



REFERENCE ARCHITECTURE

Rubrik and VMware Cloud Foundation

TABLE OF CONTENTS

3 INTRODUCTION

3 AUDIENCE

4 SOLUTION OVERVIEW

- 4 VMware vSphere Storage APIs - Data Protection
- 5 Rubrik's Approach to Adaptive Consistency
- 5 Recovery Methods
 - 6 Instant Recovery
 - 7 Live Mount
 - 8 Export
 - 9 File-Level Recovery
 - Direct Restore
 - Restore by Download
- 10 Declarative Policies and SLA Domains
 - 10 SLA Domains
 - 13 Assigning SLA Domains
 - 13 Protection Overview
 - Performance and Scalability
 - Backup Processes

15 ARCHITECTURAL OVERVIEW

- 16 VMware Cloud Foundation Architecture
 - 16 SDDC Manager
 - 17 Workload Domains
 - Management Workload Domain
 - Compute Workload Domains

- 19 Protecting and Restoring VMware Cloud Foundation

- 19 Compute Workload Domains

- 20 Management Workload Domain Components

- Platform Service Controllers and vCenter Server Recoverability

- SDDC Manager Recoverability

- NSX Recoverability

- vRealize Log Insight Recoverability

- vSAN Recoverability

23 OPERATIONAL OVERVIEW

- 23 Use Cases
 - 23 Test and Development
 - 24 Migrate to a New SDDC and Maintain Backup History
 - 24 CloudOut (Archival)
 - 25 CloudOn (Instantiation)
- 26 Day 0 - 2 Ops with API / Integrations Use Cases
 - 26 Automatic Policy-Based Data Protection with New Compute Workload Domains
 - 27 Integration with vRealize Automation for Complete VM Lifecycle Management

28 CONCLUSION

28 ABOUT THE AUTHORS

29 APPENDIX A - EXAMPLE INVENTORY AND CONFIGURATION GATHERING SCRIPT

INTRODUCTION

VMware Cloud Foundation is an integrated software platform that provides a complete set of software-defined services for compute (VMware vSphere), storage (VMware vSAN), networking (VMware NSX for vSphere and NSX-T), and cloud management (VMware vRealize Suite).

VMware Cloud Foundation provides fully automated deployment and lifecycle management of the VMware Software Defined Data Center (SDDC), and is designed to be deployed on premise or consumed as a service in the public cloud.

Cloud Foundation is engineered to be simple, unlocking greater agility and productivity by eliminating the operational bottlenecks associated with traditional administrative silos of legacy infrastructure. It accomplishes this by delivering:

- **Integrated Stack**—Engineered integration into a single solution for the entire software-defined stack with guaranteed interoperability. No more complex interoperability matrixes to deal with.
- **Software-Defined**—Infrastructure is entirely defined in software and inherently highly dynamic, scalable, and hardware independent. Underlying physical hardware is completely abstracted into logical pools that can be flexibly allocated to individual [JS1] tenants or applications.
- **Automated Deployment of a Standardized Architecture**—The Cloud Foundation initial deployment is fully automated through the use of the VMware Cloud Builder appliance. Cloud Foundation deployments are deployed in accordance with the standardized architecture defined in [VMware's Validated Designs \("blueprint" documentation for building the VMware SDDC\)](#). Cloud Builder deployment validation and automation capabilities ensure quick and repeatable deployments while eliminating risk of faulty configurations.
- **Automated Resource Provisioning**—Cloud Foundation creates and maintains logical infrastructure pools of compute, storage and network resources, making it easy to scale as business needs grow.
- **Automated Lifecycle Management**—Cloud Foundation includes unique lifecycle management (automated patching and upgrades) for all the products that it deploys. Validated and tested upgrade bundles are downloaded by SDDC Manager and provide intuitive, click button deployment of patches and upgrades. Common operational tasks such as password rotation and SSL certificate creation and replacement are automated through workflows in SDDC Manager.

Pairing Rubrik with VMware Cloud Foundations allows for automated data protection of both the management and compute workloads deployed within VCF, ensuring resources are protected and available in the event disaster strikes. Rubrik's simplistic and automated processes work together with VCF lifecycle management to truly provide a holistic data management solution for customers.

AUDIENCE

This reference architecture is intended to provide CTOs, solutions architects, and administrators with information about the architecture, implementation, and benefits of an integrated Rubrik and VMware Cloud Foundation solution.

For the remainder of this document, "virtual machines" will be referred to as "VMs," "disaster recovery" as "DR," and "VMware Cloud Foundation" as VCF.

SOLUTION OVERVIEW

VMware Cloud Foundation is the unified Software-Defined Data Center (SDDC) platform that brings together vSphere, vSAN, vRealize, and NSX into a natively integrated stack to deliver enterprise-ready cloud infrastructure for both private and public cloud.

Pairing Rubrik with VCF brings automated data protection to the SDDC and provides a variety of use cases and methods to ensure that both the management and workload VMs are protected and available. Utilizing an automated policy-driven approach through the assignment of Rubrik SLA Domains, consumers of Cloud Foundation not only gain simplicity around the management of the underlying infrastructure, but can be rest assured that the workloads consuming cloud resources can be restored with near-zero RTOs, both at a file and image level.

Rubrik and VMware Cloud Foundation work in harmony to provide a true cloud infrastructure platform, delivering software-defined services built around compute, storage, network, security, and data management / protection.

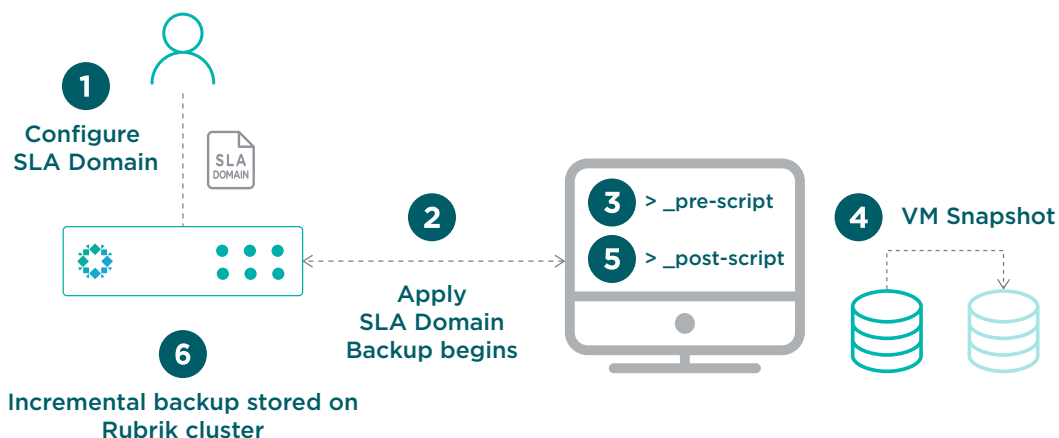
VMWARE VSPHERE STORAGE APIS - DATA PROTECTION

The vSphere Storage APIs for Data Protection (VADP) is a framework put forth by VMware that allows products to request and retrieve VM data in a consistent manner. Rubrik fully utilizes VADP in order to backup vSphere VMs without the use of third-party agents, offloading the backup processing of the individual VMs to the ESXi hosts and the Rubrik cluster. Leveraging the snapshot capabilities within vSphere, VADP and Rubrik are able to perform backups non-disruptively, no longer requiring extended backup windows or downtime. Ultimately, VADP offload reduces production overhead, allowing organizations to run more VMs per host, in turn reducing the overall costs associated.

In the case of Rubrik and VMware Cloud Foundation, leveraging the vSphere Storage APIs for Data Protection also ensures that architectural decisions around storage are largely made moot. Regardless of the underlying storage decisions made with VMware vSAN, the usage of VADP abstracts the nuances present with the selected datastore storage provider.

When data is requested by the Rubrik cluster, the vSphere API layer is used to negotiate with the ESXi hypervisor hosting the protected workload to retrieve the needed data from the datastore and transmit it back to the Rubrik cluster. Additionally, the individual node choice within the Rubrik cluster is handled in an automated fashion in which the CDM software determines the primary candidate(s) for establishing a session and receiving data.

At a high level, the successful request to begin a backup of a VM kicks off a workflow requesting that the ESXi host running that particular workload initiate a VMware snapshot. This snapshot action is intended to redirect storage IO to a snapshot disk, thus freeing the hypervisor's lock on the underlying base disk(s). Subsequent actions are focused on retrieving the data to be parsed and stored by the requesting Rubrik cluster. Once finished, the VMware snapshot is consolidated (removed), and storage IO returns to the base disk(s). While there are other decision points available in the workflow, such as adaptive consistency and pre- and post-scripts, the basic flow has been represented.



It is worth noting that the process of negotiation, session instantiation, and data transmission is secured using an SSL encrypted Network Block Device protocol and SSL encryption (NBDSSL) transport mode.

RUBRIK'S APPROACH TO ADAPTIVE CONSISTENCY

When dealing with consistency within recovery points, backups can be classified into three broad categories: inconsistent, crash consistent, and application consistent.

Inconsistent backups are taken by copying data only located on the disks of the VMs. In-memory changes or any transactions in progress are not captured within an inconsistent recovery point. Inconsistent backups may only be suited for the most basic of workloads and are not recommended for high-transactional, complex systems.

When crash-consistent backups are taken, the complete state of the VM is captured at a given point in time, resulting in an exact duplicate of the data from which the backup began. Although crash-consistent backups are sufficient for most modern workloads, in-memory and transactions in progress are still not captured.

Application-consistent backups, like crash-consistent, capture all of the VM's data at a given point in time. The difference, however, is that application-consistent backups will pause and wait for applications within the system to flush I/O operations and complete any transactions in progress.

Rubrik takes a stepped approach to perform application-consistent backups of VMs. First, Rubrik polls the VM to see if the Rubrik Backup Service (RBS) is listening. RBS is a lightweight service that can orchestrate application-consistent backups by utilizing an included custom VSS provider. This VSS provider enables functionality around application-consistent features such as flushing SQL Server and Exchange logs along with waiting for in-memory transactions to complete. Rubrik can also enable the automatic installation of the RBS service if it is not found, but this functionality is not enabled by default.

Note: When RBS is installed on VMs, it is utilized only for application-consistent, VSS purposes during a vSphere backup. The processes around the movement of data remain the same as if RBS wasn't installed, utilizing VADP and NBDSSL.

If RBS is not found or cannot be installed, Rubrik will push an ephemeral agent containing a custom VSS provider into the VMs through the vSphere Virtual Infrastructure eXtension (VIX) APIs during the backup process. VIX is an API developed by VMware that allows Rubrik to leverage and perform guest management operations within the VMs being backed up. After the ephemeral agent has been deployed, Rubrik will first attempt to perform an application-consistent backup. If an application-consistent backup is not possible, Rubrik will create a notification around the event and will proceed with processing a crash-consistent backup.

Rubrik's approach to adaptive consistency marries the options of having flexibility within the platform to perform application-consistent backups with the simplicity of providing a solution that is easy to use with minimal configuration. Rubrik will automatically and adaptively conform to the best path possible for the VM as it pertains to data consistency, ensuring that even if the optimal method is not chosen, the backup is still performed with the properly associated notifications.

RECOVERY METHODS

Rubrik provides a variety of methods to recover VMs and restore protected data. Recoverable data within the Rubrik CDM platform can exist in three locations:

- Local snapshots
- Replicated snapshots
- Archived snapshots.

Note: While the term snapshot exists both within the vSphere and Rubrik platform, they represent entirely different underlying technologies. A vSphere snapshot exists within the source production environment, while a Rubrik snapshot always represents a point-in-time copy of your production data located within the Rubrik CDM. The remainder of this section references Rubrik snapshots.

When snapshot data exists in a local snapshot and in an archived snapshot, the Rubrik cluster always uses the local snapshot to recover a VM or to restore data. By using the local snapshot, the Rubrik cluster reduces network impact and eliminates any archival data recovery charges associated with a recovery operation or a restore operation.

INSTANT RECOVERY

Rubrik's Instant Recovery can be used to recover VMs that are no longer functioning correctly because of:

- Corruption or malware
- Accidental deletion
- Any other service disruption

This functionality allows mounting restored VMs data directly off the Rubrik system, thus reducing the recovery time.

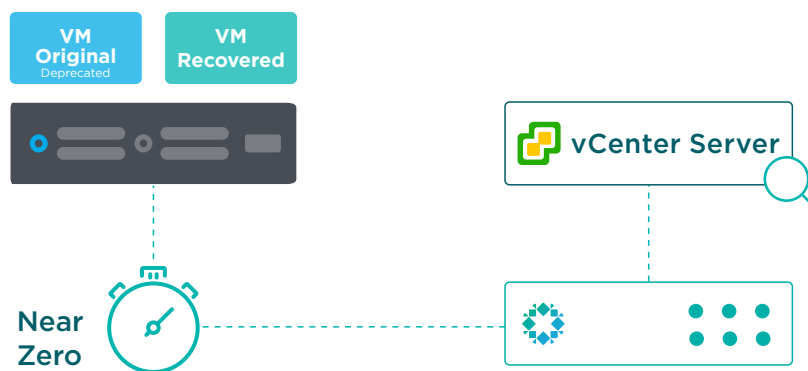
Let's visualize the Instant Recovery workflow:

The process first begins by selecting the VM, snapshot date, and recovery host. You may choose to remove a virtual network device if any networking changes or issues would prevent the VM from successfully powering on. This methodology also enables validation of certain services after recovery but before restoring the service.

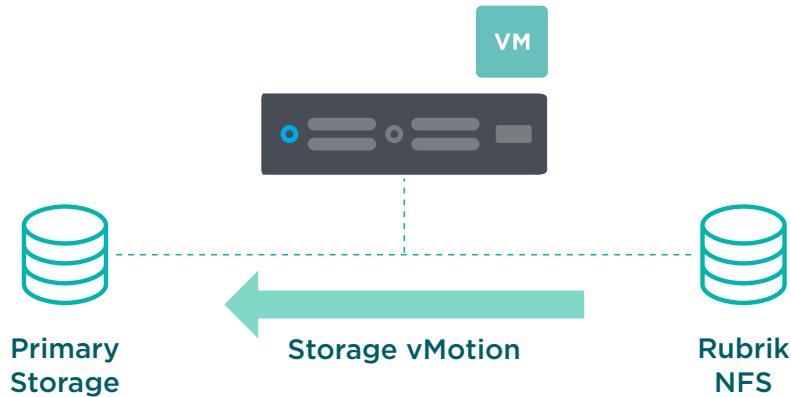
Additionally, you may select to preserve the VM-managed object ID (MoRef). This is a managed object ID, which is applicable to vSphere VMs. It will ensure that the VM is recovered using the same MoRef as a part of VM linking, rather than it being recovered as a new object. This method can be important for preserving workflows built around this VM. See the [VM Linking](#) section for more information.

At this point, the Rubrik system presents itself as an [NFS v3 datastore](#) to ESXi. If the original VM still exists within the vCenter Server inventory, it will be deprecated (renamed) before the process continues.

Rubrik coordinates the addition of the newly recovered VM into the vCenter Server inventory. A new copy of the VM running on Rubrik is presented and powered on and services resume.



Post-recovery, users can utilize VMware’s Storage vMotion to migrate the workload back to the primary storage array.



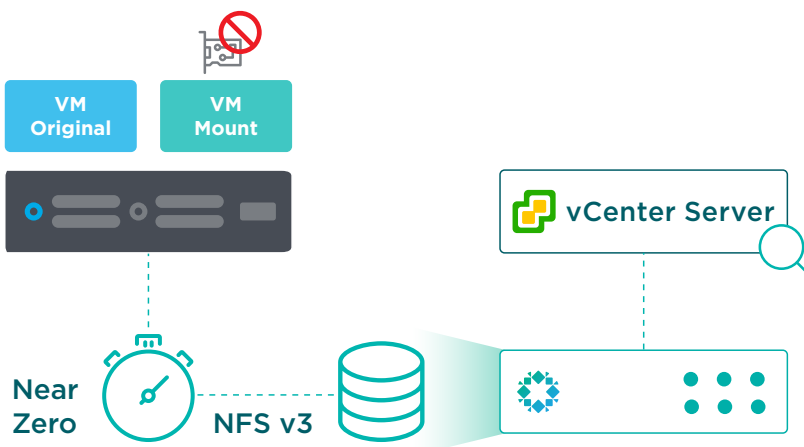
Ultimately, Rubrik serves as a storage endpoint to recover as many vSphere VMs as needed, thus eliminating the complexity and time wasted in transferring data back into the production system. This functionality provides a near-zero recovery time and restores user access near instantly.

During the process, messages about the recovery status appear in the Notifications section of the Rubrik UI. The Rubrik cluster records the final result of the task in the Activities Log, available via the Rubrik UI.

The instantly recovered VM derives protection from parent objects. When the recovered VM does not derive protection from any parent objects, add it to an SLA Domain. To protect it using the same SLA rules and policies as the source VM, add the recovered VM to the original SLA Domain or to another SLA Domain. With VM linking, the new VM is linked with the old VM, which preserves the entire snapshot history.

LIVE MOUNT

Like Instant Recovery, Rubrik becomes an NFS v3 datastore from vSphere ESXi hypervisor perspective. Instead of deprecating the original VM, a VM similar to the original is created from the point in time selected but with a trailing date timestamp appending the VM name. The original VM is not altered. Additionally, in order to avoid IP or MAC address conflicts, the Live Mount VM has its NIC disabled by default.



This functionality appeals to application owners and operations teams in order to conduct:

- Functional or regression testing
- Application development
- Software release testing (upgrade the actual applications)

Build isolated environments and leverage the Live Mount feature to instantiate an identical environment in moments. Test VMware Tools or hardware version upgrade, failure scenario, and other use cases using your backup storage. When done, simply throw it away. The production VM, along with any of its associated backups, remain unscathed.

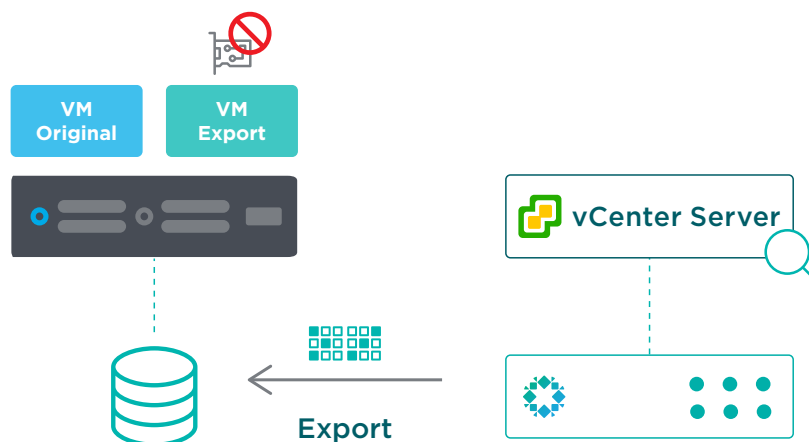
No additional configurations are needed on the hypervisor side for Live Mount functionality to work. Simply provide a service account with the documented permissions required for the type of virtualization environment. Rubrik automates the entire process. VMs of any size can be recovered in the amount of time it takes for the OS to boot. Imagine having the ability to spin up an 8 TB VM in under 2 minutes so that a recovery point can be validated by an administrator or application owner.

EXPORT

While Instant Recovery and Live Mount utilize the Rubrik resources to facilitate a fast, near-zero RTO recovery, there may be times when a traditional restore method is preferred. The Export recovery method within Rubrik provides just this, recovering the VM by transferring data from the Rubrik cluster back to a production environment.

Export creates a new VM within vCenter from a point-in-time copy of the source VM located on the Rubrik cluster, allowing a new ESXi host to be chosen along with the selection of any target datastore to host the recovered VM.

The Rubrik cluster assigns a new name to the recovered VM and powers it on. However, as a safeguard, the recovered VM is not automatically connected to a network. This enables validation of certain services or applications after the recovery but before completely restoring access to the service.



This functionality appeals to application owners and operations teams in order to conduct:

- Recovery after a primary storage failure to another location
- Restoration of services to a different host/cluster
- Functional or regression testing requiring the VM to be available on a long-term basis

Coupling Instant Recovery together with VMware's Storage vMotion will almost always result in lower recovery times, but Export is an option for those with a use case revolving around traditional restores.

FILE-LEVEL RECOVERY

The Rubrik cluster provides file-level restore (FLR) of files and folders from any local snapshot, replica, or archival snapshot that was successfully indexed.

Note: When performing file-level restores from an archived snapshot, either on-prem or in the cloud, only those required blocks comprising the file are retrieved. Rubrik eliminates the need to download entire virtual disks (.vmdk) by storing the metadata with the archived snapshots. This saves organizations high egress charges associated with the chosen cloud provider.

To restore a file or folder, search for it by name across all local snapshots. You can also browse for the file or folder on a selected snapshot.

Files and folders may be restored directly to the source system or downloaded to the local workstation.

DIRECT RESTORE

For supported Windows and Linux guest operating systems, the Rubrik cluster can restore files and folders directly to the source file system.

When restoring from a snapshot of a supported guest operating system, the Rubrik UI provides the option to restore a file or folder directly to the source file system. When this option is selected, the Rubrik UI provides a choice to overwrite the source file or folder, or to restore the file or folder to another location.

A restored file or folder inherits the access control of the parent folder and the same owner as the parent folder. The restored file or folder retains the modification time (mtime) of the source file or folder at the time of the snapshot.

To successfully restore directly to the source file system, the Rubrik cluster must be provided the following information:

- Resolvable hostname or IP address of the authentication server
- Username of an account with administrator privileges for the target
- Password for the account

When the Rubrik cluster has previously accepted the service credentials of a guest operating system, the restore job does not require additional credential information. This feature requires that the Rubrik cluster has successfully used the service credentials for at least one backup prior to the restore task. Otherwise, the credentials can be provided through the Restore File dialog during the restore task.

RESTORE BY DOWNLOAD

The Rubrik cluster generates download links to use for file-level restore (FLR) of files and folders from any local snapshot, replica, or archival snapshot that was successfully indexed. The guest OS of the source VM must have a current version of VMware Tools running to enable successful indexing.

Restore a file from a data protection object through the Rubrik cluster Rubrik UI. Once the file is selected, the Rubrik cluster processes the request and provides a link for download of the file.

When restoring a folder, the Rubrik cluster generates a `.ZIP` file containing the folder and all its contents. The `.ZIP` file retains the hierarchy of the selected folder. The Rubrik cluster provides a link for downloading the `.ZIP` file.

DECLARATIVE POLICIES AND SLA DOMAINS

Traditional architecture has long been ruled by the imperative operational model. Historically, administrators have taken some piece of infrastructure and then told it exactly what to do to meet the desired end state. In terms of data lifecycle management, this translates to defining what objects to protect, target destinations, creation and expiration schedules, storage requirements, and so on. Each job requires a non-trivial amount of daily management to function. If there are issues with the job, an administrator must triage the job to determine where the failure occurred (along with re-running the job at a later date).

One of the most positive and impactful shifts in enterprise architecture has been the move towards the declarative model. This refers to the ability to express business needs directly to the systems that run applications with the intent of allowing an intelligent fabric of components to make real-time decisions on your behalf.

The declarative model allows technical professionals to plug in their desired state for an object – in this case, the data protection policy for VM workloads – into a policy engine. This engine is elegantly simple because all of the imperative details are abstracted away and handled by an incredibly smart, scale-out system. The resulting input fields are reduced to:

- The Recovery Point Objective (RPO) requirement
- Retention periods for the aforementioned RPOs
- Any archive targets, if desired
- Any replication targets for near-zero RTO requirements, if desired

Policy is logically assigned to vSphere objects: VMs, folders, data centers, clusters, or even entire vCenter Servers, as well as constructs outside of vSphere, such as physical workloads, SQL databases, etc. Any of the “jobs,” per se, are completely abstracted away by the system. The declarative policy engine funnels your RPO, RTO, availability, and replication requirements into system-level activities. This is where the true value of the system resides – the ability to control end-to-end ingest, placement, and archive for all protected pieces of data. Just set a policy and allow the system to do all of the heavy lifting. This is how the technology industry as a whole is going to tackle the ever-increasing demands for doing more with less, faster and more efficiently.

As an example, imagine you have invited someone over to your house. In order for the person to arrive at your home, you must give exact directions -- “start by going straight down Main Street, then right at the In-N-Out Burger, ensure to follow the stop light instructions at the intersection of 1st Ave and A Street. My house is the ninth house on the left past that intersection.” This is the imperative model of thinking. Alternatively using a declarative model, I could say “my address is 16 National Ave; input it into a GPS app -- it will navigate you using the best route.”

Rubrik is firmly rooted in the declarative approach; as an administrator, you simply define the desired end state (RPO, retention, replication, archival, etc.) and allow the intelligent software to make it reality. In essence, govern infrastructure and applications using declarative policy rather than imperative jobs.

SLA DOMAINS

Rubrik orchestrates the movement of data from initial ingest and propagation of that data to other data locations, such as replicating to remote clusters or Rubrik Cloud Cluster, as well as data archival. A single SLA policy is used to dictate all data lifecycle specifications, and the data control plane does the rest.

For this section, an example SLA policy is:

- Take a backup:
 - Run a snapshot every 4 hours and retain hourly backups for a day
 - Run a snapshot every month and retain monthly backups for 7 years

Create SLA Domain

SLA Domain Name
VCF_VMs

Advanced Configuration

Service Level Agreement

Choose how often we take snapshots and the length of time we keep them.

| Take Snapshots: | | Keep Snapshots: |
|---------------------------|-------------------------|------------------|
| Every (Hours) <u>4</u> | | <u>1</u> Days ▾ |
| Every (Days) _____ | | _____ Days ▾ |
| Every (Weeks) On ▾ | | _____ Weeks ▾ |
| Every (Months) On | | |
| <u>1</u> | Last day of the month ▾ | <u>7</u> Years ▾ |
| Every (Quarters) On ▾ | Begin Quarter in ▾ | _____ Quarters ▾ |
| Every (Years) On ▾ | Begin Year in ▾ | _____ Years ▾ |

Local retention set to 30 days.

Snapshot Window

Take snapshots from: _____ : _____ ▾ to _____ : _____ ▾

Take first full between: First Opportunity ▾ at _____ : _____ ▾

[Remote Settings](#)

Cancel Create

- Archive to Amazon S3 after 30 days
- Replicate data to another Rubrik cluster and retain for 45 days

Create SLA Domain

Remote Storage Configuration

Retention On Brik

0 30 days 6 years 364 days

Archival

S3:rubrik-tm-s3-or Enable Instant Archive ⓘ

Archival starts after **30 days** and is retained on the archival location for **6 years 334 days**.

Replication

tmazusw2

0 45 days 6 years 364 days

Replication starts immediately, and is retained for **45 days**.

[SLA Domain Creation](#)

Cancel
Create

Data is ingested and retained according to the frequency specified in the SLA policy. The example policy is configured to store 30 days of data within the Rubrik cluster. Once that period has elapsed, data is archived to another location for long-term retention. In this case, data is archived to Amazon S3 for another 6 years and 334 days. There is no need for an administrator to manage, prune, or validate that data has been archived; these activities are all handled natively by Rubrik to reflect how they were expressed in the SLA.

The policy also specifies to replicate data from one Rubrik instance to another. For example, a remote office/branch office (ROBO) may replicate workloads into the main data center using Rubrik or a primary site may replicate to a DR site. Eliminate configuring and managing this functionality at the storage layer. Apply policy-based management to workloads and stop babysitting data residing across multiple data centers.

Note: Rubrik provides three built-in SLA Domains by default—each representing a set level of protection:

- Gold (highest protection)
- Silver (medium protection)
- Bronze (lowest protection)

Administrators may choose to use the built-in SLA Domains or to create additional SLA Domains.

Regardless of where the data is archived, Rubrik ensures instant accessibility of data with real-time predictive search. Metadata is included in the archive to ensure the most cost-efficient way to recover data by removing the need for recovering full backups from archive before restoring. This provides the ability to recover archived data at a snapshot or file-level selectively without having to download the entire workload to restore a single file and reduces egress charges.

ASSIGNING SLA DOMAINS

Once the policy has been created, provide protection for a VM by assigning an SLA Domain.

A VM can be protected by assigning an SLA Domain setting individually to the VM. A VM can also be protected by deriving an SLA Domain setting through automatic protection.

Automatic protection occurs in one of the following ways:

- An administrator assigns an SLA Domain to an object that contains the VM.
- An administrator moves the VM into the hierarchy of an object that is assigned to an SLA Domain.

This means that VMs will be protected through inheritance of the SLA policy assigned to a parent object. If the vCenter Server or a folder has an SLA assigned to it, the VM underneath will automatically inherit the policy. The data control plane detects the newly added VM and automatically applies a protection policy, eliminating the need for any manual administrator interaction. This resolves the common issue of new workloads being brought online and going days or weeks without being protected.

In the event that an SLA policy has been assigned to an individual VM that auto-inherits the policy from a high-level object, conflict resolution occurs. When a conflict is detected, the Rubrik cluster opens the SLA Conflicts dialog box to permit the conflict to be resolved.

In addition to overriding SLA policies, if desired, inheritance may also be blocked by applying a “Do Not Protect” policy at the object level.

SLA policies may be hierarchically assigned to:

- vCenter Server
- Clusters
- Folders
- ESXi hosts
- VMs
- Tags

Once the policy is assigned, Rubrik will ensure adherence to user-defined policies such as frequency, retention, archival, etc. as described above. All manual configuration is eliminated by the [data control plane](#), which applies intelligent algorithms to ensure efficiency and performance for the entire backup workload. These intelligent algorithms assist with balancing the workload as more VMs are created and added into the system. The automatic scheduling of tasks ensures that all workloads are evenly distributed across the Rubrik cluster, preventing cluster resource contention.

PROTECTION OVERVIEW

Rubrik provides backup protection for VMs by combining native vSphere snapshot technology with the fast and scalable converged data management platform of the Rubrik cluster.

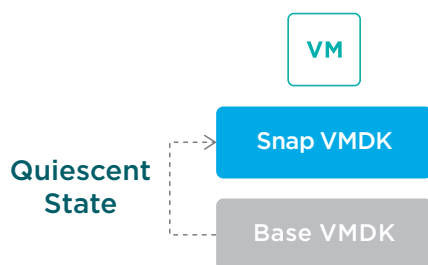
PERFORMANCE AND SCALABILITY

The Rubrik cluster provides a high-performance, highly-scalable integration with vSphere APIs for Data Protection to backup VMs hosted on ESXi hypervisors.

By efficient use of vSphere APIs for Data Protection calls and by providing very fast data ingestion, the Rubrik cluster minimizes the time that a VM is quiescent during a backup. This reduces and, in most cases, eliminates the application time-outs caused by many other backup products. The time that a VM is quiescent, sometimes referred to as VM stun or application stun, is the time between the following:

- The point where execution of the VM is paused, at an instruction boundary, and all in-flight disk input/output operations are completed
- The point where execution resumes

The period a VM is quiescent is very brief--just long enough to create a snapshot. The VM does not remain quiescent during the processing and ingestion of the snapshot data.



To help minimize the time that a VM is quiescent, the Rubrik cluster maintains multiple concurrent connections with a vSphere environment and opens five threads for each ESXi host in that environment.

The Rubrik cluster also efficiently uses the 10 Gigabit Ethernet connection to the vSphere environment. It provides a very high rate of data ingestion to the flash-based write cache that is the initial storage of the Rubrik cluster.

The result is an extremely short time that a VM is quiesced.

For best performance, use a 10 Gigabit Ethernet connection between the Rubrik cluster and the vSphere environment. Also, for replication, it is recommended to provide a 10 Gigabit Ethernet connection between the source Rubrik cluster and the target Rubrik cluster.

The Rubrik cluster uses a distributed task scheduler that permits the Rubrik cluster to schedule tasks to run on any node and on multiple nodes as needed. Since the distributed task scheduler can seamlessly schedule tasks on all available nodes and across multiple nodes, adding nodes to a Rubrik cluster further increases ingestion and processing efficiency.

BACKUP PROCESSES

A Rubrik cluster backs up a VM by using vSphere APIs for Data Protection to create a snapshot of the VM. When a Rubrik cluster begins protecting a VM, the Rubrik cluster starts by creating a first full snapshot of it. This first full snapshot is a complete backup of the VM.

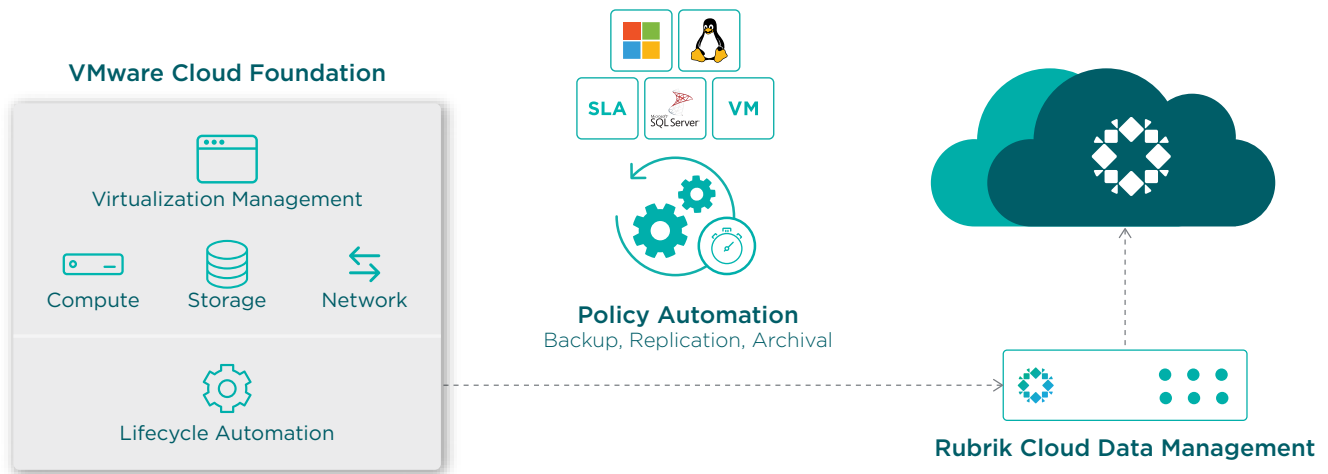
After the first full snapshot, the Rubrik cluster continues protection by creating incremental snapshots based on the change information provided by change block tracking (CBT). The Rubrik cluster creates each incremental snapshot very quickly because the snapshot only includes the data blocks that have changed since the last snapshot.

The vSphere environment transmits the snapshot data to the Rubrik cluster using the most efficient available transport mode. Normally, the vSphere environment uses the NBDSSL transport mode. The high efficiency of the Rubrik cluster eliminates data bottlenecks, allowing the NBDSSL transport mode to provide data transmission rates that minimize the time that a VM is quiescent.

ARCHITECTURAL OVERVIEW

Rubrik is a vendor-agnostic platform that is built on an API-first architecture. The SLA policy is the heart of every Rubrik configuration. Rubrik reduces daily operational management, providing a step-function change in simplicity by enabling a single-policy engine to orchestrate SLAs across the entire data lifecycle. SLA policies can be applied anywhere in the vSphere hierarchy stack: vCenter Server, the cluster, host, folder, or VM levels. The Rubrik cluster provides a variety of methods to recover VMs and restore protected data. Rubrik recovers VMs and restores data by using snapshots, replicas, and archival snapshots.

End-to-end data management is provided by Rubrik for all applications running on vSphere. Users can securely access data instantly, automate protection policies, and orchestrate data across their VMware environments.



The following details a few requirements for this integration described in the reference architecture:

- VMware Cloud Foundation 3.7.x
 - VMware vSphere 6.7 or later
 - » VMware ESXi 6.7 or later
 - » VMware vCenter Server 6.7 or later
 - » Minimum of virtual hardware version 7 for CBT
 - Primary Storage
 - » vSAN 6.7 or later
 - vCenter Server privileges
 - » It is recommended that a custom vCenter role be created with only the minimum required privileges.
 - VMware Tools
 - » The Rubrik cluster requires the current version of VMware Tools to perform administrative operations and enable application-consistent snapshots.
 - » If VMware Tools is out of date, the backup will proceed but may not be application consistent.

- Ethernet (IP) Network
 - A VMkernel port is required for NFS if using Live Mount or Instant Recovery functionality.
 - Additionally, a separate VMkernel network may be configured for Rubrik data traffic.
- Rubrik Cloud Data Management (CDM) 5.0 or later

Additionally, the following assumptions have been made in the writing of this document:

- Workloads are supported by VMware vSphere.
- Workloads are using supported versions of their operating system and application release(s).
- 10 GbE network connectivity exists between the ESXi Host(s) and Rubrik Cluster.
- 4 Node Rubrik Cluster
- 4 Node VCF Management Cluster

Lastly, a few constraints since Rubrik cannot protect data that exists on any of the following using native vSphere integration through APIs:

- VMDKs that are set to Independent-Persistent mode or to Independent-Nonpersistent mode
- Network drives that are mounted on the file system of a protected VM
- Any VM for which the Rubrik cluster does not have snapshot creation permission because of settings on the VM or on a vSphere folder that contains the VM
- Any VM data that resides on raw disk mappings (RDMs), where the compatibility mode of the RDMs is set to Physical

That being said, these constraints apply to protection using native vSphere APIs but can be backed up using RBS.

These requirements and assumptions should be taken into account for the remainder of the document.

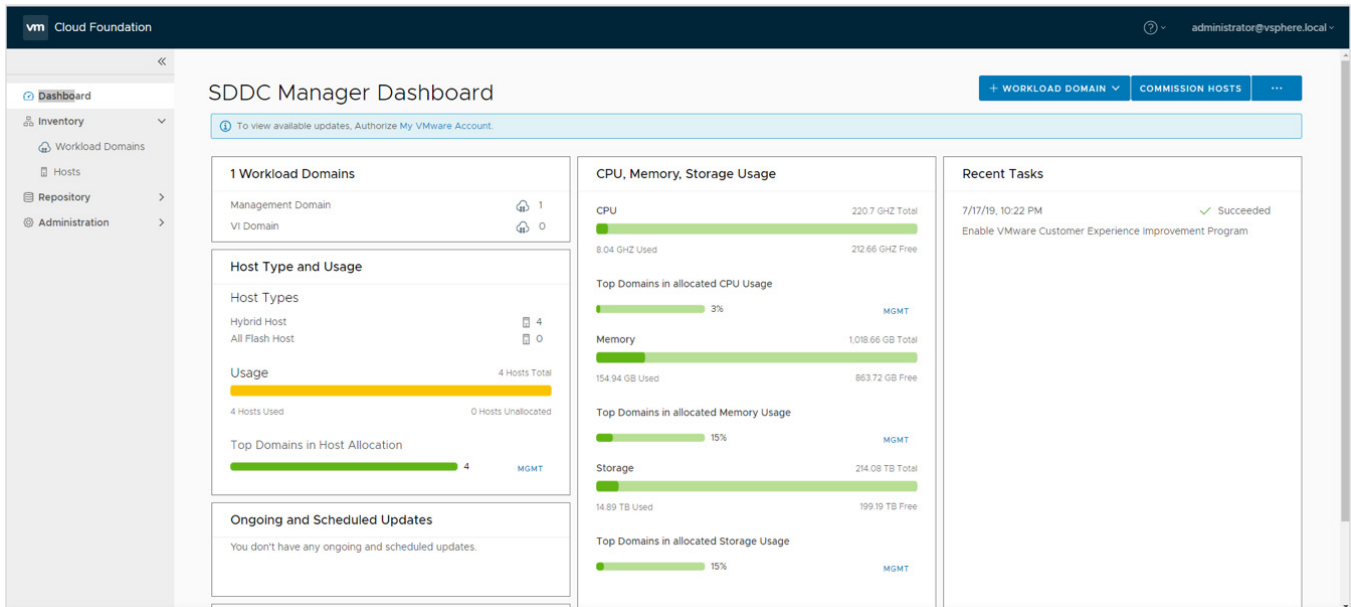
VMWARE CLOUD FOUNDATION ARCHITECTURE

VMware Cloud Foundation is an integrated software stack that bundles compute virtualization (VMware vSphere), storage virtualization (VMware vSAN), network virtualization (VMware NSX), cloud management and monitoring (VMware vRealize Suite) into a single platform that can be deployed on premises as a private cloud or consumed as a service within a public cloud.

The Cloud Foundation components are packaged into a single “Cloud Builder” virtual appliance. Cloud Builder takes deployment input and fully deploys the standardized SDDC architecture, including the deployment of SDDC Manager, a unique software component of Cloud Foundation.

SDDC MANAGER

SDDC Manager, is used to provision new cloud resources, initiate and monitor workflow automation, manage lifecycle, and more. Additionally, SDDC Manager is responsible for the addition and removal of infrastructure components, as well as creating and managing abstractions referred to as “workload domains”. A workload domain is a logical pool of compute, storage and networking, consisting of one or more vSphere clusters, including a dedicated vCenter, NSX Management components, all provisioned automatically by SDDC Manager.



All processes are operationalized by SDDC Manager using workflows. Each workflow is comprised of a series of tasks that are executed by SDDC Manager. In this manner, SDDC Manager ensures that in the event of failure, it can accurately pinpoint the failure and continue the process from that point forward.

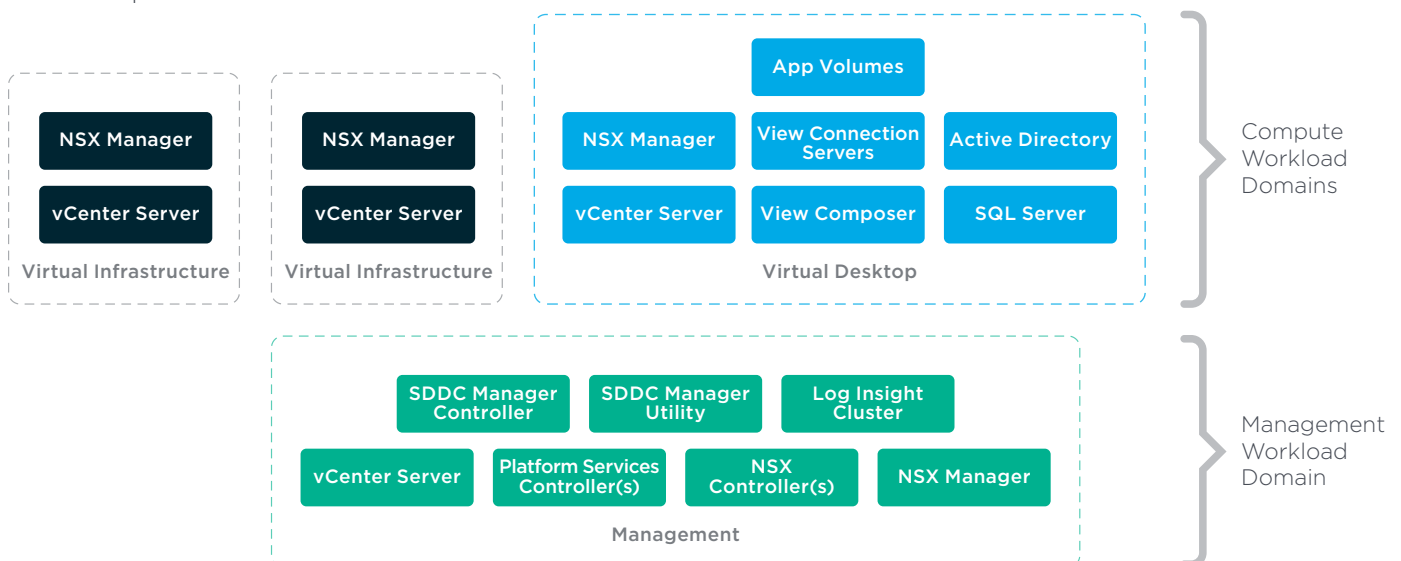
The SDDC Manager VM contains multiple services, each responsible for the different APIs providing lifecycle management for the SDDC.

WORKLOAD DOMAINS

A workload domain is a policy-based resource container with specific availability and performance attributes that combines compute (vSphere), storage (vSAN), and networking (NSX) into a single consumable entity.

This workload domain may then be used to serve two distinct functions:

- Management
- Compute



Regardless of compute workload domain location, Rubrik can protect the VMs contained within it.

MANAGEMENT WORKLOAD DOMAIN

A management workload domain is a specialized workload domain with the specific objective of providing management functionality across the SDDC.

At a minimum, it contains a core set of VMs to manage the Cloud Foundation infrastructure:

- Platform Service Controllers
- vCenter Server
- SDDC Manager
- NSX for vSphere Manager
- vRealize Log Insight cluster

Optional management components may also be deployed by the SDDC Manager within the default management domain after the core components have been established. These may include:

- vRealize Suite Lifecycle Manager
- vRealize Automation
- vRealize Operations

Finally, the management workload domain also may contain VMs which support other workload domains as well as user-provisioned VMs to support any management type functionality such as database or directory services. These may include:

- vCenter Server(s)
- NSX for vSphere Manager(s)
- NSX-T Manager(s)
- Database server(s)
- Directory services server(s)

While it's important to have a backup strategy encompassing all of the services and VMs deployed within a management workload domain, this guide will focus on protecting the core management components.

COMPUTE WORKLOAD DOMAINS

Compute workload domains can be either Virtual Infrastructure (VI) or Virtual Desktop Infrastructure (VDI) and are created on-demand by Cloud Foundation administrators. Each workload domain is created according to user-specified size, performance, and availability. For example, a cloud administrator can create one workload domain for test workloads that have balanced performance and low availability requirements, while creating a separate workload domain for production workloads requiring high availability and performance.

PROTECTING AND RESTORING VMWARE CLOUD FOUNDATION

All management components of Cloud Foundation can and should be backed up. In the event of data corruption or loss, the workload domain VMs are restorable from the backup copies.

It is recommended that regular backups are scheduled for all management domain VMs and Cloud Foundation components.

It is also recommended that the management domain VMs are backed up prior to and after upgrading Cloud Foundation; creating, deleting, or modifying the domain; and rotating the passwords.

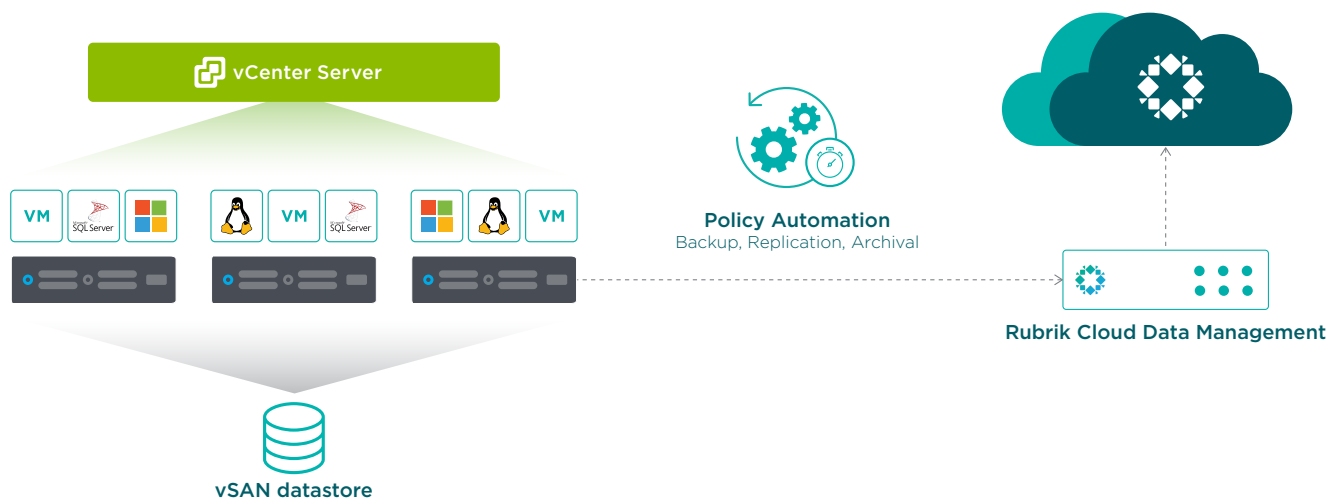
COMPUTE WORKLOAD DOMAINS

Dare to imagine never again having to define backup jobs, selecting media agents, configuring backup storage settings, or creating backup chains. Eliminate spending days or weeks mapping out backup jobs, designing and deploying the required infrastructure. Rubrik removes one of the most inefficient and storage wasting tasks in backup and recovery -- scheduling periodic full backups.

Rubrik eliminates the imperative approach and replaces it with declarative, taming the circus of manually juggling backup tasks with simple data management by defining SLA protection policies. An SLA Policy provides the choice of Rubrik snapshots frequency and the length of retention, effectively translating business logic into an automated task. Rather than manually creating a policy and applying per workload, an SLA policy may be applied at a broad level--such as the management server (vCenter Server), folder, host, cluster, etc.--or granularly (per VM) to achieve specific data protection objectives.

The automatic protection mechanism simplifies assigning protection to large numbers of VMs and provides an easy method to uniformly assign specific SLA Domains to groups of functionally similar VMs.

Rubrik's CDM platform accesses VM data through an API connection with the VMware vCenter Server that manages the hypervisor running the VM.



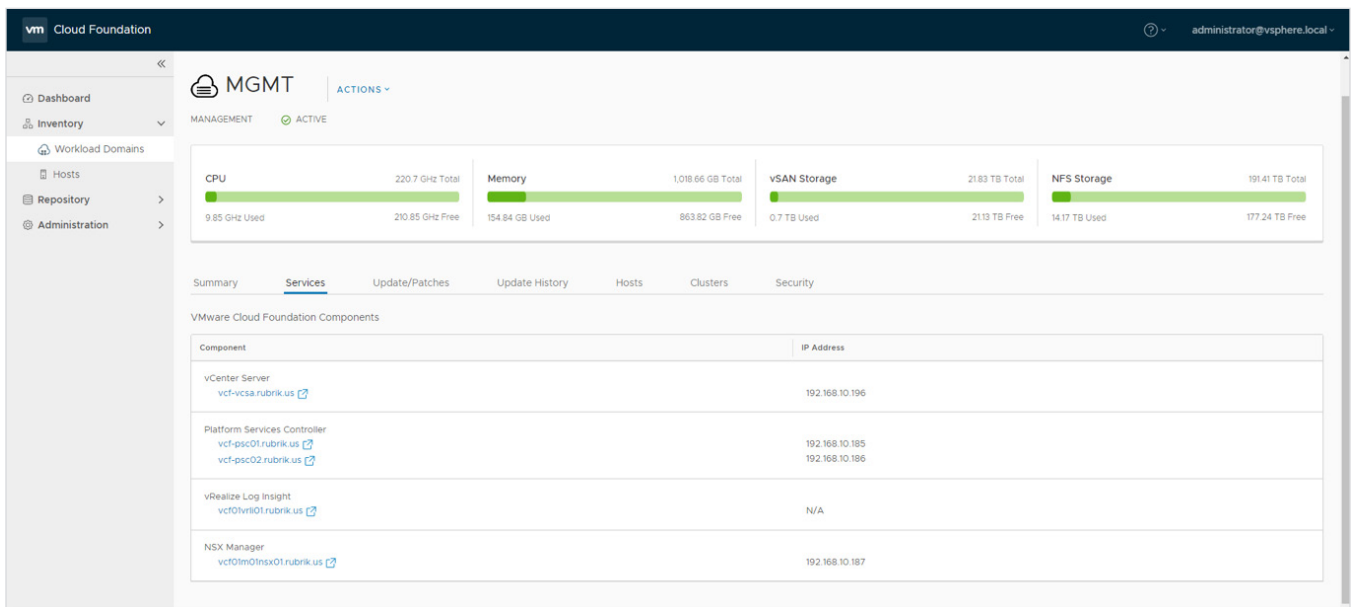
Rubrik CDM provides Auto Protect of VMs through inheritance of the SLA policy assigned to a parent object. If the vCenter Server or a folder has an SLA assigned to it, the VM underneath will automatically inherit the policy. Rubrik detects the newly-added VM and automatically applies a protection policy, eliminating the need for any manual administrator interaction. This resolves the common issue of new workloads being brought online and going days or weeks without being protected. If the newly-added VM requires different protection than the inherited SLA provides, administrators can directly assign the VM to any desired SLA, at which point it will take precedence.

Once the policy is assigned, Rubrik adheres to user-defined policies such as frequency, retention, archival, etc. as described above. Intelligent algorithms assist with balancing the workload as more VMs are created and added into the system. The automatic scheduling of tasks ensures that all workloads are evenly distributed across the Rubrik cluster, preventing cluster resource contention.

MANAGEMENT WORKLOAD DOMAIN COMPONENTS

Before restoring any management VMs, the following core components must be operational:

- Platform Services Controller(s)
- vCenter Server
- SDDC Manager
- NSX Manager
- vRealize Log Insight



If any of the above components are impacted, they must first be restored before restoring any other management services. For example, the management vCenter Server must be restored before the SDDC Manager VM as it manages the SDDC Manager and all other management VMs.

This following table provides an overview of protection for management components.

| Management Component | Backup Method | Restore Method |
|--|---------------------------------|--------------------------|
| Platform Services Controller(s) | Image level / file level | Image level / file level |
| vCenter Server Appliance | Image level / file level | Image level / file level |
| SDDC Manager | Image level | Image level |
| NSX Manager | Files backed up by SDDC Manager | File level |
| vRealize Log Insight | Image level | Image level |

The Rubrik cluster accesses the core management VMs data through a connection to the management VMware vCenter Server. To successfully connect with a vCenter Server, the Rubrik cluster requires connection information for that vCenter Server. This information includes:

- vCenter Server FQDN or IP Address
- Username
- Password

After connection information for a vCenter Server is added, the Rubrik cluster requests relevant metadata from the vCenter Server, such as folder, cluster, and host information. The Rubrik cluster uses the metadata to display and work with the VMs on the vCenter Server.

The Rubrik cluster automatically refreshes the metadata from a vCenter Server every 30 minutes. This is referred to as a light refresh. The Rubrik cluster automatically refreshes the metadata and rescans the VMDK files of a vCenter Server every two hours. This is referred to as a full refresh. VMDK files are also automatically scanned as part of every create snapshot job. A full refresh can be manually initiated at any time.

PLATFORM SERVICE CONTROLLERS AND VCENTER SERVER RECOVERABILITY

The most efficient and simplest way to recover from a Platform Service Controller (PSC) or vCenter Server VM failure is to perform an image level recovery of the affected service. This can be achieved through Rubrik by either performing an Export, a Live Mount, or an Instant Recovery, with the latter two options not requiring any data to traverse to the source.

Depending on the number of failed vCenter components and which components have failed, multiple recovery scenarios are supported.

- If the vCenter Server fails, an image or file level restore can be initiated to bring it back online.
- If only one PSC fails, the PSC can be simply restored via image or file level or a new PSC can be redeployed and joined into the existing SSO domain.
- If both PSC's fail, the initial PSC will need to be restored first, then the secondary can either be restored or rejoined.

To learn more about protecting vCenter Servers and Platform Services Controllers, please see our [Rubrik and VMware vSphere Reference Architecture](#).

SDDC MANAGER RECOVERABILITY

The SDDC Manager is the lifeblood of VCF providing the ability to provision new resources and workload domains as well as monitor changes to existing logical architectures. It is recommended to schedule regular backups of SDDC Manager to perform restores in case of a corrupt appliance instance.

Aside from provisioning access, the SDDC Manager VM stores the complete configuration of the VCF architecture, including information around hostname, IP, network, storage, credentials, and much more. This information is critical in the event a restore of the complete management domain needs to be performed. To ensure a current copy of this information is always available, it's recommended to configure a script on the SDDC Manager to dump this information, then set the script to run as a pre-backup event within Rubrik. This will ensure that before every image level backup, the most current configuration is also encapsulated within the backup, saving many frustrations and time when a complete restore event is required.

Configure Pre/Post Scripts

Pre-Backup Script
Pre-Backup Script Path

Cancel Backup if Pre-Backup Script Fails
Timeout (Seconds)

Post-Snap Script

Post-Snap Script Path

Timeout (Seconds)

.....

Post-Backup Script

Post-Backup Script Path

Timeout (Seconds)

.....

! Scripts are responsible for ensuring application consistent snapshots

Note: An example script processing the many API endpoints, as well as obtaining relevant NSX configurations can be found in [Appendix A](#).

NSX RECOVERABILITY

NSX is the network virtualization platform for the SDDC, delivering the operational model of a VM for entire networks. With NSX, network functions, such as switching, routing, and firewalling, are embedded in the hypervisor and distributed across the environment. Proper backup of all NSX components is crucial to restore the system to its working state in the event of a failure.

The NSX Manager backup contains all of the NSX configuration, including controllers, logical switching and routing entities, security, firewall rules, and everything else that can be configured within the NSX Manager interface or API. The core NSX Manager appliance deployed during bring-up is preconfigured to send backups to an NFS export on the SDDC Manager VM, which is in turn protected by Rubrik. Because of this, there is no need to provide any additional protection of NSX with Rubrik.

Important: Because the NSX Manager backup is auto-configured, do not modify the backup configuration settings in the NSX Manager interface, or the NSX Manager backup will not be included in the Cloud Foundation process.

To recover the core NSX Manager appliance, a new appliance of the same version and build is deployed. The backup settings are configured to point to the SDDC Manager VM and a file-level restore is initiated.

Note: The NSX version, build, and backup settings can be obtained from the configuration dump provided in the example script highlighted in [Appendix A](#)

Once the NSX Manager appliance has completed the restoration process, any failed NSX controllers can be simply removed and redeployed within the configuration of the appliance.

Refer to the official [VMware documentation](#) for more information.

VREALIZE LOG INSIGHT RECOVERABILITY

vRealize Log Insight delivers log management with intuitive, actionable dashboards, sophisticated analytics and broad third-party extensibility, providing deep operational visibility and faster troubleshooting.

Rubrik may be used to provide automated image-level protection and restoration through Export, Live Mount, or Instant Recovery. When restoring the vRealize Log Insight cluster, the master node must first be processed, followed by any worker nodes afterwards.

VSAN RECOVERABILITY

Because Rubrik leverages the vSphere APIs for Data Protection (VADP), no special considerations are required in order to successfully integrate a Rubrik cluster with a VMware vSAN deployment.

To learn more about using vSAN and Rubrik, please see our [VMware vSAN and Rubrik Reference Architecture](#).

OPERATIONAL OVERVIEW

Rubrik and VMware Cloud Foundation product lines leverage a complementary progressive hybrid cloud enterprise architecture with the goal of accelerating applications and business requirements. This section aims to highlight how robust the Rubrik and VMware Cloud Foundation joint solution is.

USE CASES

This section is intended to provide a few examples of how Rubrik and VMware Cloud Foundation may be used together. It is not intended to be an exhaustive list but merely a set of sample use cases.

TEST AND DEVELOPMENT

Upgrades can be scary! What if it was possible to instantiate an environment that looked exactly like production (at the time of snapshot) automatically through APIs and push code into that environment to test? With Rubrik, that possibility is reality.

Build isolated environments and leverage the Live Mount feature to instantiate an identical environment in moments. Additionally, VMs may be instantiated using CloudOn in AWS or Azure from any point-in-time without modifying the underlying immutable data. Test an application upgrade, failure scenario, or other use cases using your backup storage. When done, simply throw it away.

Rather than tinkering with a production VM, use Live Mount to spin up a copy from a specific point in time to modify. As an example, Active Directory work, such as rehearsing an upgrade to the Forest Level, may be done during the day in a bubble network.

This functionality appeals to application owners and operations teams in order to conduct:

- Functional or regression testing
- Application development
- Software release testing (upgrade the actual applications)

No additional configurations are needed on the hypervisor side for Live Mount functionality to work. Simply provide a service account with the documented permissions required for the type of virtualization environment. Rubrik automates the entire

process. VMs of any size can be recovered in the amount of time it takes for the OS to boot. Imagine having the ability to spin up an 8 TB VM in under 2 minutes so that a recovery point can be validated by an administrator or application owner.

MIGRATE TO A NEW SDDC AND MAINTAIN BACKUP HISTORY

Architecting a greenfield VMware Cloud Foundation deployment requires no maintenance in terms of existing backups and VMs since everything is created from scratch. That said, organizations building out an SDDC with the intent to migrate existing workloads may experience challenges with some third-party integrations detecting the migrated VMs as new workloads and losing any historical data or statistics with the original object. VMs are uniquely identified within a vCenter Server instance by a MoRef. A MoRef is a unique key that is never duplicated within the vCenter Server inventory. All of the Cloud Foundation [migration utilities](#) ultimately cause the VM to obtain a new MoRef upon successful migration to the new SDDC.

Although the MoRef will have no effect on the performance or operations within vSphere or Cloud Foundation, third-party tools depending on MoRefs will identify any migrated VM as new, and all object history is lost. In the case of many data protection solutions, this means that they will need to be re-added to any backup jobs and all backups and archives are no longer tied to the VM. This results in a lot of wasted capacity as a new full VM backup is created.

Rubrik provides the ability to maintain linkage to the VM's original MoRef after it is migrated into a new SDDC. This ensures compliance to existing backup policies and the protection history is maintained. A new full backup is not required because this is a known MoRef, and incrementals will continue.

A linked VM occurs when Rubrik joins two VM MoRefs into the same back history. This functionality can be configured to take place automatically and is set at the vCenter Server level within the Rubrik UI. The steps to ensure backup history and compliance is maintained within Rubrik during a Cloud Foundation migration are:

1. Cloud Foundation administrator creates a Compute Workload Domain. Subsequently, a vCenter Server is automatically deployed and configured within the Management Workload Domain
2. During the addition of the newly created vCenter Server within Rubrik, select to 'Let Rubrik resolve conflicts automatically'
3. Rubrik will then perform VM Linking on all objects created within the vCenter Server as part of the vCenter Server refresh task

This results in Rubrik automatically performing VM linking on any newly migrated VMs within the vCenter Server for the Compute Workload Domain. As administrators migrate VMs from their original infrastructure into their new SDDC, Rubrik will maintain all of the backup history, archive history, SLA Domain membership, and metadata related to the VM--ensuring there is no wasted capacity or performance within the Rubrik cluster.

CLOUDOUT (ARCHIVAL)

CloudOut is a capability within Rubrik CDM used to archive data to the cloud for short and long-term retention. Users may leverage Rubrik to intelligently and cost-effectively store backup data in Amazon S3, Microsoft Azure Blob storage, or Google Cloud Storage. More importantly, Rubrik is optimized to provide rapid and efficient data restores both on-premises and in public cloud. Data is indexed by Rubrik CDM before it is stored in the cloud archive, enabling customers to quickly browse, search, and restore any file. During restores, Rubrik only retrieves the specific files that need to be recovered to minimize bandwidth and egress costs.

Rubrik customers typically leverage CloudOut as a solution to replace their tape storage infrastructure, eliminating the need to copy backup data to tapes that would then need to be manually stored offsite. Rubrik with cloud archive provides a tape-replacement solution that is more durable, available, cost-effective, and agile.

If on-premises archive solutions are preferred, Rubrik also supports NFS, tape, and object storage.

CLOUDON (INSTANTIATION)

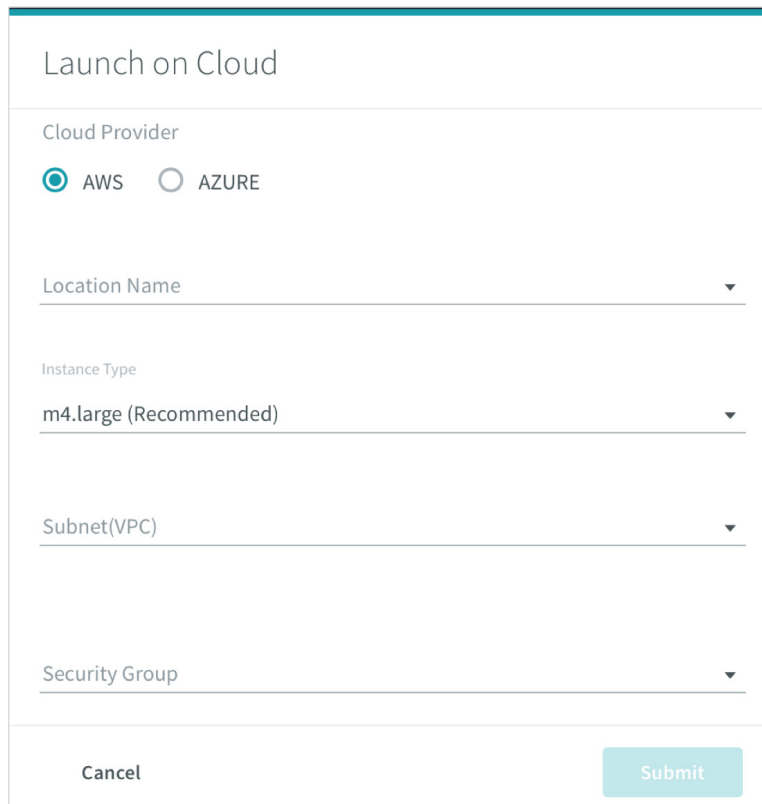
CloudOn, or instantiation, allows users to migrate existing on-premises workloads to the cloud for test/development or even DR purposes. Rubrik's CloudOn feature converts workload data (VM image) sent to the cloud into a compute instance. There is no need to run Rubrik in the cloud to migrate workloads to the cloud for test/development, increasing overall cloud savings.

Better yet, imagine not needing a separate cluster for test/development workloads or an identical physical infrastructure for DR. Using Rubrik CloudOn, workloads can be migrated at a VM level from on-premises to AWS or Azure.

Rubrik offers three options that can be applied to on-premises workloads that customers choose to instantiate in AWS or Azure:

- **On-Demand**—The default configuration in which Amazon Machine Images (AMIs) or Azure Virtual Hard Drives (VHDs) are created only at the time of a “power on in the cloud” request.
- **Auto Convert Latest Snapshot - Keep One**—Rubrik will automatically construct an AMI or VHD reflecting the latest snapshot to be archived into S3 or Azure. When a new snapshot is sent to the archive, a new AMI or VHD is constructed with the new archive data. Once completed, the older AMI or VHD is removed.
- **Auto Convert Latest Snapshot - Keep All**—Rubrik will automatically construct an AMI or VHD reflecting the latest snapshot to be archived into S3 or Azure. When a new snapshot is sent to the archive, a new AMI or VHD is constructed with the new archive data. The older AMI or VHD is retained if desired (configurable via policy), creating a series of AMIs or VHDs representing each snapshot.

The following screenshot demonstrates the required information to instantiate a workload in AWS:



The screenshot shows a 'Launch on Cloud' configuration form. At the top, the title 'Launch on Cloud' is displayed. Below the title, there are four sections, each with a label and a dropdown menu:

- Cloud Provider:** Two radio buttons are present: 'AWS' (selected) and 'AZURE'.
- Location Name:** A dropdown menu with a downward arrow.
- Instance Type:** A dropdown menu showing 'm4.large (Recommended)' with a downward arrow.
- Subnet(VPC):** A dropdown menu with a downward arrow.
- Security Group:** A dropdown menu with a downward arrow.

At the bottom of the form, there are two buttons: 'Cancel' on the left and 'Submit' on the right.

Whether instantiating workloads on-demand or automatically with the latest snapshot, spinning up copies of workloads in the cloud results in faster development cycles, as developers are unblocked from the constraints of physical infrastructure. Picture the cost savings garnered when avoiding a dedicated on-premises infrastructure for test/development. Developers can spin up instances when required and shut down when not in use.

DAY 0 - 2 OPS WITH API / INTEGRATIONS USE CASES

VMware Cloud Foundation takes an automated lifecycle approach - automating many day 0 - 2 operations and streamlining processes around bring up (installation), setup, configuration, and patching/upgrades of the entire integrated vSphere stack. Rubrik, with its API-first architecture, works in harmony with Cloud Foundation to further provide automation into the data protection realm in a number of areas, some of which are outlined below.

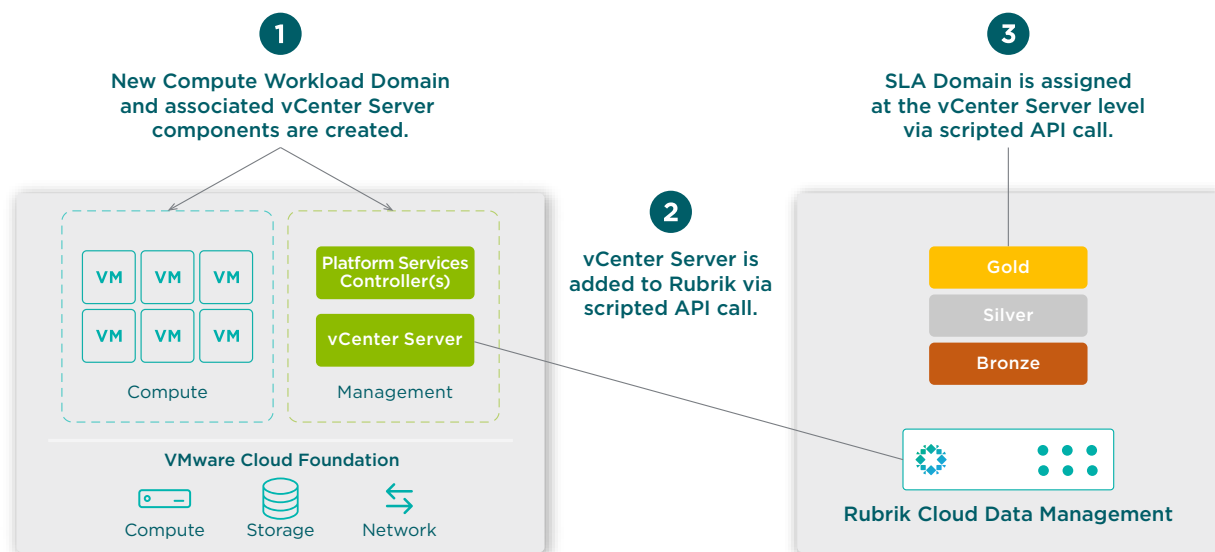
AUTOMATIC POLICY-BASED DATA PROTECTION WITH NEW COMPUTE WORKLOAD DOMAINS

Part of this streamlined process involves the ability to create workload domains. A workload domain is a policy-based resource container that adheres to specific availability and performance attributes. Simply put, a Workload Domain is a container for VMs--a consumable entity, which includes compute (vSphere), storage (vSAN), and networking (NSX).

During the setup of Cloud Foundation, a management workload domain is created by default. This domain is used to house all of the management type VMs (SDDC Manager, vCenter Server, Platform Service Controllers, vRealize Log Insight, NSX, etc). To better learn how to protect the infrastructure contained within a management workload domain, see [Management Workload Domain Components](#).

Compute workload domains are created after the initial configuration of Cloud Foundation, on demand, by administrators. Similar to the management workload domains, compute workload domains pool resources and utilizing vSAN technology for storage and NSX for network/security purposes. A compute workload domain falls into one of two categories: VI for traditional vSphere workloads and VDI for Virtual Desktop delivery. Each compute workload domain contains a set number of resources, including its own dedicated vCenter Server. The complete process, from configuring the vSAN protection policies and NSX to the deployment and configuration of vCenter, is completely automated and abstracted away from the end-user.

Rubrik's API-first architecture and declarative approach to data protection pair well with the VMware Cloud Foundation policy-based architecture. Utilizing raw API calls to Rubrik, administrators are not only able to quickly deploy VMware Validated Designs based compute workload domains, but can also ensure that workloads deployed within the domains are automatically protected. This may also be accomplished by using one of the many automation tool sets supporting RESTful APIs.



The process to integrate policy-based data protection to Cloud Foundation workload domains is as follows:

1. Cloud Foundation administrator creates a compute workload domain. Subsequently, a vCenter Server is automatically deployed and configured within the Management Workload Domain.
2. The newly created vCenter Server is added to Rubrik CDM using an API call within a script or integrated toolset (illustrated below with curl).

```
curl -X POST \  
    -d '{  
        "hostname": "$vcenter_address",  
        "username": "$vcenter_admin",  
        "password": "$vcenter_password" }' \  
    "https://$cluster_address/api/v1/vmware/vcenter"
```

3. A default SLA Domain is assigned to the vCenter Server (or any preferred lower-level construct), ensuring any new workloads created will adhere to a default policy providing data protection. Adding a default SLA domain to vCenter Server is illustrated below with curl.

```
curl -X PATCH \  
    -d '{  
        "configuredSlaDomainId": "$sladomain_id"\  
    }' \  
    "https://$cluster_address/api/v1/vmware/vcenter/$vcenter_id"
```

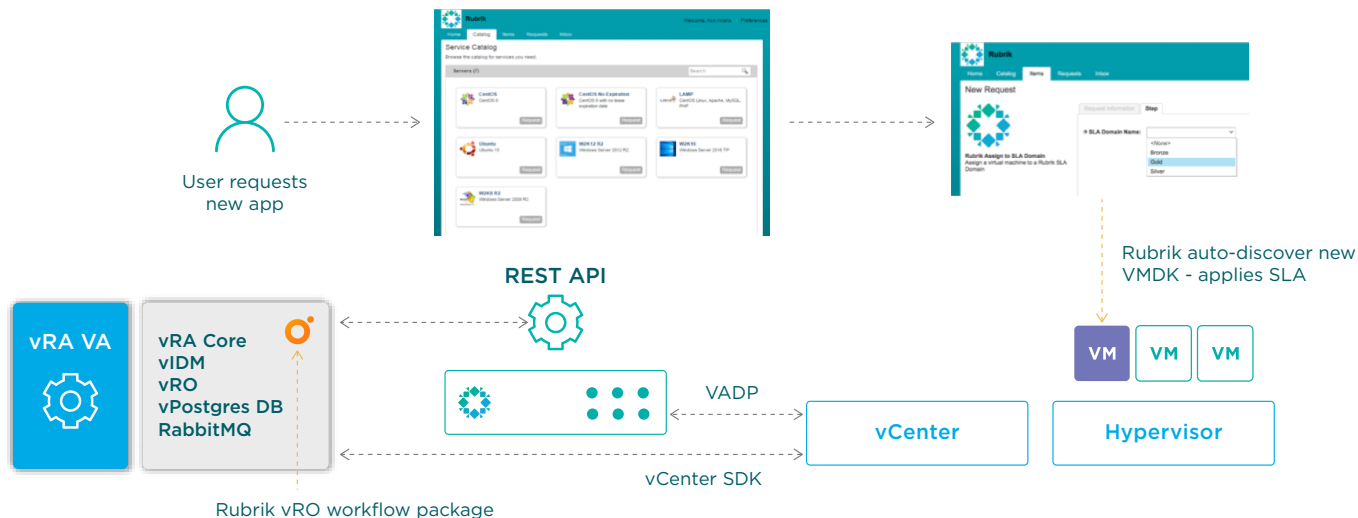
Administrators are also able to apply SLA Domains to any lower-level construct such as folders, ESXi hosts, resource pools, or individually on the VMs. This may be done manually using the Rubrik UI or through API calls, either raw or within one of the many Rubrik integrations such as vRealize Orchestrator (vRO) or PowerShell. For organizations that have paired Cloud Foundation with vRealize Automation (vRA), Rubrik's SLA Domain selection can be presented as a property included within any VM provision and protect workflows.

INTEGRATION WITH VREALIZE AUTOMATION FOR COMPLETE VM LIFECYCLE MANAGEMENT

VMware Cloud Foundation brings automated, built-in lifecycle management around deploying and managing a complete SDDC. By implementing an architecture built around VMware Validated Designs, Cloud Foundation provides a robust approach to providing an easy way to build a consistent and repeatable infrastructure integrating vSphere, vSAN, and NSX.

Since the main focus of Cloud Foundation is on the underlying VMware software stack, it is often paired with vRA to handle the delivery and management of the VMs and applications deployed on top of it. Similar to the other VMware components, vRA can be completely deployed and managed utilizing SDDC Manager, maintaining the simplicity and abstraction that Cloud Foundation is built upon.

vRA provides complete automation around the VM lifecycle, allowing end users to manage and consume resources from within the workload domains created inside Cloud Foundation. Through a self-service portal, end users are able to request, approve, and deploy VMs, abstracting away and orchestrating most of the lifecycle management tasks. With its open-API architecture, vRA integrates with many ecosystem partners - allowing third parties to create workflows around their solutions.



Rubrik integrates with vRA via a vRealize Orchestrator (vRO) package. Once installed, end users can automate and execute workflows using Rubrik features. Organizations may provide a number of vRA hooks into the VM lifecycle such as:

- The assignment of a Rubrik SLA Domain upon provisioning a new VM, ensuring data is protected from day 1.
- Self-service recovery, including VM and file-level restores.
- Requests to utilize Rubrik Live Mount to provision a duplicate copy of production VMs within seconds for testing and development purposes.
- Requests to take on-demand backups of VMs before upgrades or major reconfigurations.

Leveraging Rubrik's API-first architecture, organizations are able to integrate Rubrik functionality into vRA in the form of actions or properties. This provides a holistic design approach to the SDDC - automating not only provisioning and management of the infrastructure stack, but providing a complete VM lifecycle management solution, including deep integration with Rubrik's Cloud Data Management platform. This on-demand [webinar](#) will provide more information about Rubrik's RESTful API approach.

CONCLUSION

Rubrik's support for VMware Cloud Foundation protection is robust and full-featured while extending Rubrik's market-leading focus on simplicity. Using Rubrik and VMware Cloud Foundation together helps accelerate companies on their journey to meet hybrid cloud business requirements by protecting on-premises workloads, providing archival and replication to public cloud, and providing the ability to instantiate vSphere workloads in AWS or Azure.

ABOUT THE AUTHORS

Mike Preston is a Developer Advocate at Rubrik. He blogs on blog.mwpreston.net as well as various other tech related news sites. He is a [Toronto VMUG Leader](#) and author of [Troubleshooting vSphere Storage](#). You can find him on Twitter [@mwpreston](#).

Jason Shaw is a Technical Marketing Architect at VMware who focuses on VMware Cloud Foundation. Jason's responsibilities include educating customers, partners, and field staff on the benefits of using VMware Cloud Foundation. Jason works closely with Product Management and Engineering Teams to provide feedback and help influence product direction.

APPENDIX A - EXAMPLE INVENTORY AND CONFIGURATION GATHERING SCRIPT

During the recovery of the VCF management workload domain, there are a number of configurations that must be applied in the exact same manner as before the failure.

It is important to gather this information during each backup. To do this, a script can be configured on the SDDC Manager connecting to the applicable API endpoints and storing the configuration information in files. Rubrik can be configured to run this script prior to processing backups of the SDDC Manager. This ensures that all relevant configurations are captured and indexed within the Rubrik cluster with each backup of the SDDC Manager.

The following shows an example script that can be used to gather the configurations from the SDDC Manager.

```
#!/bin/bash

targetfolder="/nfs/vmware/vcf/nfs-mount/backup/inventory"

endpoints=( clusters domains esxis hosts nsxmanagers pcs sddcmanagercontrollers
vcenters vcfservices vds vras vrlis vrops vrslcms )

for endpoint in "${endpoints[@]}"
do

    curl -H 'Application: accept/json' http://localhost/inventory/$endpoint | python
-m json.tool > $targetfolder/$endpoint.json

done

# URI's not matching above endpoint array
curl -H 'Application: accept/json' http://localhost/licensing/licensekeys | python
-m json.tool > $targetfolder/licensekeys.json

curl -H 'Application: accept/json' http://localhost:7100/networkpools | python -m
json.tool > $targetfolder/networkpools.json

curl -H 'Application: accept/json' http://localhost/security/password/vault |
python -m json.tool > $targetfolder/passwords.json

# Get NSX IP, username and password
NSXUSER=$(cat $targetfolder/passwords.json | python -c "exec(\"import
sys, json;\njsonarray = json.load(sys.stdin);\nfor dict in jsonarray:\n if
dict['entityType'] == 'NSX_MANAGER' and dict['credentialType'] == 'API':\n print
dict['username']\")")

NSXPASS=$(cat $targetfolder/passwords.json | python -c "exec(\"import
sys, json;\njsonarray = json.load(sys.stdin);\nfor dict in jsonarray:\n if
dict['entityType'] == 'NSX_MANAGER' and dict['credentialType'] == 'API':\n print
dict['password']\")")

NSXIP=$(cat $targetfolder/passwords.json | python -c "exec(\"import sys,
json;\njsonarray = json.load(sys.stdin);\nfor dict in jsonarray:\n if
dict['entityType'] == 'NSX_MANAGER' and dict['credentialType'] == 'API':\n print
dict['entityIpAddress']\")")
```

```

#DUMP NSX Backup Settings
curl --user $NSXUSER:$NSXPASS -H 'Application: accept/json' -k https://$NSXIP/
api/1.0/appliance-management/backuprestore/backupsettings | xmllint --format - >
$targetfolder/nsxbackupsettings.xml

#Domain specific settings
cat $targetfolder/domains.json | python -c "exec(\"import sys, json, urllib2;\n
njsonarray = json.load(sys.stdin);\nfor dict in jsonarray:\n url='http://
localhost/inventory/domains/' + dict['id'] + '/inventory';\n req = urllib2.
Request(url);\n f = urllib2.urlopen(req);\n for x in f:\n print(x);\n
nf.close();\")" | python -m json.tool > $targetfolder/domainspecific.json

#Network Pool specific inventory
cat $targetfolder/networkpools.json | python -c "exec(\"import sys, json,
urllib2;\njsonarray = json.load(sys.stdin);\nfor dict in jsonarray:\n
url='http://localhost:7100/networkpools/' + dict['id'] + '/networks';\n req =
urllib2.Request(url);\n f = urllib2.urlopen(req);\n for x in f:\n print(x);\n
nf.close();\")" | python -m json.tool > $targetfolder/networkpoolspecific.json

exit

```



Global HQ

1001 Page Mill Rd., Building 2
Palo Alto, CA 94304
United States

1-844-4RUBRIK
inquiries@rubrik.com
www.rubrik.com

Rubrik, the Multi-Cloud Data Control™ Company, enables enterprises to maximize value from data that is increasingly fragmented across data centers and clouds. Rubrik delivers a single, policy-driven platform for data recovery, governance, compliance, and cloud mobility. For more information, visit www.rubrik.com and follow @rubrikInc on Twitter. © 2019 Rubrik. Rubrik is a registered trademark of Rubrik, Inc. Other marks may be trademarks of their respective owners.