CHAPTER 2

System Architecture

This chapter includes the following major topics:

DRaaS 1.0 System Architecture

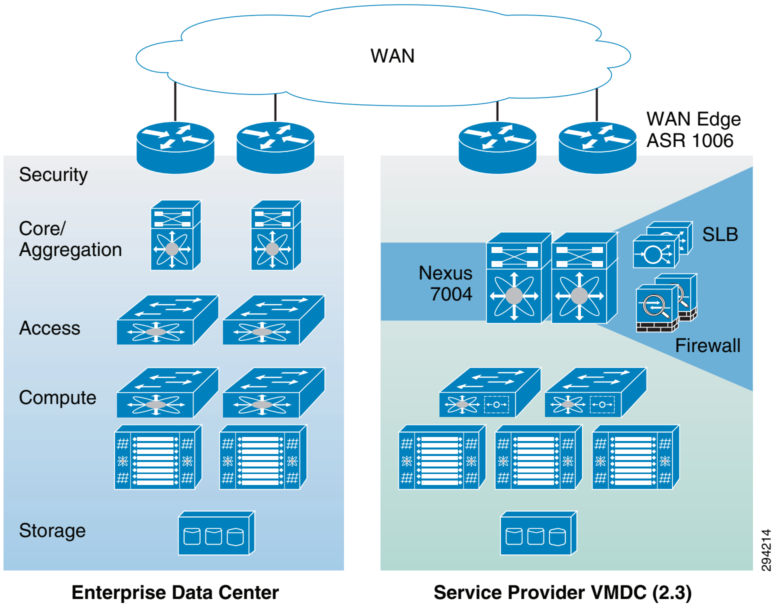
This section includes the following topics:

System High Level Architecture

This section describes the high level architecture of the DRaaS System. The system provides disaster recovery for customer physical/virtual servers by deploying recovery VMs in the VMDC 2.3-based container on the provider side.

Figure 2-1 illustrates the high level architecture of DRaaS System.

Figure 2-1 DRaaS High Level Architecture



The physical system architecture consists of the following building blocks:

Provider Cloud

The provider cloud within the DRaaS System will be based on VMDC 2.3. The VMDC 2.3 design is based on the earlier VMDC 2.2 design, with changes to optimize the design for lower cost, fewer layers, and increased tenancy scale. The Cisco VMDC System provides vPC-based L3 hierarchical virtual routing and forwarding (VRF)-Lite DC design, multi-tenancy, secure separation, differentiated service tiers, and high availability in a data center environment. It also provides secure separation between replicated workloads and provides shared network services for customers in DRaaS.

The VMDC 2.3 architecture works with Vblock, FlexPod, or any other integration stack. Integrated stacks can be added as required to scale the SP cloud environment.

Based on the customer's production environment and needs, a specific tenancy model can be selected to provide similar services in the cloud-matching production environment. VMDC architecture and deployment models will be covered in detail in this chapter.

Enterprise Data Center

The DR solutions should address enterprise customer requirements for various vertical industries and geographies. The enterprise data center design is therefore expected to vary from customer to customer. The intent of the DRaaS System is to keep the enterprise DC architecture generic so as to provide the greatest coverage. While the DC architecture is almost irrelevant and the solution supports heterogeneous replication across any-to-any infrastructure, a typical three tier (core/aggregation and access) DC architecture is suggested in the system.

WAN Connectivity

The WAN connectivity design principles provided by VMDC are maintained and supported without requiring any additional components and technologies. The replicated data between the enterprise and SP data center can be encrypted with the help of Cisco technologies like IPsec VPN based on Cisco ASA firewalls. Optionally, for low cost implementation to support a small number of servers, inflightreplicated data encryption can be provided by InMage partner software.

To support partial failover of customer's environment, technologies like Overlay Transport Virtualization (OTV) can be used for L2 extension between the customer's data center and the cloud. L2 connectivity allows customers to use the same IP from enterprise network in the cloud without the need to change for accessing workloads in the cloud after recovery.

Partner Solution for Providing Disaster Recovery

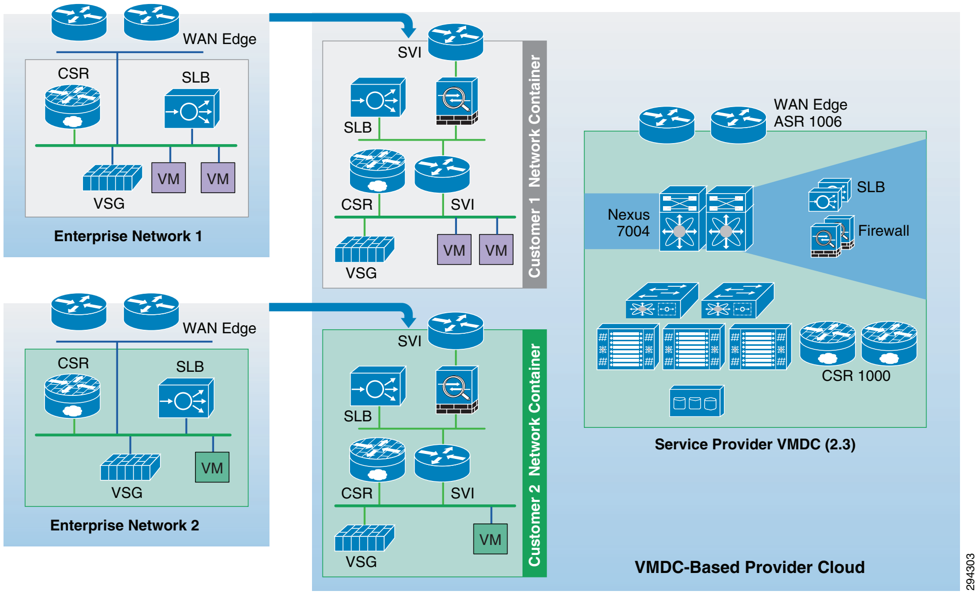
Data replication and recovery of the production servers will be provided by InMage ScoutCloud technology. InMage ScoutCloud is a software-based replication and recovery technology, which can protect both physical and virtual servers into the cloud. InMage ScoutCloud is being integrated into the DRaaS System, providing the following functionality:

- Heterogeneous data replication
- Continuous data protection
- Application consistency
- Recovery automation
- DR Drill

System Logical Topology

Figure 2-2 covers the logical topology of the DRaaS System.

Figure 2-2 DRaaS Logical Topology



As shown in Figure 2-2, each customer will have a dedicated network container created on the SP VMDC cloud. The network containers will be created based on the necessary security and network services required by the enterprise customers. Any network topology on the customer's data center can be matched on the VMDC cloud using network containers. Predefined containers provide examples for different types of deployments. Automated provisioning and management logic for each customer type is pre-defined in the management and orchestration software. Customers can choose from existing models or define their own customized models. The production workloads from each enterprise data center will be replicated to the corresponding network container on the VMDC cloud and will be available for recovery purposes.

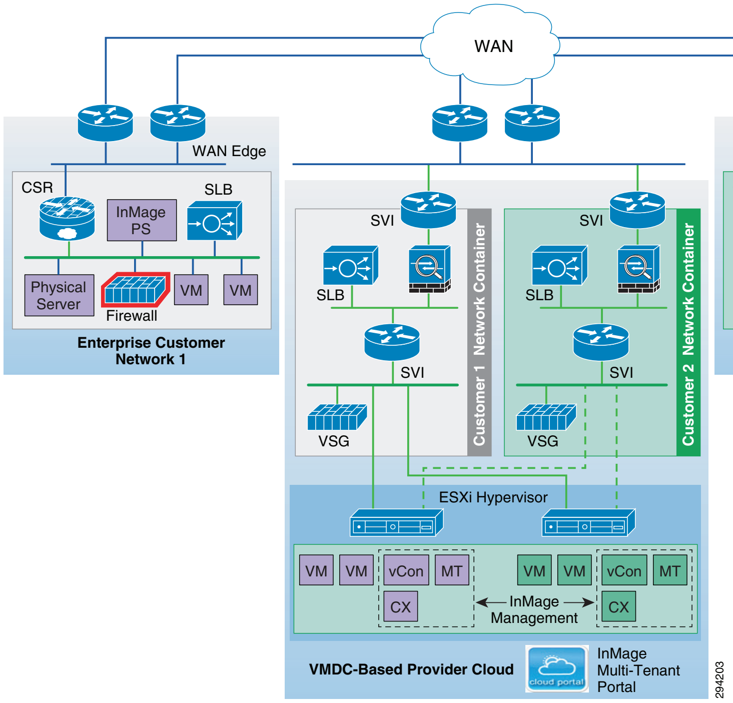
End-to-End Architecture

The DRaaS System addresses the following design principles and architectural goals:

- Secure Multi-Tenancy
- Secure, modular, and highly available cloud
- Continuous Data Protection (CDP)
- Physical-to-Virtual (P2V) and Virtual-to-Virtual (V2V) Disaster Recovery
- Near zero RPO and RTO-capable DRaaS
- Automated run book automation
- Self-Service Multi-Tenant Portal

By utilizing the architecture above, DRaaS in a multi-tenant environment can be supported as shown in Figure 2-3.

Figure 2-3 End-to-End Architecture



In a multi-tenant environment, each customer is mapped as a separate VMDC tenant where the necessary network security is provided and traffic segregation is maintained. Figure 2-3 depicts the end-to-end architecture of the DRaaS System based on VMDC.

With the deployment of lightweight components as shown in Figure 2-3 and utilizing the network security provided by VMDC architecture, customers can replicate their data into a secure cloud environment for recovery.

Data changes are collected from the production servers as they occur, directly in memory before they are written to disk, and sent to a software appliance within an enterprise data center. Because of this approach, absolutely no additional I/O load is induced on production servers due to replication. The appliance is responsible for further offloading compute-intensive tasks from production systems, such as compression, encryption, WAN acceleration, and consolidated bandwidth management.

The system provides CDP for the customer's production servers. The customers will be able to recover their environments to any point in time before the disaster occurred. The servers are not only protected from the physical disasters, but also from logical disasters due to CDP.

Application consistency is enforced at regular intervals through VSS integration on Windows and native application-specific mechanisms on Linux and Solaris systems. Application consistency is also enforced at the guest level in virtual environments such as VMware ESX, Xen Server, and Hyper-V. These application-consistent points are tagged by a bookmark and archived as part of the CDP data. They can be leveraged to perform application consistent recoveries within stringent recovery time objectives.

The following use cases are covered as part of the DRaaS System and will be discussed in more detail in the following sections:

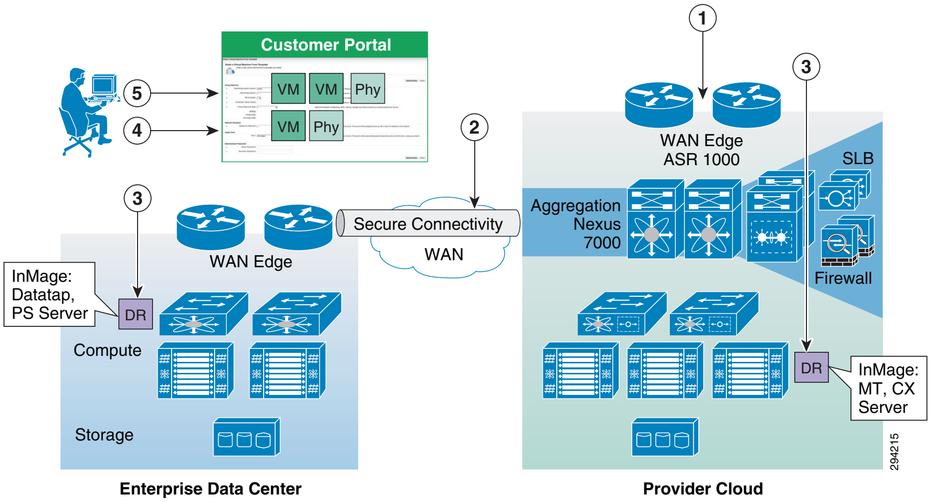
- Protection Workflows, page 4-2
- Recovery Workflows, page 4-37
- Failback Protection Workflows, page 4-46
- Resume Protection Workflows, page 4-78
- DR Drill Workflows, page 4-84

DRaaS Operational Workflows

Following are the workflows for protecting and recovering the customer's production workloads into the cloud. The workflows describe the process of creating the network containers for customers within the SP cloud, replication of workloads into the network containers, and recovery of workloads in the event of a disaster.

The workflow in Figure 2-4 is used for protection and failover scenarios.

Figure 2-4 New Customer Protection Workflow



Step 1 Based on the customer requirements, deploy a VMDC Network Container using BMC.

Step 2 Secure IPsec connectivity is manually set up between the Enterprise and the VMDC-based cloud provider setup.

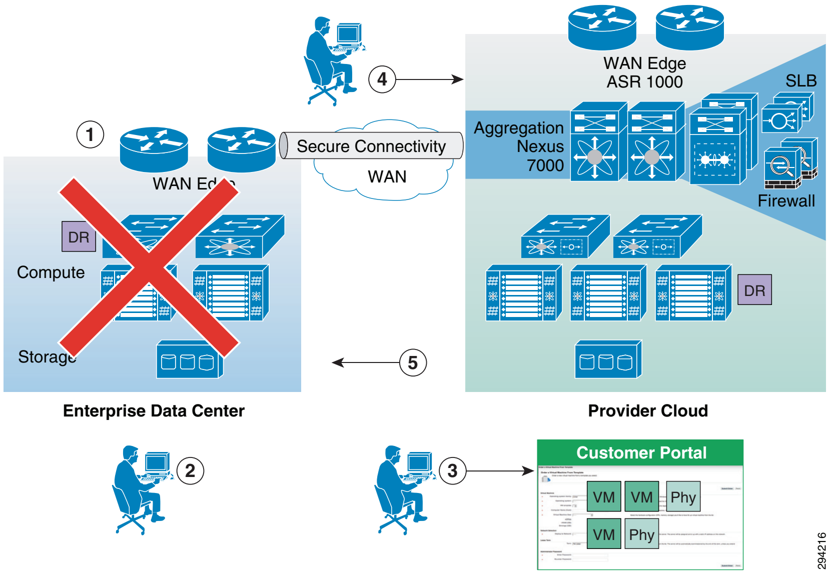
Step 3 At both enterprise and SP data centers, deploy and configure the necessary DR components.

Step 4 Use the InMage management wizard to select the machines to be protected and set up the recovery plans.

Step 5 Allow customers to monitor the status of DR and RPO/RTO utilizing the Partner Product portals.

The workflow in case of a failure scenario is shown in Figure 2-5.

Figure 2-5 Failure Scenario



Step 1 When the customer DC goes down, customer declares a disaster and communicates to SP what VMs to restore and what checkpoints to use. SP can use the recovery plan (which could be preconfigured), which details the list of protected VMs, the startup order, and any custom steps.

Step 2 SP logs into the DR product portal and brings up the required VMs in its environment. Customers with self-service capabilities will be able to recover VMs in the cloud themselves using the self-service portal.

Step 3 Customer works with its DNS provider to direct the client traffic to the SP DC. If the customer is utilizing a Global Site Selector (GSS)-based DNS solution or has a L2 extension, this step will be automatic or not required.

Step 4 When the Enterprise DC is back up, customer works with the SP during a maintenance window to bring up the VMs in customer DC, failback the VMs from SP to enterprise, and update the DNS so that the client traffic is re-routed to the customer DC.

Network Deployment Considerations to Support Recovery Environment

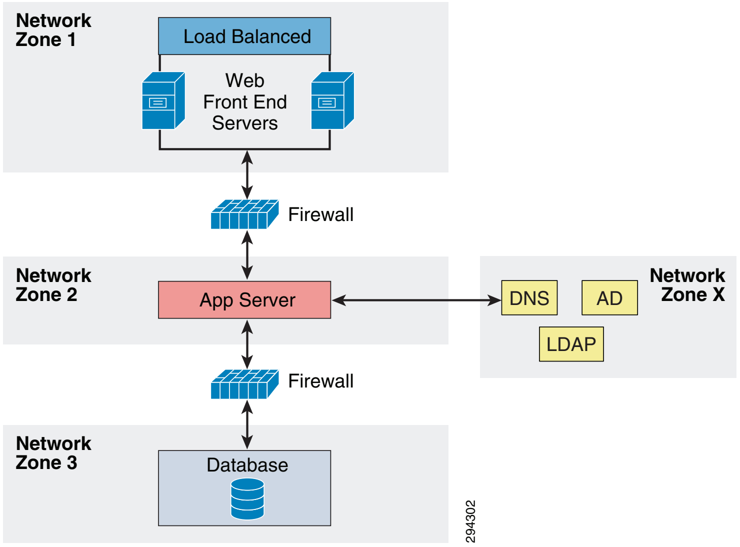
Table 2-1 shows the considerations in matching the networks between the enterprise's and SP's VPC. Logically, the enterprise network will consist of VLANs and network services, including firewall rules and load balancing. Based on the requirements of enterprise, which depend on the type of applications that are protected, network containers can be created on the VMDC to meet those requirements.

Table 2-1 Network Containers Available on VMDC

Container	VLANs	Network Services
Gold	3	Tenant firewall, intra-tenant firewall, and load balancer
Silver	3	Load balancer
Bronze	1	Intra-tenant firewall, load balancer
Copper	1	Intra-tenant firewall

The typical deployment of a multi-tiered application running in the enterprise is shown in Figure 2-6.

Figure 2-6 Application Deployment Example



The following is the onboarding procedure of a customer running the application shown above:

- The enterprise IT admin needs to coordinate with the SP to have the network container created on the VMDC, based on the requirements and dependencies of the application being protected. The options of creating the network container and maintaining consistency on the SP side are as follows:
  - The container is pre-configured by SP with the necessary VLANs and network services. The firewall rules and the load balancing rules pre-configured based on the pre-determined IPs of recovery servers on VMDC.
  - The container is preconfigured and the firewall and load balancing rules are configured dynamically by the SP using BMC orchestration or manually through CLI during the failover process of the servers. Any changes done with the network services after replication has been set up on the enterprise data center have to be communicated to the SP. This ensures network consistency during recovery of the servers. Optionally, the SP can enable the Enterprise customer to manage the firewall and load balancing services on the VMDC cloud. This can be done by providing access to the BMC orchestrator or to the specific devices directly for modifications.
- For the application to run properly on the VMDC cloud after recovery, all the components of the application from different tiers needs to communicate with each other. All the servers needs to be brought up in an order based on the application dependencies.

- The other important dependency is the IP addressing. Two types of configurations are done on the servers within an application for intra-component communication:
  - Configuration based on IP address
  - Configuration based on DNS names

Legacy application servers configured based on IP address can run seamlessly as long as they have the same IPs on the VMDC cloud. This may or may not be the case for all the customers. Customers who have different network subnets available on the VMDC need to reconfigure the servers to have the new IPs part of the application configuration. The task mentioned about can be performed by a SP administrator in a managed recovery use case or by the customer after the servers are available on the VMDC cloud.

The re-IPing of the servers can be eliminated if the servers with in the applications are using DNS names for communicating, in which case the DNS entries can be updated to reflect the new IPs. An SP can also perform the modification of DNS entries if the customer is using SP DNS servers. Optionally, DNS entries can be modified automatically using scripts.

In cases of the customer protecting the enterprise DNS servers, the DNS servers can be brought online during the recovery of application and based on the new IP addresses the configuration can be updated by the customer manually or can be automated.

Use of Cisco GSS

To accelerate the disaster recovery service and the dynamic distribution of the workload between the primary and secondary data centers, Cisco provides different network services to optimize the access and the distribution of the user traffic to the remote sites using a Global Site Load Balancing (GSLB) solution. This global GSLB solution for traditional L3 interconnection between sites relies on three major technologies:

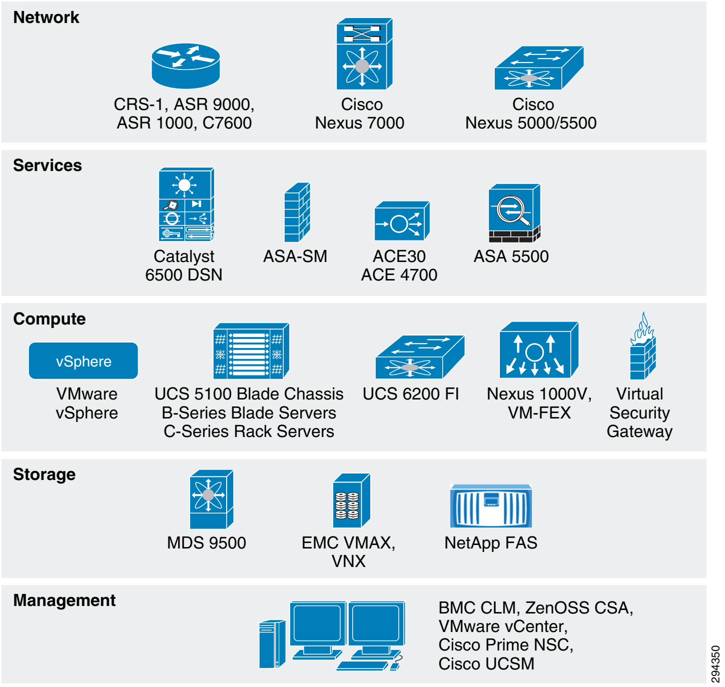
- Intelligent Domain Name System (DNS): A DNS known as the Global Site Selector (GSS) redirects the requests from end-users to the physical location where the application is active.
- HTTP Traffic Redirection between Sites: In case of resource unavailability, the local Server Load Balancing (SLB) device will return an HTTP redirection message type (HTTP status code 3xx) to the end-user so that the web browser of the client can be automatically and transparently redirected to the elected backup data center where resources and information are available.
- Route Health Injection (RHI): RHI provides a real-time, very granular distribution of user traffic across multiple sites based on application availability. This method is initiated by an SLB device that will inform the upward router about the presence or absence of selected applications based on extremely accurate information. This information is usually related to the status of the services that it supports. Therefore, the redirection of the user request to a remote site occurs in real time.

VMDC Cloud Infrastructure

The VMDC System is the Cisco reference architecture for IaaS cloud deployments. This Cisco cloud architecture is designed around a set of modular DC components consisting of building blocks of resources called PoDs, or Points of Delivery. These PoDs comprise the Cisco UCS, SAN and NAS storage arrays, access (switching) layers, and aggregation (switching and routing) layers connecting into the DSN-based services layer or connecting directly to service appliances; and multiple 10 GE fabric using highly scalable Cisco network switches and routers. The VMDC system is built around the UCS, Nexus 1000V, Nexus 5000 and Nexus 7000 switches, Multilayer Director Switch (MDS), ASR 1000, ASR 9000, ASA 5585-X or Adaptive Security Appliance Services Module (ASASM), Catalyst 6500 DSN, ACE, Nexus 1000V VSG, VMware vSphere, EMC VMAX, VNX and NetApp FAS storage arrays.

Cloud service orchestration is provided by the BMC Cloud Lifecycle Management (CLM) suite and cloud service assurance is provided by the ZenOSS Cloud Service Assurance (CSA) suite. Figure 2-7 provides a synopsis of the functional infrastructure components comprising the VMDC system.

Figure 2-7 VMDC Infrastructure Components



This section includes the following topics:

- VMDC 2.3 Architecture, page 2-9
- VMDC 2.3 Network Containers, page 2-13
- Modifications in VMDC Network Containers for DRaaS, page 2-17

VMDC 2.3 Architecture

The VMDC System utilizes a hierarchical network design for high availability and scalability. The hierarchical or layered DC design uses redundant switches at each layer of the network topology for device-level failover that creates highly available transport between end nodes using the network. DC networks often require additional services beyond basic packet forwarding, such as SLB, firewall, and intrusion prevention. These services might be introduced as modules populating a slot of one of the switching nodes in the network or as stand-alone appliance devices. Each service approach also supports the deployment of redundant hardware to preserve the HA standards set by the network topology. This layered approach is the basic foundation of the VMDC design to provide scalability, performance, flexibility, resiliency, and service assurance. VLANs and VRF instances are used to provide tenant isolation within the DC architecture, and routing protocols within the VRF instances are utilized to interconnect the different networking and service devices. This multilayered VMDC architecture is comprised of core, aggregation, services, and access layers. This architecture allows for DC modules to be added as demand and load increases. It also provides the flexibility to create different logical topologies utilizing device virtualization, the insertion of service devices, and traditional L3 and L2 network configurations.

The VMDC 2.3 System is the latest released version of the VMDC architecture, with VMDC 2.2 being the previous release. Architecturally, VMDC 2.3 is based on VMDC 2.2 (and 2.0), but with several optimizations to reduce cost and footprint and increase tenancy scale. The key differences between VMDC 2.3 and 2.2 include:

- VMDC 2.3 includes an ASR 1000 as the DC Edge (PE) router, while VMDC 2.2 uses the ASR 9000.
- VMDC 2.3 includes a collapsed core/aggregation layer, while VMDC 2.2 includes a separate Nexus 7000 core layer and Nexus 7000 aggregation layers.
- VMDC 2.3 includes an ASA 5585-X for the perimeter firewall, while VMDC 2.2 uses the ASA5585-X or ASASM module on Catalyst 6500 DSN.
- VMDC 2.3 includes an ACE 4710 for Server Load Balancing, while VMDC 2.2 uses the ACE-30 module on the Catalyst 6500 DSN.
- VMDC 2.2 optimizes the Enhanced Gold, Silver, and Bronze network containers to consume fewer resources on the platforms, compared to VMDC 2.3.
- VMDC 2.3 utilizes the ACE 4710 in One-Arm mode, while VMDC 2.2 uses the ACE30 in Two-Arm mode.

Note For detailed information on VMDC 2.3 System architecture, refer to the following documents:

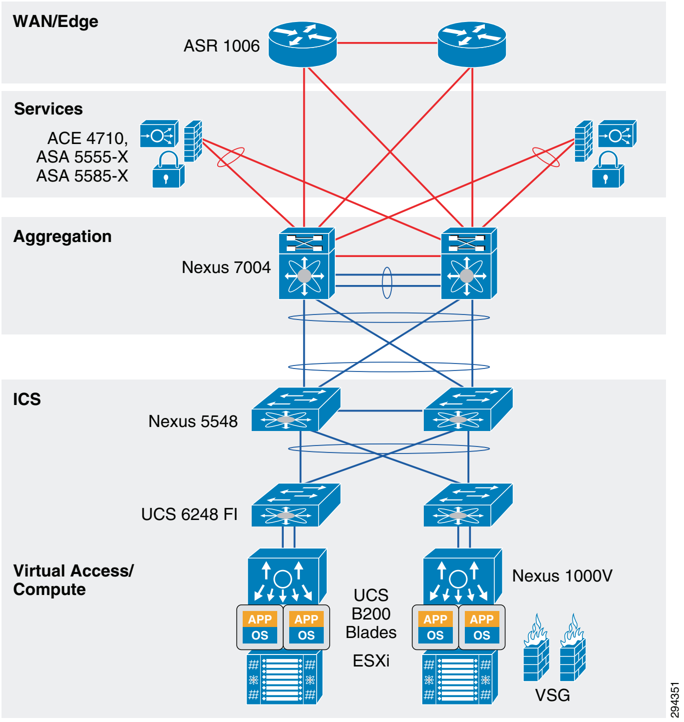
- VMDC 2.3 Design Guide
- VMDC 2.3 Implementation Guide

For information on the previous VMDC 2.2 System architecture, refer to the following documents:

- VMDC 2.2 Design Guide
- VMDC 2.2 Implementation Guide

Figure 2-8 provides a representation of the VMDC 2.3 physical architecture.

Figure 2-8 VMDC 2.3 System Architecture



VMDC 2.3 Modular Components

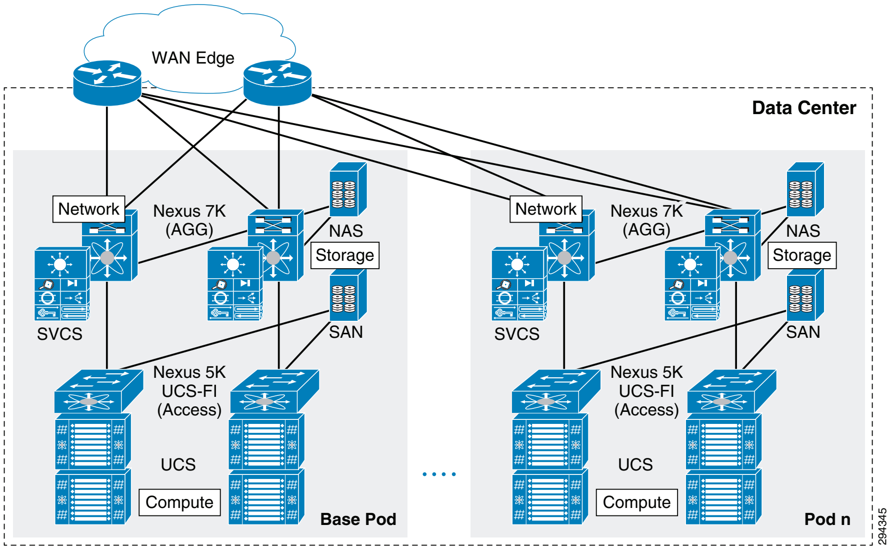
The VMDC System architecture provides a scalable solution that can address the needs of Enterprise and SP cloud data centers. This architecture enables customers to select the design that best suits their immediate needs while providing a solution that can scale to meet future needs without retooling or redesigning the DC. This scalability is achieved using a hierarchical design with two different modular building blocks, Point of Delivery (PoD), and ICS.

Point of Delivery (PoD)

The modular DC design starts with a basic infrastructure module called a PoD. A PoD is a repeatable, physical construct with predictable infrastructure characteristics and deterministic functions. A PoD identifies a modular unit of DC components and enables customers to add network, compute, and storage resources incrementally. This modular architecture provides a predictable set of resource characteristics (network, compute, and storage resource pools, power and space consumption) per unit that are added repeatedly as needed.

In this design, the aggregation layer switch pair, services layer nodes, and one or more Integrated Compute and Storage (ICSs) are contained within a PoD. The PoD connects to the WAN/PE layer device in the DC, in the VMDC 2.3 architecture, and connects to the core layer in previous VMDC 2.2 and 2.0 architectures. To scale a PoD, providers can add additional ICSs and can continue to scale in this manner until the PoD resources are exceeded. To scale the DC, additional PoDs can be deployed and connected to the core layer devices. Figure 2-9 illustrates how PoDs can be used to scale compute, network, and storage in predictable increments within the DC.

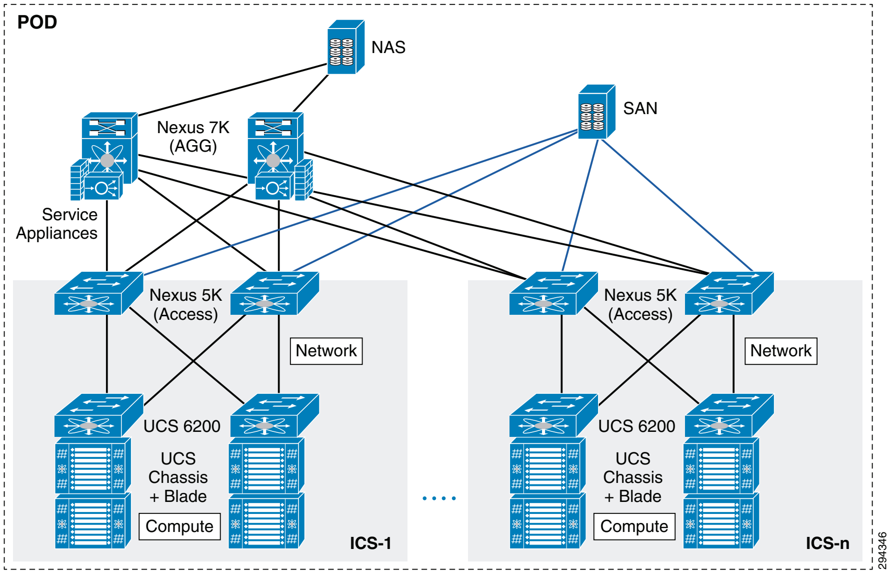
Figure 2-9 VMDC 2.3 PoDs for Scaling the Data Center



ICS

The second modular building block utilized is a generic ICS based on existing models, such as the VCE Vblock or Cisco/NetApp FlexPod infrastructure packages. The VMDC architecture is not limited to a specific ICS definition, but can be extended to include other compute and storage stacks. An ICS can include network, compute, and storage resources in a repeatable unit. In this guide, the access layer switch pair, storage, and compute resources are contained within an ICS. To scale a PoD, customers can add additional integrated compute stacks and can continue to scale in this manner until the PoD resources are exceeded. Figure 2-10 illustrates how integrated compute stacks can be used to scale the PoD.

Figure 2-10 VMDC 2.3 ICS for Scaling the Data Center



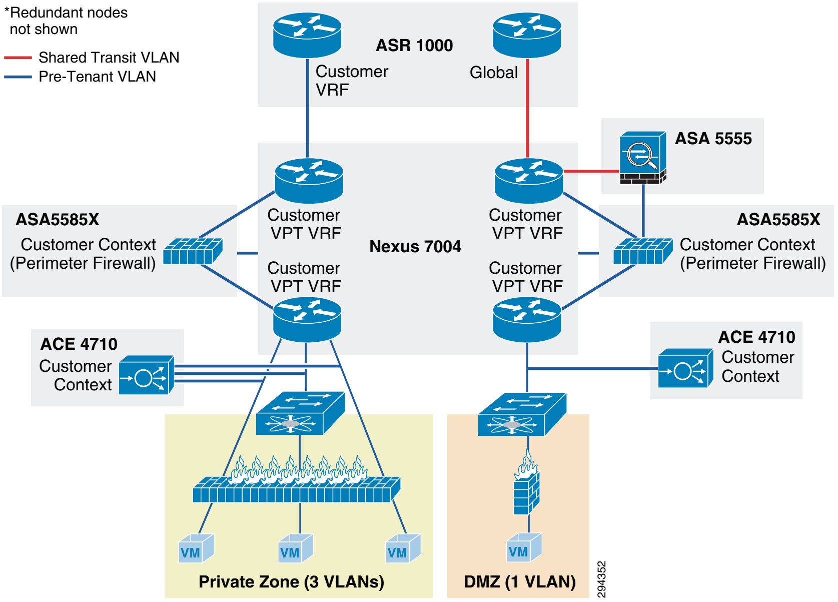
VMDC 2.3 Network Containers

The VMDC 2.3 solution defines a reference three-tier Infrastructure as a Service (IaaS) model of Gold, Silver, and Bronze tiers. These service tiers define resource and service levels for compute, storage, and network performance. This is not meant to be a strict definition of resource allocation, but to demonstrate how differentiated service tiers could be built. These are differentiated based on the following features:

- **Network resources**. Differentiation based on network resources and features:
  - **Application tiers.** Service tiers can provide differentiated support for application hosting. In some instances, applications may require several application tiers of VMs (web, application, database). VMDC 2.3 Gold and Silver services are defined with three application tiers on three separate VLANs to host web, application, and database services on different VMs. The Bronze service is defined with one VLAN only so if there are multi-tiered applications, they must reside on the same VLAN or potentially on the same VM (Linux, Apache, MySQL, PHP, Perl, or Python (LAMP)/Windows Apache, MySQL, PHP, Perl or Python (WAMP) stack). All three services, Gold, Silver, and Bronze, are defined with separate VRF instances to provide security and isolation.
  - **Stateful services.** Tenant workloads can also be differentiated by the services applied to each tier. The Gold service is defined with an ASA 5585-X virtual firewall context, ACE 4710 Virtual Server Load Balancer (vSLB) context, and secure remote access (IPSec VPN and SSL-VPN) on the ASA 5555-X. The Silver tier is defined with an ACE vSLB. The Bronze tier is defined with no services on ASA or ACE. All three services include the Nexus 1000V Virtual Security Gateway (VSG) for compute firewall services.
  - **Quality of Service (QoS)**. Bandwidth control during periods of network congestion can be a key differentiator. QoS policies can provide different traffic classes to different tenant types and prioritize bandwidth by service tier. The Gold tier supports VoIP/real-time traffic, call signalling and data class, while the Silver, Bronze, and Copper tiers have only data class. Additionally, Gold and Silver tenants are given bandwidth guarantee with Gold getting more bandwidth (2x) than Silver.
- **VM resources.** Service tiers can vary based on the size of specific VM attributes, such as CPU, memory, and storage capacity. The Gold service tier is defined with VM characteristics of four vCPUs and 16 GB memory. The Silver tier is defined with VMs of two vCPUs and 8 GB, while the Bronze tier VMs have one vCPU and 4 GB.
- **Storage resources.** To meet data store protection, RPOs, or RTOs, service tiers can vary based on provided storage features, such as redundant array of independent disks (RAID) levels, disk types and speeds, and backup and snapshot capabilities. The Gold service is defined with 15k FC disks, Silver tier on 10k FC disks, and Bronze tier on SATA disks.

Figure 2-11 shows a representation of a VMDC 2.3 Gold service tier network container.

Figure 2-11 VMDC 2.3 Expanded Gold Network Container



The network container is a logical (virtual) segment of the shared (common) physical network resources (end-to-end through the DC) that represents the DC network domain carrying tenant traffic. The physical infrastructure is common to all tenants, but each network device (routers, switches, firewalls, and so forth) is virtualized such that each tenant's virtual network container is overlaid on the common physical network.

The Gold tenant gets two network (and compute/storage) zones to place workloads into. Each zone has its own set of VLANs, VRF instances, and firewall/load balancer contexts. Figure 2-11 shows a logical representation of a two-zone VMDC 2.3 Expanded Gold network container.

This Gold service tier provides the highest level of sophistication by including secure remote access, firewall, and load balancing to the service. The vFW (on the ASA 5585-X60) provides perimeter security services, protecting tenant VMs. The vSLB (ACE 4710 appliance) provides load balancing across VMs in each tier of the tenant. The ASA 5555-X provides virtualized secure remote access (IPsec-VPN and SSL-VPN) to tenant VMs from the Internet. The ACE and ASA service module/ appliance are utilized in routed (L3) virtual mode in the VMDC 2.3 design. The Gold service tier also includes the Nexus 1000V VSG for providing virtual security services to the VMs. The Gold service provides higher QoS SLA and three traffic classes - real-time (VoIP), call signaling, and premium data.

The two zones can be used to host different types of applications, to be accessed through different network paths. The two zones are discussed below.

- **PVT Zone:** The Private Zone (PVT) and its VMs can be used for cloud services to be accessed through the customer MPLS-VPN network.
  - The customer sites connect to the provider MPLS-core and the customer has their own MPLS-VPN (Cust-VRF).
  - The VMDC DC ASR 1000 PE connects to the customer sites through the MPLS-VPN (Cust-VRF in Figure 2-11).
  - This Cust-VRF is extended through the VMDC network to the Nexus 7004 aggregation switch.
  - On the agg/access Nexus 7004, the Cust-VRF connects to the ASA Cust-vFW, and then is connected back into a Cust-PVT-VRF on the Nexus 7004 agg/access device (VRF sandwich to insert service nodes), and then to the compute layer on the UCS.

  - For the VMDC 2.3 Gold tenant, the PVT zone is defined with three server VLANs.
  - In addition, each tenant is assigned a separate Nexus 1000V VSG instance. The tenant is defined as an ORG in the VSG (PNSC), with the three VLANs placed into separate VSG sub-zones.
  - The VSG is used to provide security policies to monitor and protect traffic between the VLANs (sub-zones).
- **DMZ Zone**: The VMDC 2.3 Gold container supports a DMZ Zone for tenants to place VMs into a DMZ area, for isolating and securing the DMZ workloads from the PVT workloads, and also to enable users on the Internet to access the DMZ-based cloud services.
  - The ASR 1000 PE WAN router is also connected to the Internet and a shared (common) VRF (usually global routing table) exists for all Gold tenants to connect to (either encrypted or unencrypted).
  - Encrypted (SSL or IPsec Remote Access VPN) traffic is sent to an ASA 5555-X, and based on the VPN policy, is mapped to a particular tenant and the corresponding tenant VPN VLAN.
  - The tenant VPN VLAN then connects to the tenant DMZ-vFW (different vFW context on the ASA 5585-X than the tenant PVT-vFW), then to the tenant DMZ-VRF (different VRF on the Nexus 7004 agg/access than the tenant PVT-VRF), and then to the Compute layer for the DMZ Zone.
  - Similarly, unencrypted traffic from the Internet, based on the destination VM/VIP address, is sent to the tenant DMZ-vFW, then to the DMZ-vSLB, DMZ-VRF, and the DMZ Compute Zone.
  - The DMZ Zone can be used to host applications like proxy servers, Internet-facing web servers, email servers, etc. The DMZ Zone consists of one server VLAN in this implementation.

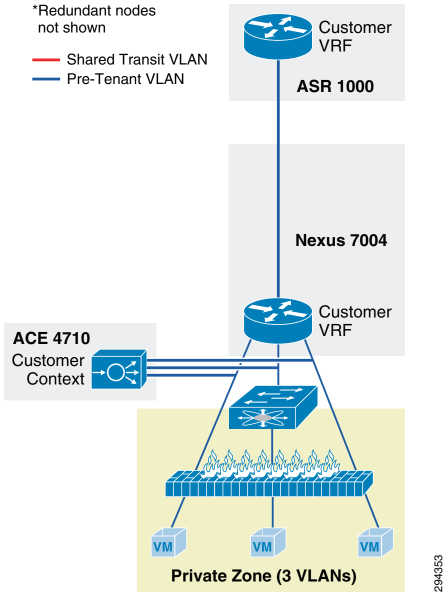
In VMDC 2.3, a Gold tenant can choose to have only the PVT Zone, or both the PVT and DMZ Zones. If the tenant has both PVT and DMZ Zones, then the Gold tenant will consume three VRF instances (Cust, Cust-PVT, and Cust-DMZ) on the Nexus 7004 Agg, two VFW instances, two vSLB instances, two VSGs, and four server VLANs. To facilitate traffic flows between the DMZ and PVT Zones (for example, proxy or web servers in the DMZ Zone, application and database servers in the PVT Zone), the DMZ-vFW and PVT-vFW are interconnected. Configuring appropriate security policies (routing, NAT, firewall rule, ACLs) on the DMZ-vFW and PVT-vFW can allow or disallow communication between the two zones.

Load-balanced traffic for all tiers of Gold tenants is implemented using the ACE 4710, which has one interface in each of the tiers.

- MPLS-VPN to PVT Zone
- Unsecured (clear) Internet to DMZ Zone
- Secure (Remote Access SSL/IPsec VPN) Internet to DMZ Zone
- DMZ to PVT Zone
- MPLS-VPN to DMZ Zone
- PVT to Internet Zone is via an HTTP proxy hosted in the DMZ Zone

Figure 2-12 is a representation of a VMDC 2.3 Silver network container.

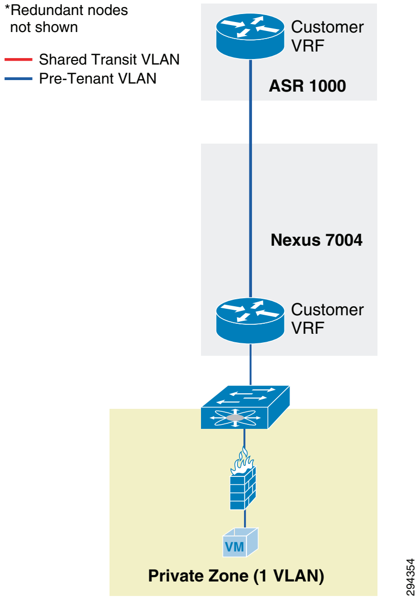
Figure 2-12 VMDC 2.3 Silver Network Container



The Silver service tier includes one VRF instance per Silver tenant and three server VLANs (three-tiered applications) for each tenant. The Silver service includes a load-balancing service for more sophistication over the Bronze tier. The vLB (ACE 4710 appliance) provides load balancing across VMs in each tier of the tenant. The ACE service load balancer is utilized in one arm, routed (L3), virtual mode in the VMDC 2.3 design, and one context is used per Silver tenant. The context has links on each of the server VLANs and works in one-arm mode. The Silver service tier also includes the Nexus 1000V VSG to provide virtual security services to the VMs. The Silver service provides medium QoS SLA and one traffic class, premium data.

Figure 2-13 is a representation of a VMDC 2.3 Bronze network container.

Figure 2-13 VMDC 2.3 Bronze Network Container



The Bronze service tier includes one VRF instance and one server VLAN for each tenant. The Bronze service is the least sophisticated tier and does not include any perimeter security services. The Bronze service tier does include the Nexus 1000V VSG for providing virtual security services to the VMs. The Bronze service provides lower QoS SLA and one traffic class, standard data.

Note Additionally, VMDC 2.3 also defines a Copper network container, which has the similar characteristics as Bronze, but has only Internet-based access and no L3VPN-based access. The Copper container also uses a shared perimeter firewall (ASA vFW context) for all tenants. However, the VMDC 2.3 Copper network container has not been validated with the DRaaS System.

Modifications in VMDC Network Containers for DRaaS

The VMDC 2.3-based infrastructure and Gold, Silver, or Bronze network containers (specific container used by a tenant based on FW, SLB services needed) can be used for DR services, but the following modifications need to be made:

- Utilize a new ASA context per tenant for IPsec-VPN services to encrypt the communication between InMage control servers in the DR site and the Enterprise site. This ASA context needs to be added whether the tenant is using Gold, Silver or Bronze container on the DR site.This context will logically reside close to the server VLANs.

  Note In the case of Silver or Bronze VMDC containers, no existing ASA context is being used in the network container for firewall or VPN services. Therefore inserting this ASA context for InMage VPN purposes will be a new addition to the network container. In the case of the VMDC Gold container, an ASA context (on the multi-context ASA5585X) is utilized for perimeter firewall services, and a shared ASA (single-context ASA5555) is utilized for remote access VPN purposes. However, these existing ASA contexts in the VMDC Gold container cannot be used for the InMage VPN purposes since they logically sit in a different part of the network container. This new ASA context for the tenant can be created on the existing ASA5585-FW device (if enough capacity for contexts and throughput exists) or a new ASA device can be utilized. It is recommended to use a new physical ASA device (ASA5555 or ASA55585 based on VPN throughput needed) for the InMage VPN purposes. Thus, the VMDC 2.3 infrastructure for DRaaS would have three separate physical ASA devices: one eachg for FW, RA-VPN, and one for Inmage Site-Site VPN. The VMDC 2.3 Gold container for DRaaS would have three logical ASA devices: one per-tenant context for FW, one shared/global ASA for RA-VPN, and one per-tenant context for InMage Site-Site VPN

- In the case of Gold containers, the tenant ASA context performing perimeter firewall services needs to have a security policy (ACL) configured to permit the IPsec-VPN traffic from the ENT site to the DR site. This ACL should be specific to allow IPsec traffic only between the IPsec tunnel endpoints (local ASA Site-Site VPN endpoint in the DR site, and remote VPN endpoint in the ENT site) used to encrypt the InMage traffic.
- Create a new VLAN for Bronze container to host the InMage control servers. To insert an ASA context to encrypt the InMage traffic, we need to create an outside and inside interface on the ASA context. Since the VMDC 2.3 Bronze container is defined with only one VLAN, we need to define a new VLAN to host the InMage servers. The first VLAN (where recovered VMs will be placed) will serve as the outside interface for the ASA VPN context and the second (new) VLAN (where the InMage servers are placed) will serve as the inside interface for the ASA VPN context.

Note For the VMDC 2.3 Gold and Silver containers, three server VLANs already support three-tier applications so there is no need to create a new VLAN to host the InMage servers or for InMage VPN purposes. Instead, the InMage servers can be hosted in the second VLAN (App tier). The first VLAN (Web tier, where recovered Web-tier VMs will be placed) will serve as the outside interface for the ASA VPN context, and the second VLAN (App tier, where the recovered App-tier VMs and also the InMage servers will be placed) will serve as the inside interface for the ASA VPN context.

Figure 2-14, Figure 2-15, and Figure 2-16 show logical representations of the modified VMDC 2.3 Gold, Silver, and Bronze network containers for DRaaS.

Figure 2-14 Modified VMDC 2.3 Gold Container for DRaaS

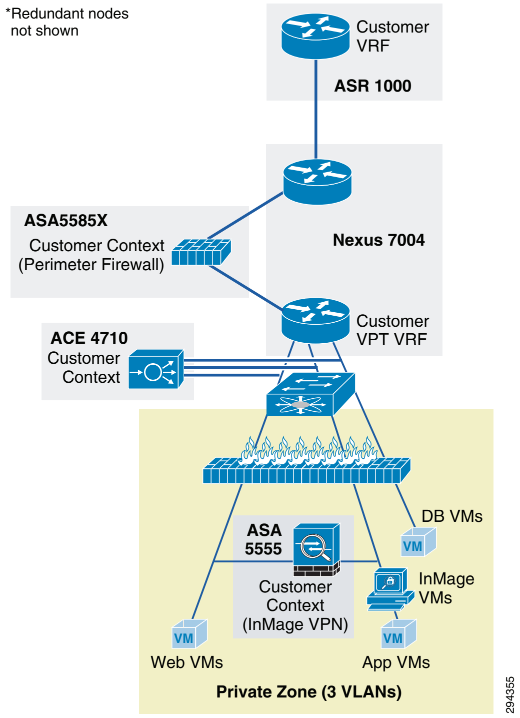


Figure 2-15 Modified VMDC 2.3 Silver Container for DRaaS

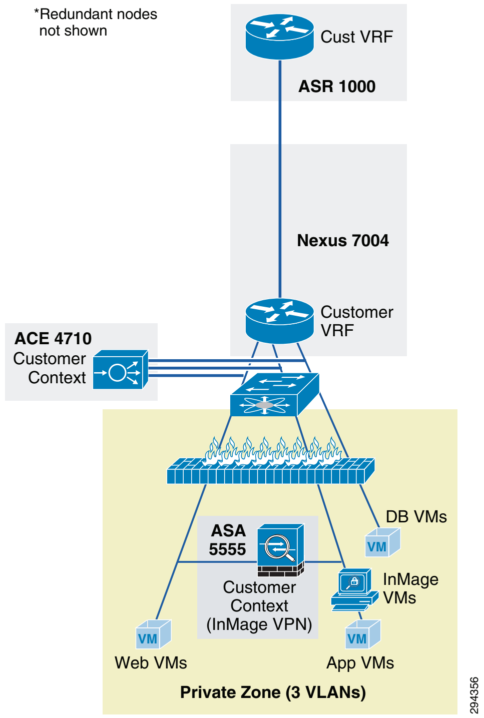
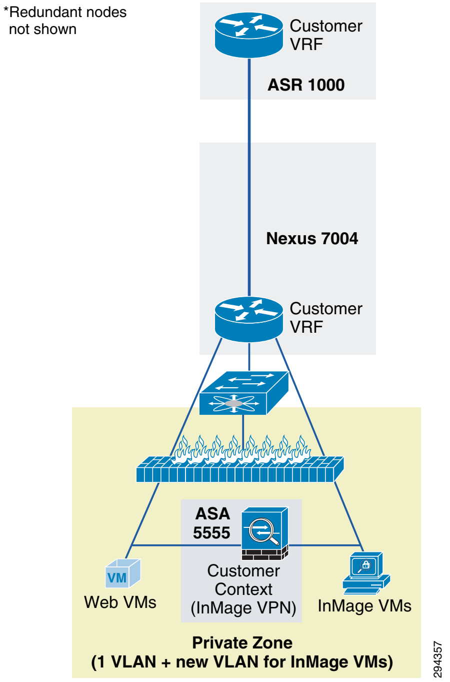


Figure 2-16 Modified VMDC 2.3 Bronze Container for DRaaS


*Redundant nodes not shown
Customer VRF
ASR 1000
Nexus 7004
Customer VRF
ASA 5555
Customer Context (InMage VPN)
VM
Web VMs
InMage VMs
Private Zone
(1 VLAN + new VLAN for InMage VMs)
294357

VMDC Orchestration using BMC CLM

The Cisco-BMC cloud management architecture for VMDC is designed to meet the growing needs of today's data center and cloud deployments. BMC Cloud Lifecycle Management (CLM) provides an end-to-end automated lifecycle management solution for cloud-based IT hosting environments.

The architecture focuses on the planning, governance, provisioning, operation, administration, and maintenance of cloud services, the runtime environments and infrastructure resources needed to sustain them, and the management services that comprise CLM.

The VMDC 2.3 architecture and network containers have been validated to be orchestrated by CLM 3.1 Service Pack 1 (SP1). CLM 3.1 SP1 includes all of the elements that are essential to enabling a VMDC 2.3-based cloud environment:

- **Self-service Portal and Service Catalog.** Provides the ability to order and track deployed services.
- **Service delivery automation.** Automates provisioning of services. CLM can also provide usage metering of services, by using additional BMC components.
- **Resource management.** Provisions and manages resources as per-service needs. This includes network, compute, and storage resources.
- **Operational process automation.** Automates operational processes such as user management, service desk integration, and alerting. Capacity management and service level management can also be provided by additional BMC components like BMC Capacity Optimization (BCO) and BMC ProactiveNet Performance Management (BPPM).

CLM 3.1 SP1 enables onboarding and pooling of resources for compute, storage, and networking, and creation of policies to manage those pools. It provides functionality to provision network containers, physical servers, and virtual server instances. It also provides the ability for end users, through a portal, to place service requests to create and manage server instances. CLM 3.1 SP1 is fully multi-tenant/ multi-service aware. It can support simultaneous use of the cloud environment by multiple tenants that can request, deploy, and operate services independently.

Note For detailed information on using BMC CLM 3.1 SP1 for orchestrating VMDC 2.3 architecture, refer to the following document:

- Orchestrating VMDC 2.3 with BMC CLM 3.1 SP1 Design & Implementation Guide

This section includes the following topics:

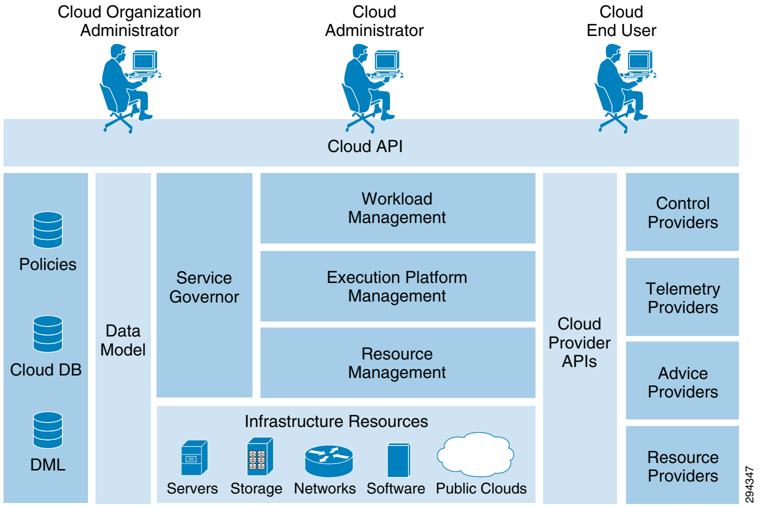
- CLM 3.1 SP1 Architecture, page 2-20
- Container Pre-Provisioning for DR Services, page 2-24

CLM 3.1 SP1 Architecture

CLM 3.1 SP1 is a general-purpose, one-size-fits-all management solution for cloud hosting environments. CLM 3.1 SP1 can manage environments that reside entirely on-premise or off-premise and hybrid environments that are hosted partially on-premise and off premise. CLM 3.1 SP1 can manage hosting environments that use physical or virtual compute, storage, and network resources. CLM 3.1 SP1 can now manage multi-hypervisor environments that include Microsoft Hyper-V and VMware vSphere. It can also manage environments that use cloud resources, including resources and services offered by other Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) clouds.

CLM 3.1 SP1 is fully multi-tenant aware. It can support simultaneous use of the cloud by multiple tenants that can request, deploy, and operate multiple services independently. CLM 3.1 SP1 has an architecture that provides the foundation for scaling the cloud and for configuring multiple data centers.

Figure 2-17 CLM 3.1 SP1 Architecture



User Roles

CLM 3.1 SP1 supports three different classes of users:

- **Cloud Administrator.** A Cloud Administrator is an IT professional responsible for the full lifecycle of the cloud environment, including initial planning, deployment and configuration, and continued administration, operation, and maintenance. The Cloud Administrator uses the Administration console for most tasks.
- **Cloud Organization (Tenant) Administrator.** A Cloud Organization (Tenant) Administrator is responsible for managing a subset of the cloud that is provisioned for a particular Tenant or Service.
- **Cloud End User.** Cloud End Users request services made available to them by the Cloud Administrator through the BMC My Services console. Cloud End Users can request virtual as well as physical resources, view and manage their commissioned resources, monitor the health of their commissioned resources, and decommission resources.

Consoles

CLM 3.1 SP1 has three consoles that cloud users can use to manage the VMDC cloud infrastructure:

- **BMC CLM My Cloud Services console.** This enables users and administrators to request, deploy, and operate Service Offerings from the Service Catalog.
- **BMC CLM Administration console.** This enables Cloud Administrators to manage the cloud and the services that it hosts.
- **BMC Tenant Administration console.** This is an enhanced My Cloud Services console. This console offers additional administration capabilities to Tenant Admins, such as, capabilities around network management.

Service Catalog

The Service Catalog contains the Service Offerings that are available for consumption by cloud users. Cloud Administrators maintain the Service Catalog by creating, modifying, and deleting Service Offerings. They can also control which offerings in the Service Catalog are available to each Tenant.

Cloud Database

The Cloud DB contains operational state and configuration information about the objects managed by the cloud. These managed objects include the Service Offering Instance (SOI), virtual cloud resources, and physical and virtual infrastructure resources.

Product Catalog and Definitive Media Library

The Product Catalog and Definitive Media Library (DML) list all software that can be provisioned in the cloud. The Product Catalog does not store the software itself. Instead, it contains a unique reference to each piece of software, while the software itself remains in native repositories such as the BMC Server Automation (BSA) software repository or the Amazon AWS Amazon Machine Images (AMI) repository. The Product Catalog also contains software metadata, such as software and hardware requirements pertaining to software provisioning, as well as other data used during software configuration. Cloud Administrators create and maintain entries in the Product Catalog by using interfaces provided by the Product Catalog.

Cloud Blueprints

Cloud blueprints define cloud services and resources that can be provisioned in the VMDC infrastructure. CLM 3.1 SP1 uses the following cloud blueprints:

- **Service blueprints** describe the functional structure of a given Service Offering, including its functional components and communication paths. They also define how a Service Offering is to be deployed under different circumstances. Each Service Offering in the Service Catalog has a Service Blueprint that is used for its instantiation. When creating a Service Blueprint, the user can define the service and how it is deployed:
    - *Service definitions* of applications or server instances specify the topology (number of tiers), configuration, operating systems, and software packages that need to be provisioned to "stand up" an application or server.
    - *Service deployment definitions* for each Service Blueprint specify a set of one or more ways in which the blueprint could be instantiated when it is provisioned.

      For example, in a blueprint for an application, one service is related to three deployments, Small, Medium, and Large, that are mapped to a Service Offering in the Service Catalog. The Small deployment definition for the application might use a single Resource Set that consists of one VM to support all three tiers, web, business logic, and database. In contrast, the Large deployment definition might distribute the application component to three different Resource Sets, each corresponding to a different application tier.
- **PoD blueprints** define the physical topology of the VMDC infrastructure.
- **Network container blueprints** define the logical segmentation of the VMDC cloud infrastructure.

Infrastructure Resources

Cloud infrastructure resources represent physical or virtual DC resources that host Service Offerings in the cloud. All compute, storage, network, and software resources that are part of the VMDC infrastructure are considered to be infrastructure resources. In the VMDC 2.3 system, the following components comprise the infrastructure resources:

- ASR 1006 (WAN Router)

- Nexus 7004 (DC Aggregation layer)
- Nexus 5548 (DC Access layer)
- ACE 4710 (Server Load Balancer appliance)
- ASA 5585-X (Firewall appliance)
- UCS B-series blade and C-series rack servers
- UCS 6248 (Fabric Interconnect)
- Nexus 1000V (Distributed Virtual Switch)
- Virtual Security Gateway (Compute Firewall)
- NetApp FAS and EMC VMAX, VNX (NAS and SAN devices)
- VMware Virtual Center
- VMware ESXi clusters and hosts
- Microsoft Hyper-V 2012
- Microsoft System Center Virtual Machine Manager 2012

Cloud Providers

Cloud providers are software programs that act as element managers for different types of resources, platforms, and services consumed by the cloud. At installation, CLM 3.1 SP1 includes the following providers:

- **BMC Server Automation (BSA).** BSA is a control and resource provider for various types of infrastructure compute resources, such as physical servers, VMs, VM clusters, and virtual cluster resource pools.
- B**MC Network Automation (BNA).** BNA is a control and resource provider for network resource providers, such as IP addresses, routers, firewalls, load balancers, and VLANs
- **Amazon EC2.** This provider facilitates integration with the Amazon EC2 cloud.
- **vCloud Director.** This provider facilitates integration with the VMware vCloud Director-based cloud environment.

Cloud Workload Manager

The cloud Workload Manager (WLM) instantiates Service Offerings selected from the Service Catalog. It also administers and maintains those services based on policies defined in the cloud policy DB.

Cloud Platform Manager

The cloud Platform Manager provisions, operates, administers, and maintains runtime platform instances.

Cloud Resource Manager

The cloud Resource Manager (RM) manages cloud infrastructure resources, including onboarding, Organization, assignment, and allocation.

Cloud Service Governor

The cloud Service Governor orchestrates workload management, platform management, and resource management operations based on policies and rules defined by Cloud Administrators to provision hosted Service Offerings. The Service Governor makes placement decisions on where resources need to be placed, based on resource management/tracking and service policies/tagging. The cloud Service Governor also distributes and subsequently enforces operational policies across CLM 3.1 SP1 components.

CLM 3.1 SP1 Northbound API

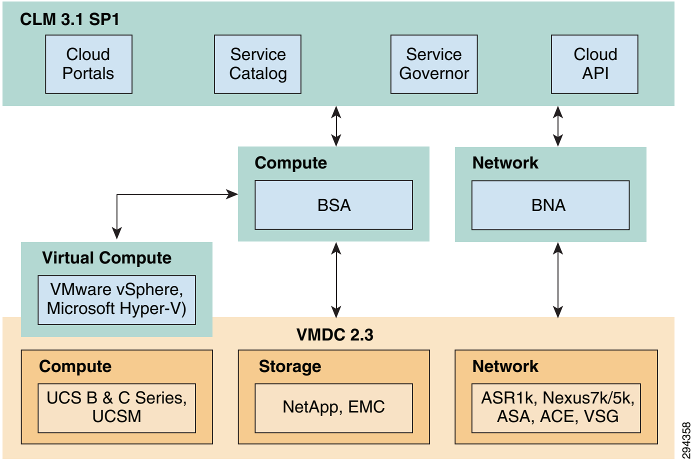
The solution architecture also defines a set of APIs that can be used by third-party application developers to enhance the CLM 3.1 SP1 implementation. The API can be used to do the following tasks:

- Integrate with CLM 3.1 SP1
- Automate repetitive processes
- Create a customized portal in CLM 3.1 SP1

The API is a model-based, object-oriented RESTful web service that features an easy-to-use interface facilitated by standard HTTP requests and response messages that carry JSON-format payloads.

Figure 2-18 illustrates the BMC CLM 3.1 components and interactions with infrastructure components.

Figure 2-18 CLM 3.1 SP1 Components and Interactions



Container Pre-Provisioning for DR Services

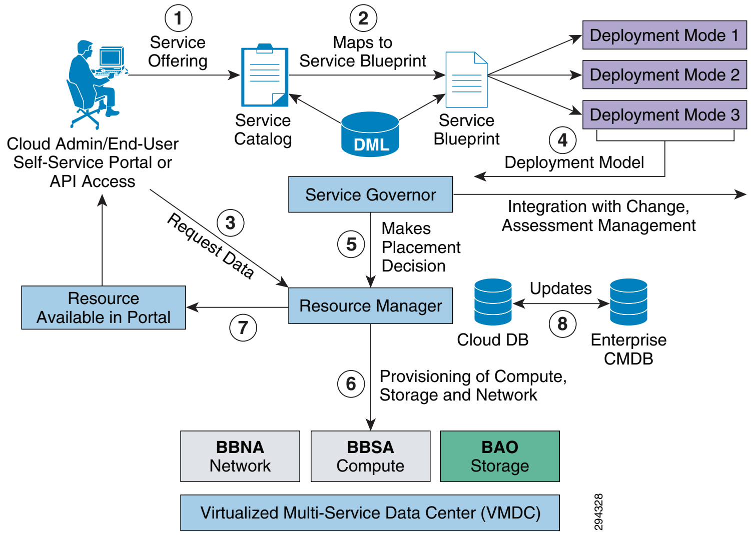
To enable the DRaaS, the VMDC-based network containers need to be pre-provisioned in the DR site. Depending on the services (firewalls, load balancers) needed, the DR customer can choose to use the VMDC 2.3 Gold, Silver, or Bronze Network containers. BMC CLM can be used to automatically provision the VMDC Gold, Silver, or Bronze network containers on a VMDC 2.3 infrastructure. This capability for orchestrating VMDC 2.3 Gold, Silver, or Bronze network containers is available out-ofthe-box in BMC CLM 3.1 SP1.

Note For more information on using BMC CLM 3.1 SP1 to orchestrate VMDC 2.3 network containers, refer to the following document:

Cloud Orchestration for VMDC with BMC Cloud Lifecycle Management 3.1 SP1 Design and Implementation Guide

Figure 2-19 illustrates the high-level CLM orchestration workflow.

Figure 2-19 CLM 3.1 SP1 End-to-End New Request Flow



The workflow steps needed for container pre-provisioning and preparing the DR service are listed:

Step 1 VMDC 2.3-based infrastructure, topology, and base infrastructure configurations are created in the DR site.

Step 2 BMC CLM 3.1 SP1 are installed in the DR site. The VMDC 2.3 PoD and Network Container Blueprints are on-boarded into CLM. The CLM portal is set up and running.

Step 3 Tenant (ENT customer requiring DR services) uses the BMC CLM portal to request the Network Container from the Service Catalog. The tenant can choose Gold, Silver, or Bronze network container.

Step 4 CLM Cloud Admin (DR Provider) approves the change request, and the requested Network Container for DR services is created on the VMDC infrastructure for the tenant. Appropriate Compute and Storage resources are set aside for the Tenant.

Step 5 Necessary modifications are made in the VMDC network container to facilitate the DR service are made. This includes setting up any additional VLANs, creating the ASA context for IPsec VPN tunnels (for secure InMage communication from customer site to DR site, etc., as documented in VMDC container. Adding VLANs can be done by the Cloud Admin through the CLM portal. Creating the ASA context for VPN, and connecting to tenant's VMDC network container and VLANs has to be done manually by the DR site network administrator.

a. Install the InMage control plane servers into the tenant's VMDC network container in the appropriate VLANs. This task can be done through CLM by the CLM Cloud Admin, if the InMage applications are created as Service Offerings in the CLM Service Catalog. Otherwise, these InMage applications have to be manually installed by the DR site server administrator.

b. Set up the InMage DR components, storage and other necessary steps as documented in Chapter 3 Implementation and Configuration. These InMage DR components have to be set up at both the Enterprise and Provider DR sites.

c. Verify IPsec connectivity between the DR components in the Enterprise and SP DR sites.

d. Verify DR functionality and create protection and recovery plans through InMage.

As part of the VMDC network container creation, BMC CLM also creates the firewall and load balancer services for the relevant containers. For example, in the VMDC 2.3 Gold container, perimeter FW services are provided by an ASA context, load balancing services are provided by an ACE context, and back-end or compute FW services are optionally provided by a VSG. When BMC CLM is used to create a VMDC 2.3 Gold network container for DR services, the ASA context, ACE context, and VSG are created for the tenant container. CLM also provisions some base security rules on the ASA an VSG. In addition, the CLM portal can be used to provision more specific FW security rules through network paths. These can be done by the Cloud Admin or by the Tenant Admin, and the security policies can be done at a VLAN (IP subnet and protocol/port) level or at a VM (specific IP address and protocol.port) level. In addition, CLM portal can be used (by Cloud Admin or Tenant Admin) to create load balancing virtual servers (VIPs) for specific protocols on the tenant ACE context, and associate servers or server pools (VMs) to the VIPs.

In the DRaaS scenario, after a failure, the recovered VMs will be brought online in the VMDC network container in the provider DR site. As such, these VMs need to have associated security policies on the ASA (and VSG if tenant is using compute FW), and associated load balancing policies on the ACE context. The following steps need to be followed to accomplish this once the workflow steps described above for creating the DR service have been completed.

Step 1 Network Container Pre-provisioning

a. Follow the workflow steps described above to pre-provision the tenant network container through CLM in the DR site, install the DR components on the ENT and DR sites, and identify VMs to be protected/recovered.

Step 2 vFW Policies

a. Use the CLM portal to create Network Paths to allow/deny specific traffic flows to/from the VMs that will be recovered on the DR site. These Network Paths will be translated into appropriate ASA and VSG security policies and provisioned on the tenant ASA and VSG by CLM.

b. The Network Paths and security policies can be based on VM subnets or specific VM IPs. It is recommended to use security policies based on VM subnets, so as to minimize the changes necessary when additional VMs get added into the protection plans (within the same VLAN or subnet).

c. It is recommended to configure these Network Paths and underlying security policies before DR declaration, so that all services are in place when VMs get recovered to the DR site. Post-recovery, the CLM portal can again be used to tweak the Network Paths and security policies as needed.

Step 3 vSLB Policies

a. Use the CLM portal to create Virtual Servers (VIP) and associate to real servers (VMs), and define the protocols to be koad balanced and the mechanisms for probing the real servers. CLM will provision these policies on the tenant ACE context. The real servers have to reflect the specific IP addresses to which the VMs to be load balanced.

b. It is recommended to configure these SLB policies before DR declaration, so that all services are in place when VMs get recovered to the DR site. Post-recovery, the CLM portal can again be used to tweak the load balancing policies as needed.

Available Replication Types

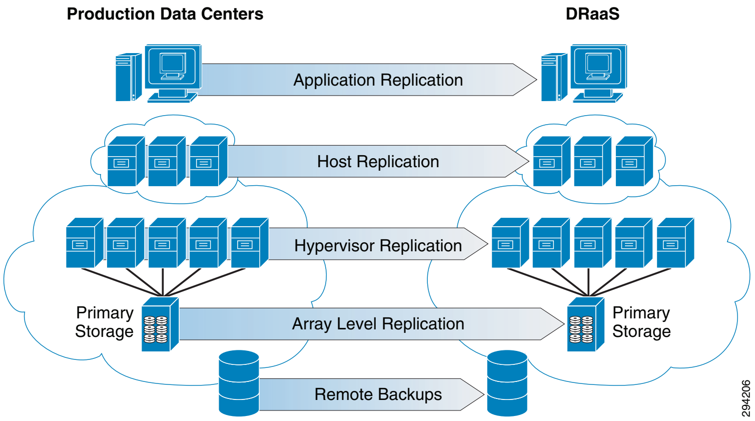
This section includes the following topics:

- Hypervisor-based vs. Guest OS vs. Storage
- InMage Software Architecture, page 2-28

Hypervisor-based vs. Guest OS vs. Storage

Figure 2-20 shows the different types of replication technologies that can be used for disaster recovery purposes.

Figure 2-20 Disaster Recovery Replication Technologies



Storage Array Level Replication

The most popular replication method used by most of the organizations today is storage array-level replication. Array-based replication is expensive and lacks granularity. You need to purchase from a single storage vendor the exact type, brand, and model number of a storage array on both the source and target side of your DR solution. You need to budget for exactly the same storage class and tier. One of those storage arrays will stay dormant until a recovery situation requires it to be active. An array-based solution typically replicates an entire volume even if there is only one VM in the volume that needs to be replicated. It does not provide the flexibility of replicating a single VM. It also requires multiple points of management while performing disaster recovery tasks and needs a separate run book management tool along with the storage array management console.

Hypervisor-Based Replication

Hypervisor-based replication is a good option for organizations who has all of their environment virtualized. The agent that captures the changes on the production servers sits at the hypervisor layer. Since hypervisor-based replication is "VM-aware," it is possible to select the VMs that need to be replicated, while saving storage space at the secondary site by avoiding replicating the ones that don't. Hypervisor-based replication allows you to be much more granular in what you protect, and it also allows you to group VMs by defining protection groups. And it can be managed from virtualization management suites like VMware's vCenter or Microsoft's System Center. The main limitation of hypervisor-based replication is that it's specific to a hypervisor and using the same solution physical environments cannot be protected.

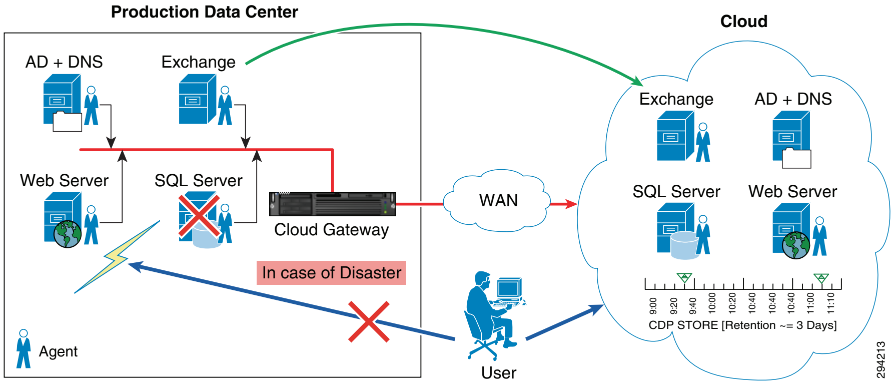
Guest OS/Host Based Replication

Many enterprises use host-based replication because it is relatively inexpensive. The process involves installing a replication agent onto the operating systems of the servers to be replicated. This agent processes and replicates I/O traffic on any storage systems (NAS, DAS, SAN, etc.) to a secondary replication target system, which use storage of any type, from any vendor. It saves money compared to array-based replication because licensing host-based replication software is much less expensive than for most array-based replication systems. Also, there's no need to go to the expense of purchasing a second storage array that's identical to the primary one. SPs can deploy any storage type in their cloud while offering DRaaS. This allows them to offer DRaaS to customers using any storage and infrastructure.

Though many organizations are embracing virtualization, most organizations are still not 100% virtualized and still have critical and legacy applications running on physical environments. Using host-based replication, both physical and virtual environments can be protected and the solution is agnostic to the server, operating system, and storage. The DRaaS system uses host-based replication for its simplicity and for providing greater coverage of protecting physical and virtual environments.

Figure 2-21 shows OS/Host-based replication in the DRaaS System.

Figure 2-21 OS/Host-Based Replication



InMage Software Architecture

This section includes the following topics:

- InMage Overview, page 2-28
- Components, page 2-31
- How InMage ScoutCloud Works, page 2-37
- Component Flow, page 2-38

InMage Overview

InMage ScoutCloud Enables Recovery as a Service

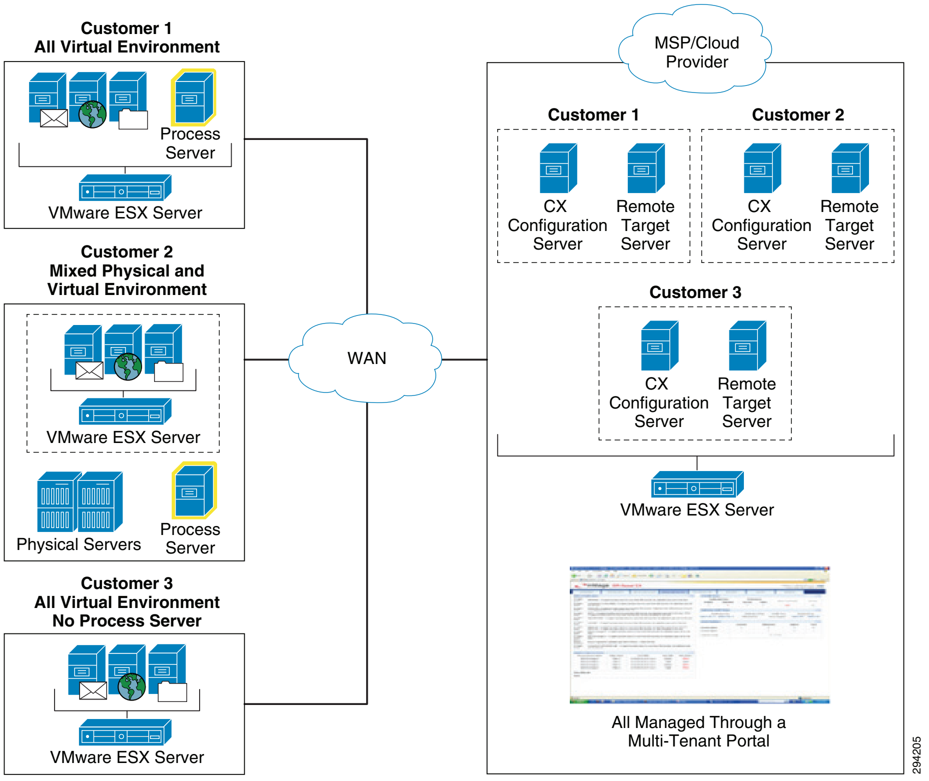
The InMage ScoutCloud platform addresses the growing market for cloud-based disaster recovery products, also referred to as the Recovery as a Service (RaaS) market. InMage ScoutCloud leverages next generation recovery technologies including disk-based recovery, CDP, application snapshot API integration, asynchronous replication, application awareness, and WAN optimization. These next generation recovery technologies are wrapped up in a single product offering, enabling MSPs and cloud providers to have the fastest time-to-market when offering customers a near zero RPO and RTOcapable RaaS with:

- Best-in-class data protection.

- A comprehensive P2V and V2V recovery engine that supports all applications.
- A provisioning manager that automates provisioning of recovery for VMs and associated storage combined with a full-fledged multi-tenant portal.

Figure 2-22 shows the InMage ScoutCloud architecture in a DRaaS environment.

Figure 2-22 InMage ScoutCloud Architecture



InMage ScoutCloud Concepts

Continuous Data Protection (CDP): CDP refers to a technology that continuously captures or tracks data modifications by saving a copy of every change made to your data, essentially capturing every version of the data that you save. It allows you to restore data to any point in time. It captures the changes to data and sends them to a separate location. CDP-based solutions can provide fine granularities of restorable objects ranging from crash-consistent images to logical objects such as files, mail boxes, messages, and database files and logs.

Traditional backups require a schedule and restore data to the point at which it was backed up. CDP does not need a schedule because all the data changes on the primary server are tracked and sent to a secondary server asynchronously.

Most CDP solutions save byte or block-level differences rather than file-level differences. This means that if you change one byte of a 100 GB file, only the changed byte or block is saved. CDP technology has the following fundamental attributes:

- Data changes of primary server are continuously captured or tracked.
- All data changes are stored in a separately located secondary server.
- It enables data recovery in much lesser time as compared to tape backup or archives.

Disaster Recovery (DR): DR is the process of preparing for recovery or continuation of technology infrastructure critical to an organization after a natural or human-induced disaster. DR solution using CDP technology replicates your data to a separately located secondary server. In case of disaster, you can get immediate access to a primary server's data, which is up-to-the minute of disaster.

Application Protection Plan: An efficient Application Protection Plan can protect customer's critical applications from natural as well as human-interfered disaster. Every individual application of an organization should have a unique protection plan where the application can have single or multiple protections; i.e., the application can be protected locally for backup purpose or it can be protected to remote locations for DR purposes.

Replication Stages*:* InMage ScoutCloud replicates drive level data in three stages:

- Resyncing (Step I): In this step, data at the primary server is replicated to the secondary server. This is done only once for each drives that you want to replicate to a secondary server drive.
- Resyncing (Step II): All data changes during Resyncing (Step I) are replicated to the secondary server in this step.
- Differential Sync: Differential Sync is a continuous process where any change in the primary server volume is copied to the Secondary server volume simultaneously.

Consistent Data: In case of DR, the restored data should be consistent with the original data. To ensure the consistency of backup data, the consistent tags/bookmarks are issued at the primary server at periodic intervals of time or on demand.

Journal/Retention or CDP Logs: The retention or CDP logs store information about data changes on primary server within a specified time period on a separately located secondary server. This timeframe is referred to as the retention window. Consistent points are stored as bookmarks/tags in retention window. An application can be rolled back to one of the bookmarks/tags in this retention window. Alternately, an application can be rolled back to any point in time of this retention window. Applications that are rolled back to any of the bookmarks/tags in this retention window will only be consistent. Three types of retention policy are associated with this retention window:

- Time-based: The data in the retention window will be overwritten after the specified time period.
- Space-based: The data in the retention window will be overwritten once the size is exhausted.
- Time and space-based: The data in the retention window will be overwritten once the time specified or space specified qualifies first.

Sparse Retention: For long term data retention purposes, the sparse policy is used, which helps to save disk space on retention volumes and makes it possible to afford a wider retention window. Depending on the type of policy enforced, the retention window is maintained by discarding older data changes within the retention log files to make rooms for new data changes.

Failover: This is the process of switching production server to secondary server. The failover process can be a planned or an un-planned operation. The planned failover is used for periodic maintenance or software upgrades of primary servers wherein the data writes to primary server are stopped. An unplanned failover happens in case of actual failure of the primary server.

Failback: This is the process of restoring the primary server from the secondary server after a planned or un-planned failover. A failover operation is usually followed by a failback operation. In this failback process, the data writes on the secondary server are also restored to the primary server. Scout also supports fast failback where the data changes of the secondary server are not applied to the primary server while restoring.

Snapshot: A snapshot is an exact replica of a primary server's data as it existed at a single point in time in retention window. The two types of snapshot are Physical Snapshot and Virtual Snapshot:

- For Physical Snapshot, you can take a snapshot on a physical volume. It requires the intended snapshot volume to be equal or larger than the Secondary server volume (in the replication pair).

- For Virtual Snapshot, you can take a snapshot on a virtual volume. It is also known as "vsnap," which requires minimal system resources and are faster in loading or unloading. These snapshots can be accessed in one of following modes:
  - Read-Only: As the name indicates, read only snapshots are for informative purposes and are not capable of retaining writes on to them.
  - Read-Write: Read/write virtual snapshots retains writes on to them; this is done by maintaining an archive log on some part of the local disk as specified.
  - Read-Write Tracking: Read/write tracking virtual snapshots goes a step forward; this is especially useful if a new virtual snapshot has to be updated with the writes of an unmounted virtual snapshot.

Application Consistency: Application Consistency ensures the usability of the application when DR copies of the application's primary server data are used in place of the original data. An application can be rolled back to any bookmark/tag in the retention window. Consistency bookmarks are of the following three types:

Application bookmarks: This bookmark ensures consistency at the application level. This is issued after flushing the application buffers to the disk.

File System bookmarks: This bookmark ensures consistency of the data at the file system level. This is issued after flushing the file system cache to the disk.

User-defined bookmarks: This is a user-defined name for a bookmark which is associated with application bookmark or a file system bookmark or both. These are human readable bookmarks unlike the application or file system bookmarks, which are used by the DR administrators to recover the data.

Components

This section includes the following topics:

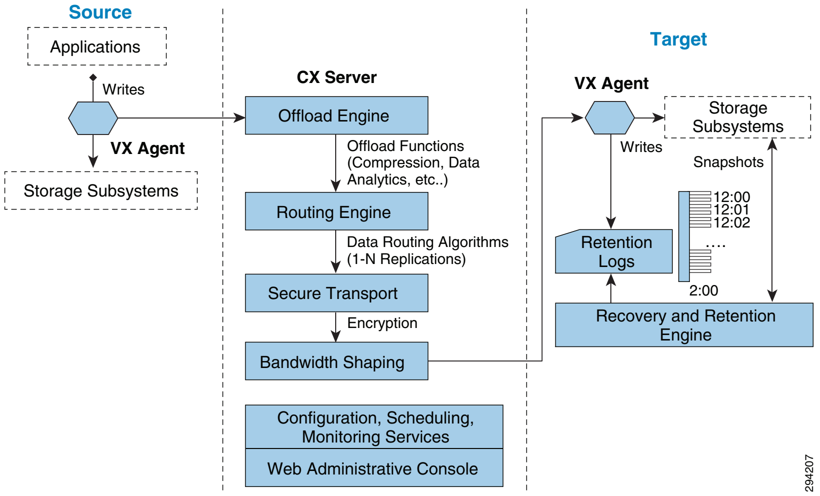
- Unified Agent, page 2-31
- Master Target, page 2-33
- CX Server, page 2-33
- RX Server, page 2-35
- Management Console, page 2-36
- Self Service, page 2-37

Unified Agent

Unified Agent (aka VX Agent) is a lightweight agent that is installed on to each VM or physical server protected. It offloads the data changes to the CX appliance. Unified Agent is installed automatically by the vContinuum wizard.

Figure 2-23 shows the VX Agent Theory of Operation.

Figure 2-23 VX Agent Theory of Operation

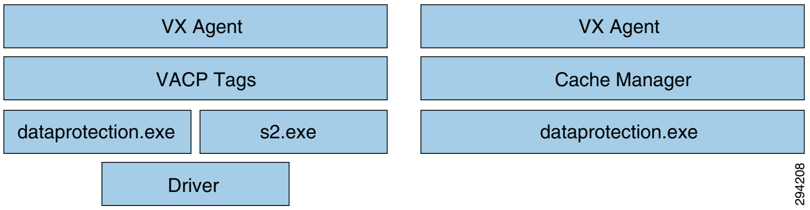


VX Agent - Responsible for Volume Level Replication and CDP

- Sentinel: The Sentinel software is installed on the protected hosts. It is responsible for keeping track of the data changes that occur. It is also necessary for the Initial Sync and Resync processes.
- Outpost Agent: The Outpost Agent software is installed on the CDP/DR hosts. It is responsible for keeping the CDP/DR host volumes in sync with the replicated host volumes. It is necessary for the Initial Sync and Resync processes. It includes the snapshot functionality.

Figure 2-24 shows the VX Agent Architecture.

Figure 2-24 VX Agent Architecture



The components required by the VX Agent differ depending on the role played by the VX Agent; i.e., source VX or target VX. VACP consistency tags, s2.exe and the driver will not be used on the target VX agent. The target VX Agent uses cache manager and dataprotection.exe to update the data changes from the source.

VX Components

Once the VX Agent is installed, a service named "svagents" is created. Svagents (Svagents.exe) is the Windows service that is responsible for launching and managing all other user space components of the VX Agent. This service runs two threads "dataprotection.exe" and "s2.exe":

- Dataprotection.exe: On the production server, dataprotection.exe is responsible for replicating all data on disk to the DR server. It is used while the replication pair is in "Initial Sync Step 1." On the target server, dataprotection.exe is responsible for both replication and recovery.
- S2.exe: This process runs only on the production server and starts along with dataprotection.exe. S2.exe works in sync with the driver to replicate real time writes happening to the production volume.

Master Target

A dedicated VM created on secondary vSphere server to act as a target for replication is called master target (MT). It is used as a target for replicating disks from primary VMs and MT contains the retention (CDP) data. Retention data is the log of prior changes using which you can recover a VM to prior point in time or to a prior application consistent point.

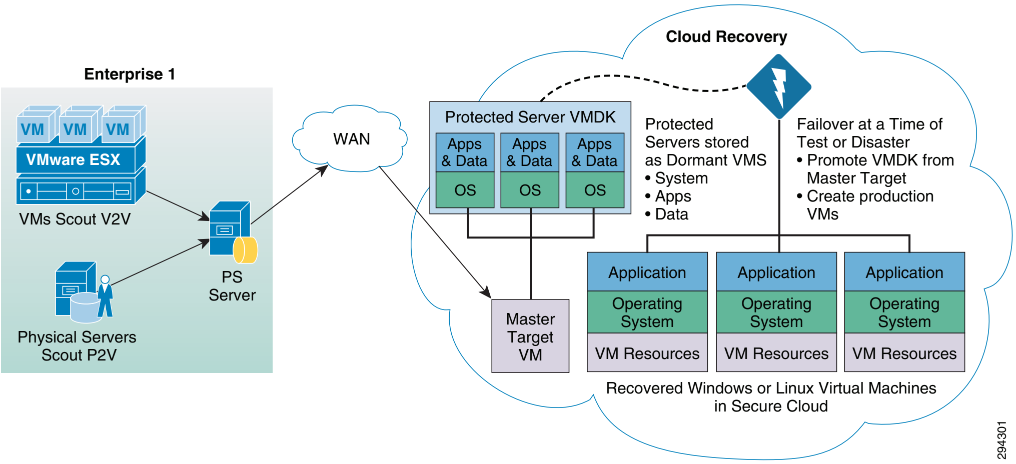
When the initial replication plan is set up for a server or group of servers. The data volumes on the production servers are created as VMDKs on the target site and get mounted to the MT server for writing data. In the event of a disaster, the VMDKs gets released by the MT server and will get mounted to the actual recovery servers.

The MT should be of the same OS family as that of primary servers. If primary VMs are Windows, MT has to be Windows. For Linux primary VMs, MT must be a Linux VM.

Win2k8R2 is recommended to protect Windows VMs. You can have more than one master target on secondary vSphere servers. To perform failback protection and failback recovery, a MT VM is required on the primary vSphere server. In case of failback, replication is set in reverse direction from recovered secondary VMs back to MT on the primary vSphere server.

Figure 2-25 shows the MT functionality.

Figure 2-25 Master Target Functionality



CX Server

CX Server is the combination of Configuration Server (CX-CS) and Process Server (CX-PS).

The Scout PS/CS server is an appliance that performs the data movement between primary and secondary servers. It offloads various CPU intensive tasks from the primary server, such as bandwidth management, caching, and compression. It is also used to monitor protection plans by the vContinuum wizard.

CX Server is:

- Responsible for data offload and WAN optimization functions such as:
  - Compression
  - Securing the data over WAN
  - Data Routing
  - Bandwidth Optimization

- Provides centralized UI for configuration and monitoring.
- Provides centralized error reporting using logs, SNMP, and email alerts.
- A mandatory component for all environments.

CX-PS server is deployed on the Enterprise data server which receives data from the production servers and sends to the target site.

CX-CS server is deployed on the cloud SP's datacenter dedicated per customer to manage the replication and recovery.

Figure 2-26 shows the CS/PS server theory of operation.

Figure 2-26 CS/PS Server Theory of Operation

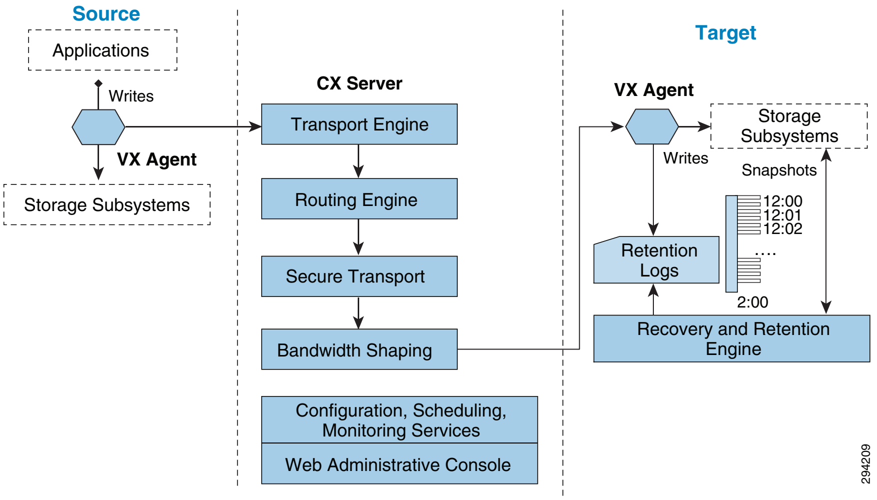
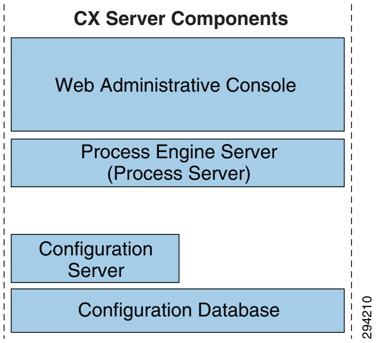


Figure 2-27 shows the CS/PS server architecture.

Figure 2-27 CS/PS Server Architecture



The CS/PS server has the following components:

- MySQL
- Common tables
- Hosts
- logicalVolumes
- srcLogicalDestinationLogicalVolume

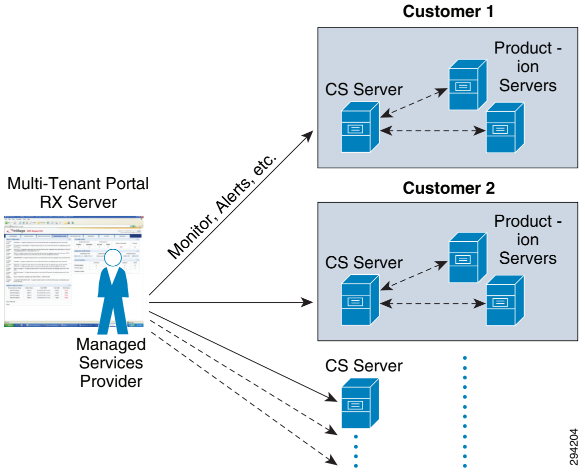
- Scripting
- Apache
- Configuration
- Transport Server
- HTTP/FTP
- Tmanagerd
- Executive functions
- Monitor.pl
- Keeps the GUI up to date
- Gentrends.pl
- Graphing
- Amethyst.conf
- Bootstrap configuration

RX Server

RX is the multi-tenant portal that enables the management of all customer services through a single portal and provides:

- Centralized monitoring across all customers.
- Fully re-brandable ready to use customer-facing dashboard.
- A full-fledged API stack for deeper integration into partner portals.
- Replication health and statistics for each CS server.
- License statistics for each CS server.
- Alerts and notifications.
- Provision to create logical groups for multiple CS servers to enforce policies on multiple CS servers in one shot.
- Custom reports on bandwidth usage for each CS server.

Figure 2-28 RX: Multi-Tenant Portal



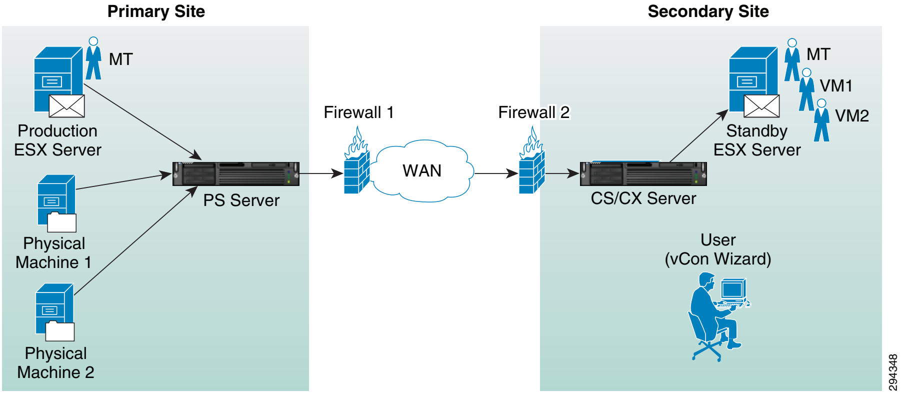
Management Console

The management console/GUI Wizard is a Windows 32 bit-based GUI wizard that progresses through the protection and recovery steps:

- In the case of Windows CX, it is installed along with the CX server.
- In the case of Linux CX, the Wizard has to be installed on Windows 2008 R2, Windows7, XP or Vista desktop. The vContinuum wizard can be installed on the MT in the case of Windows.

vContinuum is stateless and does not have the current information regarding the replication status; it talks to the CX-CS server to get this information.

Figure 2-29 vContinuum



The vContinuum wizard helps the cloud provider to perform the following tasks:

- Push agents to source production servers
- Create protection plans to protect servers
- Modify existing protection plan.
- Perform DR drill.
- Resume protection
- Failback

- Offline Sync

Self Service

Self Service can be enabled for the customers in the following ways:

- RX Portal: RX multi-tenant portal also allows the customers to perform recovery of their environments. This can be controlled by enabling the recovery option for the customer user account within the RX.
- vContinuum: vContinuum is a dedicated component deployed one per customer on the SP cloud. The SP can provide access to the vContinuum GUI to their customers who wants to have total control of the disaster recovery process. vContinuum allows customers to perform all the operations required for protecting and recovering the workloads into cloud.

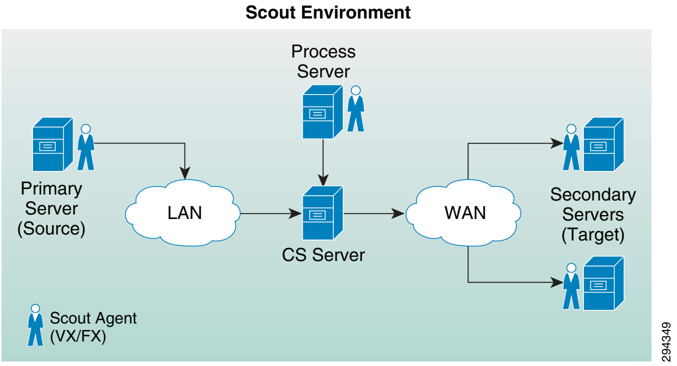
How InMage ScoutCloud Works

InMage ScoutCloud is based on CDP technology that gives it granular DR capabilities to meet most stringent DR requirements. InMage ScoutCloud can be configured to support long distance DR requirements as well as operational recovery requirements and supports heterogeneous servers running on Windows, Linux, or UNIX. ScoutCloud supports a web browser-based management UI that allows all management operations for both application and data recovery across different production servers and applications to be tracked and managed using a common management paradigm. A CLI to do the same is available as well. Management capabilities are protected through the use of a multi-level security model.

InMage ScoutCloud replicates a production server's data to one or more secondary servers that can be either local or remote, which can also be virtual or physical systems. ScoutCloud can be deployed into existing environments without disrupting your business continuity.

To understand how ScoutCloud works, let's look at a basic configuration with a single primary server and multiple secondary servers communicating to CX-CS server through a single CX-PS. The CX-CS is deployed in the primary server LAN network component whose failure and/or replacement do not impact production server's operation. The VX and FX component of ScoutCloud are deployed on your primary server, which utilizes negligible resource on your primary server. They asynchronously send writes as they occur on primary server to CX-CS. The VX and FX component of Scout are deployed on your secondary server, as well, to communicate continuously with CX-CS.

Figure 2-30 Scout Environment



ScoutCloud protects data by setting replication between primary server drive/file and secondary server drive/file. The replication process at drive level happens through stages. At the beginning of the replication process, a baseline copy of primary server's drive that you want to protect is created at the secondary server. This step is known as Resyncing (Step I). Data changes during this step are sent to the secondary server Resyncing (Step II). Thereafter, Scout captures and sends only the changes in primary server drive. This stage is known as Differential Sync. This differential sync is a continuous process which is archived through VX agents. Scout supports fast resync, where the replication process starts directly from differential sync instead of replication stages. Unlike drive level replication, file/ folder level replication between primary and secondary server are one time activity, which is archived through FX agents.

For maintenance activities on the primary server or actual failure of the primary server, Scout switches the primary server to secondary server through failover. A failover operation is always followed by a failback operation; i.e., restoring the primary server from the secondary server. Scout uses CDP technology to replicate data, so that it can restore data to any point in time. To ensure the consistency of primary server drive data, the consistent tags/bookmarks are issued at the primary server at periodic intervals of time or on demand. The secondary server can be rolled back to any of the consistency bookmarks to ensure consistency of backup/DR data.

InMage ScoutCloud also supports the storing of Snapshots (exact replica of primary server's drive data as existed in single point in time) on physical or virtual volumes. These snapshots are stored as per consistency bookmarks applied on the secondary server.

Component Flow

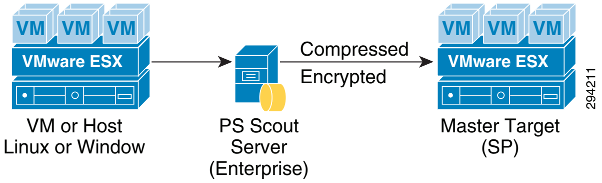
This section includes the following topics:

- Process Server to Master Target, page 2-38
- Process Server to Master Target (Reverse Protection), page 2-38

Process Server to Master Target

Figure 2-31 shows the flow of data from protected servers to the PS server at the Enterprise to the MT at the SP.

Figure 2-31 Data Flow: Protected Servers to the PS Server



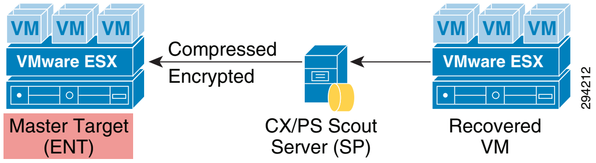
The Agent running on the source-protected servers collects data from the servers as it is created and sends to the local PS server, which would then send the data to the MT server residing at the SP premise were the data will be stored.

The PS server is also responsible for compressing and encrypting the data before sending over to the MT server. It's also capable of caching the data for extended periods of time in any WAN failure scenarios.

Process Server to Master Target (Reverse Protection)

Figure 2-32 shows the data flow from the recovered servers after a failover into the Enterprise data center.

Figure 2-32 Data Flow: Reverse Protection



The data flow is similar to the scenario of protecting a server. The changed data from the recovered server is collected by the Datatap agent on the server and is sent to the CX/PS Scout Server on the SP side. The server compresses and encrypts data and in turn sends the data to the MT server on the Enterprise side. A MT server is required on the Enterprise side for the failback scenario.

Deployment Considerations

This section includes the following topics:

- Journal Sizing, page 2-39
- Storage, page 2-41
- Compression, page 2-43
- Encryption, page 2-44
- Compute, page 2-45

Journal Sizing

Journal volume holds the data changes happening on the production servers. All the changes during a retention period will be stored in the journal volume for recovery purposes. This gives the ability to recover the production servers to any point in time within the retention period.

InMage uses the copy-on-write method to capture changes in to the journal. After the initial copy of data to the target site, InMage tracks the changing blocks on the recovery volumes. The original data that is being written to is copied into the journal for point in time recovery.

Before a write is allowed to a block, copy-on-write moves the original data block to the journal space and updates the original block.

Capacity and performance sizing for journal is a very important consideration. The journal should have correct performance characteristics to handle the total write performance of the data being protected. The journal will have the same amount of writes that will occur at the source protection site. All the changed blocks have to be written into the journal.

Journal Volume Capacity Considerations

The two important considerations for capacity planning of journal are:

- The amount of change rate of the source servers.
- The retention window required.

Retention window is the period during which the changes will be stored. When setting up a CDP configuration, the administrator will create a baseline copy of the data to be protected (effectively the state of the data at initial synchronization), take the average change rate of the data per day into account, determine how many hours, days, or weeks of writes they want to retain, and size the retention log accordingly. The log operates in a circular manner in the sense that it will keep all writes within the defined time period but discard all writes older than that. For example, if the administrator defines a retention window of 3 days, then the retention log will retain all writes for the first 3 days of operation, and then begin to discard writes that are older than 3 days as newer writes are logged.

For recovery purposes, administrators generally want the most recent data. In cases where data corruption was the problem, or where root cause analysis will be performed, one or more older recovery points may be desired, but generally even these points are no older than 24 hours. Many customers establish a retention policy of one to four weeks, balancing the availability of granular recovery points against the size of the retention log. With a retention window of only a couple of days, you may not have access to certain older points, such as a quarterly close, that may be of interest because it may already have aged out of the retention window. In the case of the quarterly close, you probably aren't interested in any of the points around it, just that point which marked the state of the database of record when the quarterly close was completed and before any new transactions from the next quarter came in. It's probably not worth it in terms of storage capacity to extend the size of the retention log out so that older points like this could be retained, but it would be nice if certain older points could be saved without saving all the rest of the data.This is where the concept of sparse retention policies comes into play. The default retention log policy is "retain all writes for x time", but multiple policies with different retention periods could be defined.

For example, four policies might be defined:

- For the most recent 24 hours, keep all writes and bookmarks.
- For data that is between 24 and 48 hours old, keep a recovery point every 4 hours plus all bookmarks.
- For data that is older than 48 hours, keep only bookmarks of a certain type.
- Keep no data longer than 4 weeks.

A policy like that initially keeps all data, but then begins pruning unneeded metadata (for potential recovery points that are being discarded) to reclaim retention log space as data ages. Once the metadata for a particular point is discarded, that point can no longer be retroactively created.

By being able to define different retention period policies within the same CDP timeline, you get the recovery granularity needed for recent data while still being able to save certain key older points for easy access without having to use too much storage.

Journal Volume Performance Considerations

For every write on the Source Production Volume, three I/Os exist on the target side:

- Write to the Replica volume
- Read from the Replica volume
- Write to the Journal volume

The pattern of the IO, therefore, is 1*Read and 2*Write on the target side, which is split between Journal and Replica Volumes. The exact breakdown of IO type is:

- Journal Volume = 1 sequential write
- Replica Volume = 1 random read and 1 random write (Production Volume IO Pattern)

The Journal size is therefore dependent on the retention period that a customer wants to recover their workloads from and should be able to support the write IOPS from the production site of the customer.

Storage

Storage is the main component in the DRaaS System. Proper storage sizing and deployment is very critical for delivering optimized service to customers. The following storage efficiency feature is recommended at the SP recovery site:

- **Thin Provisioning**: Thin provisioning is a good method for optimizing utilization of available storage. It relies on on-demand allocation of blocks of data versus the traditional method of allocating all the blocks up front. This method eliminates all the unused space, which helps avoid poor utilization rates. The best practice is to enable thin provisioning at the storage level or at the hypervisor level to avoid management challenges. In the DRaaS System, as InMage is capable of creating VMs using thin provisioning in the cloud, it is recommended to implement it on the hypervisor layer.

The following storage efficiency features are specific to EMC VNX when using vBlock as the ICS:

- **FAST Cache:** FAST Cache technology is an extension of your DRAM cache where it allocates certain flash drives to serve as FAST Cache. The benefit is that hotter data from applications running inside the VM will be copied to FAST Cache. Hence, these applications will see improved response time and throughput since the I/O is now serviced from flash drives. In DRaaS environments, FAST Cache will be useful during concurrent customer site failovers and during the on-boarding of new customers. In general, FAST Cache should be used in cases where storage performance needs to improve immediately for I/O that is burst-prone in nature.
- **FAST VP:** Data has a lifecycle. As data progresses through its lifecycle, it experiences varying levels of activity. When data is created, it is typically heavily used. As it ages, it is accessed less often. This is often referred to as being temporal in nature. FAST VP is a simple and elegant solution for dynamically matching storage requirements with changes in the frequency of data access. FAST VP segregates disk drives into the following three tiers: **Extreme Performance Tier** — Flash drives; **Performance Tier** — Serial Attached SCSI (SAS) drives for VNX; and **Capacity Tier** — Near-Line SAS (NL-SAS) drives for VNX platforms.
  - You can use FAST VP to aggressively reduce TCO and/or to increase performance. A target workload that requires a large number of Performance Tier drives can be serviced with a mix of tiers and a much lower drive count. In some cases, an almost two-thirds reduction in drive count is achieved. In other cases, performance throughput can double by adding less than 10 percent of a pool's total capacity in flash drives.
  - FAST VP and FAST Cache can be used together to improve storage system performance. Customers with a limited number of flash drives can create FAST Cache and storage pools consisting of performance and capacity drives. For performance, FAST Cache will provide immediate benefits for any burst-prone data, while FAST VP will move warmer data to performance drives and colder data to capacity drives.
  - FAST Cache is storage system aware where storage system resources are not wasted by unnecessarily copying data to FAST Cache if it is already on flash drives. If FAST VP moves a slice of data to the extreme performance tier, FAST Cache will not promote that slice into FAST Cache - even if the FAST Cache criteria is met for promotion.
  - When initially deploying flash drives in a storage system, use them for FAST Cache. FAST Cache will track I/Os smaller than 128 KB and requires multiple cache hits to 64 KB chunks. This will initiate promotions from performance or capacity drives to Flash Cache and as a result, I/O profiles that do not meet this criteria are better served by flash drives in a pool or RAID group.

The following storage efficiency features are specific to NetApp when using FlexPod as an integrated stack within VMDC:

- **Flash Cache:** Flash Cache speeds access to data through real-time intelligent caching of recently read user data and NetApp metadata. It is effective for random read-intensive workloads, including databases, e-mail, and file services. The combination of intelligent caching and NetApp data storage efficiency technologies enables the virtual storage tier, which promotes hot data to performance media in real time without moving the data, allowing you to scale performance and capacity while achieving the highest level of storage efficiency in the industry.
- **Flash Pool:** Flash Pool is a technology that allows flash technology in the form of solid-state disks (SSDs) and traditional hard disk drives (HDDs) to be combined to form a single Data onTap aggregate. When SSD and HDD technologies are combined in a Data onTap aggregate, the NetApp storage system takes advantage of the latency and throughput benefits of SSD while maintaining the mass storage capacity of HDD.
  - A Flash Pool is built from a Data onTap aggregate in a two-step process. Essentially, it is the addition of SSDs into an aggregate to provide a high-bandwidth, low-latency location that is capable of caching random reads and random overwrites. ** The feature does not require a license and works with any NetApp SSDs and one type of HDD per Flash Pool. That is, SSD and SAS performance drives can be combined to make a Flash Pool or SSD and SATA capacity drives can be combined to make a Flash Pool. You cannot combine SSD, SAS, and SATA into a single Flash Pool.
  - As a key component of the NetApp Virtual Storage Tier, Flash Pool offers a real-time, highly efficient implementation of automated storage tiering. Fine-grain promotion of hot data elements, combined with data deduplication and thin cloning, enables optimal performance and optimal use of Flash technology.
- **Deduplication:** NetApp deduplication is an integral part of the NetApp Data onTap operating environment and the WAFL file system, which manages all data on NetApp storage systems. Deduplication works "behind the scenes," regardless of what applications you run or how you access data, and its overhead is low.
  - NetApp deduplication is a key component of NetApp's storage efficiency technologies, which enable users to store the maximum amount of data for the lowest possible cost.
  - NetApp deduplication is a process that can be triggered when a threshold is reached, scheduled to run when it is most convenient, or run as part of an application. It will remove duplicate blocks in a volume or LUN.
- **Steady State Replication***:*
  - FAST VP from EMC
  - Flash Pool from NetApp.
  - During the steady state replication, the target storage will have the information about the I/ O characteristics and about the data blocks.
- **Summary:**
  - Flash Cache and FAST Cache are useful in dealing with unpredicted I/O needs that can be observed during the recovery of multiple customer environments during a disaster.
  - Flash Pool and FAST VP are useful efficiency features which helps the SP to use storage space efficiently during steady state replication scenario. Warmer data gets moved to the faster drives and cold data gets moved to the capacity disks automatically.
  - Deduplication and thin provisioning reduces the total storage foot print required to support customer workloads.

Compression

The efficient way to replicate data from one site to another is to compress the data before it is sent over to the WAN Network. This helps in reducing the WAN bandwidth required for data replication. This can be accomplished by using a dedicated external device or can be done by using the components of InMage.

The advantages of going with both these approaches include:

- Use of external device provides better handling of data compression and management as it will be used only for this functionality. This offloads the load on process server from InMage which does the compression.
- Compression consumes a lot of CPU resources and will effect the ability of the process server in performing other tasks.
- Troubleshooting for events will become easier.

For more information on the utilization of resources when compression is enabled, Refer to Appendix A, “Characterization of Replication Process”.

The InMage process server can perform compression of data. This is a good option for customers who do not want to have a dedicated device for this functionality and this would be an ideal choice for customers who have fewer servers being protected.

This section includes the following topics:

- External Cisco Products, page 2-43
- DR Vendor, page 2-44

External Cisco Products

Network links and WAN circuits can have high latency and/or packet loss as well as limited capacity. WAN optimization devices can be used to maximize the amount of replicated data that can be transmitted over a link.

A WAN Optimization Controller (WOC) is an appliance that can be placed in-line or out-of-path to reduce and optimize the data that is to be transmitted over the WAN. These devices are designed to help mitigate the effects of packet loss, network congestion, and latency while reducing the overall amount of data to be transmitted over the network. In general, the technologies utilized in accomplishing this are TCP acceleration, data deduplication, and compression. WAN and data optimization can occur at varying layers of the OSI stack, whether they be at the network and transport layer, the session, presentation, and application layers, or just to the data (payload) itself.

Cisco wide area application services (WAAS) devices can be used for data optimization. The WAAS system consists of a set of devices called wide area application engines (WAE) that work together to optimize TCP traffic over your network. Cisco WAAS uses a variety of transport flow optimization (TFO) features to optimize TCP traffic intercepted by the WAAS devices. TFO protects communicating devices from negative WAN conditions, such as bandwidth constraints, packet loss, congestion, and retransmission. TFO includes optimization features such as compression, windows scaling, Selective ACK, increased buffering, BIC TCP, and TCP Initial Window Size Maximization.

Cisco WAAS uses Data Redundancy Elimination (DRE) and LZ compression technologies to help reduce the size of data transmitted over the WAN. These compression technologies reduce the size of transmitted data by removing redundant information before sending the shortened data stream over the WAN. By reducing the amount of transferred data, WAAS compression can reduce network utilization and application response times.

When a WAE uses compression to optimize TCP traffic, it replaces repeated data in the stream with a much shorter reference and then sends the shortened data stream out across the WAN. The receiving WAE uses its local redundancy library to reconstruct the data stream before passing it along to the destination. The WAAS compression scheme is based on a shared cache architecture where each WAE involved in compression and decompression shares the same redundancy library. When the cache that stores the redundancy library on a WAE becomes full, WAAS uses a FIFO algorithm (first in, first out) to discard old data and make room for new.

DR Vendor

InMage supports compression of data before it sends the data over to the WAN. This helps in reducing the need for high WAN bandwidth to carry data traffic. WAN optimization features like data compression will help customers to achieve low RPOs.

InMage collects data changes from production servers in real time, and places them in memory before they are written to disk. They changes are sent to a software appliance called the process server and then transferred to the secondary site. The server offloads compute intensive tasks from the production systems, such as compression, encryption, WAN acceleration, and consolidated bandwidth management.

The process server does the compression of data and sends the data over to the MT server residing on the SP's cloud.

Customers who do not want to bear additional cost for the WAN optimization devices can leverage InMage for achieving low RPOs.

Encryption

Encryption of data-in-transit and data-at-rest is the best method to enforce the security and privacy of data, regardless of where it resides. Data-in-transit encryption is necessary to keep the data secure while in transit. The network connection between sites must be secure and the data must be protected. The use of IPsec or SSL to encrypt WAN connections ensures that no visibility occurs at the packet level if any of the datagrams are intercepted in transit.

Encryption of data-in-transit between the sites can be accomplished in two ways:

- **InMage technology** is capable of encrypting data in flight, the CX-PS server on the enterprise encrypts the data before it sends it over to MT over the WAN. Enabling the encryption will secure the data transmission between CX-PS and a secondary server. Since the encryption is performed on the CX-PS server, any performance impact will be limited to the CX-PS server.
- The other option is to use **Cisco ASA** for encrypting data between the Enterprise and the SP's data centers. The Cisco ASA 55xx Series is a purpose-built platform that combines superior security and VPN services for enterprise applications. The Cisco ASA 55xx Series enables customization for specific deployment environments and options, with special product editions for secure remote access (SSL/IPSec VPN).

The Cisco ASA 55xx Series SSL/IPsec VPN Edition uses network-aware IPsec site-to-site VPN capabilities. This allows customers to securely extend their networks across low-cost Internet connections to the service provider cloud.

Encryption of data-at-rest can add further security to the storage environment on the cloud SP's data center. Any external key manager can be used in conjunction with SAN fabrics and storage arrays to secure data-at-rest.

Compute

This section includes the following topic:

- Oversubscription, page 2-45

Oversubscription

DRaaS utilizes shared resources on the recovery site. Since resources at failover site sit idle most of the time, DR enables high over-subscription ratios, making it ideal for cloud environments.

The SP can have fewer compute resources compared to the customer's production environments. The compute within the SP cloud is based on Cisco UCS servers, which can be rapidly deployed with the help of the service profiles to meet any unexpected or rare scenario where all the customers fail over to the cloud. In this scenario, new UCS servers can be deployed and added to the existing compute clusters for additional compute resource needs.

Every server that is provisioned in the Cisco UCS is specified by a service profile, which is a software definition of a server and its LAN and SAN network connectivity. In other words, a service profile defines a single server and its storage and networking characteristics. Service profiles are stored in theCisco UCS 6xxx Series Fabric Interconnects. When a service profile is deployed to a server, UCS Manager automatically configures the server, adapters, fabric extenders, and fabric interconnects to match the configuration specified in the service profile. This automation of device configuration reduces the number of manual steps required to configure servers, network interface cards (NICs), host bus adapters (HBAs), and LAN and SAN switches.

Key Findings

This section includes the following topics:

- Concurrency, page 2-45
- Limitations, page 2-45

Concurrency

A maximum of 40 VMDKs/RDMs per MT based on best practice. Refer to "Implementation Best Practices" section on page 5-8 for details.

Maximum supported change rate per day per Scout Process Server is 1TB. Deploy additional process servers to support additional change rate. Refer to Table 3-1 (Scout Server Storage and Compute Implementation, page 3-6) for details.

InMage relies on vCenter limits for throttling and concurrent operations. Refer to http://www.vmware.com/pdf/vsphere5/r51/vsphere-51-configuration-maximums.pdf for additional details.

Limitations

The following are current limitations for the DRaaS 1.0 System:

- For recover later plans, outside of the Multi-Tenant Portal, no mechanism exists to view VMs included in a recovery plan. The assumption is the recovery plan should map exactly to a protection plan. If the protection plan changes, a new corresponding recovery plan should be created.
- Storage vMotion of MT is not supported. Compute vMotion is supported.
- MT OS type needs to match the OS of the protected servers. Also a single protection plan can only span across a single MT. It is not possible to migrate a protection plan from one MT to another.
- Offline sync requires a deployment of MT in the primary enterprise data center. MT has to be a VM that gets shipped to a SP's secondary data center.
- Smallest configuration retention window from the vContinuum server is 1 day.
- The RX portal can have up to a 10 minute lag in display. Resource consumption for storage and compute is not available from the RX portal.
- It is not possible to set VM replication priority within a protection group. It is possible to limit available bandwidth at a MT level.
- Infrastructure masking for tenants resources relies on VMware vCenter permissions.