Executive Summary

Today’s global economy is driven by data. Businesses of all types and sizes continually find new ways to drive better decision making and profitability through large-scale analytics based on massive and ever-growing datasets. At the same time, the pace of change continues to accelerate, with greater agility taking on a major role as a driver of competitive advantage. That leaves IT organizations struggling to provide massive computing resources that can turn on a dime, even as their budgets continue to shrink.

A hybrid cloud infrastructure is an innovative approach to meeting the need to do more with less. Combining the benefits of on-premises infrastructure with those of public clouds, this model is gaining traction as a forward-looking approach for businesses as they adapt their existing computing resources to emerging needs. Co-engineered solution stacks from VMware and Intel offer a path forward through this field of opportunity.

This reference architecture (RA) demonstrates how to build security-enabled, scalable, and high-performance infrastructures and services that are easily consumable, like database as a service (DBaaS). The solution described in this RA is based on the Intel® Select Solutions for VMware Cloud Foundation* reference design. The hardware stack for the solution is built from VMware vSAN ReadyNodes* based on Intel® Server Board S2600WF platforms. The platforms include the latest generation of Intel® Xeon® Platinum processors and Intel Xeon Gold processors. For high-performance, all-flash software-defined storage, the platforms include Intel® Optane™ Solid State Drive (SSD) DC P4800X Series drives and Intel® SSD DC P4500 Series with NVMe Express* (NVMe*) drives. For accelerated software-defined network, the platforms use 10 Gb Intel® Ethernet Converged Network Adapters X710. The Intel® Data Center Blocks for Cloud (Intel® DCB for Cloud) for vSAN ReadyNodes are fully validated, pre-configured server systems that include VMware* certifications for VMware vSAN*.

From a software perspective, the solution consists of VMware Cloud Foundation, including the VMware vSphere® hypervisor with vSAN, VMware NSX®, and VMware SDDC Manager* providing infrastructure-as-a-service (IaaS) capabilities. The solution also includes VMware vRealize Automation* suite, which allows delivering platform-as-a-service (PaaS)-like capabilities to end users through a self-service portal.
Table of Contents

Executive Summary ................................................................. 1
Authors .............................................................................. 1
Reference Architecture Overview ...................................... 3
VMware vRealize Automation* ................................................ 4
   Basic Concepts .................................................................. 4
   Logical Architecture and Multi-Tenancy .............................. 4
   Default User Accounts ....................................................... 4
   Service Catalog and Catalog Items ..................................... 5
   Services .......................................................................... 5
   Actions ........................................................................... 5
   Entitlements ...................................................................... 5
   Blueprints ........................................................................ 5
   Hardware Specification ...................................................... 5
Software Specification ............................................................. 7
VMware Cloud Foundation Deployment ................................. 8
   Hardware Configuration and Wiring ................................. 8
   Imaging Infrastructure Preparation .................................... 8
   VMware Cloud Foundation Imaging ................................ 8
   VMware Cloud Foundation Bring-Up ............................... 9
   Creating Workload Domains ............................................ 10
vRealize Automation Deployment ........................................ 13
   vRealize Automation Infrastructure Overview .................. 13
   IaaS Web Server ............................................................ 13
   Manager Service ............................................................. 13
   Microsoft SQL Server* Database .................................... 13
   Distributed Execution Manager (DEM) ............................. 13
   Agents ............................................................................ 13
   Load Balancers ................................................................ 13
   vRealize Automation Deployment Prerequisites .................. 13
   Active Directory* Service and Account .............................. 14
   Time Synchronization ....................................................... 14
   SQL Server ..................................................................... 14
   Multi-SAN Certificates for vRealize Automation Components 14
   Network Configuration ..................................................... 15
   Windows 2012 R2* Template for IaaS .............................. 16
   vRealize Automation Deployment in VMware SDDC Manager* 16
   Starting the VMware vRealize Orchestrator* Service ......... 21
   Connecting Workload Domains to vRealize Automation .... 23
   vRealize Automation Initial Configuration and Blueprints Deployment ........................................ 24
   Initial Configuration .......................................................... 24
   Fabric Group Creation ....................................................... 24
   Linux* Template VM Preparation ....................................... 24
   Data Collection .................................................................. 25
   Configure the vRealize Orchestrator Plug-in for VMware vCenter* ........................................ 25
   Blueprint Import .............................................................. 26
   Obtaining the Script and Blueprints .................................. 26
   Script Configuration .......................................................... 26
   Script Execution ............................................................... 26
   Script Functionality Overview .......................................... 26
   Final Configuration ........................................................... 27
   Machine Prefixes .............................................................. 27
   Testing ........................................................................... 28
   Example Blueprints .......................................................... 29
   Workload Migration ........................................................... 30
   A Consumable and Scalable Hybrid Cloud ........................ 32
   Learn More ..................................................................... 32
Reference Architecture Overview

The RA consists of three key layers: the hardware infrastructure upon which the solution is constructed, VMware Cloud Foundation with its suite of products and services, and vRealize Automation suite with supporting automation script and service blueprints (see Figure 1).

The hardware-infrastructure layer consists of servers and network switches. The solution can scale from a single rack with eight servers up to eight racks and 256 total servers—all of which can be managed from a single instance. Heterogeneous server configurations are supported, enabling users to run different workload types on the same set of resources while still meeting specific workloads’ requirements. For example, you can create a workload domain optimized for database servers, another one for virtual desktop infrastructure (VDI), and others for software development, integration, and testing, or any other purpose—all using the same set of resources.

VMware Cloud Foundation runs on top of this hardware-infrastructure layer. VMware Cloud Foundation includes a suite of products and services, built on top of the vSphere cloud-computing platform, whose components work together to provide a managed, hyper-converged, and security-enabled cloud infrastructure for various workloads. The main components of the VMware Cloud Foundation suite are vSphere, NSX, vSAN, vRealize Automation, and SDDC Manager.

VMware Cloud Foundation delivers a new abstraction, workload domains, to create logical resource pools. A workload domain can be treated as a policy-based resource space with a specific type of availability and performance chosen by the operator during creation. From an infrastructure perspective, each workload domain is a cluster of VMware ESXi® servers with separate vSAN, NSX, and dedicated instances of VMware vCenter Server® for management purposes.

A key component of VMware Cloud Foundation is VMware SDDC Manager, which automates the entire cloud-foundation platform lifecycle from initial bring-up, configuration, and provisioning to upgrading and patching. SDDC Manager simplifies day-to-day management in operations. It is also used to scale the infrastructure, provision and scale logical workload domains, and handle infrastructure lifecycle-management tasks including software upgrades. The solution supports heterogeneous hardware configurations allowing users to better meet specific workloads’ requirements. SDDC Manager is also used to deploy additional components like vRealize Automation, which is used in this solution to automate infrastructure and services consumption.

vSphere is VMware’s cloud computing platform, consisting of a type-1 hypervisor (ESXi), a virtual management server (VMware vCenter Server) for each workload or management domain, and an administration interface (vSphere Web Client) to manage virtual machines (VMs) and other virtualized components in the SDDC. vSphere also includes Platform Services Controller*—a service that handles network-wide security functions such as single sign-on (SSO), licensing, and certificate management.

VMware NSX is VMware’s network virtualization software. NSX allows you to define network connectivity among virtualized elements running on vSphere and to harden
network security through micro-segmentation rules. Virtual network functions (VNFs) defined by NSX include switching, routing, firewalling, load balancing, and VPNs (IPsec and Secure Sockets Layer [SSL]).

VMware vSAN provides high-performance, hyper-converged storage for VMs in the vSphere environment. It serves as a key building block of the hyper-converged infrastructure that is integrated with vCenter Server. Through vSAN, all flash storage is pooled from multiple hosts, and virtual disks are provisioned as needed from the available space.

The third layer of the RA consists of the VMware vRealize Automation suite, supported by automation script and service blueprints. vRealize Automation provides a dedicated, security-enabled on-premises portal from which authorized administrators, developers, or business users can request new IT services. The services provided can include infrastructure components (like VMs, virtual networks, and network appliances), applications (such as database instances), VDIs, containers, and many others. All requests are realized through a common service catalog to provide a consistent user experience. Users can also manage resources delivered to them using actions like provisioning, backup, and recovery, or scaling on demand.

This RA provides several scripts and blueprints demonstrating how to build a DBaaS solution, along with an example WordPress* application consuming that database. Please note that the scripts and blueprints are not production-ready and serve only as an example of how to build such services.

**VMware vRealize Automation* Basic Concepts**

**Logical Architecture and Multi-Tenancy**

VMware vRealize Automation suite supports multiple tenants that can consume resources provided by various infrastructure providers. To facilitate a high level of flexibility and better control, vRealize Automation provides several layers of abstraction and grouping.

Figure 2 depicts general components and the dependencies between them. These components and dependencies are briefly described below; for more information, refer to the “Foundations and Concepts” document for vRealize Automation, available at https://docs.vmware.com/en/vRealize-Automation/7.3/vrealize-automation-73-foundations-and-concepts.pdf.

VMware vRealize Automation supports multiple tenants, as shown Figure 2, but only a single default tenant is configured after deployment in the VMware Cloud Foundation environment; additional tenants can be created after deployment.

A tenant is an organizational unit that can represent a business unit in an enterprise or a company that subscribes to cloud services from a service provider. The default tenant is always used to manage system-wide configuration (including the creation of additional tenants).

Infrastructure resources provided by hypervisors or cloud providers can be configured and shared among all tenants by exposing a set of API endpoints to the fabric.

For management purposes, all the resources can be grouped into fabric groups, with separate administrators managing the resources within each group. In the VMware Cloud Foundation environment, the fabric group represents endpoints exposed by one or more workload domains (which are separate vSphere clusters with dedicated storage provided by vSAN and networking provided by NSX). Fabric-group administrators can then allocate a portion of the available resources to the business groups by creating reservations.

Finally, end users belonging to specific business groups can consume resources available to each group through service catalogs or APIs provided by vRealize Automation.

**Default User Accounts**

During the vRealize Automation deployment process, the system creates three different user accounts:

- A default tenant administrator, with a user name and password provided by the user during deployment
- “administrator,” with a password automatically generated by the system (and obtainable by using the password-lookup tool in SDDC Manager)
- “configurationadmin,” with the same password as the administrator account

The administrator account can be used to create and manage tenants, configure branding, set up e-mail servers, and browse logs, whereas configurationadmin is the administrative account for the default tenant, similar to the account provided by the user during the deployment process. However, these two accounts have different roles assigned by default: the default tenant administrator serves only as the tenant administrator, whereas configurationadmin has approval administrator, catalog administrator, and tenant administrator rights, among others.

Additional local users and groups can be created later through the vRealize Automation dashboard. You can also integrate vRealize Automation with external directory services and authentication providers.

Service Catalog and Catalog Items

The service catalog is a self-service portal that provides a central place for end users from which they can request and manage various IT services and resources. Only services to which a user is entitled are presented. You can build several service catalogs and assign them to different business groups.

Figure 3 shows the basic catalog with single DBaaS instance—MariaDB SQL database with backup and restore functionalities—and a sample web application that also uses the MariaDB SQL database as one of its components.

The service catalog is built from catalog items, which provide typically complete specification of the resource to be provisioned and the process responsible for delivering it. The service catalog can also define optional arguments available to the user allowing for customization. Catalog items can be made only available to specific business groups, or they can be shared between all groups in the same tenant.

Services

Services are used to organize catalog items into related offerings to make it easier for users to find the services and resources they need.

Actions

Actions are additional operations that users can perform on already provisioned services and resources. Some standard actions, like “Power On” or “Restart,” are built-in and can be available to all users if enabled, but administrators can also build and provide more advanced custom actions for services, such as on-demand backup and recovery.

Entitlements

Entitlements determine which users and groups can request specific catalog items or perform specific actions. Entitlements are specific to a business group.

Blueprints

A blueprint is a complete specification of a resource that can include predefined and custom VMware vRealize Orchestrator workflows, which are used to create the requested service or resource. Blueprints can also be associated with actions to allow users to perform various post-provisioning operations. Blueprints can utilize other blueprints, so the architect can build more advanced services based on simpler components. When a blueprint is published, it becomes a catalog item and can be used in a service catalog.

Hardware Specification

The solution can scale from a single rack with just eight servers up to eight racks with 256 servers in total. This RA
uses 12 Intel® servers, two top-of-rack (ToR) Arista* switches, and a single Quanta Cloud Technology* management switch in a single rack. Additional racks can be added at the purchase time or later.

The initial software imaging requires an additional server or laptop running virtualization software and a privately managed switch. These components are not part of the solution and are not needed after completing the VMware Cloud Foundation imaging and bring-up process.

To demonstrate support for heterogeneous hardware configurations, this RA uses two types of servers differing in the CPU model, memory size, and number of drives. Customers can modify the base vSAN ReadyNode configuration to some extent, adding more memory or drives or replacing the CPU with a higher core-count or better clock-speed model. The general rules are described in https://blogs.vmware.com/virtualblocks/2017/03/14/can-cannot-change-vsan-readynode/.

Each rack requires a set of two ToR switches and a single out-of-band management switch. In multi-rack deployments, an additional set of spine switches would be required (usually installed in the second rack). A list of ToR switches that can be used with VMware Cloud Foundation can be found at vmware.com/resources/compatibility/info.php?deviceCategory=vcf&mode=vcf_introduction%20. For this RA, Intel used two Arista DCS-7280SRA-48C6 switches.

---

### Table 1. Configuration 1: Management cluster and first workload domain—eight nodes total

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>DESCRIPTION</th>
<th>PART NUMBER</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base SKU</td>
<td>Intel® Server System VRN2208WFAF82</td>
<td>VRN2208WFAF82</td>
<td>1</td>
</tr>
<tr>
<td>Mainboard</td>
<td>Intel® Server Board S2600WF0</td>
<td>S2600WF0</td>
<td>1 (included)</td>
</tr>
<tr>
<td>CPU</td>
<td>Intel® Xeon® Gold 6130 processor (16 cores, 2.10 GHz)</td>
<td>CD8067303409000</td>
<td>2</td>
</tr>
<tr>
<td>Memory</td>
<td>32 GB RDIMM DDR4-2666</td>
<td>J47951-001</td>
<td>12 (included)</td>
</tr>
<tr>
<td>Caching Tier</td>
<td>375 GB Intel® Optane™ SSD DC P4800X Series (Peripheral Component Interconnect Express* [PCIe*] x4 U.2)</td>
<td>SSDPE21K375GA01</td>
<td>2 (included)</td>
</tr>
<tr>
<td>Capacity Tier</td>
<td>2 TB Intel® SSD DC P4500 Series (2.5” NVM Express* [NVMe*] U.2)</td>
<td>SSDPE2X020T701</td>
<td>6 (included)</td>
</tr>
<tr>
<td>Boot device</td>
<td>480 GB Intel SSD DC S3520 Series (M.2, 80 mm)</td>
<td>SSDSCKB480G701</td>
<td>1 (included)</td>
</tr>
<tr>
<td>Network Interface Controller (NIC)</td>
<td>Intel® Ethernet Converged Network Adapter X710-DA2</td>
<td>X710DA2</td>
<td>1</td>
</tr>
</tbody>
</table>

### Table 2. Configuration 2: Second workload domain—four nodes total

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>DESCRIPTION</th>
<th>PART NUMBER</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base SKU</td>
<td>Intel® Server System VRN2208WFAF83</td>
<td>VRN2208WFAF83</td>
<td>1</td>
</tr>
<tr>
<td>Mainboard</td>
<td>Intel® Server Board S2600WF0</td>
<td>S2600WF0</td>
<td>1 (included)</td>
</tr>
<tr>
<td>CPU</td>
<td>Intel® Xeon® Platinum 8160 processor (24 cores, 2.10 GHz)</td>
<td>CD8067303405600</td>
<td>2</td>
</tr>
<tr>
<td>Memory</td>
<td>32 GB RDIMM DDR4-2666</td>
<td>J47951-001</td>
<td>24 (included)</td>
</tr>
<tr>
<td>Caching Tier</td>
<td>375 GB Intel® Optane™ SSD DC P4800X Series (Peripheral Component Interconnect Express* [PCIe*] x4 U.2)</td>
<td>SSDPE21K375GA01</td>
<td>4 (included)</td>
</tr>
<tr>
<td>Capacity Tier</td>
<td>2 TB Intel® SSD DC P4500 Series (2.5” NVM Express* [NVMe*] U.2)</td>
<td>SSDPE2X020T701</td>
<td>12 (included)</td>
</tr>
<tr>
<td>Boot device</td>
<td>480 GB Intel SSD DC S3520 Series (M.2, 80 mm)</td>
<td>SSDSCKB480G701</td>
<td>1 (included)</td>
</tr>
<tr>
<td>NIC</td>
<td>Intel® Ethernet Converged Network Adapter X710-DA2</td>
<td>X710DA2</td>
<td>1</td>
</tr>
</tbody>
</table>

Software Specification

The RA consists of two main software component suites, as shown in Figure 1: VMware Cloud Foundation and vRealize Automation suite, supplemented by automation scripts and blueprints developed by Intel engineers for demonstration purposes. vRealize Automation suite requires additional components and supporting infrastructures like Active Directory*, the Windows Server* operating system template for IaaS, and an instance of a Microsoft SQL Server* database. In addition, several networking services like enterprise Network Time Protocol (NTP) server and Domain Name System (DNS) server are needed for seamless integration with the external networks and global time synchronization. For a complete list of requirements and prerequisites, refer to the official VMware documentation at [1](https://docs.vmware.com/en/VMware-Cloud-Foundation/index.html) and [2](https://docs.vmware.com/en/vRealize-Automation/index.html).

Table 3. Network infrastructure

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>DESCRIPTION</th>
<th>PART NUMBER</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>ToR Switch</td>
<td>Arista 7280SRA-48G6*</td>
<td>DCS-7280SRA-48G6-R</td>
<td>2 per rack</td>
</tr>
<tr>
<td>Management Switch</td>
<td>QuantaMesh BMS T1048-LB9*</td>
<td>T1048 LB-9</td>
<td>1 per rack</td>
</tr>
</tbody>
</table>


Table 4. VMware Cloud Foundation* products and services

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>VERSION</th>
<th>BUILD</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware Cloud Foundation* bundle</td>
<td>2.3.2</td>
<td>8318074</td>
</tr>
<tr>
<td>VMware Imaging Appliance* (VIA)</td>
<td>2.3.2</td>
<td>8408982</td>
</tr>
<tr>
<td>VMware ESXi* hypervisor</td>
<td>6.5 EP6</td>
<td>7967591</td>
</tr>
<tr>
<td>VMware vSAN*</td>
<td>6.6.1</td>
<td>7547709</td>
</tr>
<tr>
<td>VMware NSX* for VMware vSphere*</td>
<td>6.3.5</td>
<td>7119875</td>
</tr>
<tr>
<td>VMware vCenter Server* on VMware vCenter Server Appliance*</td>
<td>6.5 U1g/EP6</td>
<td>8024368</td>
</tr>
<tr>
<td>VMware SDDC Manager*</td>
<td>2.3.2</td>
<td>8317783</td>
</tr>
<tr>
<td>VMware vCenter Lifecycle Manager*</td>
<td>2.3.2</td>
<td>8261646</td>
</tr>
</tbody>
</table>

Note that only the main components are listed in Table 4. For information on other components, see the VMware Cloud Foundation release notes at [5](https://docs.vmware.com/en/VMware-Cloud-Foundation/2.3.2/rn/VMware-Cloud-Foundation-232-Release-Notes.html).

Table 5. VMware vRealize Automation* suite

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>VERSION</th>
<th>BUILD</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware vRealize Automation*</td>
<td>7.3</td>
<td>5610496</td>
</tr>
</tbody>
</table>

Table 6. VMware vRealize Automation* support components

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Directory* on Windows Server 2016*</td>
<td></td>
</tr>
<tr>
<td>Microsoft SQL Server 2016*</td>
<td></td>
</tr>
<tr>
<td>Windows Server 2012 R2* (for the VMware vRealize Automation* IaaS template)</td>
<td></td>
</tr>
</tbody>
</table>

Note that only main components are listed in Table 6. For information on other components and prerequisites, refer to the official VMware vRealize Automation release notes and other supporting documentation at [6](https://docs.vmware.com/en/vRealize-Automation/7.3/rn/vrealize-automation-73-release-notes.html).
VMware Cloud Foundation Deployment

The VMware Cloud Foundation deployment process consists of several steps. First, you have to obtain all necessary hardware components, install them in a rack, and provide the necessary power, cooling, and uplink connection to the data center infrastructure.

Hardware Configuration and Wiring

The VMware Imaging Appliance* (VIA) user’s guide (part of the VMware Cloud Foundation documentation) provides detailed wiring diagrams and describes how to connect all the components together. It also describes how to set up the environment for the imaging process (including initial installation of the required software). The document is available at https://docs.vmware.com/en/VMware-Cloud-Foundation/index.html.

Unlike previous VMware Cloud Foundation versions, the current version (as of this writing) does not manage or validate server BIOS configuration. The VIA user’s guide document contains a set of recommended settings that must be configured manually or using an external management tool on all servers prior to starting the imaging process. All drives in the servers should be empty—although the imaging process tries to clear unnecessary data, in some cases that might not handle all configurations properly and the process might fail unexpectedly.


Imaging Infrastructure Preparation

The imaging process requires an additional server or laptop with virtualization software installed. This server is not a part of the solution, and it is not needed after the initial installation. This RA fills this role by using another Intel server running the VMware ESXi 6.5 hypervisor. VIA must be deployed on the imaging server along with the VMware Cloud Foundation installation bundle ISO image. The VIA user’s guide document describes the preparation process in detail. You also need a Windows® workstation (either physical or running on the imaging server as a VM) to run and monitor the installation process.

VMware Cloud Foundation Imaging

The imaging process is managed from the Windows workstation through a standard web browser pointing to the web interface exposed by the VIA appliance.

When you have the imaging infrastructure ready and all required hardware properly installed and configured, navigate in your web browser to http://192.168.100.2:8080/via/.

On the first screen, you need to mount and activate the ISO image with the VMware Cloud Foundation software bundle. You also have the option to provide additional drivers in the form of vSphere Installation Bundle (VIB) packages. This RA uses that option to automatically install the latest driver for the Intel Ethernet Converged Network Adapter X710 and provide the “i40en-1.5.8-1OEM.650.0.0.4598673.x86_64.vib” package obtained from the VMware support website at https://my.vmware.com/group/vmware/details?productId=614&downloadGroup=DT-ESXI65-INTEL-I40EN-158.

Note that some VMware Cloud Foundation versions do not contain a configuration profile for the models, so you won’t be able to select them for imaging. This is a known limitation of VMware Cloud Foundation and the VIA appliance, and it is described in the VMware Knowledge Base article 50354, along with workaround. For details, visit https://kb.vmware.com/s/article/50354.

The JavaScript Object Notation* (JSON®) manifest file mentioned in the VMware Knowledge Base article mentioned above should contain the following vendor and server model definition:

```json
{
    "vendorName": "Intel Corporation",
    "vendorCode": "intel",
    "scriptsFolder": "intel_scripts",
    "vibsFolder": "intel_vibs",
    "isoFolder": "",
    "serverModels": [
        {
            "modelCode": "S2600WF",
            "scriptsFolder": "",
            "vibsFolder": "",
            "isoFolder": "",
            "platform": "x86",
            "supportsAllFlash": "true"
        }
    ]
}
```

After preparing the image bundle, switch to the Imaging tab, which allows selecting a deployment type (full deployment or just a single server), the rack number (first or additional), the management switch vendor and model, the ToR and spine switch vendor and model, and finally the vendor and number of ESXi servers. When you are ready, click Start Imaging to begin the imaging process.

The VIA appliance is responsible for the entire flow of the imaging process, which can be monitored from the web browser running on the Windows workstation. The process starts with imaging the management switch, and it then continues with the ToR switches. When the ToR switches are ready, the system waits for the servers to boot from the network using PXE protocol; note that you might need to manually reset the servers and force them to boot from the network in the event that they are not configured in order to retry the boot sequence.
For this RA, the whole imaging process took about 2–3 hours; note that actual timing might vary depending on the number of hosts and the hardware configuration. In case of failure, you can restart the whole process for any specific device. Properly imaged devices will be marked with a green check mark.

When the imaging process completes, VIA performs additional verification and uploads Secure Shell (SSH) keys and certificates to the appropriate servers, in addition to uploading the inventory database built during the imaging process to the SDDC Manager installed on the primary ESXi server.

**VMware Cloud Foundation Bring-Up**

Once the imaging process is complete, you can move on to the second phase—bringing up VMware Cloud Foundation. The bring-up process is handled by SDDC Manager. A detailed explanation of this procedure is described in the "VMware Cloud Foundation Overview and Bring-Up Guide" document, located at https://docs.vmware.com/en/VMware-Cloud-Foundation/index.html.

The SDDC Manager interface is accessible through a standard web browser, similar to the VIA appliance. This process is responsible for the deployment of the first workload domain (management) and for configuring the networking.

Before you start the bring-up process, first collect all the information needed to configure network connectivity, including a list of virtual LANs (VLANs), network addresses, and uplinks. You should also plan for the DNS infrastructure and subdomain name reserved for VMware Cloud Foundation.

When you access the SDDC Manager interface for the first time, the system automatically sets the proper time and time zone on all components. It also performs several basic checks to verify that all components are accessible and in the expected state.

The Host Selection page displays hosts that are suitable for the management domain. You should select at least three identical hosts (although selecting four is recommended for higher availability). The primary ESXi host is selected by default.

The next step of the bring-up process is to supply a new account name and password for the administrator account, in addition to some general information like a physical rack name, a root DNS domain name, the VMware Cloud Foundation subdomain, the SSO domain, and finally the VMware Cloud Foundation license key.

Next is network configuration, for which you must provide VLAN and IP addresses for management, VMware vSphere vMotion*, vSAN, a Virtual Extensible LAN (VXLAN) overlay, an external data center network, and data center uplinks.

Finally, there is a short summary page that displays all the information you have provided. The bring-up process starts when you accept all settings by clicking **Finish**.

The bring-up process can take several hours and is orchestrated by SDDC Manager. You can monitor the current progress using a web browser.

After the bring-up process completes, you can check the IP address assignment of various VMware Cloud Foundation components by clicking **SEE IP ALLOCATION**. The Component IP Allocation page is important; be sure to record the names and IP addresses listed for **vCenter** and **SDDC_MANAGER**, as these are the two main interfaces that you will use to manage the whole infrastructure.

The VM running SDDC Manager is running an authoritative DNS server for that subdomain, and all VMware Cloud Foundation components have their hostnames registered in that DNS server. You should configure proper delegation on your corporate (external) DNS servers so that clients in external networks can properly access the VMware Cloud Foundation infrastructure using hostnames.

The passwords for all components except the SDDC Manager SSH root account are dynamically generated. To view the current password, you have to log on using SSH to the SDDC Manager VM as the user **root** with default password, **vmware1234**, and then use the command **/home/vrack/bin/lookup-passwords**.
Note: At the time of this writing, there was an issue with the current version of VMware Cloud Foundation incorrectly configuring Multi-Chassis Link Aggregation (MLAG) on port-channel interfaces on Arista ToR switches. After the bring-up process completes, the servers’ interfaces are reconfigured on the ToR switches, so they are members of port channels numbered from 121 onwards. Because the servers are connected to two switches, MLAG should be configured on all port-channel interfaces on both switches, but this configuration is missing. Consequently, the network connectivity might fail and some components will be impacted and work improperly.

To work around this issue, you have to log on to both ToR switches (using credentials obtained from the lookup-password command run on SDDC Manager), enter switch-configuration mode using the `config` command, and manually configure proper MLAG association on each port-channel interface using the same port-channel number as MLAG number. For example:

```
interface Port-Channel121
  mlag 121
```

The same issue applies not only to the servers assigned to the management workload domain but also to other workload domains. You can prepare all port-channel interfaces in advance for all servers, so if you have 12 servers in the setup, you should create port-channel interfaces numbered up to 132 and assign appropriate MLAG association to them so you will not have to do that again after workload-domain creation.

Remember to save the switch configuration using the `write memory` command when you are done with all configuration changes.

Creating Workload Domains

When the bring-up process completes, the infrastructure is ready for the creation of workload domains. The whole infrastructure is managed using the web interface of SDDC Manager.

The management workload domain is already created during the bring-up process, and it contains all the components needed to manage the whole infrastructure. It will also host additional components like vRealize Automation, which will be installed later on. You should not deploy any user applications on this management cluster. Instead, you should create one or more workload domains that comprise a separate vSphere cluster with vSAN and NSX preinstalled and configured, along with a dedicated instance of vCenter Server for management purposes. All instances of vCenter Server and VMware NSX Manager* are deployed on the management workload domain, whereas VMware NSX Controller* instances are deployed on the target workload domain.

In addition to the management domain, there are two other types of workload domains that can be created in SDDC Manager:

- Virtual infrastructure (VI): General-purpose domain
- VDI: Dedicated to virtual desktop environments; these workload domains rely on the VMware Horizon* suite, which is not part of this RA
The VI workload domain type represents a cluster of resources that can contain up to 64 servers with its own vCenter Server appliance, integrated vSAN, and NSX. A VI workload domain can span multiple racks, so if you later add more racks to this single-rack implementation, you can scale any existing VI workload domains to the additional racks as needed.

All the tasks related to the workload domains are performed using the SDDC Manager web interface. This includes the creation, expansion, and deletion of workload domains, along with physical-infrastructure monitoring and management.

Creating a new workload domain is simple. After you provide the name, you need only to specify the desired availability level (none, normal, or high), and then select the servers that will be added to the workload domain. The system automatically checks if the number of servers meets the required availability level and whether they have an identical configuration. It will also show total CPU, memory, and storage resources provided by a selected set of machines.

Different workload domains can use different types of servers with hardware configurations best suited to the workload requirements. For a general-purpose workload domain, you can use servers with Intel Xeon Gold processors and Intel SSDs, whereas for mission-critical applications you can use CPUs with higher core counts, like Intel Xeon Platinum processors, and more Intel SSDs with NVMe, providing more performance and capacity.

This RA used the “Base” and “Plus” configurations of Intel Select Solutions for VMware Cloud Foundation. These configurations were implemented with two workload domains named WD1 and WD2, each containing four servers. WD1 used servers with Intel Xeon Gold 6130 processors and 384 GB RAM, two Intel Optane SSD DC P4800X Series drives for cache, and six Intel SSD DC P4500 Series with NVMe drives for data. WD2 used servers with Intel Xeon Platinum 8160 processors and 768 GB RAM, four Intel Optane SSD DC P4800X Series drives for cache, and 12 Intel SSD DC P4500 Series with NVMe drives for data.

The next step is selecting which data center connection (uplink) should be used for your workload domains. You can share a single connection for all workload domains or have separate connections for each.

Finally, at the Review screen, you can verify all selected options and initiate the workload-creation process by clicking Finish.

The provisioning process starts in the background and is orchestrated by SDDC Manager. You can monitor the progress in the “Status” section. SDDC Manager installs and configures all the needed components (including vCenter Server, vSAN, and NSX). It then performs all the steps needed to integrate these components, including creating new VLANs on the switches and reconfiguring the ports leading to selected servers.

The time needed to create a workload domain depends on the server configuration and the requested infrastructure size. In the example environment, the process took about one hour and 40 minutes to provision the complete infrastructure of a workload domain with four servers. This means that, depending on the system and network configuration, the system can take as little as two hours to provision what used to require weeks—this reduction is enabled by the automation provided by SDDC Manager in VMware Cloud Foundation.
Moreover, because the whole process is automated, there is a much lower risk of the kind of misconfiguration that can often occur during manual installation. Such configuration errors could historically lead to serious issues or additional delays in infrastructure provisioning.

Later, you can use SDDC Manager to add new servers to an existing workload domain as needed. And when you no longer need the infrastructure used previously by the workload domain, you can remove it and return unused servers to the pool of available resources.

Because each new workload domain has its own vCenter Server and NSX instances, you should use the lookup-passwords tool on SDDC Manager to find the new set of passwords. The tool also displays the IP addresses and domain names associated with all components.

Figure 7. Selecting a host in the VMware SDDC Manager* web interface
vRealize Automation Deployment

The ability to automatically deploy vRealize Automation is a new feature available in VMware Cloud Foundation versions 2.3 onwards. The vRealize Automation product supports several scenario deployments for both small- and large-scale environments. Large-scale environments require installation and configuration of many different components and supporting infrastructure (like load balancers), which can make the process complicated and requires lots of knowledge and effort. Thanks to integration with VMware Cloud Foundation and SDDC Manager, the whole process is now largely automated and requires only preparation of several prerequisites and filling in the form with parameters. SDDC Manager handles the creation, installation, and configuration of all VMs with necessary vRealize Automation components and supporting infrastructure (such as virtual switches and VMware NSX Edge* gateways with load balancers).

vRealize Automation Infrastructure Overview

vRealize Automation is a powerful tool that can scale to more than hundreds of thousands of users and many concurrent workflows. To achieve this level of scalability and high availability, the architecture consists of several components responsible for different tasks, along with a set of load balancers. All the main components are deployed in multiple instances and can be scaled for even larger deployment if necessary.

vRealize Automation Appliance

The vRealize Automation appliance is a preconfigured Linux* VM delivered in Open Virtualization Format (OVF). It hosts vRealize Automation portal, where users can provision and manage IT services, and it provides SSO for user authorization and authentication. The appliance also offers a portal from which an administrator can manage vRealize Automation appliance settings. It also includes a preconfigured instance of vRealize Orchestrator, which vRealize Automation can use to extend its capabilities, and a downloadable management-agent installer. The management agent registers Windows IaaS servers with the vRealize Automation appliance, automates the installation and management of IaaS components, and collects support and telemetry information.

IaaS Web Server

The IaaS web server provides infrastructure administration and service authoring to the vRealize Automation product interface. The web server component communicates with the manager service, which provides updates from the Distributed Execution Manager (DEM), SQL Server database, and agents.

Manager Service

The manager service is a Windows service that coordinates communication between IaaS DEMs, the SQL Server database, agents, and Simple Mail Transfer Protocol (SMTP). In addition, the manager service communicates with the web server through the model manager, and it must be run under a domain account with local administrator privileges on all IaaS Windows servers.

Microsoft SQL Server* Database

IaaS uses a SQL Server database to maintain information about the machines it manages and its own elements and policies. Most users allow vRealize Automation to create the database during installation. Alternatively, you could create the database separately if site policies require it. The SQL Server database is not the part of vRealize Automation itself, but it is a required external component.

Distributed Execution Manager (DEM)

The IaaS DEM component runs the business logic of custom models, interacting with the IaaS SQL Server database and with external databases and systems. A common approach is to install DEMs on the IaaS Windows Server system that hosts the active manager service, but it is not required. Each DEM instance acts as a worker or orchestrator—the roles can be installed on the same or separate servers.

A DEM worker is responsible for running workflows. Multiple DEM workers increase capacity and can be installed on the same or separate servers.

A DEM orchestrator is responsible for monitoring DEM workers, scheduling workflows, and enabling that only one instance of each scheduled workflow is running at a given time. If a worker stops or loses its connection to the model manager, the DEM orchestrator moves the workflows to another DEM worker. DEM orchestrator also preprocesses workflows before they run, which includes checking preconditions for workflows and creating the workflow execution history. The active DEM orchestrator needs a strong network connection to the model-manager host.

Agents

vRealize Automation IaaS uses agents to integrate with external systems and to manage information among vRealize Automation components. Multiple agents increase capacity and can be installed on the same or separate servers.

Load Balancers

Multiple instances of each component are deployed for enhanced high availability and performance. Each set of services has a load balancer provided by NSX, which is responsible for exposing needed services and distributing traffic to back-end servers. The load balancers are also deployed in a highly available manner, eliminating a single point of failure.

vRealize Automation Deployment Prerequisites

Although the vRealize Automation deployment is mostly automated, there are still several prerequisites and requirements that must be prepared before installation. Detailed requirements are described in the “Deploy vRealize Automation in Cloud Foundation” section of the “Administering VMware Cloud Foundation” document, available at https://docs.vmware.com/en/VMware-Cloud-Foundation/.

This section provides only an overview and points out the most important aspects.
**Active Directory** Service and Account

First, all IaaS components require access to the Active Directory service. You need to obtain the IP addresses of Active Directory controllers and prepare an account that has permission to join Windows VMs to the domain. This account will also be used to install and run the management agents and IaaS components for the vRealize Automation infrastructure proxying agents. In simple proof-of-concept (PoC) installations, you can use the domain admin account; but for production environments, you need to follow policies established within the company and create a separate account with the right set of permissions.

As mentioned previously, SDDC Manager runs the DNS service hosting subdomain for the VMware Cloud Foundation environment (“vcf.example.com” in this example). You have to configure DNS zone forwarding from the VMware Cloud Foundation subdomain to the external IP address of SDDC Manager. To verify configuration, make sure you can resolve the hostnames assigned to the VMware Cloud Foundation (for example, sddc-manager-controller.vcf.example.com) from the Active Directory Domain Controller* and SQL Server. You should obtain the list of all names and the IP addresses of various VMware Cloud Foundation components during the bring-up process. Alternatively, you can use the command-line tool lookup-passwords on the SDDC Manager VM (located at /home/vrack/bin/lookup-passwords).

**Time Synchronization**

It is critical that you have synchronized time using NTP between the Active Directory domain and SDDC Manager (all other VMware Cloud Foundation components should have time synchronized to SDDC Manager). This is an important step because even a slight difference in time might cause certificates or authentication tokens to be treated as invalid, resulting in a broken installation or unexpected behavior.


**SQL Server**

You also need to prepare SQL Server as described in the section “Configure Microsoft SQL Server for vRealize Automation in Cloud Foundation” of the “Administering VMware Cloud Foundation” document. For a list of compatible versions, refer to the vRealize Automation 7.3 compatibility matrix document available at [https://docs.vmware.com/en/vRealize-Automation/7.0/vrealize-automation-6x7x-support-matrix.pdf](https://docs.vmware.com/en/vRealize-Automation/7.0/vrealize-automation-6x7x-support-matrix.pdf).

Note that SQL Server 2016 130 compatibility mode is not supported for vRealize Automation in VMware Cloud Foundation. If you separately create an empty SQL Server 2016 database for use with IaaS, use 100 or 120 compatibility mode.

The host running SQL Server must be a member of the Active Directory domain. vRealize Automation uses domain authentication when connecting to the database, so you should create a separate account for that purpose and configure it as a database owner. In simple PoC deployments, you can use the domain administrator account for that purpose.

For other required configuration options, refer to the “Administering VMware Cloud Foundation” section of the “Configure Microsoft SQL Server for vRealize Automation in Cloud Foundation” documentation, located at [https://docs.vmware.com/en/VMware-Cloud-Foundation/](https://docs.vmware.com/en/VMware-Cloud-Foundation/).

Additionally, make sure that the Microsoft Distributed Transaction Coordinator* (DTC*) is enabled in Component Services. The DTC should be installed by default, but in some cases, like when the system was cloned without using Sysprep*, the DTC might not work correctly. In that case, you might need to reinstall it. To do so, you can refer to the simple procedure described at [https://thecloudxpert.net/2017/07/pro-tip-configuring-msdtc-vmware-vrealize-automation/](https://thecloudxpert.net/2017/07/pro-tip-configuring-msdtc-vmware-vrealize-automation/).

The last step is to make sure that SQL Server can communicate using the TCP/IP protocol. Using the SQL Server Configuration Manager tool, navigate to SQL Server Network Configuration > Protocols for MSSQLSERVER > TCP/IP > Properties, and then set Network Interfaces to Yes.

**Multi-SAN Certificates for vRealize Automation Components**

The network communication between all components is secured using Transport Layer Security (TLS) and requires proper certificates installed on all nodes. Because all Windows IaaS VMs are installed from a single template, you have to prepare a single digital certificate containing all required hostnames (with and without domain names) in a Subject Alternative Name (SAN) field. The certificate should be signed by the trusted root certificate authority (CA). If you do not have any existing CA in your infrastructure, you can install the Active Directory Certificate Authority* service associated with your Active Directory and use it for signing your certificate request for IaaS components. For detailed instructions, refer to the respective Microsoft documentation at [https://docs.microsoft.com/en-us/windows-server/networking/core-network-guide/cncg/server-certs/install-the-certification-authority](https://docs.microsoft.com/en-us/windows-server/networking/core-network-guide/cncg/server-certs/install-the-certification-authority).

Please notice that Windows IaaS VMs will belong to the Active Directory domain, whereas vRealize Automation appliances running Linux and all load balancers will belong to the VMware Cloud Foundation subdomain hosted on SDDC Manager.
Assuming you are going to use typical naming convention for hostnames, the Active Directory domain is vraiaas.local, the VMware Cloud Foundation subdomain is vcf.example.com, and the SAN field in the digital certificate should contain the following values:

vra-01
vra-01.vcf.example.com
vra-02
vra-02.vcf.example.com
vra-web-01
vra-web-01.vraiaas.local
vra-web-02
vra-web-02.vraiaas.local
vra-mgr-01
vra-mgr-01.vraiaas.local
vra-mgr-02
vra-mgr-02.vraiaas.local
vra-dem-01
vra-dem-01.vraiaas.local
vra-dem-02
vra-dem-02.vraiaas.local
vra-agent-01
vra-agent-01.vraiaas.local
vra-agent-02
vra-agent-02.vraiaas.local
vra-lb
vra-lb.vcf.example.com
vra-web-lb
vra-web-lb.vcf.example.com
vra-mgr-lb
vra-mgr-lb.vcf.example.com

Depending on the public-key infrastructure (PKI) you are going to use, you should generate a private key and prepare a certificate signing request (CSR) using an appropriate tool; you should then send the CSR to the CA for signing. The private key should be password-protected. After approval, the CA should provide a signed certificate that will be used for IaaS components.

In the official documentation, you might find that VMware often refers to the vRealize Certificate Generation tool available at https://kb.vmware.com/s/article/2107816. There is also another tool that you might find useful, named Certificate Generation Utility for VMware Validated Design for Software-Defined Data Center, which is available at https://kb.vmware.com/s/article/2146215.

It doesn’t matter which tool is used, as long as the required fields (especially the SAN) are properly configured and the CA is able to properly sign the CSR created using the chosen tool. Note that VMware Cloud Foundation requires the certificate chain that contains both the concatenated CA certificate and your certificate. Both the certificate chain and certificate private key should be in a Privacy-Enhanced Mail (PEM) format, so they can be copied and pasted in the form during the deployment process.

**Network Configuration**

vRealize Automation deployed by SDDC Manager requires a separate VLAN along with a subnet, which will be used for various vRealize Automation components and load balancers. Most of the network configuration is done during the installation process—the user needs only to provide the VLAN ID, subnet address, netmask, and gateway.

It is expected that the VLAN and the subnet already exist on the corporate network where the uplink ports from VMware Cloud Foundation ToR switches are connected, and also that the gateway provides necessary connectivity (routing) to the VMware Cloud Foundation management and external network. SDDC Manager is only responsible for the VMware Cloud Foundation ToR switches configuration.

The test environment used VLAN ID 2103 and subnet 100.68.0.0/16. The VLAN was created on the uplink switch and added to the trunk ports leading toward VMware Cloud Foundation ToR switches. The test environment also connected the additional interface to an external router and configured IP address 100.68.0.1, which acts as the default gateway for vRealize Automation components. Additionally, the environment includes configured firewall policies to allow traffic between all required networks.

In production environments, you should provide similar functionality using existing data center infrastructure, or you should deploy and configure an additional virtual router using NSX on VMware Cloud Foundation.

**Note:** At the time of this writing, there was an issue with the current version of VMware Cloud Foundation incorrectly configuring VLAN assigned to the vRealize Automation subnet on ToR switches. Because various vRealize Automation components can be placed on different nodes within the management workload domain cluster, all nodes should have access to the vRealize Automation VLAN. Unfortunately, the system configures the VLAN only on the port-channel interfaces for a single server in the cluster.

To work around this issue, you have to log on to both ToR switches (using credentials obtained from the lookup-password command ran on SDDC Manager), enter switch-configuration mode using the `config` command, and manually configure an additional VLAN assigned to vRealize Automation on each port-channel interface for hosts in the management workload domain (usually port channel 121 to 124 if you have a four-node management cluster). For example, to add VLAN 2103, configure:

```
interface Port-Channel121-124
switchport trunk allowed vlan add 2103
```

Remember to save the switch configuration using the `write memory` command when you are done with all configuration changes.
### Windows 2012 R2 Template for IaaS

The final prerequisite is the Windows VM that will be used as a template for the IaaS components of vRealize Automation. The currently recommended version is Windows Server 2012 R2.

The official minimum requirements for the VM are as follows: two vCPUs, 8 GB memory, and 40 GB disk storage. For larger deployments, you might need to increase memory size or vCPU count. You should also consider increasing disk size to at least 64 GB to accommodate for additional space required by Microsoft patches and additional components.

It is extremely important that the system has all patches installed. During the development and validation of this RA, the system experienced some unexpected behavior and failing installations when some recent patches were missing.

The VM must have the latest version of VMware Tools installed and running. Due to the way VMware Tools reports IP addresses assigned to the interfaces, you have to manually disable the 6to4 tunnel interface created automatically in Windows using the following command:

```bash
netsh interface 6to4 set state disable
```

You also need to configure the firewall and open the required ports, as described in the installing and upgrading vRealize Automation > Installing vRealize Automation > Preparing for vRealize Automation Installation > IaaS Windows Servers sections of the “Installing or Upgrading vRealize Automation” document for vRealize Automation version 7.3, available at [https://docs.vmware.com/en/vRealize-Automation/](https://docs.vmware.com/en/vRealize-Automation/).

Several components require that Java Runtime Environment (JRE) be working, so you need to install the latest version of JRE 1.8 64-bit edition (do not use the 32-bit version), and you must then configure the system environmental variables JAVA_HOME and JRE_HOME to point to the main Java installation directory. To verify, check whether the `%JAVA_HOME%\bin\java.exe` binary is available.

Similar to with SQL Server, you should ensure the DTC is enabled in Component Services and working properly.

Also, you need to install Internet Information Services (IIS) 8.5* with a set of additional components described in the “IaaS Web Server” section of the “Installing and Upgrading vRealize Automation” document mentioned previously. Although the vRealize Automation installer tries to install additional components if missing, it is not guaranteed to work every time, especially when your infrastructure is behind a firewall or proxy without direct access to the Internet. After installation of IIS with required components, you should run Windows Update and make sure all recent patches are installed.

Finally, when you finish installing all prerequisites, you should delete all temporary and unnecessary files to minimize the template size and speed up deployment. When you are ready, shut down the VM properly and export it to the OVA template. This file will be used later on during vRealize Automation deployment.

### vRealize Automation Deployment in VMware SDDC Manager

After preparing all prerequisites, you can start the deployment process by navigating to the vRealize section under the Settings tab in the SDDC Manager web graphical user interface (GUI).

On the first page, you will be asked to specify all hostnames (without domain names) for vRealize Automation appliances, IaaS web VMs, IaaS manager VMs, IaaS DEM VMs, IaaS agent VMs, and load balancers. You need to provide the same names that were embedded in the SAN field in the digital certificate prepared before; otherwise, the installation will fail. Double-check that you have not misspelled any hostname, as the form is not validated and any mistake will result lately in a failed installation.
You also need to provide certificate details, a vRealize Automation license, and default tenant administrator details. In the **Certificate Chain** and **Certificate Private Key** fields, you need to copy and paste the content of the appropriate files prepared earlier. In the **Certificate Passphrase** field, you have to provide the password protecting the private key.

In the **Default Tenant Administrator** section, you have to provide details including the user name and password for a vRealize Automation account that will be created during the deployment process. Note that this is not an Active Directory, SSDC Manager, or vSphere account. vRealize Automation uses its own authentication database by default.
The last section contains load balancers details—you should provide hostnames for NSX load balancers that will be created. You have to use the names provided in the SAN field in the certificate you have previously prepared.

In the next section, you will need to upload a previously prepared Windows OVA template for IaaS components and provide a proper Windows license key and a local administrator password. You also have to fill in details regarding Active Directory domain controller and SQL Server. The **Domain Name** field should contain the Active Directory domain name where all Windows IaaS components will be placed (in this example, `vraiaas.local`), whereas the **Sub-domain Name** field should contain the VMware Cloud Foundation domain controller provisioned by SDDC Manager (in this example, `vcf.example.com`). The **Username** field doesn't have to be Domain Administrator, but it needs the right to join new computers (VMs with IaaS components) to the domain.
The Username and Password fields in the MS SQL Server section are optional—if you don’t provide them, the Active Directory credentials will be used, but in that case, you need to make the same user an owner of the database.

Figure 10. Configuring IaaS in VMware vRealize Automation*
The last section contains network-configuration details. Remember that the VLAN, subnet, and gateway need to be prepared on the upstream network infrastructure before the deployment process begins, otherwise the deployment will fail.

After verifying all the information given on the summary screen, you can initiate the deployment. The operation will take several hours and can be monitored in the Status section of SDDC Manager. During the process, SDDC Manager will deploy all necessary components and configure networking and NSX load balancers. All components will be located on the management workload domain. To verify deployment status, navigate to the Settings > vRealize > vRealize Automation page in SDDC Manager and confirm the vRealize Automation status is ACTIVE.

Figure 11. Network configuration details
Starting the VMware vRealize Orchestrator* Service

vRealize Automation uses the embedded vRealize Orchestrator, which is basically a workflow engine. Because all components are deployed in a highly available manner, several additional steps are required for configuring proper configuration synchronization.

After deploying vRealize Automation in VMware Cloud Foundation, you must manually start the vRealize Orchestrator configuration service to access the vRealize Orchestrator configuration interface. The process is briefly described in the “Start the vRealize Orchestrator Configuration Service” section of the “Administering VMware Cloud Foundation” document.

First, you have to log on to the vRealize Automation Linux appliance console as root and enter the service vco-configurator start command. Repeat the same process on the second appliance (in this example, the appliances are named “vra-01” and “vra-02”). You can obtain the password using the same password-lookup command run on SDDC Manager as before.

Then you have to access the vRealize Orchestrator configuration console pointing your browser to the following URL: (replace the hostname with the fully qualified domain name [FQDN] of the first vRealize Automation Linux appliance) https://vra-01.vcf.example.com:8283/vco-controlcenter.

Log on using the administrator account with the password obtained from the password-lookup command. The system will try to redirect you to the load balancer (vra-lb), but the current version of VMware Cloud Foundation does not configure the load balancer for the vRealize Orchestrator configuration console running on the 8283 port. Instead, you have to manually replace the URL to point back to the first vRealize Automation Linux appliance (in this example, “vra-01.vcf.example.com”) to continue.

After successfully logging on, you should see the following console:

![Figure 12. VMware vRealize Orchestrator* console](image-url)
Click **Validate Configuration** to verify that all components are running and the cluster is properly synchronized. You might see warnings like "A server restart is required because of a configuration change that is not yet applied" or "The Orchestrator cluster is in an inconsistent state."

In the first case, refer to the following Knowledge Base article: [https://kb.vmware.com/s/article/2151288](https://kb.vmware.com/s/article/2151288). After restarting the services, wait a couple of minutes before you run the “Validate Configuration” process again.

*Figure 13. Warning messages in VMware vRealize Orchestrator*
In case the cluster is in an inconsistent state, click the **Orchestrator Cluster Management** icon from the main screen to view the cluster state. You should see both vRealize Automation appliances listed with their respective states. First, you should deal with the local node (vra-01) and use the method described above to bring it to a Synchronized state. In some rare cases, you might need to reboot the whole VM and wait until it starts up and initializes all components.

When the local node is fixed, you should proceed with the second node (vra-02) using the same method. Eventually, both nodes should report the Synchronized state and the **Validate Configuration** option should report no issues.

**Connecting Workload Domains to vRealize Automation**

The vRealize Automation environment deployed so far contains all main components, but it does not have any vCenter endpoints created. To connect an existing workload domain and automatically create endpoints associated with vCenter Server and NSX Manager for that workload domain, you need to navigate to the **Settings > vRealize > vRealize Automation** page in SDDC Manager and click **Connect Workloads**. Alternatively, you can navigate to the particular workload domain properties window and then click **Connect vRealize** in the top right corner.

In the simple wizard, you just need to select workload domains that you want to connect and then start the process. This will trigger a task that can be monitored in the Status section. The whole process takes just a couple of minutes.

![Configuration validated in VMware vRealize Orchestrator](https://vra-01.vcloud.example.com:8080/vcloud-controlcenter/#/control-app/ha)
vRealize Automation Initial Configuration and Blueprints Deployment

After finishing vRealize Automation deployment and adding workload domains, there are few additional configuration steps required to finish the initial configuration process. In the current state of deployment, you should be able to access the vRealize Automation dashboard at https://vra-lb.vcf.example.com/vcac, where “vra-lb” is the hostname given to the load balancer and “vcf.example.com” is the subdomain name for VMware Cloud Foundation hosted on SDDC Manager. The system should also have three user accounts created as stated in the “VMware vRealize Automation Basic Concepts” section of this document, including the configurationadmin account. vSphere and NSX endpoints for each workload domain associated with vRealize Automation should also be already created. The default tenant is named “vsphere.local.”

Initial Configuration

The following steps are needed to finish the configuration process. This RA includes an automation script that handles most of the steps to make the deployment easier, but, due to limitations in available APIs or for the sake of simplicity, some of the steps must be done manually.

Fabric Group Creation

First, you need to create at least one fabric group and associate workload domains. Refer to the “VMware vRealize Automation Basic Concepts” section of this document for a general overview of various vRealize Automation components and their roles.

Using a web browser, log on to the vRealize Automation administration panel at https://vra-lb.vcf.example.com/vcac with the configurationadmin user account. Navigate to the Infrastructure tab, select Endpoints, and then click Fabric Groups.

Click the New button with the green plus symbol to create a new fabric group. Provide the fabric group name, and then select workload domains you want to associate and specify which user will be the fabric administrator.

In the example shown in Figure 15, the fabric group is named “cluster01” and only the vSphere cluster WD1-0-cluster is selected, which represents the workload domain WD1. The configurationadmin user is selected as the fabric group administrator.

Linux Template VM Preparation

vRealize Automation architects must build blueprints describing the services consumed by end users. You need to prepare a Linux CentOS template that can be utilized by these blueprints as a base for deploying additional software components and applications. The template VM requires a guest agent and bootstrap agent to be installed and the operating system to be customized using the special script provided by vRealize Automation.

During template creation, you have to choose virtual disk size, which needs to be large enough to hold not only the operating system itself, but also the applications and any necessary data, so you have to choose wisely. The simplest solution is to specify a large enough disk (such as 250 GB) that should cover most of the cases. You might also want to create several templates with different disk sizes or use a single template with secondary virtual disks added during deployment of specific blueprints if needed. In any case, use thin provisioning so the template does not use unnecessary disk space. This will also speed up cloning and blueprint deployment.

First, you need to log on to vCenter Server for the workload domain where you want to deploy your blueprints (in this example, this is WD1, and the vCenter Server should be accessible at https://rack-1-vc-1.vcf.example.com).

Figure 15. Configuring a new fabric ground in VMware vRealize Automation*
In vCenter Server, create a new Linux VM named centos—this name will be used in the blueprint as a base template. The guest operating system version should be CentOS 7 (64-bit). You should customize the virtual hardware (the number of vCPUs, memory size, and disk space) according to your needs. The template VM also needs a single virtual network adapter; for the preparation phase, it should be connected to the network having access to the Internet and the vRealize Automation appliance. At the end of the preparation, network configuration is cleared so the template can be used with other networks as needed.

When you have a VM created, install the latest version of the CentOS 7 (64-bit) operating system. You should choose minimal installation and then install only necessary additional tools and components (that way the template will be much smaller). In case additional software packages are needed later, you can add them during the blueprint deployment phase.

You should install the following tools and packages:

- Yellowdog Updater, Modified* (yum*)
- GNU Wget*
- Python 2*
- dmidecode
- sed, awk, perl, chkconfig, unzip
- OpenSSH Server* (optional, for easier remote troubleshooting)

You also need to install VMware Tools so that administrators can allow users to perform various actions (like reconfigure, snapshot, and reboot) against VMs.

Next, you have to install a vRealize Automation agent. The agent script is available on the vRealize Automation appliance at https://vra-lb.vcf.example.com/software/download/prepare_vra_template.sh.

You might need to adjust the hostname to the value you have configured for the vRealize Automation load balancer.

Use the wget command (with the optional --no-check-certificate parameter if you use self-signed certificates) to download the script to the VM template you are preparing, and then make the script executable using the chmod +x prepare_vra_template.sh command.

Execute the script as a root user by running the sudo ./prepare_vra_template.sh command, and then follow the prompts to complete the installation.

The last step is network-configuration cleanup. You need to remove udev persistence rules by running the following command as a root user: /sbin/udevd -h now and then use the vCenter Server GUI to convert it to the template by right-clicking on the VM and selecting Template > Convert to Template from the menu.

Finally, power down the template VM using the shutdown -h now command, and then use the vCenter Server GUI to select the template by right-clicking on the VM and selecting Template > Convert to Template from the menu.

Data Collection
To be able to use the newly prepared template in vRealize Automation, log on to the vRealize Automation dashboard as configurationadmin, and then navigate to Infrastructure > Compute Resources > Compute Resources. Hover the mouse over the workload domain containing the template you have prepared (in this example, WD1), and then, from the menu that appears, select Data Collection.

In the Data Collection window, make sure Data Collection is set to On, and then choose the required frequency in hours (leaving the field empty defaults to once per day). To force an immediate refresh, click Request now.

The last step needed is the customization specification, which will be used during deployment of the blueprint. It is used to set the proper VM hostname, time zone, and network configuration.

In the vSphere web GUI, navigate to Policies and Profiles > Customization Specification Manager. From the drop-down menu, select your workload domain, and then, above the list, click Create new specification to start the Customization Spec Wizard. On the first screen, select Linux as the Target VM Operating System, and then name the spec Linux. This spec name will be used in the blueprints. Click Next to move to the next screen.

On the second screen select Use the virtual machine name as the Computer Name, and then enter your VMware Cloud Foundation subdomain name in the Domain name field (this example, this is vcf.example.com). Click Next to move to the next screen.

In the time zone screen, you might want to adjust your local time zone, but it is recommended that you leave the hardware clock set to UTC (the default). Click Next for network configuration.

In the Configure Network pane, do not configure anything (leave the default settings). Click Next to move forward.

Provide your enterprise or public DNS servers and optional DNS search paths, which will be used by your applications and services deployed from this template. Click Next to advance to the last screen.

Finally, review all the options provided, and then click Finish to accept and create the customization specification.

Configure the vRealize Orchestrator Plug-in for VMware vCenter*

In order to allow vRealize Orchestrator to create XaaS blueprints and manage vSphere inventory objects, you have to configure the vRealize Orchestrator plug-in for vCenter. Because each workload domain is managed by its own vCenter Server, you need to define separate endpoints for each workload domain connected to vRealize Automation.

In the vRealize Automation dashboard, as configurationadmin, navigate to the Administration tab, then select vRO Configuration and proceed to the Endpoints option.

Reference Architecture | Build an Easily Consumable and Scalable Hybrid Cloud
Click Add New Endpoint, select vSphere vCenter Plug-in from the list, and then click Next.

On the next page, specify the endpoint name (for example, WD1-vRO), and then click Next.

In the Details tab, you need to specify the IP address or hostname of the vCenter Server instance for the specific workload domain. Leave the default options for the port number and location of the software development kit (SDK), and then click Next.

On the last screen, provide the username (usually administrator@vsphere.local) and password that vRealize Orchestrator will use to connect to the vCenter Server, and then click Finish.

Repeat the same procedure for other workload domains if needed.

Blueprint Import

This RA includes an automation script that handles configuration of other required components and imports prepared blueprints and actions. Because not all functionalities are exposed through the API, some of the tasks must be performed manually as described in previous sections.

Obtaining the Script and Blueprints

The script and blueprints are available on the Intel Service-Catalog-for-VCF GitHub* repository, which is available at https://github.com/intel/Service-Catalog-for-VCF.

To begin, you have to clone the content of the repository to your local Linux workstation using either the Git* clone command or another method. Then, you must change the working directory to the Service-Catalog-for-VCF repository, where you should find the bash script file named vra_import.sh. Finally, make it executable by using the chmod +x vra_import.sh command.

The script was tested on a CentOS Linux 7 x64 distribution and uses standard Linux tools, including grep, curl, and Python 2.7, which must be already installed. You can use a VM deployed from the CentOS 7 template prepared earlier or any other Linux machine with the required tools installed.

Script Configuration

Before executing this script, there are a few configuration settings that need to be edited inside this script. All the options are located at the beginning of the script and are preceded by an appropriate comment:

- #subtenant and user: Required settings for creation of business groups and new users; for example, the name of a business group, a business group admin e-mail, a username, a user e-mail, and the first and last name of a user
- #storage: Settings including the name of the data store—with storage and the storage quota in listed in GB for vRealize Automation reservation—storage_reserve_gb
- #network: Name of the external network port group in VMware Cloud Foundation
- #External network profile: A pool of external networks that will be reserved for vRealize Automation, a subnet mask, a default gateway, and DNS servers
- #NAT network profile: A pool of the network behind NSX Edge deployment
- #memory: The memory quota in MB for vRealize Automation reservation—memory_reserve_mb

Most of these parameters have sensible default values, but you have to adjust all IP addresses according to your infrastructure.

Script Execution

The script takes three arguments:

- <vRA FQDN>: The domain name of the vRealize Automation appliance load balancer (vra-lb.vcf.example.com)
- <vRAadmin>: The administrator of vRealize Automation (configurationadmin)
- <tenant>: The tenant created in vRealize Automation (vsphere.local)

Additionally, the script will ask for two passwords: first is the vRealize Automation administrator’s password, and second is the password for the newly created test user. The name of the user is configured in the script-configuration section (#subtenant and user, above).

Both passwords must adhere to the following rules:

- Be at least 8 characters long
- Contain at least one digit
- Contain at least one uppercase letter
- Contain at least one lowercase letter
- Contain at least one special character

Script Functionality Overview

The vra_import.sh script:

- Adds the software architect role to the vRealize Automation administrator specified as a script parameter
- Creates a business group subtenant
- Creates a user with the business group consumer role
- Configures a vRealize Orchestrator endpoint (used for data collection from vRealize Orchestrator) named “vRO Endpoint”
- Creates a reservation within the business group with:
  - A storage reservation quota
  - A network reservation with external and network address translation (NAT) network profiles and associated vSphere port groups
  - A memory reservation quota
- Imports a vRealize Orchestrator package with custom actions, including database backup and database restore for DBaaS catalog items
• Uploads vRealize Automation blueprints with all required software components, including MariaDB, Linux SFTP server, and HTTP server with the WordPress web application

• Creates service named "DBaaS" that will be used to organize imported catalog items

• Creates an entitlement named "DBaaS" that will determine that only users from a given business group who can request imported catalog items and vRealize Orchestrator actions

Final Configuration

After running the script, you need to manually configure machine prefixes, which will be used to generate VM names created by vRealize Automation servicing users’ requests.

Machine Prefixes

Machine prefixes contain a common VM name and a counter—you can specify the number of digits used and the starting value. To create machine prefixes, navigate to the Infrastructure tab in the vRealize Automation dashboard, select Administration, and then click Machine Prefixes. Click New, and then provide a VM prefix name, a number of digits (this is set to 3 by default), and the next number (this is set to 1 by default). Note that for Windows VMs the prefix should not be longer than 15 characters (including the digits). It should contain only case-insensitive ASCII letters A through Z, the digits 0 through 9, and the hyphen character (-), and it cannot start with a hyphen. Also, you must ensure that the generated names do not overlap with other existing VMs within the same scope.

When you have machine prefixes configured, you must set one as the default for your business group. Once again,

---

**Figure 16. Creating machine prefixes**
in the vRealize Automation dashboard, navigate to the Administration tab, select Users and Groups from the list, and then continue to Business Groups. Finally, edit the business group created by the script. Within the Edit Business Group window, switch to the Infrastructure tab, and then set the Default machine prefix to the one you have created.

Testing

Now your infrastructure should be ready to service users’ requests. You can log on to vRealize Automation as a test user created using the vra_import.sh script. After switching to the Catalog tab, you should see the screen shown in Figure 17.

Two items will be presented:

- App, which is a WordPress web application composed from the Linux web server, a MariaDB database, and a Linux SFTP server used for backups
- MariaDB, which is a SQL database (DBaaS) that also contains a Linux SFTP server used for database backup and restore functionalities, realized as custom actions

When you request the MariaDB, the system will ask for the database’s root user name and password, in addition to the TCP port number where the service should be available (by default, this is 3306). After submitting the form, the system will schedule deployment of the service and all needed components.

![Figure 17. VMware vRealize Automation* service catalog](image)

You can monitor the status of your deployments in the Requests tab.

![Figure 18. Monitoring deployment status in the Requests tab](image)
After successful deployment, the newly provisioned database can be administered from the Items tab, where the user can check the IP address assigned to a specific component and can also request to backup or restore the content of database using custom actions.

**Figure 19. Administering a newly provisioned MariaDB® database**

**Example Blueprints**

This RA includes two example blueprints as part of the automation script in order to demonstrate vRealize Automation capabilities.

The first blueprint is a sample MariaDB deployment with a backup host. Both hosts are connected to the “External” network port group. MariaDB and backup software components consist of two bash scripts: one downloads and installs necessary software, the other one configures software running on the VMs.

**Figure 20. MariaDB® sample-deployment blueprint**
The second blueprint contains an additional VM with an Apache* HTTP server and a WordPress web application. This blueprint also showcases the deployment of NSX Edge, which serves as a load balancer and NAT gateway for an isolated network. In this scenario, only the front-end host (Apache with WordPress) will be exposed to the “External” network, while the MariaDB database and backup host will be accessible only from the front end through the private “NAT” network.

Note that all the software components, including MariaDB, Apache HTTP server, and the WordPress web application, are installed each time from external repositories on the VMs created from the previously prepared template. The blueprints themselves do not contain any software. To speed up deployment and skip the downloading and installation of required software, administrators might want to prepare customized templates with preinstalled software and modify the blueprints accordingly.

Figure 21. Sample-deployment blueprint with an Apache* HTTP server and a WordPress* web application

Workload Migration

VMware Cloud Foundation provides several tools and technologies that facilitate live workload migrations between hosts and workload domains. Traditionally, the vMotion feature has allowed migrating running VMs only within the same cluster sharing common storage. It has been enhanced to allow for migration between clusters managed by a common vCenter Server management server as long as the clusters share common storage. With the introduction of extended vMotion (often called XvMotion), the shared storage is no longer required, and customers are able to migrate workloads between separate clusters backed by separate storage (such as vSAN). This feature is orchestrated by the vCenter Server, so either both clusters must be managed by a single vCenter Server instance or the vCenter Server instances managing both source and destination clusters must be working in enhanced linked mode, which is the default configuration for VMware Cloud Foundation.

There are some cases when you need to migrate your workload from a legacy environment or other clusters managed by a separate instance of vCenter Server, and you simply cannot link that with the VMware Cloud Foundation environment, so you can’t use the aforementioned mechanisms in an easy way. The latest version of the vSphere API provides support for migrations across different vCenter Server instances, but you have to either consume it directly using third-party or in-house developed tools, or you must use a PowerCLI* cmdlet named Move-VM.

To address such cases, VMware offers the Cross vCenter Workload Migration Utility tool, which is available at https://labs.vmware.com/flings/cross-vcenter-workload-migration-utility.

This is a simple GUI tool that allows for bulk VM migrations between completely separate vSphere environments managed by different vCenter Server instances using the cross-vCenter vMotion feature.

The key features of the Cross vCenter Workload Migration Utility tool, as stated by VMware, are as follows:

- Completely user-interface (UI)-driven workflow for VM migration
- Provides a REST API for managing migration tasks
- Works with vCenter instances that are not part of the same SSO domain
- Supports batch migration of multiple VMs in parallel
- Auto-populates inventory for ease of management
- Supports both live and cold migration of VMs
- Performs storage vMotion, without requiring shared storage
- Enables flexible network mappings between source/destination sites
The same tool can be used for migrations between private and public clouds based on VMware vSphere virtualization technology such as VMware Cloud* on Amazon Web Services* (AWS*).

Note that this tool is provided by VMware under a Technical Preview license, and it is being actively developed, adding new features and fixes to work around discovered issues. Also, several prerequisites must be met to allow migrations across vCenter instances. Those prerequisites are described in the VMware Knowledge Base article 2106952, available at https://kb.vmware.com/s/article/2106952.

To use the tool, you need Java virtual machine (JVM) 8* or later installed on your workstation and a web browser.

After downloading and unpacking the tool, just follow the procedure described in the enclosed readme document. If you use Java 8, simply run the java -jar xvm.jar command and then connect to http://localhost:8080 using your web browser.

Once connected, you must register all your vCenter Server instances that will participate in the migration and provide valid credentials (user names and passwords) with sufficient permissions. If you are using self-signed certificates, you should also select the Trust server check box to skip certificate verification.

Figure 22. The Cross vCenter Workload Migration Utility tool
Next, switch to the Migrate tab, select source and destination sites, and then select the operation type—either Relocate (for a migration) or Clone. You should also select the source and target site (represented by one of the previously registered vCenter Server instances), the source Datacenter, and a list of VMs that will be migrated (you can select multiple machines). In the Placement Target section, you can select either one of the available clusters or specific hosts managed by the target vCenter Server.

Finally, select the target data store where you want to place the migrated VMs and map port groups between the source and target vCenter Server instances so that migrated VMs will be connected to the appropriate networks.

When you are ready, click Submit to start migration. You will be presented with the Task information screen, where you can monitor the migration's progress and status.

**Figure 23.** The Task information screen allows you to monitor your migration's status

---

**A Consumable and Scalable Hybrid Cloud**

Hybrid cloud infrastructures enable you to do more with less, combining the benefits of on-premises and public cloud infrastructures. By making use of Intel and VMware technology, you can create a security-enabled, scalable, and high-performance infrastructure that can help your organization make use of its ever-growing datasets for large-scale analytics, improved decision-making, and higher profitability.