VOSS Automate Installation Guide

Release 21.3

Jun 07, 2024

Legal Information

Please take careful note of the following legal notices:

Security Information

This product may contain cryptographic features that may be subject to state and local country laws that govern the import, export, transfer and use of such features. The provision of this software does not imply that third-party authorization to import, export, distribute or use encryption in your particular region has been obtained. By using this product, you agree to comply with all applicable laws and regulations within your region of operation. If you require further assistance, please contact your dedicated VOSS support person.

Contents

1 What's New **1**

2 Introduction **2**
2.1 Overview . . . 2
2.2 Command Conventions . . . 2

3 Deployment Topologies **3**
3.1 Architecture Offerings . . . 3
3.2 Multi Data Center Deployments . . . 5
3.3 Clustering Considerations . . . 8
3.4 Network Communications between Nodes within the Cluster . . . 9
3.5 Network Communications External to the Cluster . . . 10
3.6 Cluster Commands . . . 12
3.7 Geo-redundancy/Redundancy And DR Synopsis . . . 13
3.8 Unified Node Topology . . . 13
3.9 Modular Cluster Topology . . . 21

4 Prepare to Install **25**
4.1 Installation Prerequisites with Cisco HCM-F . . . 25
4.2 Network Docker Container Range . . . 25
4.3 Backup Size Considerations . . . 26
4.4 Unified Node Topology Preparation . . . 27
4.5 Modular Cluster Topology Preparation . . . 29

5 Install VOSS Automate **32**
5.1 Installation Process . . . 32
5.2 Create a New VM Using the Platform-Install OVA . . . 32
5.3 Using the `screen` command . . . 37
5.4 Installation Logs . . . 38
5.5 Primary Node Role in a Cluster . . . 41
5.6 Unified Node Topology Installation . . . 41
5.7 Modular Cluster Topology Installation . . . 54

6 VOSS Automate Azure Installation **59**
6.1 Installing Azure . . . 59
6.2 Deploying Additional Azure Nodes . . . 67

Index **69**

1. What’s New

2. Introduction

2.1. Overview

Note: For upgrade steps, refer to the Upgrade Guide.

This document provides an overview of the deployment of the VOSS Automate system on VMware.

The system can be deployed in either a test single node topology, or a multi-node cluster with High Availability and Disaster Recovery capabilities. It is aimed at Technical and Operational personnel responsible for the deployment of a VOSS Automate system.

This system supports various deployments/solutions - the document describes the product in general and is not specific to a particular deployment/solution. For cluster installations, also refer to the Health Checks for Cluster Installations Guide.

Information may vary slightly depending on the installation environment.

2.2. Command Conventions

For file transfers, either **sftp** or **scp** can be used. For example, either:

1. **sftp platform@<node_hostname>**
2. **cd media**
3. **put <upgrade_template_file>**

or:

scp <upgrade_template_file> platform@<node_hostname>:~/media

In this Guide, **scp** is used to show file transfer.

3. Deployment Topologies

3.1. Architecture Offerings

VOSS Automate offers a range of deployment topologies. The choice of a deployment topology should take into consideration the advantages and disadvantages of each as shown below.

- Single-node cluster (cluster-of-one/standalone)
- Single-node cluster (cluster-of-one/standalone) with VMWare HA
- Unified Node Cluster Topology
  - 2 Node with Web proxies
  - 4 Node with Web proxies
  - 6 Node with Web proxies
- Modular Cluster Topology (separate Application and Database nodes)
  - 3 Database nodes
  - 1 - 8 Application Nodes
  - Web Proxies
- Cloud deployments
  - Azure
  - Google Cloud Platform (GCP)
  - Support all Standalone, Unified and Modular cluster topologies
- MaaS (Management-as-a-Service)
  - VOSS hosted SaaS solution

Topology	Pro's	Con's
Single-node[1]	• Smallest hardware footprint	• No DR • Less throughput than clusters
Single-node with VMWare HA	• Smallest hardware footprint • DR available	• Less throughput than clusters
3 Unified Node Cluster	• More throughput than single-node	• More limitations with DR scenarios • More hardware than single-node
4 Unified Node Cluster	• More DR scenarios supported • More throughput than 3 Node	• More hardware than 3 Node
6 Unified Node Cluster	• Typically deployed for multi-data center deployments • Support Active/Standby	• Largest hardware footprint
Modular Cluster	• Increased processing capacity utilization on Application Nodes • Horizontal scaling by adding more Application Nodes • Improved database resilience with dedicated nodes and isolation from application • Improved database performance by removing application load from the primary database	
Cloud Deployment (Azure, GCP)	• Leverage cloud tooling like proxies (can use instead of VOSS Web Proxy)	
MaaS	• No hardware footprint or infrastructure costs • Fast setup	

3.2. Multi Data Center Deployments

Multinode clusters can be deployed in both Active/Active or Active/Standby configurations. Active/Active configurations have all the Unified nodes enabled for transaction processing. Active/Standby configurations have only the Unified nodes in the primary data center (the data center containing the Unified node with the primary database) enabled for transaction processing.

In order to run an Active/Active configuration, the latency (RTT) between data centers must not exceed 20ms. For higher latencies an Active/Standby configuration must be used.

To switch to an Active/Standby configuration, do the following on all Unified nodes in the secondary data center (the data center *not* containing the Unified node with the primary database):

1. log into platform:

   ssh platform@<ip_address>

2. set the worker count on the nodes in the secondary data center to zero:

   voss workers 0

   Setting the number of workers is persistent; in other words, this setting will still apply after upgrades and system restart.

For an Active/Standby configuration, the proxy server web weights should be set to the Unified nodes on the primary data center. This is done with the **web weight add** command.

The web weight specifies the routing and relative counts of the initial HTTP request from the Web Proxy to a Unified Node. The initial request could be a request such as a transaction, or for example a GET request. Consider web weights configuration as shown below:

```
172.0.0.158:443: 1
172.0.0.159:443: 1
172.0.0.161:443: 1
172.0.0.162:443: 1
173.0.0.163:443: 0
173.0.0.164:443: 0
```

This configuration means that the servers `173.0.0.163` and `173.0.0.164` serve as backup servers and requests are only routed to these if the other servers are not available. While the other servers are available, an equal number of requests are routed to them in a round-robin manner. Refer to the Best Practices Guide for more details on deployment models and web weight settings.

Consider an example where:

Note: The examples below show system command output where the Phone Based Registration application is running.

---

1. Primary data center has unified nodes with IP addresses and ports:
   - 172.0.0.158:443
   - 172.0.0.159:443
   - 172.0.0.161:443
   - 172.0.0.162:443

[1] This deployment should be used for test purposes only.

A *unified node* typically shows the following output for the **web weight list** command:

```
$ web weight list
Default service weights

 voss-deviceapi:
     phoneservices:
         172.0.0.158:8412: 1
     selfservice:
         172.0.0.158:5000: 1
     voss-deviceapi:
         172.0.0.158:9902: 1
     voss-portal:
         172.0.0.158:6001: 1
```

2. Secondary data center has unified nodes with IP addresses and ports:
   - 173.0.0.163:443
   - 173.0.0.164:443

The defaults of the **web weight list** command run on the *proxy servers* is as below:

1. Primary data center proxy server:

```
$ web weight list
Default service weights

    phonebasedreg:
        phoneservices:
            172.0.0.158:443: 1
            172.0.0.159:443: 1
            172.0.0.161:443: 1
            172.0.0.162:443: 1
            173.0.0.163:443: 0
            173.0.0.164:443: 0
    voss-deviceapi:
        selfservice:
            172.0.0.158:443: 1
            172.0.0.159:443: 1
            172.0.0.161:443: 1
            172.0.0.162:443: 1
            173.0.0.163:443: 0
            173.0.0.164:443: 0
        voss-deviceapi:
            172.0.0.158:443: 1
            172.0.0.159:443: 1
            172.0.0.161:443: 1
            172.0.0.162:443: 1
            173.0.0.163:443: 0
            173.0.0.164:443: 0
```

2. Secondary data center proxy server:

```
$ web weight list
Default service weights

    phonebasedreg:
        phoneservices:
            172.0.0.158:443: 0
            172.0.0.159:443: 0
            172.0.0.161:443: 0
            172.0.0.162:443: 0
            173.0.0.163:443: 1
            173.0.0.164:443: 1
    voss-deviceapi:
        selfservice:
            172.0.0.158:443: 0
            172.0.0.159:443: 0
            172.0.0.161:443: 0
            172.0.0.162:443: 0
            173.0.0.163:443: 1
            173.0.0.164:443: 1
        voss-deviceapi:
            172.0.0.158:443: 0
            172.0.0.159:443: 0
            172.0.0.161:443: 0
            172.0.0.162:443: 0
            173.0.0.163:443: 1
            173.0.0.164:443: 1
```

In order to ensure that the secondary data center is configured for a Standby mode, change the web weights to show `userweights` as seen in the output on the *secondary* data center proxy server below:

```
$ web weight list
Default service weights

    phonebasedreg:
        phoneservices:
            172.0.0.158:443: 0
            172.0.0.159:443: 0
            172.0.0.161:443: 0
            172.0.0.162:443: 0
            173.0.0.163:443: 1
            173.0.0.164:443: 1
    voss-deviceapi:
        selfservice:
            172.0.0.158:443: 0
            172.0.0.159:443: 0
            172.0.0.161:443: 0
            172.0.0.162:443: 0
            173.0.0.163:443: 1
            173.0.0.164:443: 1
        voss-deviceapi:
            172.0.0.158:443: 0
            172.0.0.159:443: 0
```

(continues on next page)

(continued from previous page)

```
            172.0.0.161:443: 0
            172.0.0.162:443: 0
            173.0.0.163:443: 1
            173.0.0.164:443: 1

Customized service weights
    userweights:
        172.0.0.158:443: 1
        172.0.0.159:443: 1
        172.0.0.161:443: 1
        172.0.0.162:443: 1
        173.0.0.163:443: 0
        173.0.0.164:443: 0
```

As can be seen, the load balancing web weights have been changed to the unified nodes on the *primary* data center.

3.3. Clustering Considerations

The cluster contains multiple nodes which can be contained in separate firewalled networks.

Network ports need to be opened on firewalls to allow inter-node communication – these are described in more detail in the Platform Guide.

All communication between nodes is encrypted.

Node type	Ports
WebProxy	22 (ssh & sftp), 80 (http), 161 & 162 (snmp), 443 & 8443 (https)
Unified	22 (ssh & sftp), 80 (http), 161 & 162 (snmp), 443 & 8443 (https), 27020 & 27030 (database)

- 22/ssh is used for remote administration
- 80 and 443 is used for the web server
- 161 and 162 are used for sending and receiving snmp
- 8443 is used for inter-cluster communication
- 27020 and 27030 is used for database queries and replication

3.4. Network Communications between Nodes within the Cluster

The cluster contains multiple nodes which can be contained in separate firewalled networks. Network ports need to be opened on firewalls to allow inter-node communication.

All communication between nodes is encrypted.

The following details are all based on the default settings. These can vary depending on the application setup and network design (such as NAT) of the solution, so may need adjustment accordingly. Where a dependant is noted, this is fully dependant on the configuration with no default.

These communications are all related to communications between application nodes within the cluster. There are a few different deployment models so the details below cover the different models and relevant ports. So review and implement according to the deployment model in use.

Note that Standalone is only a single node so this section is not relevant for that deployment model.

- Proxy to Proxy Node

  This is relevant if the proxy node is present in the system.

Communication	Protocol	Port
Cluster Communications	HTTPS	TCP 8443 bi-directional

- Proxy to Unified/Application Node

  This is relevant if the proxy node is present in the system.

Communication	Protocol	Port
User access	HTTPS	TCP 443
Cluster Communications	HTTPS	TCP 8443 bi-directional

- Unified Node to Unified node

  This is relevant to the communications between the unified nodes (application and database combined). If the application and database nodes are split, then see the relevant application and database node details below. Database arbiters run on port 27030.

Communication	Protocol	Port
Database access	database	TCP 27020 and 27030 bi-directional
Cluster Communications	HTTPS	TCP 8443

- Application node to Application node

  This is relevant to the communications between application nodes nodes in the system. This is only relevant where the database node is separate from the application node (in other words, not Unified node).

Communication	Protocol	Port
Cluster communications	HTTPS	TCP 8443 bi-directional

- Application Node to Database node

  This is relevant to the communications between the application node and the database node. This is relevant if the database node is separate from the application node. Database arbiters run on port 27030.

Communication	Protocol	Port
Database access	database	TCP 27020 and 27030 bi-directional
Cluster Communications	HTTPS	TCP 8443

- Database Node to Database node

  This is relevant to the communications between the application node and the database node. This is relevant if the database node is separate from the application node. Database arbiters run on port 27030.

Communication	Protocol	Port
Database access	database	TCP 27020 and 27030 bi-directional
Cluster Communications	HTTPS	TCP 8443

3.5. Network Communications External to the Cluster

The following details are all based on the default settings. These can vary depending on the application setup and network design (such as NAT) of the solution, so may need adjustment accordingly. Where a dependant is noted, this is fully dependant on the configuration with no default.

These communications are all related to communications with devices external to the cluster.

3.5.1. Outbound Communications to Devices from the Application/Unified Nodes

Communication	Protocol	Port
Cisco Unified Communications Manager (UCM)	HTTPS	TCP 8443
Cisco Unity Connection (CUXN)	HTTPS	TCP 443
Webex	HTTPS	TCP 443
LDAP directory	LDAP	TCP/UDP 389 and/or 636(TLS/SSL)
Cisco HCM-F	HTTPS	TCP 8443
MS PowerShell Proxy Node	HTTP, HTTPS	TCP 5985, 5986

3.5.2. Outbound to External Systems from the Proxy Node

Communication	Protocol	Network Protocol and Port
API Sync and Async responses	HTTPS	TCP 443
Northbound Notification messages	HTTPS	dependant
Microsoft 365/Online	HTTP HTTPS	80 443
VOSS Cloud Licensing Service	HTTP HTTPS	80 443

3.5.3. Outbound to External Systems from All Nodes

Communication	Protocol	Port
SNMP	SNMP	TCP/UDP 162
SFTP as required for backup destinations	SFTP	TCP 22
NTP	NTP	UDP 123

3.5.4. Inbound Communications From External Systems to the Proxy Node

Communication	Protocol	Port
Web Access	HTTPS	TCP 443
API Request	HTTPS	TCP 443

3.5.5. Inbound Communications to All Nodes

Communication	Protocol	Port
SSH and SFTP for management and files transfers	SFTP/SSH	TCP/UDP 22

3.6. Cluster Commands

Note: From release 21.1 onwards, a standalone topology is considered a single node cluster (cluster-of-one). This means that commands such as **cluster provision**, **cluster list** and **cluster status** should be run on a standalone topology.

The following Command Line Interface console display shows the available commands for clustering.

```
cluster add <ip>                    - add a new node to join the existing cluster
cluster check <verbose>             - Display pre-upgrade readiness.
                                      For each node, test if ports
                                      are available, the time taken to connect, drive space
                                      percentage and lastly checks if NTP is running
cluster del <ip>                    - remove a node from the existing cluster
cluster job kill <pid>              - Kill a detached job <pid>
cluster job list                    - List detached jobs in the cluster
cluster job reconnect <pid>         - Reconnect to a detached job <pid>
cluster list                        - display the list of nodes associated with the cluster
cluster maintenance-mode            - Display the status, start or stop maintenance
        <start|stop|status>           mode across the cluster
cluster prepnode                    - Prepares the system so that it can be joined to a␣
↪cluster
cluster primary                     - Check if the system is considered the primary by␣
↪patch
                                      or delta install scripts
cluster primary role                - Check if the system is considered the
        <application|database>        'primary' in the given role
cluster provision                   - perform cluster-wide provisioning
        [datacentre <location>]
        [role <role>]
cluster run <where> <command>       - run the command on a particular host.
```

(continues on next page)

(continued from previous page)

```
                                  <where> can either be a name prefix, ip, role, or
↪'all'
cluster status                  - display the status of the cluster
cluster upgrade <iso/url>       - upgrade all applications from iso image <iso-name>.
        [datacentre <location>]   <iso-name> can be a URL for upgrading
        [backup <location>]       from a remote server.
cluster where <application>     - determine on which nodes the application is installed
```

3.7. Geo-redundancy/Redundancy And DR Synopsis

High Availability (HA) is an approach to IT system design and configuration that ensures VOSS Automate is operational and accessible during a specified time frame. This is achieved using redundant hardware and resources. If there is a failure, an automatic failover will occur to a second node.

Web server proxy nodes perform load-balancing between application roles, so that load is distributed. During provisioning, the web server proxy is provided with all the IP addresses of the application nodes. The web server software then does load balancing among these nodes, according to its configuration. If a node fails to respond in a set time, the proxy will send the transaction to another node. This means that in the event that an Application role is lost, the WebProxy will transparently bypass the faulty Application role.

The proxy web server that is configured to be located in the primary site normally load balances to the two unified nodes in the primary site. The proxy web server falls back to the two nodes in the Disaster Recovery site if the nodes in the primary site are down. The web proxy nodes in the secondary site defaults load balancing to the two unified nodes configured for the secondary site.

Data is replicated between Database roles, and role failure is recoverable. This is done using the database replication facility. Automatic failover between Database roles occurs while there is greater than 50% Database role availability. Once there is insufficient role availability, the system needs to be manually re-provisioned.

HA can be increased by adding nodes to the cluster. Application performance and availability can be increased by adding additional application role servers.

Backups can be scheduled to run automatically across the cluster. Backups include application data, configuration and software. Backups can take place to both local disk and remote network location. Every node upgrade is preceded by a snapshot backup which allows any upgrade to be rolled back. Refer to the Platform Guide for details.

3.8. Unified Node Topology

3.8.1. Unified Node Deployment Topology

Important: Choose between a unified node deployment or a modular architecture deployment.

In a Unified Node Deployment, VOSS Automate is deployed either as a single node cluster, 2 unified nodes, or a cluster of multiple nodes with High Availability (HA) and Disaster Recovery (DR) qualities.

Each node can be assigned one or more of the following functional roles:

- WebProxy - load balances incoming HTTP requests across unified nodes.
- Single node cluster - combines the Application and Database roles for use in a non-multi-clustered test environment.
- Unified - similar to the Single node cluster role Application and Database roles, but clustered with other nodes to provide HA and DR capabilities

The nginx web server is installed on the WebProxy, Single node cluster, and Unified node, but is configured differently for each role.

In a clustered environment containing multiple Unified nodes, a load balancing function is required to offer HA (High Availability providing failover between redundant roles).

VOSS Automate supports deployment of either the WebProxy node or a DNS load balancer. Here are some considerations in choosing a WebProxy node vs. DNS:

- The Proxy takes load off the Unified node to deliver static content (HTML/JAVA scripts). When using DNS or a third-party load balancer, the Unified node has to process this information.
- DNS does not know the state of the Unified node.
- The WebProxy detects if a Unified node is down or corrupt. In this case, the WebProxy will select the next Unified node in a round robin scheme.

We recommend that you run no more than two Unified nodes and one WebProxy node on a physical server (VMware server). Also recommended is that the disk subsystems be unique for each Unified node.

The following deployment topologies are defined:

- Test: a standalone, single node cluster with Application and Database roles combined. No High Availability/Disaster Recovery (HA/DR) is available.

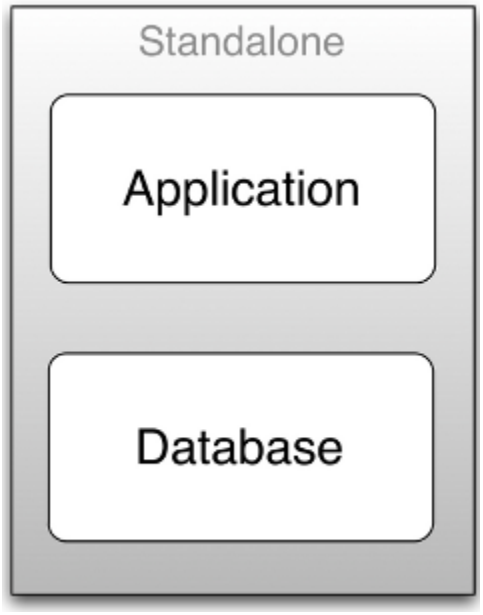
  Important: This deployment should be used for test purposes only.

- Production with Unified Nodes: in a clustered system, comprising:
  - 2, 3, 4 or 6 Unified nodes (each with combined Application and Database roles)
  - 0 to 4 (maximum 2 if 2 Unified nodes) WebProxy nodes offering load balancing. The WebProxy nodes can be omitted if an external load balancer is available.

3.8.2. Single-node cluster (cluster-of-one)

Important: This deployment should be used for test purposes only.

- No High Availability or Disaster Recovery capability is offered in this topology.



3.8.3. Multinode Cluster with Unified Nodes

In order to achieve Geo-Redundancy using the Unified nodes, you need the consider the following:

- Either four or six Unified nodes - each node combining Application and Database roles - are clustered and split over two geographically disparate locations.
- Two Web Proxy nodes to provide High Availability that ensure an Application role failure is gracefully handled. More may be added if Web Proxy nodes are required in a DMZ.

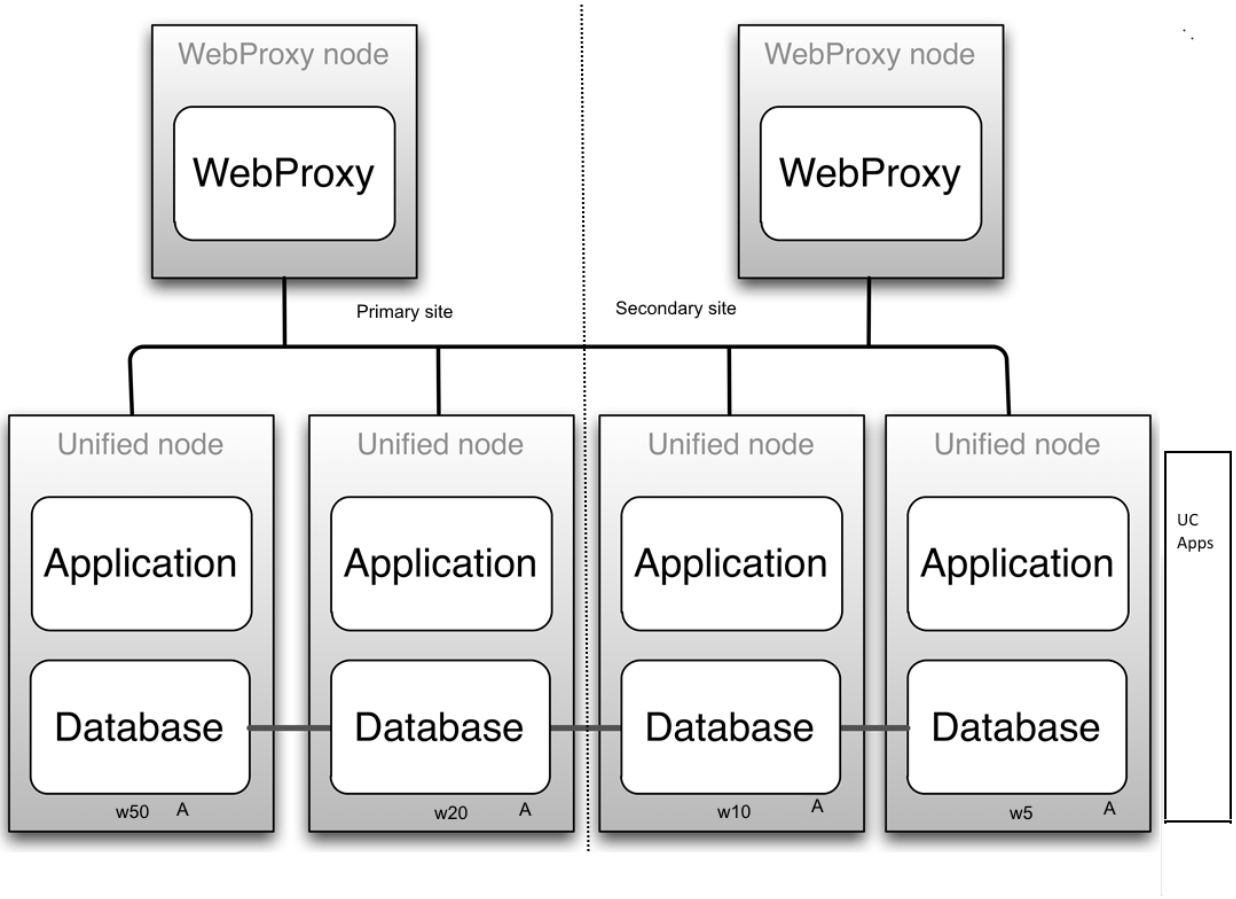
  It is strongly recommended *not* to allow customer end-users the same level of administrator access as the restricted groups of provider- and customer administrators. This is why Self-service web proxies as well as Administrator web proxies should be used.

  Systems with Self-service only web proxies are *only* recommended where the system is customer facing, but where the customer does not administer the system themselves.
- Web Proxy and Unified nodes can be contained in separate firewalled networks.
- Database synchronization takes places between all Database roles, thereby offering Disaster Recovery and High Availability.
- For 6 unified nodes, all nodes in the cluster are active. For an 8 node cluster (with latency between data centers greater than 10ms) , the 2 nodes in the DR node are passive, in other words, the **voss workers 0** command has been run on the DR nodes.

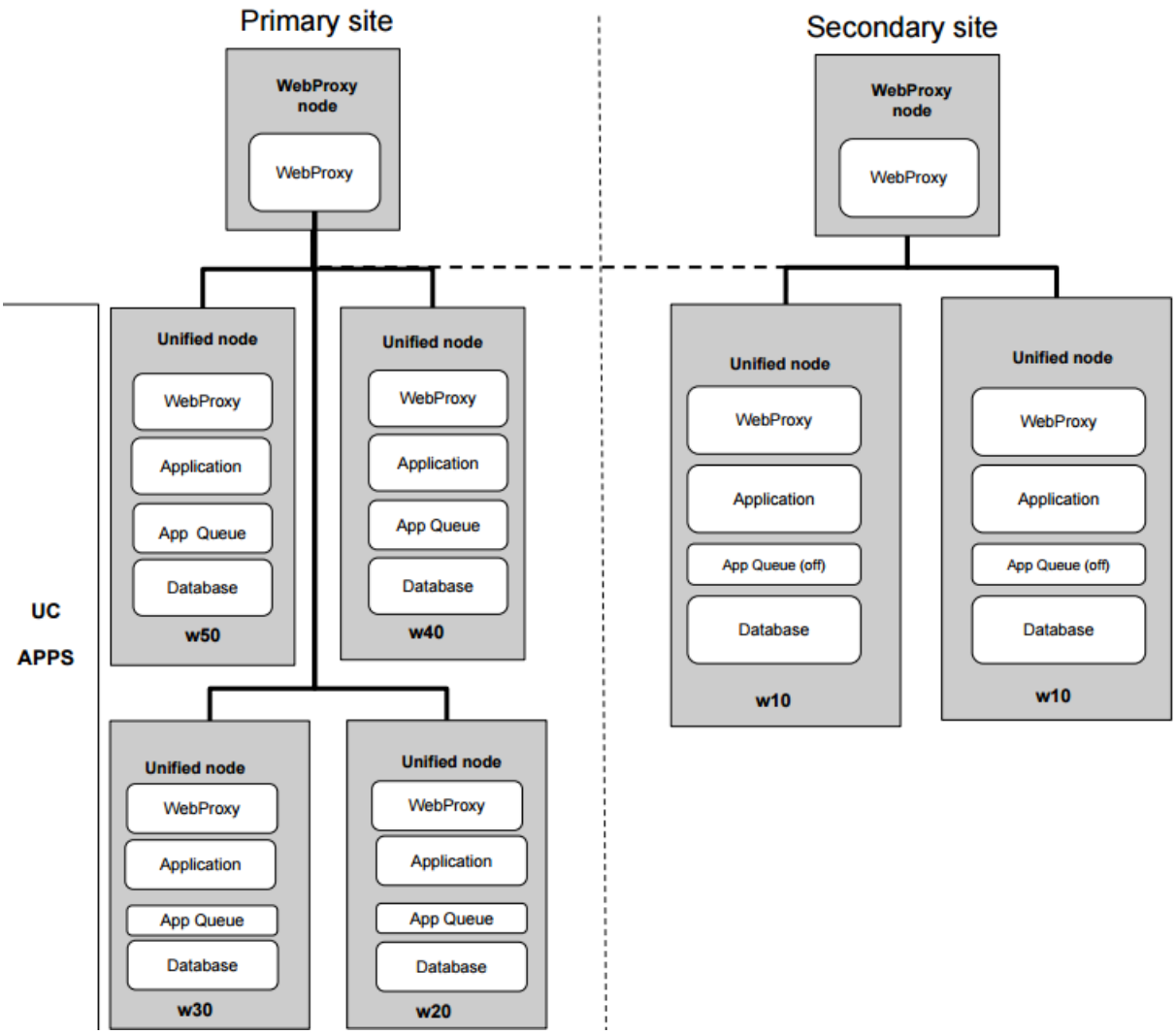
Primary and fall-back Secondary Database servers can be configured manually. Refer to the Platform Guide for further details.

The diagrams in this section illustrate:

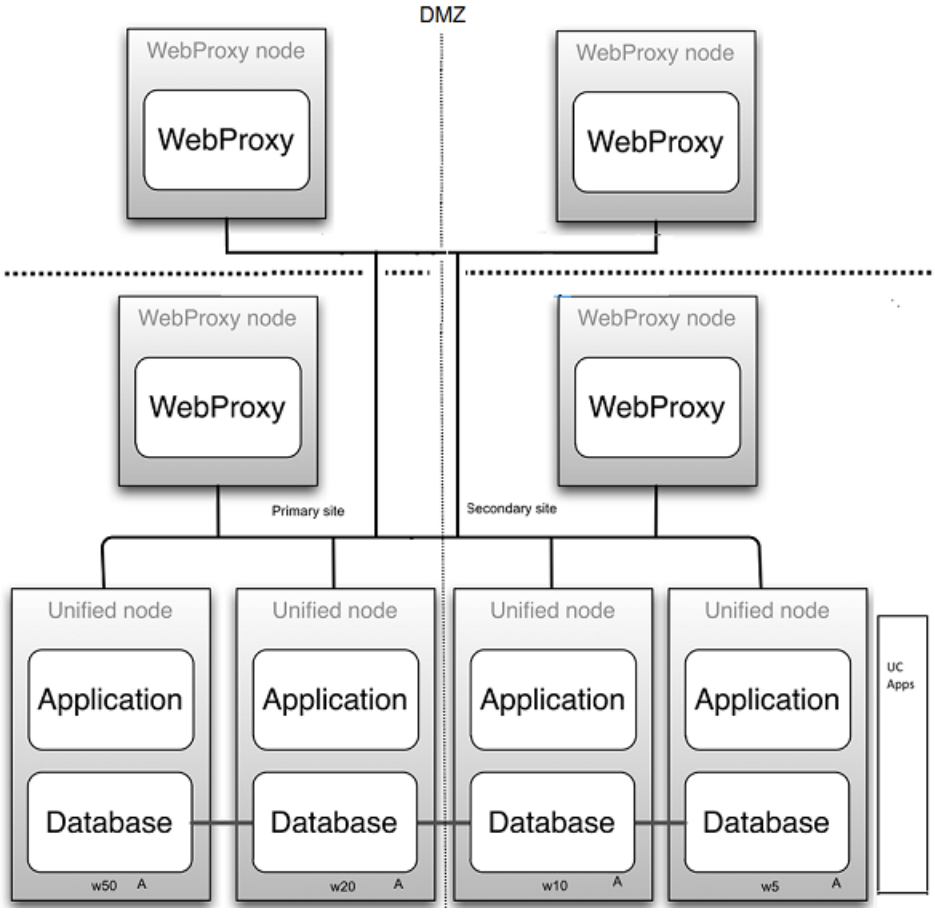
- the six node cluster



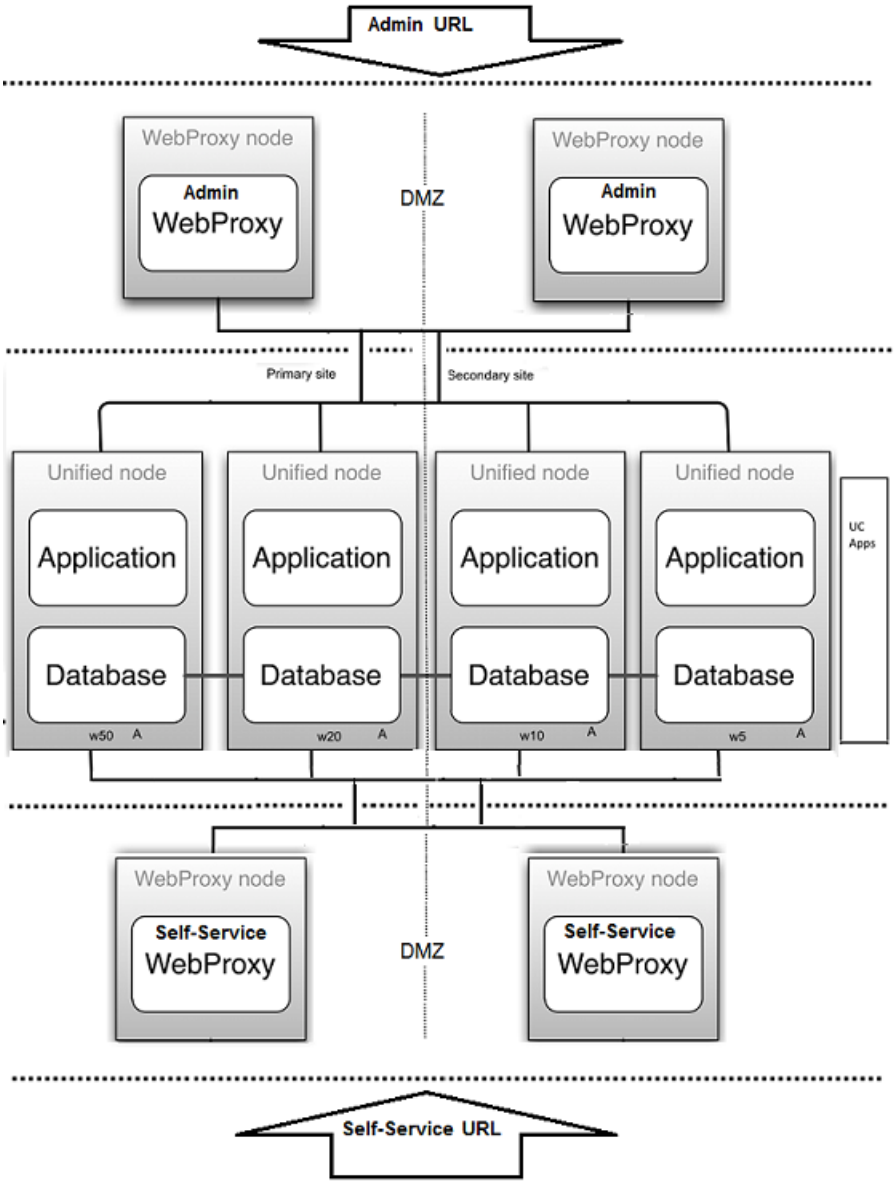
- the eight node cluster



- 2 Web Proxy nodes in a DMZ



- 4 (2 admin, 2 Self-service) Web Proxy nodes in a DMZ



3.8.4. Two-node Cluster with Unified Nodes

In order to achieve Geo-Redundancy using the Unified nodes, you need the consider the following:

- Two unified nodes - each node combining application and database roles - are clustered and optionally split over two geographically disparate locations.
- (Optional) Two web proxy nodes can be used. It may be omitted if an external load balancer is available.
- Web proxy and unified nodes can be contained in separate firewalled networks.
- Database synchronization takes place from primary to secondary unified nodes, thereby offering Disaster Recovery if the primary node fails.
- If the secondary unified node has *more than 10ms latency* with the primary unified node, it must be configured to be in the *same* geographical location.

Important:

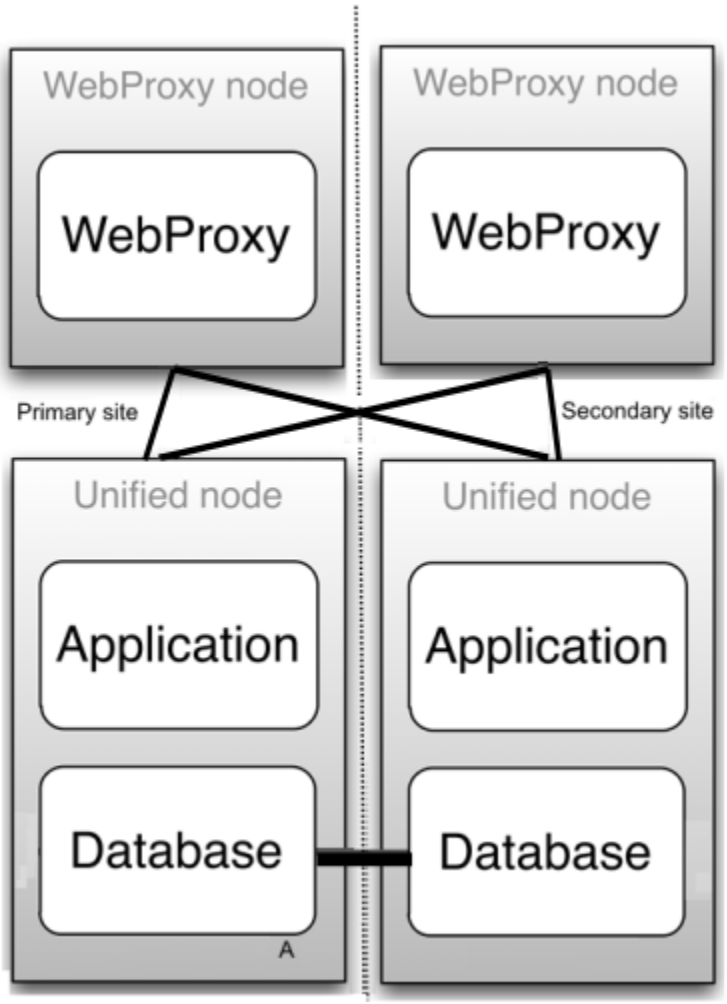
With only two Unified nodes, with or without Web proxies, there is no High Availability. The database on the primary node is read/write, while the database on the secondary is read only.

Only redundancy is available.

- If the primary node fails, a manual delete of the primary node on the secondary and a cluster provision will be needed.
- If the secondary node fails, it needs to be replaced.

Refer to the topic on DR Failover and Recovery in a 2 Node Cluster in the Platform Guide.

The diagram below illustrates the two node cluster:



3.9. Modular Cluster Topology

3.9.1. Modular Cluster Deployment

Important: Choose between a unified node cluster deployment or a modular cluster deployment.

VOSS Automate is deployed as a modular cluster of multiple nodes with High Availability (HA) and Disaster Recovery (DR) qualities.

Each node can be assigned one or more of the following functional roles:

- WebProxy - load balances incoming HTTP requests across nodes.
- Application role node, clustered with other nodes to provide HA and DR capabilities
- Database role node, clustered with other nodes to provide HA and DR capabilities

The nginx web server is installed on the WebProxy and application role node, but is configured differently for each role.

Note: Also refer to the following topics:

Multinode Modular Cluster with Application and Database Nodes in the Install Guide.

Modular Cluster Deployment in the Install Guide.

Modular Architecture Multinode Installation in the Install Guide.

Migrate a Unified Node Cluster to a Modular Cluster in the Platform Guide.

A load balancing function is required to offer HA (High Availability providing failover between redundant roles).

VOSS Automate supports deployment of either the WebProxy node or a DNS load balancer. Here are some considerations in choosing a WebProxy node vs. DNS:

- The Proxy takes load off the application role node to deliver static content (HTML/JAVA scripts). When using DNS or a third-party load balancer, the application role node has to process this information.
- DNS does not know the state of the application role node.
- The WebProxy detects if an application role node is down or corrupt. In this case, the WebProxy will select the next application role node in a round robin scheme.

We recommend that you run no more than one application role node and one database role node and one WebProxy node on a physical server (VMware server). When selecting disk infrastructure, high volume data access by database role replica sets must be considered where different disk subsystems may be required depending on the performance of the disk infrastructure.

The following modular cluster topology is recommended (minimum):

Important: Single node cluster topologies are not available for modular cluster deployments.

- Production with nodes: in a clustered system of 2 data centers:
  - DC1 = primary data center containing primary database node (highest database weight)

  - DC2 = data recovery data center

The system comprises of the following nodes:

  - 3 nodes with application roles (2 in DC1; 1 in DC2)
  - 3 nodes with database roles (2 in DC1; 1 in DC2)
  - Maximum 2 WebProxy nodes if 2 data centers; offering load balancing. The WebProxy nodes can be omitted if an external load balancer is available.

3.9.2. Multinode Modular Cluster with Application and Database Nodes

In order to achieve Geo-Redundancy using Application and Database nodes, you need the consider the following:

- Six Application and Database nodes - 3 nodes with an application role and 3 nodes with a database role - are clustered and split over two geographically disparate locations.
- Two Web Proxy nodes to provide High Availability that ensure an Application role failure is gracefully handled. More may be added if Web Proxy nodes are required in a DMZ.

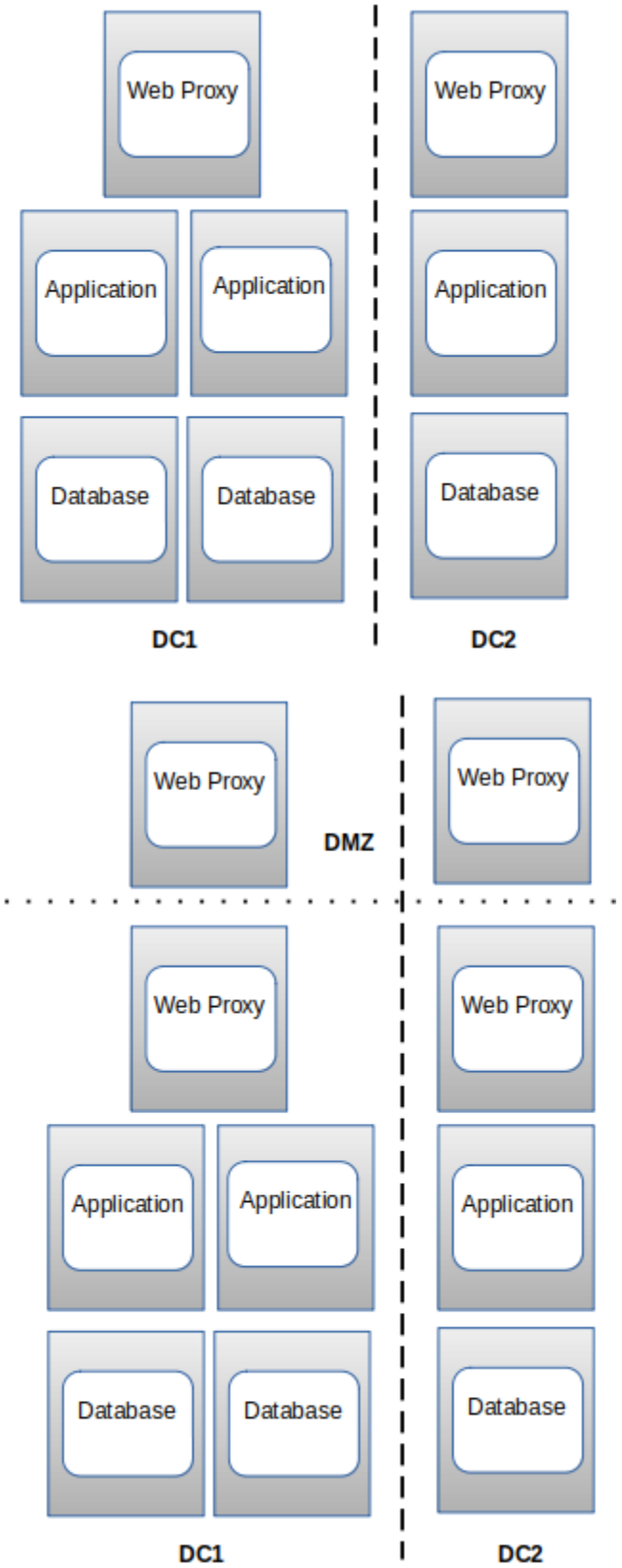
  It is strongly recommended *not* to allow customer end-users the same level of administrator access as the restricted groups of provider- and customer administrators. This is why Self-service web proxies as well as Administrator web proxies should be used.

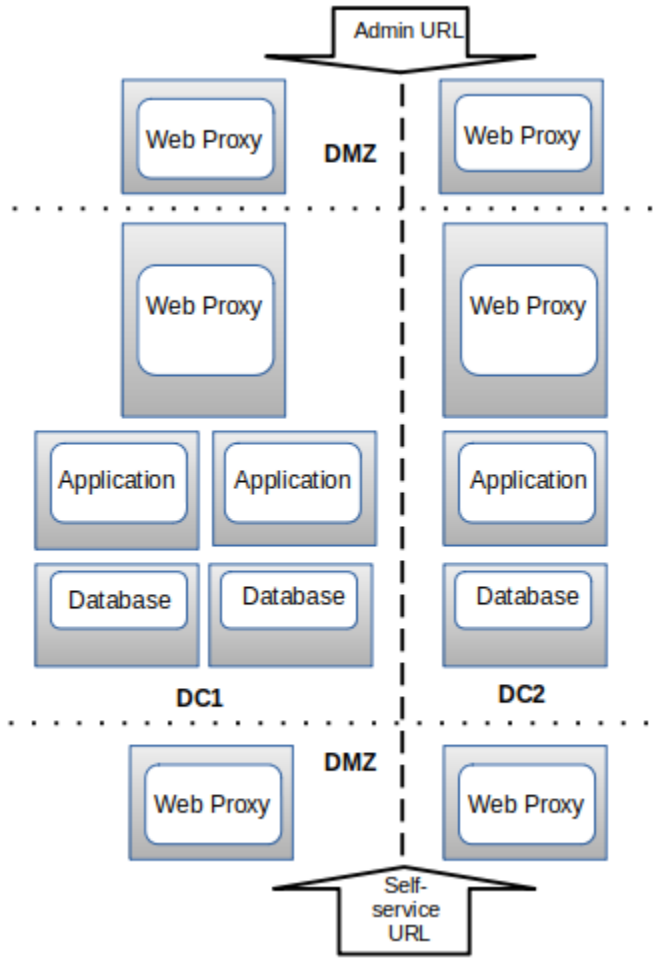
  Systems with Self-service only web proxies are *only* recommended where the system is customer facing, but where the customer does not administer the system themselves.
- Web Proxy, application and database nodes can be contained in separate firewalled networks.
- Database synchronization takes places between all database role nodes, thereby offering Disaster Recovery and High Availability.
- All nodes in the cluster are active.

Primary and fall-back Secondary Database servers can be configured manually. Refer to the Platform Guide for further details.

The diagrams in this section illustrate:

- the six node cluster
- 2 Web Proxy nodes in a DMZ
- 4 (2 admin, 2 Self-service) Web Proxy nodes in a DMZ





4. Prepare to Install

4.1. Installation Prerequisites with Cisco HCM-F

Install VOSS Automate in the same domain as Cisco Hosted Collaboration Mediation Fulfillment (HCM-F).

Installation has the following prerequisites:

- HCM-F services are activated and running.
- Network connectivity is available between Unified CDM nodes and the HCM-F, UC application servers, and WebEx servers.

4.2. Network Docker Container Range

Important: When troubleshooting network issues, verify that the address range is not in use.

If it is in use, modify the Private Address Space using the **network container range add <private IP>** command as described below.

RFC-1918 states that the following three blocks of the IP address space are reserved for private internets:

```
10.0.0.0        -   10.255.255.255  (10/8 prefix)
172.16.0.0      -   172.31.255.255  (172.16/12 prefix)
192.168.0.0     -   192.168.255.255 (192.168/16 prefix)
```

This subnet block address range can be modified to another Private Address Space if needed.

Use the command **network container range list** to see the current Private Address Space.

For example:

```
$ network container range list
    range: 10.1.2.1/24
```

Use the command **network container range add <private IP>** to modify the Private Address Space.

Important: A valid Private Address IP is required as input.

The range `/24` is appended to the IP. For example, if `192.168.0.6` is used, the Private Address range `192.168.0.0/24` is used.

In a clustered environment, you could use **cluster run all network container range add <private IP>**, but if required, the Private Address Space can be also set to be different on each node by running the **add** command on each individual node.

For example:

```
$ network container range add 192.168.2.3
You are about to restart all services. Do you wish to continue?y
Application processes stopped.  (note this line changes dynamically)

Reconfiguring applications....
Application  processes started. (note this line changes dynamically)
```

4.3. Backup Size Considerations

The default backup partition size is 50GB for the default 250GB database partition size. These are the default partition installation sizes.

Important: Backups should be created and restored in a **screen** session - see: *Using the screen command*.

To determine the required space for a specific backup partition, carry out and consider the following:

1. Run **backup create <location-name>** from the CLI. The command output indicates the required space needed to do the backup.

   If the current backup partition size is too small, the command will fail and suggest the size of the partition required. The size can be increased in size with the **drives add** command.

   See the Drive Control topic in the Platform Guide.

   If there is sufficient space but only a size check is required, the backup command can be canceled (Ctrl-C), if needed.

2. Run **voss db_collection_stats all** to show the size of the current database. This command validates the size of the database. This total will be smaller than the suggested backup size.

- A local backup requires free space of at least twice the size of the database. Preferably add another 30% of this. At any given time it is therefore recommended that available backup space should be approximately twice the database size plus 30% of the database size.

  For remote backups, the backup requires free space of at least the size of the database plus 30% of this.

- Database growth over time needs to be considered and allowed for in the backup partition size.
- Space for multiple local backups also needs to be considered and added to the calculated backup partition size.

4.4. Unified Node Topology Preparation

4.4.1. Single-node cluster (cluster-of-one) System Hardware Specification

Virtual machine requirements are specified in the table below.

Node type	Quantity	VM	Memory	CPU	Disk	Network
Single node cluster	1	>= VMware 5.1	16 GB with 16 GB reservation	4 vCPU @ 2 GHz with 4000 MHz reservation	370 GB partitioned: • 20 GB for OS • 50 GB for application: 10 GB for logs, 40GB for our apps • 50 GB for compressed backups • 250 GB for database	1 Gbit/s minimum

For Memory and CPU, the Resource Allocation Reservation on VMware is indicated in the table. Hyper-threading is supported.

Note: If memory allocations are customized, ensure that the memory reservation remains equal to the allocated memory in order to prevent possible negative side-effects due to memory reclamation.

For VMware details, refer to the VMware Support topic in the Compatibility Matrix.

The maximum number of users for a single node cluster is 50,000.

4.4.2. Multinode Cluster Hardware Specification

Virtual machine requirements are specified in the table below.

Node type	Quantity	VM	Memory	CPU	Disk	Network
Unified	4 or 6	>= VMware 5.1	16 GB with 16 GB reservation	4 vCPU @ 2 GHz with 4000 MHz reservation	370 GB partitioned: • 20 GB for OS • 50 GB for application: 10 GB for logs, 40GB for our apps • 50 GB for compressed backups • 250 GB for database	1 Gbit/s minimum
WebProxy	2	>= VMware 5.1	4 GB with 4 GB reservation	2 vCPU @ 2 GHz with no reservation	70 GB partitioned: • 20 GB for OS • 50 GB for application	1 Gbit/s minimum

For Memory and CPU, the Resource Allocation Reservation on VMware is indicated in the table. Hyperthreading is supported.

Note: If memory allocations are customized, ensure that the memory reservation remains equal to the allocated memory in order to prevent possible negative side-effects due to memory reclamation.

The OS disk requirement is fixed and logs are rotated to ensure that 10 GB is sufficient. 40 GB for applications is a generous allocation and does not scale with the number of users.

The Database storage partition is sized to support 250 K users. Database backups are compressed and the partition is sized to ensure that sufficient space available to support backup of 250 GB database.

Note: To change the TRANSACTION_LOG cap size to greater than 10GB at larger providers for operational reasons, for example for diagnostics and a longer transaction replay window, the **voss db_collection_cap TRANSACTION_LOG <10-50GB>** command can be used from the command line.

Refer to Database Commands in the Platform Guide for more details.

The resize operation will impact the usage on the disk size allocated for the database (typically, 250GB is reserved upon installation). Consider a larger database disk size allocation upon installation if a larger cap size is set.

The backup disk should be Thick Provisioned and Eager Zeroed for better performance immediately after installation.

Web Proxies are optional, but if Web Proxies are used, then they form part of the cluster to allow sharing of static data and other content as needed (for example, themes).

To set up the disk requirements, the disk should be set up on the VMware GUI Resources tab where a disk can be created. This task should be done after the OVA import but prior to the boot of the system.

For VMware details, refer to the VMware Support topic in the Compatibility Matrix.

4.4.3. Two-node Cluster Hardware Specification

Virtual machine requirements are specified in the table below.

Node type	Quantity	VM	Memory	CPU	Disk	Network
Unified	= 2	>= VMware 5.1	16 GB with 32 GB reservation	4 vCPU @ 2 GHz with 4000 MHz reservation	370 GB partitioned: • 20 GB for OS • 50 GB for application: 10 GB for logs, 40GB for our apps • 50 GB for compressed backups • 250 GB for database	1 Gbit/s minimum
WebProxy	>= 0	>= VMware 5.1	4 GB with 4 GB reservation	2 vCPU @ 2 GHz with no reservation	70 GB partitioned: • 20 GB for OS • 50 GB for application	1 Gbit/s minimum

For Memory and CPU, the Resource Allocation Reservation on VMware should correspond with these requirements.

Note: If memory allocations are customized, ensure that the memory reservation remains equal to the allocated memory in order to prevent possible negative side-effects due to memory reclamation.

For VMware details, refer to the VMware Support topic in the Compatibility Matrix.

4.5. Modular Cluster Topology Preparation

4.5.1. Multinode Modular Cluster Hardware Specification

Virtual machine requirements are specified in the table below.

Node type	Quantity	VM	Memory	CPU	Disk	Network
Application	3	>= VMware 5.1	16 GB with 16 GB reservation	4 vCPU @ 2 GHz with 4000 MHz reservation	80 GB partitioned: • 20 GB for OS • 50 GB for application: 10 GB for logs, 40GB for our apps	1 Gbit/s minimum
Database	3	>= VMware 5.1	32 GB with 32 GB reservation	4 vCPU @ 2 GHz with 4000 MHz reservation	380 GB partitioned: • 20 GB for OS • 50 GB for compressed backups • 50 GB for application: – 10 GB for logs – 40GB for our apps • 250 GB for database	1 Gbit/s minimum
WebProxy	2	>= VMware 5.1	4 GB with 4 GB reservation	2 vCPU @ 2 GHz with no reservation	70 GB partitioned: • 20 GB for OS • 50 GB for application	1 Gbit/s minimum

For Memory and CPU, the Resource Allocation Reservation on VMware is indicated in the table. Hyperthreading is supported.

Note: If memory allocations are customized, ensure that the memory reservation remains equal to the allocated memory in order to prevent possible negative side-effects due to memory reclamation.

The OS disk requirement is fixed and logs are rotated to ensure that 10 GB is sufficient. 40 GB for an applications role node is a generous allocation and the size will not have to be increased with the number of users.

The Database storage partition is sized to support 250 K users. Database backups are compressed and the partition is sized to ensure that sufficient space is available to support backup of 250 GB database.

Note: To change the TRANSACTION_LOG cap size to greater than 10GB at larger providers for operational reasons, for example for diagnostics, the **voss db_collection_cap TRANSACTION_LOG <10-50GB>** command can be used from the command line.

Refer to Database Commands in the Platform Guide for more details.

The resize operation will impact the usage on the size of the disk allocated for the database (typically, 250GB is reserved upon installation). Consider a larger database disk size allocation upon installation if a larger cap

size is set.

The backup disk should be Thick Provisioned and Eager Zeroed for better performance immediately after installation.

Web Proxies are optional, but if Web Proxies are used, then they form part of the cluster to allow sharing of static data and other content as needed (for example, themes).

To set up the disk requirements, the disk should be set up on the VMware GUI Resources tab where a disk can be created. This task should be done after the OVA import but prior to the boot of the system.

For VMware details, refer to the VMware Support topic in the Compatibility Matrix.

5. Install VOSS Automate

5.1. Installation Process

The installation process is divided into:

1. The VMWare installation of a node (*Create a New VM Using the Platform-Install OVA*).
2. Node setup for single node or multinode installations (*Multinode Installation* and *Single-node cluster (cluster-of-one) Installation*).

The node setup stage requires one or more prerequisite VMWare node installations.

5.2. Create a New VM Using the Platform-Install OVA

Note: If an OVA file is not available for your current release, you need to obtain the most recent release OVA for which there is an upgrade path to your release.

The steps below show the common setup of a *single node* from the OVA file - either for the purposes of:

- a standalone installation

  If an OVA file is not available for your current release:

  1. Obtain and install the most recent release OVA for which there is an upgrade path to your release.
  2. Apply the Delta Bundle Upgrade steps for the current release to the OVA to upgrade it.

- a node installation during multinode installation - see *Notes on Multi-Node Installation*

  If an OVA file is not available for your current release:

  1. Obtain and install the most recent release OVA for which there is an upgrade path to your release.
  2. Apply the Delta Bundle Upgrade steps for the current release *to the cluster* to upgrade it. Refer to the *Upgrade Guide with Delta Bundle*.

- or during a failover recovery

  If an OVA file is not available for your current release:

  1. Obtain and install the most recent release OVA for which there is an upgrade path to your release.
  2. Add it to your cluster. Use the same configure options in the table below as were applied to the lost node.

Note that the node version mismatch in the cluster can be ignored, since the next upgrade step aligns the versions.

3. Apply the Delta Bundle Upgrade steps for the current release *to the cluster* to upgrade it.

   For details, refer to the *Upgrade Guide with Delta Bundle* and to the specific scenario Disaster Recovery steps in the *Platform Guide*.

The steps will therefore be followed either once or multiple times during installation - in accordance with the required topology.

The downloaded OVA file is imported into VMware vCenter Server. Only one OVA file is used to deploy all the functional roles. You choose the specific node `role` when the installation wizard is run.

1. Log in to vSphere to access the ESXi Host.
2. Choose **File > Deploy OVF Template**.
3. Choose Source, browse to the location of the .ova file, and click **Next**.
4. On the Name and Location page, enter a Name for this server.
5. On the Deployment Configuration page, select the appropriate node type. (Refer to the list at the `role` option below.)
6. Choose the resource pool in which to locate the VM.
7. Choose the data store you want to use to deploy the new VM.
8. Choose the disk format to use when deploying the new VM.
   - For non-SSD-based drives in production environments, “thick provisioning” is mandatory. Thick Provision Eager Zeroed is recommended.
   - For SSD-based drives, “thin provisioning” is supported.
9. On the Network Mapping, choose your network on which this VM will reside.
10. Do not select Power on after deployment.
11. On the Ready to Complete page, click **Finish** to start the deployment.
12. After the VM is created, select the CD ROM configuration and verify the **Connect at Power On** check box is enabled. Also, verify the memory, CPU, and disk settings against the requirements shown in either the Single-node cluster (cluster-of-one) System Hardware Specification or Multinode Cluster Hardware Specification section in the Install Guide.
13. Power on the VM.
14. Configure the options in the installation wizard:

Option	Option name	Description
1	IP	The IP address of the server.
2	netmask	The network mask for the server.
3	gateway	The IP address of the network gateway.
4	DNS	The DNS server is optional. Ensure that the DNS server is capable of looking up all hostnames referred to, including NTP server and remote backup locations.
5	NTP	The NTP server is mandatory to ensure that time keeping is accurate and synchronized among nodes in the same cluster.
6	boot password	Enable boot loader configuration password. See the example below.
7	hostname	The hostname, not the fully qualified domain name (FQDN).
8	role	Note: only WebProxy, Application and Database nodes are used for a modular architecture installation. • A WebProxy role installs only the front-end web server together with ability to distribute load among multiple middleware nodes. • An Application node is the main transaction processing engine and includes a web server which can operate by itself, or route transactions from a web node. • A Database node provides persistent storage of data. • A Standalone node consists of the Web, Application, and Database roles on one node. For Single-node cluster (cluster-of-one). • A Unified node consists of the Web, Application, and Database roles on one node. On installation, the system needs to be clustered with other nodes and the cluster provisioned. • A General node used for M2UC, NBI.
9	data center	The system's geographic location (data center name, city, country that a customer can use to identify the system location). You cannot change this setting once set.
10	platform password	Platform password must be at least eight characters long and must contain both uppercase and lowercase letters and at least one numeric or special character.

Note: On a fresh installation, if you run the install on a network with a DHCP server and encounter an error:

"Error: DNS server <DNS server> is either invalid or cannot be reached on the network"

you can enter a valid DNS server address to continue the installation.

Once all details are entered, installation will commence. When installation is complete, the system will reboot. Since all services will be stopped, this takes some time.

5.2.1. Notes on Passwords and Security

The default security protocol for the web server is TLSv1.2.

Password protection can be enabled on the VOSS Automate boot loader configuration from the install wizard upon first install and also from the CLI - see the topic on System Boot Passwords in the Platform Guide for commands to enable, disable or reset the boot password.

Important: The boot password is non-recoverable.

The console example below shows the `boot password` configuration output:

```
(1)              ip    (199.29.21.89)
(2)         netmask    (255.255.255.0)
(3)         gateway    (199.29.21.1)
(4)             dns    (199.29.88.56)
(5)             ntp    (199.29.88.56)
(6)   boot password    (disabled)
(7)        hostname    (atlantic)
(8)            role    (UNDEFINED)
(9)     data centre    (earth)
(10) platform password (UNDEFINED)
Select option ? 6
Valid passwords must contain:
   at least one lower- and one upper-case letter,
   at least one numeric digit
   and a special character eg. !#@$%&^*
Password: Please enter platform user password:
 Please re-enter password
Password:
NOTE: The system boot password is now set for user platform.
```

When the boot password is set, the wizard will show:

```
(6)   boot password    (*****)
```

5.2.2. Notes on Multi-Node Installation

According to the multi-node deployment topology and specification, the *role* of each VM installation is as indicated below.

- For each WebProxy instance, create a new VM using the platform-install OVA. For *role*, select **(3) WebProxy**. Specify the appropriate data center (Primary/DR site) for each WebProxy instance.
- *Standard Topology only*: For each Unified instance, create a new VM using the platform-install OVA. For *role*, select **(2) Unified**. Specify the appropriate data center (Primary/DR Site) for each Unified instance.

  The following Unified nodes are required in the cluster:

  - One Unified node as the Primary node at the Primary site
  - One Unified node as the Secondary node at the Primary site

  - Two Unified nodes as the Secondary nodes at the Disaster Recovery (DR) site

  Note:

  - For a six Node Multi Cluster deployment there are; two Unified nodes (one Primary and one Secondary) and one WebProxy node at the Primary site, and two Unified nodes (both Secondary) and one WebProxy node at the DR site.
  - For an eight Node Multi Cluster deployment, there are four Unified nodes (one Primary and three Secondary) and one WebProxy node at the Primary site. Two Unified nodes (both Secondary) and one WebProxy node are at the DR site.

- *Modular Architecture Topology*

  The following nodes are required in a typical Modular Architecture cluster:

  - One Application node as the Primary node at the Primary site
  - One additional Application node at the Primary site
  - One Database node as the Primary Database node at the Primary site
  - One additional Database node at the Primary site
  - One Application node at the Disaster Recovery (DR) site
  - One Database node at the Disaster Recovery (DR) site

Note: For a typical Modular Architecture cluster there is one WebProxy node at the Primary site and one WebProxy node at the DR site.

For each *Database* instance, create a new VM using the platform-install OVA. For *role*, select **(2) Database**. Specify the appropriate data center (Primary/DR Site) for each Database instance.

For each *Application* instance, create a new VM using the platform-install OVA. For *role*, select **(2) Application**. Specify the appropriate data center (Primary/DR Site) for each Application instance.

Also refer to Multinode Installation section in the Install Guide.

Detailed configuration can be applied from the Command Line Interface (CLI). Use **network help** or **network** for details. For example, domain can be configured using **network domain add <domain-name>**. For a geo-redundant deployment, the `data center` information entered in the wizard is equivalent to the location information.

5.2.3. Finalize the Installation

When the installation of the OVA is complete, a sign-in prompt for the platform user is displayed. The system is ready for use.

Connect to newly deployed server CLI as the platform user.

The login message would for example looks the same as below:

```
Last login: Wed Nov  2 11:12:45 UTC 2016 from thwh on pts/6
Last failed login: Wed Nov  2 11:19:53 UTC 2016 from iza on ssh:notty
There were 2 failed login attempts since the last successful login.

host: dev-test, role: webproxy,application,database, load: 0.21, USERS: 3
date: 2016-11-02 11:19:57 +00:00, up: 14:19
```

(continues on next page)

(continued from previous page)

```
network: 172.29.253.14, ntp: 172.29.1.15
HEALTH: NOT MONITORED
database: 31Gb
Failed logins: 2 since Wed Nov 02 11:19:53 2016 from iza

    mail - local mail management          keys - ssh/sftp credentials
 network - network management           backup - manage backups
    voss - voss management tools           log - manage system logs
database - database management          notify - notifications control
schedule - scheduling commands        selfservice - selfservice management
    diag - system diagnostic tools      system - system administration
    snmp - snmp configuration             user - manage users
 cluster - cluster management           drives - manage disk drives
     web - web server management             app - manage applications
security - security update tools
```

If the user failed to log in prior to a successful login, the count, date and origin of the attempts are shown as `Failed logins`. A successful login resets this login count.

Run **app status** and ensure the services are all running and reporting the correct version before continuing.

Note: Return to Multinode Installation, Standalone Installation or Failover step to complete the overall installation or failover recovery procedure.

5.3. Using the screen command

The **screen** command is available to execute long-running commands (for example, when upgrading) in the background.

The following commands require the running of **screen**:

- **cluster provision**
- **cluster upgrade**
- **app template**
- **voss export type <args>**
- **voss export group <args>**
- **voss subscriber_data_export**

A message is displayed to indicate that **screen** should be run first:

```
This is a potentially long-running command and should be executed in a screen session
Run `screen` and then execute the command again
```

The use of **screen** is *not affected* by the use of the `--force` parameter with any of these commands.

The commands then run in a screen session that can be reconnected. The standard screen command parameters are available, in particular:

- **screen** - start a new session

- **screen -ls** - show sessions already available
- **screen -r [screen PID]** - reconnect to a disconnected session

The version of **screen** used in VOSS Automate also supports the creation of a log file. If long-running commands will be run, the log file captures screen console output up to the session timeout. A message shows:

```
timed out waiting for input: auto-logout
```

To create a screen log file:

1. Run **screen** and wait for screen to open.
2. Press **<Ctrl>-a** then **:** (colon). This will enter screen command mode at the bottom of the console.
3. Create your screen logfile in the `media/` directory:
   a. In screen command mode, type **logfile media/<screen-logfilename>.log**
   b. Press **<Enter>**
   c. Press **<Ctrl>-a** and then **H** to start writing to the log file
   d. Run your commands.

If the **screen** session times out, you can obtain console output from the log file, for example:

$ sftp platform@<host>:media/<screen-logfilename>.log

5.4. Installation Logs

To troubleshoot an installation, log files can be inspected. For example, detailed platform commands show in the `execute.log` file. Here, log entries for the command execution have a `ui` column. Log entries that follow these show related commands.

To only see the commands in `execute.log` example, you can open a new console and run:

log follow execute.log | grep " ui "

Note: Logs are rotated and install commands may not show after log rotation.

The list below shows examples of installation commands and corresponding `ui` and following entries in `execute.log`.

- **app install vmware**.

  `execute.log`:

  ```
  <timestmp><user><execnum>: ui - /opt/platform/bin/scripts.py install 'vmware'
  ```

- **app list**.

  `execute.log`:

  ```
  <timestmp><user><execnum>: ui - /opt/platform/bin/execute get /scripts/
  <timestmp><user><execnum>: run: /opt/platform/bin/execute get /scripts/
  <timestmp><user><execnum>: run: /opt/platform/bin/scripts.py list
  ```

- **database config**

  execute.log:

```
<timestmp><user><execnum>: ui - /opt/platform/apps/mongodb/bin/database_
↪helper.py config
<timestmp><user><execnum>: run: /opt/platform/apps/mongodb/bin/database_
↪helper.py config
<timestmp><user><execnum>: run: /opt/platform/apps/mongodb/bin/database_
↪helper.py config returned 0
```

- **cluster list**.

  execute.log:

```
<timestmp><user><execnum>: ui - /opt/platform/bin/execute get /apps/cluster/engine/
↪list
<timestmp><user><execnum>: run: /opt/platform/bin/execute get /apps/cluster/engine/
↪list
<timestmp><user><execnum>: run: /opt/platform/bin/execute --app=cluster   get /list
<timestmp><user><execnum>: run: /opt/platform/apps/cluster/cluster.py list
<timestmp><user><execnum>: run: /opt/platform/apps/cluster/cluster.py list returned␣
↪0
```

- **cluster status**.

  execute.log:

```
<timestmp><user><execnum>: ui - /opt/platform/bin/execute get /apps/cluster/
↪engine/status
<timestmp><user><execnum>: run: /opt/platform/bin/execute get /apps/cluster/
↪engine/status
<timestmp><user><execnum>: run: /opt/platform/bin/execute --app=cluster  ␣
↪get /status
<timestmp><user><execnum>: run: /opt/platform/apps/cluster/cluster.py status
```

- **web service list**

  execute.log:

```
<timestmp><user><execnum>: ui - /opt/platform/bin/execute get /apps/nginx/
↪engine/disable
<timestmp><user><execnum>: run: /opt/platform/bin/execute get /apps/nginx/
↪engine/disable
<timestmp><user><execnum>: run: /opt/platform/bin/execute --app=nginx   get␣
↪/disable
<timestmp><user><execnum>: run: /opt/platform/bin/config.py get --app=nginx␣
↪disable
<timestmp><user><execnum>: run: /opt/platform/bin/config.py get --app=nginx␣
↪disable returned 0
```

- **log follow upgrade_db.log**

  execute.log:

```
<timestmp><user><execnum>: ui - /opt/platform/bin/execute post /apps/services/
↪process/log/engine/log/follow '{"follow":"upgrade_db.log"}'
<timestmp><user><execnum>: run: /opt/platform/bin/execute post /apps/services/
↪process/log/engine/log/follow '{"follow":"upgrade_db.log"}' --method=os.system
<timestmp><user><execnum>: run: /opt/platform/bin/execute --app=services:log   post␣
↪/log/follow '{"follow":"upgrade_db.log"}' --method=os.system
<timestmp><user><execnum>: run: /opt/platform/apps/services/logviewer.sh follow␣
↪upgrade_db.log
```

- **app template media/<VOSS Automate_template_file>**

  execute.log:

```
<timestmp><user><execnum> ui - /opt/platform/apps/template_runner/template media/
↪install.template platform
[...]
<timestmp><user><execnum> ui - /opt/platform/bin/execute --app=template_runner post␣
↪/methods/import
    '{"filename":"DummyTestImport.json","import":"DummyTestImport.json"}'


Please enter a password for ...

<timestmp><user><execnum>  ui - /usr/bin/docker exec -it voss-wsgi /opt/voss-
↪deviceapi/bin/python
   /opt/voss-deviceapi/src/deviceapi/utils/get_user_password.py set_details␣
↪sysadmin@sys

[...]

<timestmp><user><execnum>: ui - /opt/platform/bin/execute --app=template_runner␣
↪post /methods/import
'{"filename":"UpgradeChecks.json","import":"UpgradeChecks.json -p sys"}'

[...]

'{"filename":"EndToEnd.application.json","import":"EndToEnd.application.json -p sys"}
↪'
'{"filename":"SYS.json","import":"SYS.json -p sys"}'
'{"filename":"SYSnoPKG.json","import":"SYSnoPKG.json -p sys"}'
'{"filename":"SYSdotHCS_with_hcmf.json","import":"SYSdotHCS_with_hcmf.json -p sys.
↪hcs"}'

[...]
```

5.5. Primary Node Role in a Cluster

Commands are available to determine if a node is the primary database or application node.

From the CLI on a node, the **cluster primary** command can be run with additional parameters to determine if the node is the primary database or application node.

```
platform@VOSS-UN-1:~$ cluster primary
is_primary: true
platform@VOSS-UN-1:~$ cluster primary role application
is_primary: true
platform@VOSS-UN-1:~$ cluster primary role database
is_primary: false
platform@VOSS-UN-1:~$
```

This command should be used to establish and confirm the primary application node during patch, patch bundle or delta bundle installation.

5.6. Unified Node Topology Installation

5.6.1. Multinode Installation

Before You Begin

Before continuing, you should have followed the OVA installation on each node according to the steps and preliminary requirements specified in: *Create a New VM Using the Platform-Install OVA* and according to the node roles as indicated in *Notes on Multi-Node Installation*.

Optionally download or extract language pack template files to support languages other than English.

Note:

- For geo-redundant Multinode Cluster deployment with six Unified Nodes, there are four Unified nodes in the Primary Site and two Unified nodes in the Disaster Recovery (DR) Site in active-standby setup.

  The worker count (**voss workers** command) needs to be set on the DR nodes. Refer to:

  - *Multinode Cluster with Unified Nodes*
  - *Multi Data Center Deployments*
- For 2 node cluster deployment there are 2 unified nodes.
- Template installation and upgrade takes approximately two hours. You can follow the progress on the Admin Portal transaction list.
- It is strongly recommended *not* to allow customer end-users the same level of administrator access as the restricted groups of provider- and customer administrators. This is why Self-service web proxies as well as Administrator web proxies should be used.

  Systems with Self-service only web proxies are *only* recommended where the system is customer facing, but where the customer does not administer the system themselves.
- For cluster installations, also refer to the Health Checks for Cluster Installations Guide.

- If it is necessary to change an IP address of a node in a cluster, first remove it from the cluster by running the command below *on the node to be changed*:

  cluster del <IP address of node to be changed>

- Refer to *Installation Logs* for troubleshooting logs during an installation.

The standard **screen** command should be used where indicated. See: *Using the screen command*.

Procedure

1. Install VMware tools on each node.
   a. Log in to each node and run **app install vmware**.
   b. Verify that vmware is running: **app list**.
2. Prepare each node to be added to the cluster:
   a. Select a primary Unified node that will become the primary database node. The designation of primary unified node is arbitrary. The deploying administrator can pick any unified node that they see fit.
   b. On each WebProxy and Unified node, *excluding the primary node*, run **cluster prepnode**.
3. Add nodes to the cluster.
   a. Log in to the selected primary Unified node.
   b. Add the Unified and WebProxy nodes to the cluster: **cluster add <ip_addr>**.

      Note that you do not have to add the selected primary node to the cluster. It will autmatically be added to the cluster.
   c. Verify the list of nodes in the cluster: **cluster list**.
4. Add the network domain (optional if a domain name is needed). From the selected primary Unified node:
   a. Configure the domain: **cluster run all network domain <domain_name>**.
   b. Verify the configured network domain: **cluster run all network domain**. Each node shows the domain that you configured.
5. Check the network:
   a. From the selected primary Unified node, run **cluster check** to verify the status of the cluster, network connectivity, disk status and NTP.

      Note:

      - Since database weights have not been added yet, `` database: not configured`` errors can be ignored. Verification of database weights should be done when this command is run during the step following the provisioning step.
      - If a cluster has not been provisioned, port 443 errors from web proxies are raised upon this check. These errors can be ignored.
      - If errors on other ports (for example 27020) are raised by the check, verify that the firewall service has started. The following command can be run:

        ```
        cluster run database app start services:firewall --force
        ```

      ---

b. Verify the DNS configuration: **cluster run all network dns**. Each node responds with the DNS server address.

6. Create a restore point. As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
7. Configure the cluster.
   a. Provide a weight for each database server with the **database weight add <database_ip> <priority>** command.
      - Weights of 40, 30 are recommended for two Unified nodes
      - Weights of 40, 30, 20, and 10 are recommended for four Unified nodes
      - Weights of 60, 50, 40, 30, 20, and 10 are recommended for six Unified nodes

      The higher the value, the more priority.

      For Multinode Cluster deployment with four Unified Nodes in a geo-redundant system containing two data center infrastructures in two physical locations the following weights are used:
      - Specify a weight of 40 for the Primary node at the Primary site
      - Specify a weight of 30 for the Secondary node at the Primary site
      - Specify weights of 20 and 10 for the Secondary nodes at the DR site

      For Multinode Cluster deployment with six Unified Nodes in a geo-redundant system containing two data center infrastructures in two physical locations the following weights are used:
      - Specify a weight of 60 for the Primary node at the Primary site
      - Specify a weight of 50 for the Secondary node at the Primary site
      - Specify a weight of 40 for the Secondary node at the Primary site
      - Specify a weight of 30 for the Secondary node at the Primary site
      - Specify weights of 20 and 10 for the Secondary nodes at the DR site

   b. From the selected primary Unified node, now set it up as the primary Unified node. It is recommended that this step is run in a terminal opened with the **screen** command.
      i. Run **screen**.
      ii. Run **cluster provision**

      Allow approximately 2 hours for the operation to complete for two WebProxy and four Unified nodes.

   c. When provisioning is complete, check that each node is contactable and that the time server is running on each with **cluster check**.

      If a service is down, run **cluster run <node_ip> app start** to restart the service.

      If provisioning is successful, the screen session can be closed by typing **exit** in the screen terminal. If errors occurred, keep the screen terminal open for troubleshooting purposes and contact VOSS support.

   d. (Optional) If required, set the web weights configurations (Active-Active, Active-Standby, Single node cluster). From the primary Unified node, run the required **web weight** commands for the Web Proxy nodes. For details, refer to *Multi Data Center Deployments* and the VOSS Automate Best Practices Guide.

e. (Optional) If required, enable or disable Self-service or admin web services on the web proxy nodes. This may for example be needed for security purposes.

The commands must be run on the relevant web proxy node. It is not recommended that the commands be run on a single node cluster system, but only on a cluster. The commands will automatically reconfigure and restart the nginx process, so some downtime will result. Request URLs to a disabled service will redirect the user to the active service.

- To disable or enable admin or Self-service web services on the web proxy node:

  web service disable <selfservice|admin>

  web service enable <selfservice|admin>

- To list web services on the web proxy node:

  web service list

8. Create a restore point. As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
9. Initialize the database and clear all data. On the primary Unified node, run **voss cleardown**.

   Note that this step may take some time. You can follow the process by running **log follow upgrade_db.log** or **log follow voss-deviceapi/app.log** in a separate console on the primary Unified node.
10. Import the templates.

    a. Copy the VOSS Automate template file to the primary Unified node with the command:

    scp <VOSS Automate_template_file> platform@<unified_node_ip_address>:~/media

    b. Log in to the primary Unified node and and install the template. It is recommended that this step is run in a terminal opened with the **screen** command.

       i. Run **screen**.

       ii. Run **app template media/<VOSS Automate_template_file>**

       - The console will display a message:

       ```
       Deploying the template-product for VOSS Automate <<RELEASE_VERSION>> ...
       ```

    c. When prompted to select the product deployment type, provide and confirm the deployment type:

       - `Enterprise`
       - `Provider without HCM-F`
       - `Provider with HCM-F`

       In accordance with the selected deployment type, you are prompted to enter and verify:

       - a top-level administrator password:

         `Please enter a password for "sysadmin"`

       - and one administrator password - depending on the deployment:
         - Enterprise : `Please enter a password for "entadmin"`
         - Provider : `Please enter a password for "hcsadmin"`

Upon installation, the password length should be at least 8 characters.

Deployment-specific artifacts are installed according to the selected type of product deployment. A message displays according to the selected deployment type - one of:

```
"Importing EnterpriseOverlay.json"

"Importing ProviderOverlay_Hcmf.json ..."

"Importing ProviderOverlay_Decoupled.json ..."
```

Deployment specific system artifacts are imported and a message is displayed:

```
Deployment-specific Overlay artifacts successfully imported.
```

i. Python functions are deployed

ii. System artifacts are imported.

iii. You are prompted to provide administrator passwords.

The template install automatically restarts necessary applications. If a cluster the installation propagates changes throughout the cluster.

11. Review the output from the **app template** commands and confirm that the install message appears:

```
Deployment summary of UPDATED template solution (i.e. current values after␣
↪installation):
--------------------------------------------------------------------------------
↪----

Product: [PRODUCT]
Version: [UPDATED PRODUCT RELEASE]
Iteration-version: [UPDATED ITERATION]
Platform-version: [UPDATED PLATFORM VERSION]
```

You can also monitor the template installation from the Admin Portal transaction list.

- If there are no errors, as part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
- If there was an error, the install script stops with a failure message listing the problem. Contact VOSS Support.

12. Check for needed security updates by running the **cluster run all security check** command on the primary node. If at least one update is required for any node, run the **cluster run all security update** command on the primary Unified node.

    After the security update is successful, reboot the cluster:

    a. From the selected primary Unified node, run **cluster run notme system reboot**. Since all services will be stopped, this takes some time.

    b. From the selected primary Unified node, run **system reboot**. Since all services will be stopped, this takes some time.

    If a node does not properly reboot but the console shows that all processes have terminated, you can manually reboot the node without any system corruption.

13. (Optional) Install language templates for languages other than English.
    a. Copy the language template file to the primary Unified node with the command:

       scp <language_template_file> platform@<unified_node_ip_address>:~/media

    b. Log in to the primary Unified node and install the template with the command:

       app template media/<language_template_file>

       For example, to install French:

       app template media/VOSS AutomateLanguagePack_fr-fr.template

       There is no need to run this command on all nodes.
14. (Optional) If the VOSS Automate Phone Based Registration Add-on is required, follow the installation instructions in the Appendix of your Core Feature Guide:

    "Install the Phone Based Registration Web Service"
15. Run the following command:

    voss migrate_summary_attributes device/cucm/HuntPilot

5.6.2. Single-node cluster (cluster-of-one) Installation

Before You Begin

Before continuing, you should have followed the OVA installation according to the steps and preliminary requirements specified in: *Create a New VM Using the Platform-Install OVA*

Note:

- Template installation and upgrade takes approximately two hours. You can follow the progress on the GUI transaction list.

The standard **screen** command should be used where indicated. See: *Using the screen command*.

Procedure

1. Install VMware tools:
   a. Log in on the node as the platform user and run **app install vmware**.
   b. Verify that vmware is running: **app list**.
2. Prepare the node to be added as a single-node cluster ("cluster-of-one").

   Run **cluster prepnode**.
3. Add the node to the cluster: **cluster add <ip_addr>**.
   a. Verify the node is listed in the cluster: **cluster list**. Example:

```
platform@VOSS:~$ cluster list
Cluster has 1 nodes:
        application :  192.168.100.3
```

(continues on next page)

(continued from previous page)

```
        webproxy :  192.168.100.3
        database :  192.168.100.3
```

4. Initialize the database and clear all data with the **voss cleardown** command.

   Note that this step may take some time. You can follow the process by running **log follow upgrade_db.log** or **log follow voss-deviceapi/app.log**.

5. Add the network domain (optional if a domain name is needed).
   a. Configure the domain: **cluster run all network domain <domain_name>**.
   b. Verify the configured network domain: **cluster run all network domain**. The node shows the domain that you configured.
6. Check the network:
   a. Run **cluster check** to verify the status of the cluster, network connectivity, disk status and NTP.
   b. Verify the DNS configuration: **cluster run all network dns**. The node responds with the DNS server address.
7. As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
8. Import the templates.
   a. Copy the VOSS Automate template file to the node with the command:

      scp <VOSS Automate_template_file> platform@<ip_address>:~/media

   b. Log in to the node and install the template. It is recommended that this step is run in a terminal opened with the **screen** command.
      i. Run **screen**.
      ii. Run **app template media/<VOSS Automate_template_file>**
      - The console will display a message:

      ```
      Deploying the template-product for VOSS Automate <<RELEASE_VERSION>> ...
      ```

   c. When prompted to select the product deployment type, provide and confirm the deployment type:
      - `Enterprise`
      - `Provider without HCM-F`
      - `Provider with HCM-F`

      In accordance with the selected deployment type, you are prompted to enter and verify:
      - a top-level administrator password:

        `Please enter a password for "sysadmin"`
      - and one administrator password - depending on the deployment:
        – Enterprise : `Please enter a password for "entadmin"`
        – Provider : `Please enter a password for "hcsadmin"`

Upon installation, the password length should be at least 8 characters.

Deployment-specific artifacts are installed according to the selected type of product deployment. A message displays according to the selected deployment type - one of:

```
"Importing EnterpriseOverlay.json"

"Importing ProviderOverlay_Hcmf.json ..."

"Importing ProviderOverlay_Decoupled.json ..."
```

Deployment specific system artifacts are imported and a message is displayed:

```
Deployment-specific Overlay artifacts successfully imported.
```

i. Python functions are deployed

ii. System artifacts are imported.

iii. You are prompted to provide administrator passwords.

The template install automatically restarts necessary applications. The installation propagates changes throughout the cluster.

9. Review the output from the **app template** commands and confirm that the install message appears:

```
Deployment summary of UPDATED template solution (i.e. current values after␣
↪installation):
--------------------------------------------------------------------------------
↪----

Product: [PRODUCT]
Version: [UPDATED PRODUCT RELEASE]
Iteration-version: [UPDATED ITERATION]
Platform-version: [UPDATED PLATFORM VERSION]
```

You can also monitor the template installation from the Admin Portal transaction list.

- If there are no errors indicated, we recommend that you make a restore point.

  As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.

- If there was an error, the install script stops with a failure message listing the problem. Contact VOSS Support.

10. Check for needed security updates by running the **security check** command. If at least one update is required for any node, run the **security update** command.

    After the security update is successful, reboot: run **system reboot**.

    Since all services will be stopped, this takes some time.

    If the node does not properly reboot but the console shows that all processes have terminated, you can manually reboot without any system corruption.

11. (Optional) Install language templates for languages other than English.

a. Copy the language template file to the primary Unified node with the command:

scp <language_template_file> platform@<ip_address>:~/media

b. Log in to the primary Unified node and install the template with the command:

app template media/<language_template_file>

For example, to install French:

app template media/VOSS AutomateLanguagePack_fr-fr.template

12. (Optional) If the VOSS Automate Phone Based Registration Add-on is required, follow the installation instructions in the Appendix of your Core Feature Guide:

"Install the Phone Based Registration Web Service"

13. Run the following command:

voss migrate_summary_attributes device/cucm/HuntPilot

5.6.3. View Installation and Upgrade Transactions

Use this procedure to view transactions from a VOSS Automate installation or upgrade.

Procedure

1. Log in as sysadmin administrator.
2. Select **Administration Menu > Transactions**.
3. To view details on a transaction, click the transaction.

5.6.4. Installation Quick Reference

Note:

- These steps are described in depth in the VOSS Automate Install Guide.
- The standard **screen** command should be used where indicated, and the *reconnect* parameter is available if needed:
  - **screen** - start a new session
  - **screen -ls** - show sessions already available
  - **screen -r [screen PID]** - reconnect to a disconnected session

  VOSS recommends to use the **screen** command to avoid failures if the connection is interrupted whilst running the command. If the connection is interrupted whilst running the command in `screen` then the session can be retrieved by first listing the sessions PID currently running in screen: **screen -ls**, and then reconnecting to the session using **screen -r [screen PID]**.

General Steps

1. Download VOSS Automate install and patch media from:

   https://voss.portalshape.com > Downloads > VOSS Automate > XXX > New Installation

   where XXX is the release number.
2. Review sizing requirements and define the deployment model:
   - Single node cluster/cluster-of-one (Lab Only)
   - MicroCluster (Two Unified nodes clustered)
   - Cluster (4 Unified Nodes and 2 Web Proxies)
   - DR Cluster (6 Unified Nodes and 2 Web Proxies)
3. Define VMHost space on VMWare servers
4. Deploy the VOSS Automate OVA to VMHost(s)
5. After the VM is created, select the CD ROM configuration and verify the **Connect at Power On** check box is enabled.
6. Power on the VM.
7. Configure the options in the installation wizard.
8. Install VMWare Tools from VOSS Automate CLI as platform user:

   command: **app install vmware**
9. Continue below to chosen deployment model.

Single Node Cluster Deployment

1. Prepare the node for cluster command via the SSH CLI:

   command: **cluster prepnode**
2. Add the node to the cluster:

   command: **cluster add <node_ip-address>**
3. Verify the node is a member of the cluster:

   command: **cluster list**
4. Initialize the database and clear all data with the **voss cleardown** command.

   Note that this step may take some time. You can follow the process by running **log follow upgrade_db.log** or **log follow voss-deviceapi/app.log**.
5. (Optional) Set VOSS Automate Network Domain:

   command: **cluster run all network domain <yourdomain>**
6. Check application status:

   command: **cluster status**
   - Should any services be in a down state, restart all services on that affected node:

     command: **cluster run <node_ip> app start**
7. Run the command: **voss cleardown**.

8. SFTP the install templates to the VOSS Automate server `media` directory
9. Install VOSS Automate Templates via VMWare Console CLI. VOSS recommends that this step is run in a terminal opened with the **screen** command, or on the VMWare console.
    a. Run **screen**.
    b. Run **app template media/<VOSS Automate_Template_Name>**.

    - A Deployment Type choice must be made during the template install execution. Choose one of:
        - Enterprise
        - Provider without HCM-F
        - Provider with HCM-F

      In accordance with the selected deployment type, you are prompted to enter and verify:
        - a top-level administrator password:

          `Please enter a password for "sysadmin"`
        - and one administrator password - depending on the deployment:
            * Enterprise : `Please enter a password for "entadmin"`
            * Provider : `Please enter a password for "hcsadmin"`

      Upon installation, the password length should be at least 8 characters.

Multi Node Deployment

All of the following commands will be run on the primary node via the SSH CLI until specified to use ESX Console CLI. The designation of primary unified node is arbitrary. The deploying administrator can pick any unified node that they see fit.

1. On each node that is not the designated primary unified node. prepare the servers for cluster command via the SSH CLI:

   command: **cluster prepnode**
2. Add all of the other nodes to the cluster:

   command: **cluster add <non-primary-node_ip-address>**

   Repeat this command for each other node - binding each individual node IP Address to the cluster. This command does not need to be run for the primary unified node.
3. Verify all nodes are members of the cluster:

   command: **cluster list**
4. (Optional) Set VOSS Automate Network Domain:

   command: **cluster run all network domain <yourdomain>**
5. Set each unified node's database weight:

   command: **database weight <un-node_ip-address> <priority_weight>**
    - This command must be run for all unified nodes primary and secondary.
    - Priority weights of 40, 30 are recommended for *two* Unified nodes.
    - Priority weights of 40, 30, 20, and 10 are recommended for *four* Unified nodes
    - Priority weights of 60, 50, 40, 30, 20, and 10 are recommended for *six* Unified nodes.

6. Provision the VOSS Automate cluster database. VOSS recommends that this step is run in a terminal opened with the **screen** command, or on the VMWare console.
   a. Run **screen**.
   b. command: **cluster provision**
7. Check cluster application status:

   command: **cluster status**
   - Should any services be in a down state, restart all services on that affected node:

     command: **cluster run <node_ip> app start**
8. Run the command: **voss cleardown**.
9. SFTP the install templates to the VOSS Automate server `media` directory of the primary unified node.
10. Install VOSS Automate Templates via VMWare Console CLI of primary unified node. VOSS recommends that this step is run in a terminal opened with the **screen** command, or on the VMWare console.
    a. Run **screen**.
    b. Run **app template media/<VOSS Automate_Template_Name>**
11. A Deployment Type choice must be made during the template install execution. Choose one of:
    - Enterprise
    - Provider without HCM-F
    - Provider with HCM-F

    In accordance with the selected deployment type, you are prompted to enter and verify:
    - a top-level administrator password:

      `Please enter a password for "sysadmin"`
    - and one administrator password - depending on the deployment:
      – Enterprise : `Please enter a password for "entadmin"`
      – Provider : `Please enter a password for "hcsadmin"`

    Upon installation, the password length should be at least 8 characters.

Post Deployment

1. Access the VOSS Automate web interface via any web browser:

   `https://<ip_address_or_dns_name_of_VOSS Automate_PrimaryUN_or_WebProxy>`
2. Run the following security commands:
   - **cluster check** - inspect entries under `security`.
   - If needed, **security update**.
3. Run the following command:

   voss migrate_summary_attributes device/cucm/HuntPilot

5.6.5. Migrating from a 6 Node to 8 Node System

On a standard topology, to migrate a clustered 6 node system (4 unified nodes and 2 WebProxy nodes) to a clustered 8 node system (6 unified nodes and 2 WebProxy nodes), the considerations and steps below are required.

1. Check and make a restore point of the clustered 6 node system *before* adding the nodes:
   a. Run **cluster list** to ensure the node count is correct.
   b. Run **cluster status** to check all nodes are online and services reported as running.
   c. Run **cluster run database cluster list** to make sure all unified nodes are aware of the current cluster nodes.
   d. Run **cluster run all app status** to make sure all services are running on all nodes.
   e. Make a restore point of the entire 6 node cluster.

      As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
2. Add the 2 unified nodes:
   a. Create the new unified node - see: *Create a New VM Using the Platform-Install OVA*.
   b. Run **cluster prepnode** *on all nodes*, including new nodes.
   c. From the primary unified node, run **cluster add <ip>** for new nodes, excluding itself.
3. Reset the cluster database weights. When nodes are removed from and added to a cluster, remove all database weights completely and add them back in *before provisioning* to reset the configuration.
   a. Delete all database weights in the cluster. For each IP, run **database weight del <IP>**.
   b. To add database weights in, you set the weight of the intended primary, but must always specify the current primary (using **database primary**), regardless of whether the new intended primary is the same node or not. During the provision process, the role of primary will then be transferred from the existing primary to the node with the highest weight.

      Determine the current primary database node with **database primary**.
4. Run the command **database weight add <IP> <numeric>** on the primary database node for each IP, making the value of the intended primary database node the highest value.
5. Check the cluster before provisioning:
   a. Run **cluster list** to ensure the node count is correct.
   b. Run **cluster status** to check all nodes are online and services reported as running.
   c. Run **cluster run database cluster list** to make sure all unified nodes are aware of the current cluster nodes.
   d. Run **cluster run all app status** to make sure all services are running on all nodes. Fresh nodes that have not been provisioned will show a message: `suspended waiting for mongo`.
6. Run **cluster provision** to provision the cluster.
7. After a successful migration, the restore point made in step 1. can be removed.

5.7. Modular Cluster Topology Installation

5.7.1. Modular Architecture Multinode Installation

Note: A modular archtecture installation is not supported for a single node cluster ("cluster of one") topology.

Before You Begin

Before continuing, you should have followed the OVA installation on each node according to the steps and preliminary requirements specified in: *Create a New VM Using the Platform-Install OVA* and according to the node roles as indicated in *Notes on Multi-Node Installation*. Data center names are also selected at this stage.

For example, for an 8-node modular cluster in 2 data centers:

- DC1 = primary site or data center containing primary database node (highest database weight)
- DC2 = data recovery (DR) data center

Install:

- 3 nodes with Database roles (2 in DC1, 1 in DC2)
- 3 nodes with Application roles (2 in DC1, 1 in DC2)
- 2 nodes with WebProxy roles (1 in DC1, 1 in DC2)

Optionally download or extract language pack template files to support languages other than English.

Note:

- For typical modular geo-redundant multinode cluster deployment with 3 database and 3 application nodes, there are:
  - two application nodes in the primary Site
  - two database nodes in the primary Site
  - one application node in the Disaster Recovery (DR) Site
  - one database node in the Disaster Recovery (DR) Site

  The worker count (**voss workers** command) needs to be set on the DR nodes. Refer to:
  - *Multinode Cluster with Unified Nodes*
  - *Multi Data Center Deployments*
- NAT between nodes is not allowed
- If there is a firewall between nodes, then specific ports must be configured. For port configuration, refer to:
  - *Clustering Considerations*
  - *Network Communications between Nodes within the Cluster*
  - *Network Communications External to the Cluster*

- Template installation and upgrade takes approximately two hours. You can follow the progress on the Admin Portal transaction list.
- It is strongly recommended *not* to allow customer end-users the same level of administrator access as the restricted groups of provider- and customer administrators. This is why Self-service web proxies as well as Administrator web proxies should be used.

  Systems with Self-service only web proxies are *only* recommended where the system is customer facing, but where the customer does not administer the system themselves.
- For cluster installations, also refer to the Health Checks for Cluster Installations Guide.
- If it is necessary to change an IP address of a node in a cluster, first remove it from the cluster by running the command below *on the node to be changed*:

  cluster del <IP address of node to be changed>
- Refer to *Installation Logs* for troubleshooting logs during an installation.

The standard **screen** command should be used where indicated. See: *Using the screen command*.

Procedure

1. Install VMware tools on each node.
   a. Log in to each node and run **app install vmware**.
   b. Verify that vmware is running: **app list**.
2. Prepare each node to be added to the cluster:
   a. Select a database node that will become the primary database node. The *primary site* or data center will contain the primary database node. The deploying administrator can pick any database node that they see fit.
   b. On each node, run **cluster prepnode**.
3. Add nodes to the cluster.
   a. Log in to the selected primary database node.
   b. Add the other database, application and WebProxy nodes to the cluster: **cluster add <ip_addr>**.

      Note that you do not have to add the selected primary database node to the cluster. It will autmatically be added to the cluster.
   c. Verify the list of nodes in the cluster: **cluster list**.
4. Add the network domain (optional if a domain name is needed). From the selected primary database node:
   a. Configure the domain: **cluster run all network domain <domain_name>**.
   b. Verify the configured network domain: **cluster run all network domain**. Each node shows the domain that you configured.
5. Check the network:
   a. From the selected primary database node, run **cluster check** to verify the status of the cluster, network connectivity, disk status and NTP.
   b. Verify the DNS configuration: **cluster run all network dns**. Each node responds with the DNS server address.

6. Create a restore point. As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
7. Configure the cluster.
   a. From the selected primary database node, provide a weight for each database server with the **database weight add <database_ip> <priority>** command.

      The higher the value, the more priority.

      - A weight of 40 for the primary database node at the primary site (DC1)
      - A weight of 30 for the secondary database node at the primary site (DC1)
      - A weight of 10 for the secondary database node at the DR (Data Recovery) site (DC2)

   b. From the selected primary database node:

      It is recommended that this step is run in a terminal opened with the **screen** command.

      i. Run **screen**.

      ii. Run **cluster provision**

      Allow approximately 2 hours for the operation to complete for two WebProxy and four database, application nodes.

   c. When provisioning is complete, check that each node is contactable and that the time server is running on each with **cluster check**.

      If a service is down, run **cluster run <node_ip> app start** to restart the service.

      If provisioning is successful, the screen session can be closed by typing **exit** in the screen terminal. If errors occurred, keep the screen terminal open for troubleshooting purposes and contact VOSS support.

   d. On each of the new *application* nodes, set the queues to 2 with the command **voss queues 2**.

      Note: Applications are reconfigured and the `voss-queue` process is restarted.

   e. (Optional) If required, set the web weights configurations (Active-Active, Active-Standby, Single node cluster). From the primary database node, run the required **web weight** commands for the Web Proxy nodes. For details, refer to *Multi Data Center Deployments* and the VOSS Automate Best Practices Guide.

   f. (Optional) If required, enable or disable Self-service or admin web services on the web proxy nodes. This may for example be needed for security purposes.

      The commands must be run on the relevant web proxy node. The commands will automatically reconfigure and restart the *nginx* process, so some downtime will result. Request URLs to a disabled service will redirect the user to the active service.

      - To disable or enable admin or Self-service web services on the web proxy node:

        web service disable <selfservice|admin>

        web service enable <selfservice|admin>

      - To list web services on the web proxy node:

        web service list

8. Create a restore point. As part of the rollback procedure, ensure that a suitable restore point is obtained prior to the start of the activity, as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
9. Initialize the database and clear all data. On an *application node*, run **voss cleardown**.

   Note that this step may take some time. You can follow the process by running **log follow upgrade_db.log** or **log follow voss-deviceapi/app.log** in a separate console on the application node.
10. Import the templates.
    a. Copy the VOSS Automate template file to *an application node* with the command:

       scp <VOSS Automate_template_file> platform@<app_node_ip_address>:~/media

    b. Log in to *this application* node and install the template. It is recommended that this step is run in a terminal opened with the **screen** command.
       i. Run **screen**.
       ii. Run **app template media/<VOSS Automate_template_file>**
       - The console will display a message:

       ```
       Deploying the template-product for VOSS Automate <<RELEASE_VERSION>> ...
       ```

    c. When prompted to select the product deployment type, provide and confirm the deployment type:
       - `Enterprise`
       - `Provider without HCM-F`
       - `Provider with HCM-F`

       In accordance with the selected deployment type, you are prompted to enter and verify:
       - a top-level administrator password:

         `Please enter a password for "sysadmin"`
       - and one administrator password - depending on the deployment:
         – Enterprise : `Please enter a password for "entadmin"`
         – Provider : `Please enter a password for "hcsadmin"`

       Upon installation, the password length should be at least 8 characters.

       Deployment-specific artifacts are installed according to the selected type of product deployment. A message displays according to the selected deployment type - one of:

       ```
       "Importing EnterpriseOverlay.json"

       "Importing ProviderOverlay_Hcmf.json ..."

       "Importing ProviderOverlay_Decoupled.json ..."
       ```

       Deployment specific system artifacts are imported and a message is displayed:

       ```
       Deployment-specific Overlay artifacts successfully imported.
       ```

       i. Python functions are deployed
       ii. System artifacts are imported.

iii. You are prompted to provide administrator passwords.

The template install automatically restarts necessary applications. If a cluster the installation propagates changes throughout the cluster.

11. Review the output from the **app template** commands and confirm that the install message appears:

```
Deployment summary of UPDATED template solution (i.e. current values after␣
↪installation):
--------------------------------------------------------------------------------
↪----

Product: [PRODUCT]
Version: [UPDATED PRODUCT RELEASE]
Iteration-version: [UPDATED ITERATION]
Platform-version: [UPDATED PLATFORM VERSION]
```

You can also monitor the template installation from the Admin Portal transaction list.

- If there are no errors indicated, we recommend a suitable restore point is created as per the guidelines for the infrastructure on which the VOSS Automate platform is deployed.
- If there was an error, the install script stops with a failure message listing the problem. Contact VOSS Support.

12. Check for needed security updates by running the **cluster run all security check** command on the primary database node. If at least one update is required for any node, run the **cluster run all security update** command on the primary database node.

    After the security update is successful, reboot the cluster:

    a. From the selected primary database node, run **cluster run notme system reboot**. Since all services will be stopped, this takes some time.

    b. From the selected primary database node, run **system reboot**. Since all services will be stopped, this takes some time.

    If a node does not properly reboot but the console shows that all processes have terminated, you can manually reboot the node without any system corruption.

13. (Optional) Install language templates for languages other than English.

    a. Copy the language template file to the *selected application node* with the command:

    scp <language_template_file> platform@<app_node_ip_address>:~/media

    b. Log in to the *application* node and install the template with the command:

    app template media/<language_template_file>

    For example, to install French:

    app template media/VOSS AutomateLanguagePack_fr-fr.template

    There is no need to run this command on all nodes.

14. (Optional) If the VOSS Automate Phone Based Registration Add-on is required, follow the installation instructions in the Appendix of your Core Feature Guide:

    "Install the Phone Based Registration Web Service"

15. Run the following command:

    voss migrate_summary_attributes device/cucm/HuntPilot

6. VOSS Automate Azure Installation

6.1. Installing Azure

Azure for VOSS Automate is available as a docker image that installs by means of terraform scripts.

During the run, terraform saves the plan file (`terraform.tfstate`) to the folder location created under VOSS Automate Azure Installation Procedure - Step 3.

The file `terraform.tfstate` should be backed up. This will allow you to expand the installation in the future.

6.1.1. Hardware Requirements

For details on Standard and Modular Topologies, refer to the VOSS Automate Architecture and Hardware Specification Guide and Platform Guide.

Unified or Database Nodes:

- VM Size: E4ds_v4 Standard
- CPU: 4
- RAM: 32
- OS disk: 40GB, Premium_LRS
- application disk: 50GB, Premium_LRS
- backup disk: 55 GB, Standard LRS
- DB disk: 250 GB, Premium_LRS
- Total disk size: 395GB

Application Nodes:

- VM Size: E4ds_v4 Standard
- CPU: 4
- RAM: 32
- OS disk: 40GB, Premium_LRS
- application disk: 50GB, Premium_LRS
- Total disk size: 90GB

Network:

- Address space: 10.0.0.0/16

- Subnet prefix: 10.0.0.0/24

Web Proxies:

- Web Proxies are replaced by the Azure Load Balancer

6.1.2. Network Communications External to the Cluster

The following details are all based on the default settings. These can vary depending on the application setup and network design (such as NAT) of the solution, so may need adjustment accordingly. Where a dependant is noted, this is fully dependant on the configuration with no default.

These communications are all related to communications with devices external to the cluster.

- Outbound Communications to Devices from the Application/Unified nodes:

Communication	Protocol	Port
Cisco Unified Communications Manager (UCM)	HTTPS	TCP 8443
Cisco Unity Connection (CUXN)	HTTPS	TCP 443
Webex	HTTPS	TCP 443
LDAP directory	LDAP	TCP/UDP 389 and/or 636(TLS/SSL)
Cisco HCM-F	HTTPS	TCP 8443

- Unified Node to Unified node

  This is relevant to the communications between the unified nodes (application and database combined). If the application and database nodes are split, then see the relevant application and database node details below. Database arbiters run on port 27030.

Communication	Protocol	Port
Database access	database	TCP 27020 and 27030 bi-directional
Cluster Communications	HTTPS	TCP 8443

6.1.3. Ubuntu and Docker Installation Procedure

1. Deploy a LTS version of Ubuntu on Azure Portal. The instructions to follow are a guide for Ubuntu 20.04 (Focal Fossa).

   ```
   https://ubuntu.com/#download
   ```

2. Install Docker on the newly deployed Ubuntu 20.04 VM. Docker can be installed either online from the repository or offline with the required packages downloaded.

   For the online procedure - follow the instructions from step 3.

   For the offline procedure - follow the instructions from step 4.

3. Online Docker installation

   3.1. Open a terminal window.

   3.2. Check if the system is up-to-date:

   ```
   sudo apt update
   ```

   3.3. Install Docker:

   ```
   sudo apt install docker.io
   ```

   3.4. Verify that Docker Engine - Community is installed correctly by running the hello-world image:

   ```
   sudo docker run hello-world
   ```

   3.5. Manage Docker as a non-root user. Add your user to the docker group:

   ```
   sudo usermod -aG docker $USER
   ```

   3.6. Reboot the system so that your group membership is re-evaluated.

   3.7. Verify that you can run docker commands without sudo:

   ```
   docker run hello-world
   ```

4. Offline Docker Installation using downloaded packages.

   4.1. Download the latest packages from the offical Docker website.

   ```
   https://download.docker.com/linux/ubuntu/dists/<dist codename>/pool/stable/
   ↪amd64/

   For Ubuntu 20.04 (Focal Fossa) - https://download.docker.com/linux/ubuntu/
   ↪dists/focal/pool/stable/amd64/

   Files required are:
   containerd.io_<latest_version>.deb
   docker-ce-cli_<latest_version>.deb
   docker-ce_<latest_version>.deb
   ```

   4.2. Open a terminal window.

   4.3. cd to the folder with the downloaded packages.

   4.4. Install the downloaded packages of Docker Engine - Community and containerd:

   ```
   sudo dpkg -i *.deb
   ```

   4.5. Verify that Docker Engine - Community is installed correctly by running the hello-world image:

   ```
   sudo docker run hello-world
   ```

   4.6. Manage Docker as a non-root user. Add your user to the docker group:

```
sudo usermod -aG docker $USER
```

4.7. Reboot the system so that your group membership is re-evaluated.

4.8. Verify that you can run docker commands without sudo:

```
docker run hello-world
```

6.1.4. Azure Portal Configuration

1. Register an application

```
Search for and select Azure Active Directory.
Under Manage, select App registrations > New registration.
```

2. Create a new application secret

```
From App registrations in Azure AD, select your application.
Select Certificates & secrets.
Select Client secrets -> New client secret.
Provide a description of the secret, and a duration. When done, select Add.

After saving the client secret, the value of the client secret is displayed.
↪ Copy this value because you won't be able to retrieve the key later.
Store the key value where your application can retrieve it.
```

3. Assign a role to the application

```
In the Azure portal, search for and select Subscriptions, or select␣
↪Subscriptions on the Home page.
Select the particular subscription to assign the application to.
Select Access control (IAM). Select Add role assignment. Select the␣
↪Contributor role.
Assign access to Azure AD user, group, or service principal.
To find your application, search for the name and select it as done in step␣
↪1.
```

6.1.5. VOSS Automate Azure Installation Procedure

1. Download the VOSS Automate docker image along with the sha256 checksum from the provided location and verify the checksum.

```
sha256sum platform-azure.tgz
cat platform-azure.tgz.sha256
```

2. Run: **gunzip -c platform-azure.tgz | docker load**

```
77cae8ab23bf: Loading layer [======>]  5.815MB/5.815MB
f7030b11e80f: Loading layer [======>]  25.69MB/25.69MB
ebf6924cf16a: Loading layer [======>]  50.64MB/50.64MB
3cf5f2bba2c7: Loading layer [======>]  1.536kB/1.536kB
fda51b70dfb1: Loading layer [======>]  14.85kB/14.85kB
75a157f47953: Loading layer [======>]  1.897GB/1.897GB
e1dad0f87419: Loading layer [======>]  115.9MB/115.9MB
Loaded image: voss-azure:19.3.1-1583240523
```

3. Create a directory to store the plan file (`terraform.tfstate`).

   For example, **mkdir -p /home/azure/plan**

4. For installations over ssh, it is recommanded to use a screen session.

   ```
   screen - start a new session
   screen -ls - show sessions already available
   screen -r [screen PID] - reconnect to a disconnected session
   ```

5. Run **docker run -it -v <directory created>:/app/state voss-azure:<release_number>**

   For example, **docker run -it -v /home/azure/plan/:/app/state/ voss-azure:19.3.1-1583240523**

   NB: Once the install is complete, please store this directory for future use.

6. Enter values at the prompt for the terraform script variables:

   - If installing on a standard, unified node topology, select: `var.voss_count_app=0`, `var.voss_count_unified=4`.
   - If installing a for a modular topology, select: `var.voss_count_app=2`, `var.voss_count_unified=0`.
   - For details on Standard and Modular Topologies, refer to the VOSS Automate Architecture and Hardware Specification Guide and Platform Guide.

Important: The user input is in clear text.

Variable	Description
var.azure_client_id	The Azure user ID of deployer Azure portal: From App registrations in Azure AD, select your application.
var.azure_client_secret	The Azure password of deployer (secret) Azure portal: From App registrations in Azure AD, select your application. Select Certificates & secrets. This value was required to be stored upon creation in step 2 of the Azure Portal Configuration.
var.azure_location	The location/region where the VOSS Automate system is created. Changing this forces a new resource to be created. List of available regions - centralus, eastasia, southeastasia, eastus, eastus2, westus, westus2, northcentralus, southcentralus, westcentralus, northeurope, westeurope, japaneast, japanwest, brazilsouth, australiasoutheast, australiaeast, westindia, southindia, centralindia, canadacentral, canadaeast, uksouth, ukwest, koreacentral, koreasouth, francecentral, southafricanorth, uaenorth, australiacentral, switzerlandnorth, germanywestcentral, norwayeast, jioindiawest, australiacentral2
var.azure_resource_group_name	The name of the resource group in which to create the VOSS Automate system.
var.azure_subscription_id	The Azure subscription ID of deployer Azure portal: Under the Azure services heading, select Subscriptions. Your Subscription IDs are listed in the second column.
var.azure_tenant_id	The Azure tenant ID of deployer Azure portal: From App registrations in Azure AD, select your application.
var.fault_domain_count	Set the fault domain count. The number of fault domains varies depending on which Azure region you're using. Refer to the official Azure documentation.
var.ntp_name	NTP server to be used by VOSS Automate (e.g. ntp.ubuntu.com). Please make sure this is a publicly accessible NTP server.
var.update_domain_count	Set the update domain count. This should match the fault_domain_count.
var.voss_env	Tag for the environment (Example QA/Production)
var.voss_password	Password to be used for VOSS Automate 'platform' user
var.voss_count_app	Number of VOSS Automate **application nodes** to deploy
var.voss_count_unified	Number of VOSS Automate **unified nodes** to deploy

7. A terraform execution plan and resources are created (example resource):

```
An execution plan has been generated and is shown below.
Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # azurerm_availability_set.avset will be created
  + resource "azurerm_availability_set" "avset" {
      + id                           = (known after apply)
      + location                     = "southafricanorth"
      + managed                      = true
```

(continues on next page)

(continued from previous page)

```
      + name                          = "vossavset"
      + platform_fault_domain_count   = 3
      + platform_update_domain_count  = 3
      + resource_group_name           = "VOSS_Cloud"
      + tags                          = {
          + "environment" = "dev"
          + "systemname"  = "voss"
        }
    }

[...]
```

Verify the resources in the plan:

```
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.
```

8. When to execution plan is carried out successfully, the login details for the created nodes are shown, for example:

   - Unified topology

   ```
   azurerm_virtual_machine.voss_un[0]: Creation complete after 25m9s [id=...]

   Outputs:

   VM-ssh-access = tolist([
     "ssh platform@102.37.108.121 -p 50001",
     "ssh platform@102.37.108.121 -p 50002",
     "ssh platform@102.37.108.121 -p 50003",
     "ssh platform@102.37.108.121 -p 50004",
   ])
   VM-ssh-access-App_nodes = tolist([])
   VM-ssh-access-DB_nodes = tolist([])
   ```

   - Modular topology

   ```
   azurerm_virtual_machine.voss_app[1]: Creation complete after 24m23s [id=..]

   Outputs:

   VM-ssh-access = tolist([])
   VM-ssh-access-App_nodes = tolist([
     "ssh platform@102.37.107.75 -p 50011",
     "ssh platform@102.37.107.75 -p 50012",
   ])
   VM-ssh-access-DB_nodes = tolist([
     "ssh platform@102.37.107.75 -p 50021",
     "ssh platform@102.37.107.75 -p 50022",
     "ssh platform@102.37.107.75 -p 50023",
   ])
   ```

9. *On each of the newly deployed nodes*, log in as the platform user, using the password selected.

   9.1. For example `ssh platform@10.13.168.197 -p 50001`

   9.2. Run: **system reboot**. Alternatively you can reset the Virtual Machine from the Azure Portal. This is to ensure all services are running before proceeding with the cluster configuration.

10. *On each of the newly deployed nodes*, log in as the platform user, using the password selected.

    10.1. For example `ssh platform@10.13.168.197 -p 50001`

    10.2. Run: **cluster prepnode**

    10.3. Obtain the IP address. Run: **network interfaces**

    Example output:

```
$ network interfaces
    interfaces:
        eth0:
            gateway: 10.0.0.1
            ip: 10.0.0.4
            netmask: 255.255.255.0
```

11. Log in to the first node above again, for example `ssh platform@10.13.168.197 -p 50001`.

    11.1. Add the IP addresses obtained in the previous step to the cluster:

    cluster add <IP1>, cluster add <IP2>, . . .

    11.2. Add database weights to your database nodes:

    Log in to the database node.

    For *modular deployment*, use the `50021` port to log into the database, for example:

```
ssh platform@152.77.167.75 -p 50021
```

    database weight add <IP1> <priority>, database weight add <IP2> <priority>, . . .

    - Weights of 40, 30 are recommended for two Unified nodes
    - Weights of 40, 30, 20, and 10 are recommended for four Unified nodes
    - Weights of 60, 50, 40, 30, 20, and 10 are recommended for six Unified nodes

    The higher the value, the higher priority.

    11.3. Run **cluster provision**

    11.4. Run **voss cleardown**

    11.5. Upload the provided template install file via the scp (secure copy protocol) to the media folder on the primary node

    11.6. Install the templates.

    app template media/<template file name>.template

6.2. Deploying Additional Azure Nodes

1. To deploy additional nodes, make sure the docker image has been loaded with the current VOSS Automate version (for example, 19.3.1 in the example below) in the deployment.

```
gunzip -c platform-azure.tgz | docker load
Loaded image: voss-azure:19.3.1-1583240523
```

2. Run the install command and specify the absolute folder path location of the `terraform.tfstate` file.

   Example (`/home/azure/plan`):

```
/home/azure/plan$ ls -lA
-rw-r--r-- 1 root root 37232 Feb 27 17:53 terraform.tfstate

docker run -it -v /home/azure/plan/:/app/state voss-azure:<release_number>
```

3. Enter values at the prompt for the terraform script variables.
4. Total number of VOSS Automate nodes in deployment. Enter the total number of nodes required.

   Example:

   - Unified topology (`var.voss_count_unified`)

```
Initital deployment - 4 Nodes
Additional nodes required - 2 Nodes
Total Number of nodes to enter at the prompt - 6
```

   - Modular topology (`var.voss_count_app`)

```
Initital deployment - 2 app nodes
Additional nodes required - 1 app node

var.voss_count_app=3
```

5. The additional nodes will be deployed and a new/current terraform plan state will be saved. The initial plan state will be renamed to `terraform.tfstate.backup`.

```
/home/azure/plan$ ls -lA
-rw-r--r-- 1 root root 37232 Mar 1  13:11 terraform.tfstate
-rw-r--r-- 1 root root 37232 Feb 27 17:53 terraform.tfstate.backup
```

6. *On each of the newly deployed nodes*, log in as the platform user, using the password selected.

   6.1. For example `ssh platform@10.13.168.197 -p 50001`

   6.2. Run: **system reboot**. Alternatively you can reset the Virtual Machine from the Azure Portal. This is to ensure all services are running before proceeding with the cluster configuration.

7. *On each of the newly deployed nodes*, log in as the platform user, using the password selected.

   7.1. For example `ssh platform@10.13.168.197 -p 50001`

   7.2. Run: **cluster prepnode**

   7.3. Obtain the IP address. Run: **network interfaces**

Example output:

```
$ network interfaces
     interfaces:
         eth0:
             gateway: 10.0.0.1
             ip: 10.0.0.4
             netmask: 255.255.255.0
```

8. Determine which node is the primary node.
   - Run the following command on an Application/Unified node to determine the PRIMARY NODE:
   - Command : `cluster run application cluster primary role application`
   - Search for node with `is_primary:  true`

   Log in to the primary node, for example `ssh platform@10.13.168.197 -p 50001`.

   8.1. Add the newly deployed nodes IP addresses obtained in the previous step to the cluster:

   cluster add <IP1>, cluster add <IP2>, . . .

   8.2. Add database weights to your database nodes:

   For *modular deployment*, this step can be skipped.

   database weight add <IP1> <priority>, database weight add <IP2> <priority>, . . .
   - Weights of 40, 30 are recommended for two Unified nodes
   - Weights of 40, 30, 20, and 10 are recommended for four Unified nodes
   - Weights of 60, 50, 40, 30, 20, and 10 are recommended for six Unified nodes

   The higher the value, the higher priority.

   8.3. Run **cluster provision**

Index

A

app
    app install, 38, 41, 46, 54
    app template, 37, 38, 41, 54

B

backup
    backup create, 26

C

cluster, 12
    cluster add, 38, 41, 53, 54
    cluster del, 38, 41, 54
    cluster list, 53
    cluster prepnode, 38, 41, 53, 54
    cluster primary, 41
    cluster primary role application, 41
    cluster primary role database, 41
    cluster provision, 37, 38, 41, 53, 54
    cluster run, 25, 38, 41, 53, 54
    cluster status, 38, 41, 53, 54
    cluster upgrade, 37

D

database
    database weight, 38, 41, 54
drives
    drives add, 26

L

log
    log follow, 32, 38, 41, 46, 54

N

network
    network container range, 25
    network domain, 46

S

screen, 37, 38, 41, 49, 54
security
    security update, 32, 46
system
    system reboot, 46

V

voss
    voss cleardown, 32, 38, 41, 46, 54
    voss db_collection_cap, 28
    voss db_collection_stats, 26
    voss queues, 54
    voss upgrade_db, 32
    voss workers, 15, 22
voss export
    voss export group, 37
    voss export type, 37
voss subscriber_data_export, 37

W

web
    web service, 15, 22, 32
    web weight, 5, 38, 41, 54
    web weight add, 5
    web weight list, 5