server** | kubernetes api server specific metrics |
| **Kubernetes / Compute Resources / Cluster** | kubernetes compure resources metrics at cluster level |
| **Kubernetes / Compute Resources / Namespace (Pods)** | kubernetes compure resources metrics by namespace with pod level filter |
| **Kubernetes / Compute Resources / Namespace (Workloads)** | kubernetes compure resources metrics by namespace with workload level filter |
| **Kubernetes / Compute Resources / Node (Pods)** | kubernetes compure resources metrics by namespace with workload level filter |
| **Kubernetes / Compute Resources / Pod** | kubernetes compure resources metrics by pod |
| **Kubernetes / Compute Resources / Workload** | kubernetes compure resources metrics by workload |
| **Kubernetes / Controller Manager** | kubernetes controller manager performance metrics |
| **Kubernetes / Kubelet** | kubernetes - kubelet specific metrics |
| **Kubernetes / Networking / Cluster** | kubernetes networking metrics at the cluster level |
| **Kubernetes / Networking / Namespace (Pods)** | kubernetes networking metrics by namespace with pod level filter |
| **Kubernetes / Networking / Namespace (Workload)** | kubernetes networking metrics by namespace with workload level filter |
| **Kubernetes / Networking / Pod** | kubernetes networking metrics by pod level |
| **Kubernetes / Networking / Workload** | kubernetes networking metrics by workload level |
| **Kubernetes / Persistent Volumes** | kubernetes persistant volumes utilization and other relevant parameters |
| **Kubernetes / Proxy** | kubernetes proxy component metrics |
| **Kubernetes / Scheduler** | kubernetes scheduler component metrics |
| **Kubernetes / StatefulSets** | kubernetes statefulsets metrics |

| Dashboard | Description |
|---|---|
| **Node Exporter / Nodes** | kubernetes node level metrics 1 - only kubernetes nodes |
| **Node Exporter / USE Method / Cluster** | kubernetes cluster level metrics - only kubernetes nodes |
| **Node Exporter / USE Method / Node** | kubernetes node level metrics 2 - only kubernetes nodes |
| **postgres-monitoring** | This dashboard works with postgres_exporter to show postgres service specific metrics |
| **rabbitmq-monitoring** | Rabbitmq stats and alerting using RabbitMQ Exporter |
| **redis-monitoring** | Redis Dashboard for redis service specific metrics |
| **vault-monitoring** | Hashicorp Vault service specific metrics |
| **velero dashboard** | This dashboard works with velero for velero specific metrics |
| **minio-monitoring** | minio service specific metrics |
| **mysql-monitoring** | mysql service specific metrics |
| **Elasticsearch** | NA |
| **elasticsearch-monitoring** | NA |
| **Elasticsearch complete** | Elasticsearch service specific metrics |

# 17.0 Appendix-J Troubleshooting

## 17.1. Deployment

### 17.1.1. Deployment of HSOP Region fails with errors

- Run the HSOP region deployment script again.
```
./hsop --update --region <aws_region_name> --modules "*" --verbose
```
- If the issue persists, note down the module name where the error was encountered.
- Teardown the module in which the error occurred.
```
./hsop --teardown --region <aws_region_name> --modules <module_name> --verbose
```
- Restart the HSOP region deployment.
```
./hsop --update --region <aws_region_name> --modules "*" --verbose
```

### 17.1.2. Deployment of HSOP Site fails with errors

- HSOP Site deployment can be retriggered by making a modification to the site manifest file & repeating the steps 5 through 6 of section Site Deployment Instructions.
**Note:** An insignificant change, for example updating the site address attribute with a comma, can be used to retrigger the deployment.

#### 17.1.3. Deployment fails with state is locked

To avoid inconsistencies due to multiple people trying to modify same infrastructure resources Terraform state fille will be locked during deployment. At times for some reasons when deployment is abruptly stopped then there is a possibility of state lock being not released. In such cases any attempt to redeploy will fail with "Error acquiring the state lock".

Sample output will be like as follows, indicating who has locked the state and at what time:

```
Deploying module cloudfoundry ...

Acquiring state lock. This may take a few moments...

  Error: Error acquiring the state lock

  Error message: ConditionalCheckFailedException: The conditional request failed
  Lock Info:
    ID:        60d991f2-500a-947a-eeed-59210ff5656a
    Path:      hsop-us-west-2/env:/cloudfoundry-prod.cicd.hs-premise.com/tfstate
    Operation: OperationTypePlan
    Who:       root@df057ebc73a5
    Version:   1.0.6
    Created:   2022-05-17 17:30:16.459697402 +0000 UTC
    Info:


  Terraform acquires a state lock to protect the state from being written
  by multiple users at the same time. Please resolve the issue above and try
  again. For most commands, you can disable locking with the "-lock=false"
  flag, but this is not recommended.

cloudfoundry plan failed.
```

- In such cases following command can be executed:
- aws dynamodb delete-table --table-name <environment>.<domain name>

### 17.2. Teardown

#### 17.2.1. Error occurs during full teardown of a deployment

- Resume teardown by executing the teardown.sh script with -f yes argument
  **./teardown.sh -f yes**

### 17.3. kubectl command fails with the error - The connection to the server localhost:8080 was refused - did you specify the right host or port?

- Ensure openssl is installed on the system.
- Download certificate of the kubernetes cluster & save the base64 encoded certificate string to a file using the following command.

```
echo $(true | openssl s_client -connect api.k8s.<domain_name>:6443 2>/dev/null
| openssl x509) | base64 | tr -d '\n' > k8s_ca_cert
```

- Edit the kube config file, located at *~/.kube/config*, to look like below.

```
apiVersion: v1
clusters:
  - cluster:
      server: https://api.k8s.<domain_name>:6443  # domain name from profile
```

```
file
      certificate-authority-data: <base64_encoded_certificate>  # replace with
content of file k8s_ca_cert
    name: k8s.<domain_name>   # domain name from profile file
contexts:
  - context:
      cluster: k8s.<domain_name>  # domain name from profile file
      user: oidc
    name: k8s.<domain_name>   # domain name from profile file
current-context: "k8s.<domain_name>"  # domain name from profile file
kind: Config
preferences: {}
users:
- name: oidc
  user:
    exec:
      apiVersion: client.authentication.k8s.io/v1beta1
      args:
      - oidc-login
      - get-token
      - --oidc-issuer-url=https://identity.cf.<domain_name>/auth/realms/hsop  #
domain name from profile file
      - --oidc-client-id=<oidc_client_id>   # oidc client id from keycloak
      - --oidc-client-secret=<oidc_client_secret>   # oidc client secret from
keycloak
      command: kubectl
      env: null
      provideClusterInfo: false
```

- Export environment variable KUBECONFIG to point to the kube config file path as below.

```
export KUBECONFIG=~/.kube/config
```

- Rerun the kubectl command.

### 17.4. ssh to bastion host fails with error - SSH Permission denied (publickey)

- Run the ssh command as follows.

```
ssh -o PreferredAuthentications=password -o PubkeyAuthentication=no
<username>@bastions.<domain_name>
```

### 17.5. Client requests to increase disk space/storage for a managed service

**Note:** Volume expansion requires admin access to the kubernetes cluster and can only be performed by Level-3 support engineer.

**Note:** For HSOP MicroCloud v2.0, volume expansion will be performed by Tier4(R&D). This will be facilitated by Tier3(Ops) by enabling temporary Tier3 access for the assigned Tier4 engineer.

- Login to kubernetes as level3 user, refer section Accessing kubernetes Cluster.

- Enable volume expansion.
  - Run the following command.

```
kubectl get sc ebs-sc --no-headers | awk '{ print $6 }'
```

  - If the output of the previous command is *false*, run the following command.

```
kubectl patch storageclass \
  ebs-sc \
  --type='json' \
  -p='[{"op": "add", "path": "/allowVolumeExpansion", "value": true}]'
```

- Identify the persistent volume claim to be expanded from the below table.
  - For managed services, except hsdp-s3 & hsdp-vault, login to cloudfoundry as level3 user & obtain the service guid.

```
cf service <service name> --guid    # replace service name here
```

| Service | kubernetes namespace | PVC name |
|---|---|---|
| hsdp-s3 | minio-s3 | client-s3-minio |
| hsdp-vault | vault | data-postgresql-postgresql-0 |
| hsdp-redis-db | managed-redis | redisdb5-redis-<*service guid*>-0 |
| hsdp-rds (postgres) | managed-rds | postgredb-postgres-<*service guid*>-0 |
| hsdp-rds (mysql) | managed-rds | data-mysql-<*service guid*>-0 |
| hsdp-rabbitmq | managed-rmq | see command below* |
| hsdp-elastic | managed-es | see command below for service guid**<br>elasticsearch-data-es-<*service guid*>-es-default-0 |

* run the below command to get pvc name for rabbitmq.

```
export SVC_NAME=$(kubectl -n managed-rmq get statefulset \
          -l cf_instance_id=<service guid> \
          -o yaml |  yq e '.items.0.metadata.name' -)

kubectl -n managed-rmq \
  get pvc data-${SVC_NAME}-0 \
  --no-headers -o custom-columns=":metadata.name"
```

** run the below command to get service guid for elastic search.

```
cf service <service name> --guid | tr -d '-'
```

- Update the volume size with the following command.

```
kubectl patch pvc \
  <pvc name> \
  --namespace <namespace name> \
  -p '{"spec":{"resources":{"requests":{"storage":"<new size>Gi"}}}}}'
```

- Restart the corresponding statefulset or deployment as per the below table, using the following command.

```
kubectl -n <namespace name> scale <type> <name> --replicas=0
sleep 20
kubectl -n <namespace name> scale <type> <name> --replicas=1
sleep 30
```

| Service | kubernetes namespace | Type | Name |
|---|---|---|---|
| hsdp-s3 | minio-s3 | deployment | client-s3-minio |
| hsdp-vault | vault | statefulset | postgresql-postgresql |
| hsdp-redis-db | managed-redis | statefulset | redis-<*service guid*> |
| hsdp-rds (postgres) | managed-rds | statefulset | postgres-<*service guid*> |
| hsdp-rds (mysql) | managed-rds | statefulset | mysql-<*service guid*> |
| hsdp-rabbitmq | managed-rmq | | see command below* |
| hsdp-elastic | managed-es | | see command below for service guid**<br>es-<*service guid*>-es-default |

* run the below command to get the statefulset name for rabbitmq.

```
export NAME=$(kubectl -n managed-rmq get statefulset \
          -l cf_instance_id=<service guid> \
          -o yaml |  yq e '.items.0.metadata.name' -)
```

** run the below command to get service guid for elastic search.

```
cf service <service name> --guid | tr -d '-'
```

## 17.6. Cluster is out of resources – CPU/Memory

**Note:** Cluster expansion requires admin access to the kubernetes cluster and can only be performed by Level-3 support engineer.

**Note:** For HSOP MicroCloud v2.0, volume expansion will be performed by Tier4(R&D). This will be facilitated by Tier3(Ops) by enabling temporary Tier3 access for the assigned Tier4 engineer.

- Run the following commands to extend the cluster.

```
# Run the docker container
docker run --rm --mount src="$(pwd)",target=/deployment,type=bind -w
/deployment -i -t hsop/driver:1.0 sh

# Run the subsequent commands in the docker shell
```

```
# Set kubeconfig
export DOMAIN_NAME=<domain-name> # Replace the <domain-name> with value from
the profiles file
export KOPS_STATE_STORE=s3://<bucket-name>/kops # Replace <bucket-name> with
value from the profiles file
./modules/kubernetes/scripts/proxy_aware_kops_export_kubecfg.sh
k8s.$DOMAIN_NAME

# Change directory to scripts folder
cd scripts
chmod +x extend_k8s_cluster.sh

# Run cluster expansion script
./extend_k8s_cluster.sh -c all -p <profile-name>  # Replace <profile-name> with
the name of the profile file without extension

# Exit the docker container
exit
```

**Note:** If the cluster is extended already, you'll need to give a unique name for the *InstanceGroup* by editing the *extend_cluster.tpl* file under *modules/extend-cluster* folder in line numbers 9 & 16.

### 17.7. Client requests for VM or Cartel instance

TODO: with APIs

## 18.0 Appendix- L Common kubectl commands

### 18.1. Command to find top memory consuming pods in a node

```
export NODE_NAME=<node>  # Replace node name here
kubectl get po -A -o wide | grep ${NODE_NAME} | awk '{print $1, $2}' \
  | xargs -n2 kubectl top pod --no-headers --use-protocol-buffers -n $1 \
  | sort --key 3 -nr | column -t
```

### 18.2. Command to find top cpu consuming pods in a node

```
export NODE_NAME=<node>  # Replace node name here
kubectl get po -A -o wide | grep ${NODE_NAME} | awk '{print $1, $2}' \
  | xargs -n2 kubectl top pod --no-headers --use-protocol-buffers -n $1 \
  | sort --key 2 -nr | column -t
```

## 19.0 Appendix- S IAM deployment prerequisites

On completion of MicroCloud deployment following pre-requisites required for IAM will be creates

- K8S namesapce by name “iam”
- Service account in the iam namespace with the following permissions
  - Full permission for iam namespace
  - Read permission for router-public secret in certificates namespace
- KubeConfig which can be used to deploy IAM will be available in vault at cf/<secret-path>/<k8s.domainname>/*iamkubeconfig* (refer section 14)
- Profile file used to deploy the HSOP can be used to get the details required to deploy the IAM

## 20.0 Appendix – U Setup SSH tunnel through the central Bastion host in a HSOP region

This section describes how to setup a SSH tunnel through the central Bastion host in a HSOP region, to access kubernetes, cloudfoundry & other services & applications running on-premises. The same steps can be used to also access the HSOP control plane EKS cluster that is deployed in a HSOP region.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)

**Note:**

If you are using WSL2 on Windows, sshuttle needs to be installed on the WSL2 Linux environment & OpenSSH should be installed on Windows.

**Commands to start the SSH tunnel:**

**sshuttle:**

In a terminal window, run the following command. Note, you can omit the –daemon flag to run sshuttle as a foreground/blocking process.

```
sshuttle --dns --daemon -vr \
  <ldap_username>@bastion.<stage>-<aws_region_short_name>.<apex_domain> \
  10.16.0.0/16 100.64.0.0/10 \
  --ssh-cmd 'ssh -i /path/to/your/id_rsa -o "StrictHostKeyChecking no"'
```

**OpenSSH:**

In a command prompt window, run the following command.

```
ssh -o "StrictHostKeyChecking no" -N -D <proxy_port_number> ^
  -i drive_letter:\path\to\your\id_rsa ^
  <ldap_username>@bastion.<stage>-<aws_region_short_name>.<apex_domain>
```

**Note**: The above command runs a SOCKS proxy to tunnel via the Bastion host. In order to use the tunnel, you’ll need to configure Windows to use the SOCKS proxy for all HTTP communication. Run the following command in a Windows command prompt.

```
netsh winhttp set proxy ^
  proxy-server="socks=localhost:<proxy_port_number>" ^
  bypass-list="localhost"
```

**Commands to stop the SSH tunnel:**

**sshuttle:**

In a terminal window, run the following command.

```
pkill --signal SIGTERM sshuttle
```

**Note:** Do not use SIGKILL(9) as this can leave the iptables in an inconsistent state forcing a system restart. Always use SIGTERM(15).

**OpenSSH:**

- Close the windows where the OpenSSH command for starting the SOCKS proxy was run.
- Reset the Windows proxy environment settings by running the following command in a command prompt window.

```
netsh winhttp reset proxy
```

## 21.0 Appendix – V Provide access to other AWS IAM users after deployment of HSOP Control Plane (or the regional EKS cluster)

This section describes how to provide additional AWS IAM users access to the regional EKS cluster.

**Prerequisites:**

- aws-cli version 2.2.4 (for all Operating Systems)
- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)
- kubectl v1.22.3 (for all Operating Systems)

**Steps:**

1. Perform the steps described in section Appendix – W Access HSOP Control Plane (regional EKS cluster).
2. Update the **aws-auth ConfigMap** in *kube-system* namespace with the user you wish to add using the following command.

```
USER="    - userarn: <arn_of_the_aws_iam_user_to_add>\n      username: admin\n
groups:\n        - system:masters"
kubectl get -n kube-system configmap/aws-auth -o yaml | awk "/mapUsers:
\|/{print;print \"$USER\";next}1" > /tmp/aws-auth-patch.yml
kubectl patch configmap/aws-auth -n kube-system --patch "$(cat /tmp/aws-auth-
patch.yml)"
```

**Note:** The above command assumes that the mapUsers attribute exists in the **aws-auth ConfigMap** and makes the new user an admin of the EKS cluster. For a more detailed explanation, please read AWS documentation at https://docs.aws.amazon.com/eks/latest/userguide/add-user-role.html.

**Note:** If you would like to edit the aws-auth ConfigMap manually, you can run: $ kubectl edit -n kube-system configmap/aws-auth

## 22.0 Appendix – W Access HSOP Control Plane (regional EKS cluster)

This section describes how to access the HSOP regional control plane or the EKS cluster.

**Prerequisites:**

- aws-cli version 2.2.4 (for all Operating Systems)
- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)
- kubectl v1.22.3 (for all Operating Systems)

**Note:**

Access control for the EKS cluster is managed through AWS user accounts & the **aws-auth ConfigMap** in the EKS cluster. In order to access the EKS cluster you should have either the cluster_creator role for the cluster or should be added to **aws-auth ConfigMap** with the correct kubernetes username.

**Steps:**

1. Establish a ssh tunnel through the desired regional bastion host as described in section Appendix – U Setup SSH tunnel through the central Bastion host in a HSOP region.
2. Export AWS user account related environment variables in a terminal window.
```
export AWS_ACCESS_KEY_ID=<your aws access key id>
export AWS_SECRET_ACCESS_KEY=<your aws secret access key>
export AWS_REGION=<the aws region>
```
3. Update the local kube config file using the following command in the same terminal window.
```
aws eks update-kubeconfig --name hsop-eks-<stage>-<aws_region_short_name>
```

Upon completion of the above steps, you should be able to access the EKS cluster using kubectl commands.

```
# for example
kubectl get nodes
```

## 23.0 Appendix – X Access central Vault instance in a region

This section describes how to access the central Vault instance in a region.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)

**Steps:**

1. Establish a ssh tunnel through the desired regional bastion host as described in section Appendix – U Setup SSH tunnel through the central Bastion host in a HSOP region.
2. If you are using SOCKS proxy, update system proxy settings to use the SOCKS proxy using the below command.
```
netsh winhttp set proxy ^
  proxy-server="socks=localhost:<proxy_port_number>" ^
  bypass-list="localhost"
```
Alternatively, you can also update the proxy settings in your browser to use the SOCKS proxy. The specific instructions to update proxy settings is dependent on the browser used. Refer to your browser's documentation on updating proxy settings.
3. In your web browser, navigate to https://vault.<stage>-<aws_region_short_name>.<apex_domain>/login
Example: If stage name is <<nine>> and aws_region_short_name is <<usw2>> and apex domain name is <<hsop.in>> below would be the argocd URL

URL: http://vault.nine-usw2.hsop.in/ui

4. Select the LDAP method for authentication & login using your HS Cloud ldap credentials.

   Alternatively, you can also access Vault from the command line.

## 24.0 Appendix – Y Access ArgoCD in a region

This section describes how to access ArgoCD instance in a region.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)

**Steps:**

5. Establish a ssh tunnel through the desired regional bastion host as described in section Appendix – U Setup SSH tunnel through the central Bastion host in a HSOP region.
6. If you are using SOCKS proxy, update system proxy settings to use the SOCKS proxy using the below command.

```
netsh winhttp set proxy ^
  proxy-server="socks=localhost:<proxy_port_number>" ^
  bypass-list="localhost"
```

   Alternatively, you can also update the proxy settings in your browser to use the SOCKS proxy. The specific instructions to update proxy settings is dependent on the browser used. Refer to your browser's documentation on updating proxy settings.
7. In your web browser, navigate to https://argocd.<stage>-<aws_region_short_name>.<apex_domain>/login
   Example: If stage name is <<nine>> and aws_region_short_name is <<usw2>> and apex domain name is <<hsop.in>> below would be the argocd URL
   URL: https://argocd.nine-usw2.hsop.in/login
8. Obtain ArgoCD admin password from central Vault instance at path *hsop/cloud/outputs/argocd_admin_password*. See section Appendix – X Access central Vault instance in a region to access the Vault instance.
9. Login to ArgoCD using the admin credentials obtained from previous step.

## 25.0 Appendix – Z Access central LDAP instance in region

This section describes how to access the central LDAP instance in a region.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)

**Steps:**

10. Establish a ssh tunnel through the desired regional bastion host as described in section Appendix – U Setup SSH tunnel through the central Bastion host in a HSOP region.
11. If you are using SOCKS proxy, update system proxy settings to use the SOCKS proxy using the below command.

```
netsh winhttp set proxy ^
  proxy-server="socks=localhost:<proxy_port_number>" ^
  bypass-list="localhost"
```

Alternatively, you can also update the proxy settings in your browser to use the SOCKS proxy. The specific instructions to update proxy settings is dependent on the browser used. Refer to your browser's documentation on updating proxy settings.

12. In your web browser, navigate to https:// phpldapadmin.<stage>-<aws_region_short_name>.<apex_domain>/login
    Example: If stage name is <<nine>> and aws_region_short_name is <<usw2>> and apex domain name is <<hsop.in>> below would be the argocd URL
    URL: https://phpldapadmin.nine-usw2.hsop.in/
    Login using your HS Cloud credentials.

## 26.0 Appendix – AA Check status of a HSOP site

This section describes how to access ArgoCD instance in a region & check the status of a HSOP site.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)

**Steps:**

1. Login to ArgoCD as described in section Appendix – Y Access ArgoCD in a region.
2. In the Applications tab select the *microcloud* application by clicking on the card titled **microcloud**.
3. Select the site by clicking on the card with the required sitename in the title.
4. Scroll to the bottom on the page in the LIVE MANIFEST tab & verify the value of the state attribute.

## 27.0 Appendix – AB Access Cloud Foundry instance of a HSOP site

This section describes how to access Cloud Foundry instance of a HSOP site.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)
- cf cli v6 (for all OS)

**Steps:**

1. Establish a ssh tunnel through the desired regional bastion host as described in section Appendix – U Setup SSH tunnel through the central Bastion host in a HSOP region.
2. Login to Cloud Foundry using your HS Cloud LDAP credentials.

```
cf api api.cf.<sitename>.<stage>-<aws_region_short_name>.<apex_domain>
cf login
# Provide your HS Cloud LDAP credentials when prompted for Email & Password
```

## 28.0 Appendix – AC Create new Cloud Foundry Org & Space in a HSOP site

This section describes how to create a new Cloud Foundry Org & Space in a HSOP site.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)
- cf cli v6 (for all OS)

**Steps:**

1. Login to the Cloud Foundry instance of a HSOP site as described in section Appendix – AB Access Cloud Foundry instance of a HSOP site.
2. Run the following command to create a new Cloud Foundry Org. Skip this step if the Org already exists.
```
cf create-org <ORG_NAME>
```
3. Run the following command to create a new Cloud Foundry Space. Skip this step if the Space already exists.
```
cf create-space <SPACE_NAME> -o <ORG_NAME>
```

## 29.0 Appendix – AD Assign Cloud Foundry Org & Space roles to a User in a HSOP site

This section describes how to assign Cloud Foundry Org & Space roles to a user in a HSOP site.

**Prerequisites:**

- sshuttle 0.78.3 (for drawin/Linux/WSL2)
- Latest version of OpenSSH (for Windows only)
- cf cli v6 (for all OS)

**Steps:**

1. Login to the Cloud Foundry instance of a HSOP site as described in section Appendix – AB Access Cloud Foundry instance of a HSOP site.
2. Run the following command to assign Org role to a user.
```
cf set-org-role <USERNAME> <ORG_NAME> <ORG_ROLE>.
# ORG ROLES
# OrgManager - Invite and manage users, select and change plans, and set spending limits
# BillingManager - Create and manage the billing account and payment info
# OrgAuditor - Read-only access to org info and reports
```
3. Run the following command to assign Space role to a user.
```
cf set-space-role <USERNAME> <ORG_NAME> <SPACE_NAME> <SPACE_ROLE>.
# SPACE ROLES
# SpaceManager - Invite and manage users, and enable features for a given space
# SpaceDeveloper - Create and manage apps and services, and see logs and reports
# SpaceAuditor - View logs, reports, and settings on this space
```
4. Repeat steps 2 & 3 to assign additional Org & Space roles to the user.

## 30.0 Appendix – AE AWS Resource requirement & quotas

This section outlines the AWS resources & quotas required for HSOP region & site deployments.

| Resource | HSOP Region | HSOP Site | Quota Name | Minimum Quota |
|---|---|---|---|---|
| **VPC** | 2 | 1 | VPC | 20 |
| **Subnets** | 5 | 4 | Subnets | 20 |
| **CIDR blocks** | 4 | 3 | IPv4 CIDR blocks per VPC | 5 |
| **Elastic IP** | 1 | 0 | Elastic IP addresses per Region | 5 |
| **Internet gateways** | 1 | 0 | Internet gateways per Region | 5 |
| **NAT gateways** | 5 | 2 | NAT gateways per Availability Zone | 20 |
| **Route tables** | 7 | 4 | Route tables per VPC | 20 |
| **Security groups** | 5 | 3 | VPC security groups per Region | Default |
| **Gateway VPC endpoints** | 1 | 1 | Gateway VPC endpoints per Region | Default |
| **Interface VPC endpoints** | 11 | 11 | Interface and Gateway Load Balancer endpoints per VPC | Default |
| **Transit gateways** | 2 | 0 | Transit gateways per VPC | 10 |
| **TGW route tables** | 3 | 0 | Transit gateway route tables per transit gateway | Default |
| **TGW Static routes** | 4 | 2 | Static routes per transit gateway | Default |
| **TGW Attachments** | 4 | 2 | Attachments per transit gateway | Default |

## 31.0 Broker Service Plans not available in HSOP

HSOP brokers doesn't support high availability and cluster mode plans for brokers. Below is the list of service plans which are not available in HSOP but available in HSDP broker service plans catalogue.

| Broker | Plans not available in HSOP | Additional Plans in HSOP |
|---|---|---|
| Elastic Service Broker | highmem16 | |
| RDS Service Broker | *-medium-ha-dev | postgres-5th-gen-large<br>postgres-5th-gen-xlarge<br>mysql-large |

| | | |
|---|---|---|
| Redis-DB Service Broker | redis-development-cluster<br>redis-cluster | nano-dev |

For *RabbitMQ-Server Service Broker*, the service name in HSDP is “hsdp-rabbitmq-server” whereas in HSOP it is “hsdp-rabbitmq”

## 32.0 GPU and Windows instance creation

<TODO>

## 33.0 RDS Snapshot and Restore Feature

RDS Snapshot feature operations are implemented using the velero component. So following pre-requisites are important for making the RDS snapshot feature work.

Pre-Requisites:

1. Ready to use kops cluster deployed with RDS service broker
2. Velero Component installed and configured in velero namespace.

1. Create Manual Snapshot

Command:

```
cf update-service test_pg_instance -c '{"Action": "create-snapshot"}'
```

2. List Snapshot

Command:

```
cf bind-service hsdp-rds test_pg_instance_100 -b testrdsbroker -c '{"Action": "list-snapshots"}'
```

3. Delete Snapshot

Command:

```
cf update-service test_pg_instance_100 -c '{"Action": "delete-snapshot", "backupName": "postgres-e5cf12f8-48d6-4bd0-8051-09d0fba362f3-20220607101152"}'
```

4. Restore Snapshot

Command:

```
cf create-service hsdp-rds postgres-micro-dev test-pg-restore -b testrdsbroker -c '{"Action": "restore-snapshot", "backupName": "postgres-d47233df-6051-44b9-8ed3-4bfd0668886a-20220620123633"}'
```

**Velero References:**

**Backup** - Velero Docs - Backup Reference

**Restore** - Velero Docs - Restore Reference

## 34.0 Document Revision History

| Version | Date | Author | Description of changes |
|---|---|---|---|
| 2.1 | 30-Jun-2022 | Vikram Rao | Initial Draft |
| | | | |