



REFERENCE ARCHITECTURE

Rubrik and VMware vRealize Automation

TABLE OF CONTENTS

- AN INTRODUCTION TO RUBRIK CLOUD DATA MANAGEMENT (CDM) 4**
- AN INTRODUCTION TO VMWARE VREALIZE AUTOMATION 5**
- AUDIENCE 5**
- SOLUTION OVERVIEW 6**
 - Powered by APIs 6
 - Declarative Policies and SLA Domains 7
 - SLA Domains 8
 - Assigning SLA Domains 9
 - Recovery Methods 10
 - Instant Recovery 10
 - Live Mount 12
 - Export 12
 - File-Level Recovery 13
 - Direct Restore 14
 - Restore by Download 14
 - Protection Overview 15
 - Performance and Scalability 15
 - Backup Processes 16
- ARCHITECTURAL OVERVIEW 16**
 - Self-Service Architecture 18
 - vRealize Automation Architecture 18
 - vRA Appliance 19
 - IaaS Servers 19
 - Web Server 20
 - Model Manager 20
 - Manager Service 20
 - SQL Server Database 20
 - Distributed Execution Manager 20
 - Agents 21
 - Integration with vRealize Orchestrator 21
 - vRA Deployment Types 22
 - Minimal Deployment 22
 - Distributed Deployment 23

TABLE OF CONTENTS

Rubrik Integration with vRA	24
Rubrik Plugin for vRealize	24
Rubrik APIs and Added Extensibility	25
vSphere Integration.....	25
OPERATIONAL OVERVIEW	26
Protection Restoring vRA.....	26
vSphere Recoverability	28
vRO Recoverability.....	28
IaaS Recoverability	28
vRA Recoverability.....	29
Supporting Database(s) Recoverability.....	29
IaaS MS SQL Database	29
vRO Database	29
vRA Postgres Database.....	30
Protecting and Restoring vRA Provisioned Resources.....	30
Integrating Rubrik CDM with vRA	31
Use Cases and Sample Workflows	33
Provision and Protect.....	33
Creating Rubrik SLA Domain Property Definitions and Groups	34
Assigning Property Definitions to New or Existing Blueprints.....	36
Creating a Workflow Subscription.....	36
Automated Data Protection	38
Self-Serve Recovery.....	39
Self-Serve Instant Recovery	40
Self-Serve Live Mount	42
CONCLUSION.....	45
ABOUT THE AUTHORS.....	45
APPENDIX A: PORT DIAGRAM.....	46
APPENDIX B: VREALIZE AUTOMATION PRIVILEGES.....	50

AN INTRODUCTION TO RUBRIK

Rubrik, Inc. is a data management platform used by enterprise organizations to securely manage all data, physical or virtual, across all locations – on-premises, edge of the data center, and cloud.

CLOUD DATA MANAGEMENT (CDM)

The Rubrik Cloud Data Management unifies backup, instant recovery, replication, global indexed search, archival, compliance, and copy data management into a single scale-out fabric across the data center and public cloud.

Instant search delivers near-zero RTOs with predictive search. Easily locate VMs, databases, applications, or files regardless of whether they reside in the cloud or on premises.

Policy-driven engine and programmatic interface eliminates daily operational management by automating how data services are created, consumed, and retired from across the data center and cloud.

Orchestration is the core of Rubrik, providing a suite of APIs that can be used to orchestrate data from data center to cloud. Rubrik provides the freedom to provision data management services with configuration management tools and via custom portals.

Data is **secure** in transit and at rest throughout the entire lifecycle, regardless of location. Granular role-based access can be leveraged while automating compliance reporting in order to successfully meet and complete various industry audits.

Analytics and reporting are provided by Rubrik Envision, which unlocks actionable insight across all environments with customizable reports. Leverage platform analytics that detail operational efficiency, compliance, and capacity utilization across your infrastructure.

POLARIS

Rubrik Polaris acts as a unified system of record to organize all your mission critical data to make it discoverable and usable regardless of location.

Centralized management for a global, distributed Rubrik environment delivered by Rubrik Polaris. Designed for a seamless user experience, Polaris **GPS** provides a comprehensive view of your physical, virtual, and cloud topologies while transforming complex management tasks into elegantly simple and intuitive insights.

Multi-leveled defense makes it easier and faster to recover from security attacks while providing greater intelligence on how an incident impacted your global applications and data. Polaris **Radar** detects anomalies, analyzes threat impact, and accelerates recovery.

For more information, watch the [on-demand Product Demo](#) or visit the [Rubrik website](#).

AN INTRODUCTION TO VMWARE VREALIZE AUTOMATION

vRealize Automation (vRA) provides a secure self-service portal allowing administrators, developers or business users to request new IT services in the terms of both on-premises and cloud workloads. Multi-hypervisor and multi-cloud support brings current day flexibility while providing investment protection around future technology needs and choices.

Working alongside vRealize Orchestrator (vRO), vRA increases both IT agility and IT efficiencies by leveraging vRO's extensible and pluggable architecture. Providing automation around everything from multi-cloud deployments to VMware's line of business applications to on-premises software such as Active Directory and SQL Server, vRA allows administrators leverage a number of built-in workflows and components in order to orchestrate the end-to-end delivery and management of both infrastructure and applications alike.

While initially focussed on on-premises vSphere and the applications running within, vRA has since broadened the provisioning capabilities and supported platforms, becoming a true automation engine for a multi-vendor, multi-cloud infrastructure. Coupling all of the pre-built support with the ability to craft and create plugins supporting nearly any product or platform, vRA gives IT Operations the ability to automate and orchestrate the deployment, management, and lifecycle of nearly any application or workload, no matter where they wish to deploy it.

vRA also provides automation beyond the scope of technical processes, such as tackling operational and business oriented workflows. Through entitlements customers are able to add governance and additional controls to their environment, ensuring complete lifecycle management of resources deployed by their internal or external consumers. This governance includes the ability to limit or restrict the number of resources consumed, the right-sizing of workloads based on entitlement, the option to provide specific and simple approval policies which may already be in place, and the ability to expire or destroy resources based on a specified time barrier.

In the end, vRA provides a complete end-to-end lifecycle management solution to organizations, offloading and automating mundane, time-consuming processes yet still providing a robust and holistic provisioning solution. The orchestration and self-service aspect of vRA saves organizations time and reduces cost when deploying applications within on-premises environments or public and private clouds. Lifecycle management and right-sizing governance ensures that organizations are getting the most out of their hardware and limiting their cloud costs by automatically applying custom-built blueprints specifying the amount of CPU, memory and disk to which the users are entitled, along with managing the complete lifecycle of the application from provisioning to expiration. More importantly, vRA accelerates an applications deployment cycle, putting the power in the hands of the end-users, while still providing the control IT Operations needs and wants.

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 vRealize Automation solution.

For the remainder of this document, "virtual machines" will be referred to as "VMs", "disaster recovery" as "DR", "vRealize Automation" as "vRA", and "vRealize Orchestrator" as "vRO".

SOLUTION OVERVIEW

vRealize Automation accelerates the delivery of IT services by providing automation and pre-defined blueprints across clouds and on-premises infrastructure, providing a high level of flexibility and provisioning options to end-users through a self-service catalog. While placing the power of provisioning into the end-users hands, vRA provides the governance and control that IT teams and organizations require through the use of VM lifecycle management and flexible approval policies. Coupled with vRealize Orchestrator, vRealize Automation is able to deploy and manage nearly any application that supports REST API endpoints. REST is a platform independent, stateless architecture that can be used to inner join once disparate and separate software solutions. This allows vRA to provide complete end-to-end provisioning across an organization's application stack.

Pairing Rubrik and vRA together is as simple as installing the Rubrik Plugin for vRealize. This brings automated data protection to deployed resources while adding a number of use-cases to ensure provisioned workloads and applications are always protected and available. This joint solution allows organizations to set and adhere to data protection policies without manual intervention. Rubrik and vRA provide a variety of options as it pertains to protecting provisioned applications such as:

- Requiring SLA Domain selection during provisioning
- Automatically assigning SLA Domains by inheritance by deploying VMs into protected compute clusters
- A combination of both direct and indirect SLA Domain assignments

In addition to data protection, the Rubrik Plugin for vRealize enables many benefits as it pertains to data management. As an example, end-users can be allowed to request and perform self-service recoveries of data as well as granting the ability for end-users to provision point-in-time copies of their production workloads for testing and development purposes.

Rubrik and vRealize Automation work in harmony to provide complete automation around data protection and data management, delivering various self-serve options to end-users, while allowing IT teams to maintain the governance and control they require.

POWERED BY APIS

As modern data centers become increasingly more software-defined, the amount of automation being built around the infrastructure and solutions stack is rising drastically. For this reason it's crucial to select software solutions which can easily fit into an organization's existing automation and orchestration toolset. The most common interface providing this integration is the use of Application Programming Interfaces (APIs). APIs provide a stateless, common architecture that enables IT Operations to stitch together applications from all corners of their data center, whether on-premises or in cloud.

Rubrik takes an API-first approach, meaning all actions and functionality within the user interface (and more) can also be called by leveraging a language of choice consuming a back-end API. With embedded documentation and example code, IT Operations can easily integrate Rubrik functionality (such as protecting and restoring virtual machines) into their existing automation processes.

Both vRA and vRO are also heavily reliant upon an automated, API-first approach. This allows for a simple and easy integration of Rubrik related data management functionality into a vRA self service catalog, granting end-users to perform actions such as Instant Recovery, Live Mount, and backups directly from the familiar vRA interface, all the while providing automated data protection to vRA provisioned resources.

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 and vRA are firmly rooted in the declarative approach. End users within vRA simply request a catalog item containing a resource or action they would like provisioned, allowing the automation to take over and ensure everything is deployed and configured according to the organization’s requirements. Within Rubrik, 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.

In the SLA Policies section, an example SLA policy was given:

- Take a backup:
 - Perform a backup every 8 hours with the following retention:
 - 8 hour backup increments are kept for 30 days
 - Monthly snapshot increments are kept for 30 days
- Archive to Amazon S3 after 30 days
- Replicate data to another Rubrik cluster and retain for 45 days

The image displays two screenshots of the Rubrik SLA Domain configuration interface. The left screenshot shows the 'Create SLA Domain' form with the following details: SLA Domain Name: vSphere_VMs; Service Level Agreement: Choose how often we take snapshots and the length of time we keep them; Snapshot settings: Take Snapshots: Every (Hours) 8, Keep Snapshots: For (Days) 30; Local retention set to 7 years; Snapshot Window: Take snapshots from: [time] to [time]; Take first full between: First Opportunity at [time]. The right screenshot shows the 'Remote Storage Configuration' section: Retention On Brik: 30 days; Archival: Enabled, S3:se3archive, Enable Instant Archive (disabled); Archival starts after 30 days and is retained on the archival location for 6 years 335 days; Replication: Enabled, selab-emea-apj, 45 days; Replication starts immediately, and is retained for 45 days.

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 335 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) Administrators may choose to use the built-in SLA Domains
- Bronze (lowest protection) 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, [using PowerShell](#)

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.

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:

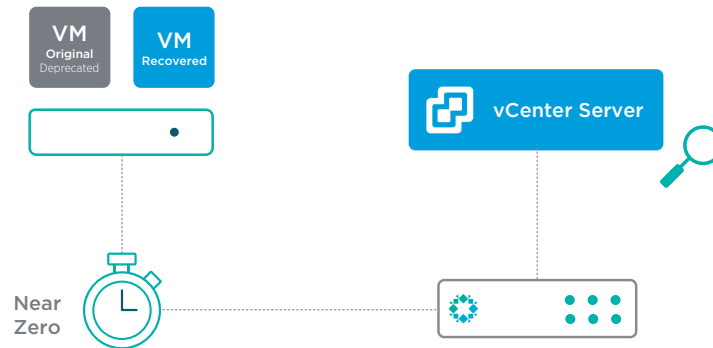
The process first begins by the selecting the VM, snapshot date, and recovery host. You may select to remove a virtual network device if any networking changes or issues would prevent the VM from successfully powering on.

This methodology also enables validation of certain services after recovery but before restoring the service.

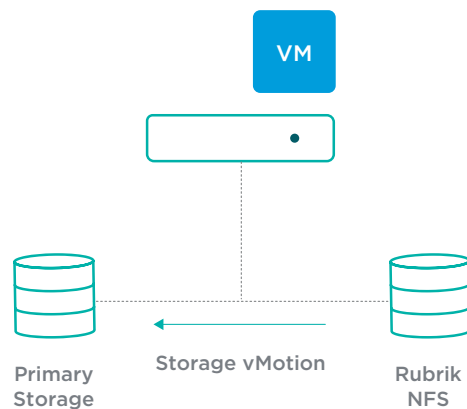
Additionally, you may select to preserve the VM-managed object ID (MoRef). This is a managed object ID, which is applicable to vSphere VMs. It will ensure that the VM is recovered using the same MoRef as a part of VM linking, rather than it being recovered as a new object. This method can be important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

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.



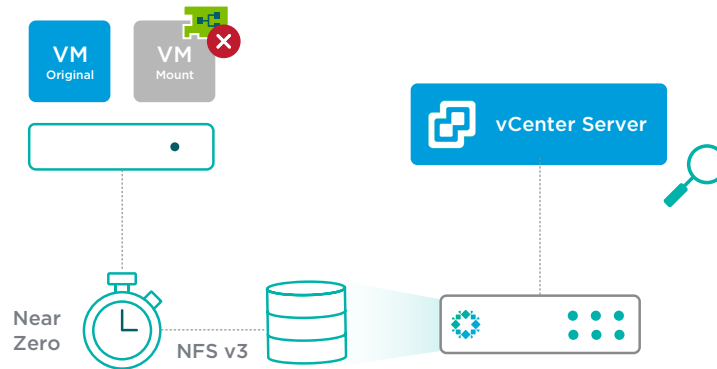
Ultimately, Rubrik serves as a storage endpoint to recover as many vSphere VMs as needed, thus eliminating the complexity and time wasted in transferring data back into the production system. This functionality provides a near-zero recovery time and restores user access near instantly.

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.

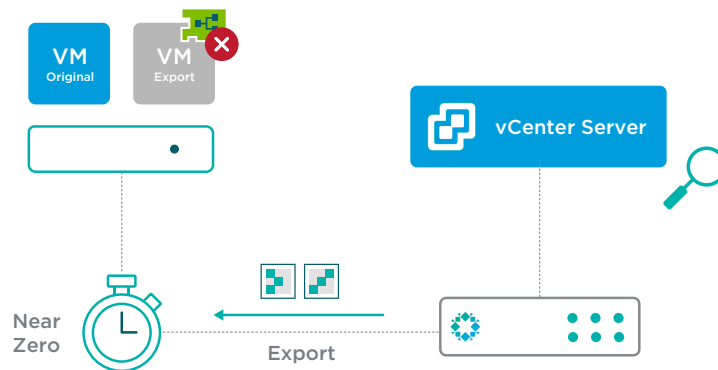
No additional configurations are needed on the hypervisor side for Live Mount functionality to work. Simply provide a service account with the documented permissions required for the type of virtualization environment. Rubrik automates the entire process. VMs of any size can be recovered in the amount 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: 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.

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.

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.

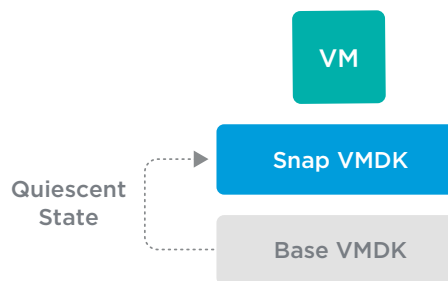
Performance and Scalability

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.

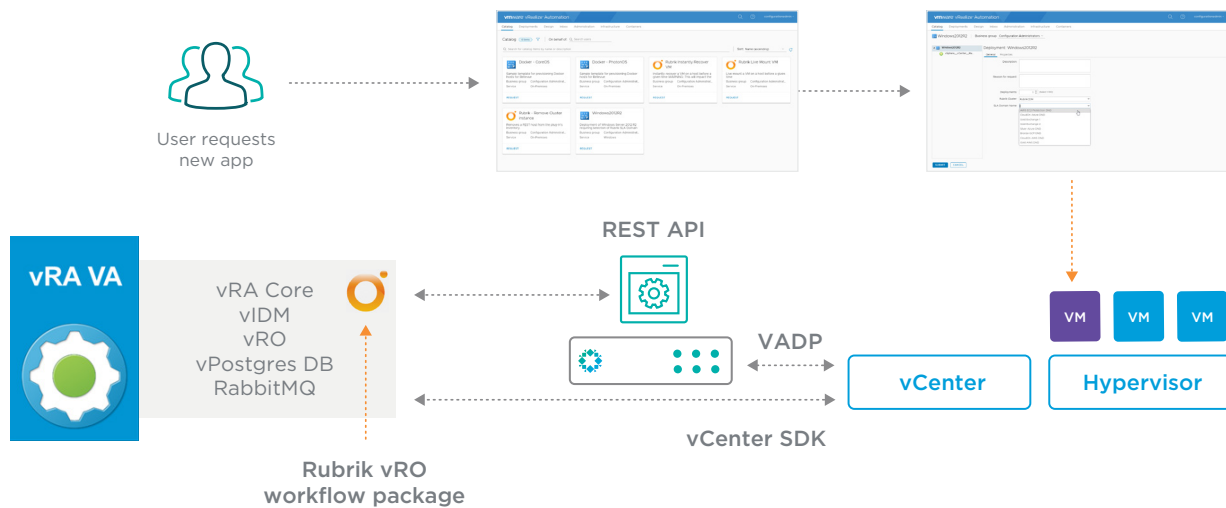
After the first full snapshot, the Rubrik cluster continues protection by creating incremental snapshots based on the change information provided by change block tracking (CBT). The Rubrik cluster creates each incremental snapshot very quickly because the snapshot only includes the data blocks that have 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 (Network Block Device over Secure Sockets Layer) 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

At the heart of Rubrik's architecture lies the SLA policy. The SLA policy reduces daily operational management by enabling a single policy engine to orchestrate the protection and management of services 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, allowing organizations to be as broad or as granular as they desire with data protection strategies.

End-to-end data management is provided by Rubrik for all vRA provisioned applications running on vSphere. End users are able to utilize a self-service catalog to securely access data instantly, automate protection policies, and orchestrate data across their VMware environments.



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

- VMware Software Solutions
 - VMware vRealize Automation 7.5.x or later
 - VMware vRealize Orchestrator 7.5.x or later (embedded with vRA)
 - [Rubrik Plugin for vRealize](#)
 - VMware vSphere 6.5 or later
 - VMware vCenter 6.5 or later
 - VMware ESXi 6.5 or later
 - vCenter Server privileges
 - It is recommended that a custom vCenter role be created with only the minimum required privileges. [Appendix B](#) details all 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.
 - All required ports must be opened to allow communication throughout the vRealize Automation and vSphere software stacks. [Appendix A](#) details all required ports.
- Rubrik Cloud Data Management (CDM) 4.1 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

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 virtual machine
- 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 of VMs using native vSphere APIs but these workloads can be backed up using RBS.

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

SELF-SERVICE ARCHITECTURE

The main benefits of vRA are delivered through a secure web-based self service portal in which administrators, developers, end-users, and customers can request new IT services. vRA's self-service architecture bring efficiencies and automation through the lifecycle of new or existing deployments. Utilizing highly automated blueprints, organizations are able to deliver more efficient and flexible processes around lifecycle management, while maintaining the control and governance they require. This document provides many high-level examples of various catalog items which can be consumed and requested by end users including:

- Deployment of new vSphere workloads
- Integrating data protection into existing and new provisioning blueprints
- Self service instant recoveries of applications and workloads
- Self service point-in-time copies of production environments for testing and development
- Automated data protection for all self-serve deployments

The self-service architecture outlined in this document is driven by the tight integration between three software solutions; VMware vRealize, Rubrik CDM, and VMware vSphere. The following sections will dive deeper into the integrations and architectures of each software component.

VREALIZE AUTOMATION ARCHITECTURE

vRA flexible installation methods can be configured to support minimal, proof of concept environments or to orchestrate the largest, distributed enterprise configurations. vRA architectures most commonly consist of the following components:

- vRA software packed as a virtual appliance
- One or more Infrastructure as a Service (IaaS) server(s) hosting a variety of services

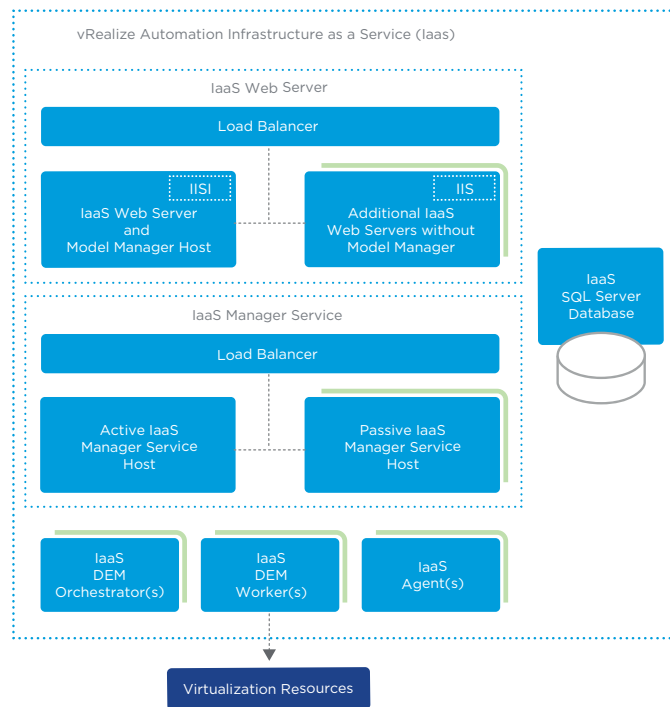
vRA Appliance

The vRA appliance, like most VMware delivered solutions, is a pre-packaged Linux system delivered as an open virtualization file (.ovf) which can be deployed in an organization's existing vSphere environment. The main functionality delivered via the vRA virtual appliance is as follows:

- Contains the server which hosts the vRA product portal that users utilize for the provisioning and management of resources
- Manages single sign-on (SSO) to authenticate and authorize valid users
- Hosts a management interface for configuring the vRA appliance settings
- Includes a preconfigured PostgreSQL database used for internal operations
- Includes a preconfigured instance of vRealize Orchestrator
- Hosts a location allowing IaaS servers to download management agents

IaaS Servers

Aside from a vRA appliance, every vRA deployment must contain one or more IaaS servers. IaaS servers work together to provide a number of different provisioning capabilities and can reside on either physical or virtual instances of Microsoft Windows Server. While installing all or a mixture of different IaaS components is possible, for larger environments it's recommended to break out the installation as much as possible to provide a highly available solution.



IaaS servers provide the following functionality, either installed together, or on separate instances:

Web Server

The IaaS Web Server provides infrastructure administration and service authoring to the vRA interface.

Model Manager

The IaaS Model Manager provides services and utilities for persisting, versioning, security and distributing model elements which facilitate integration with external systems and databases. Model Managers are hosted on one of the IaaS Web Servers.

Manager Service

The IaaS Manager Service is a Windows service which that manages the communication between all IaaS Distributed Execution Managers, the SQL Server database(s) for IaaS, installed agents, and SMTP. While the Manager Service may be installed and stopped on multiple IaaS instances in order to provide high-availability, it must only be running on one IaaS at any given time.

SQL Server Database

The SQL Database is used to maintain all information about the machines which IaaS manages. Depending on an organization's requirements, the database can be created automatically during installation or alternatively created separately to abide by any site policies.

Distributed Execution Manager

The Distributed Execution Manager (DEM) is commonly installed on the same IaaS server as the Manager Service and processes the business logic of any custom create models within vRA. Depending on the size and scope of the vRA deployment, the DEM may be installed on multiple or a single IaaS server(s). Each DEM is assigned one of two roles:

- **DEM Worker** - Used to solely run workflows and nothing else. Multiple DEM workers can be utilized to increase capacity and efficiency, and can be installed on separate or even the same server.
- **DEM Orchestrator** - Used to oversee and monitor the DEM workers deployed. This including the creation, scheduling, and failover of workflows on the DEM worker instances. The DEM Orchestrator also performs a check on various preconditions on each workflow before it is scheduled to be executed.

Agents

vRA makes use of agents in order to integrate with external systems. Like DEM, agents are commonly installed alongside the the Manager Service, however this isn't a strict requirement. Multiple agents can be installed on separate servers in order to provide maximum efficiency and availability in order to meet even the largest organizations requirements.

Agents facilitate the communication between vRA and IaaS to endpoints. For instance, a vSphere agent is used to create and manage virtual machines on VMware vSphere whereas a Hyper-V agent is utilized to manage Microsoft Hyper-V provisioning actions. Agents come in many different types, but out of the box vRA supports the following:

- **Virtualization Proxy Agents** - used to manage virtual machines on virtualization hosts such as VMware ESXi, Citrix XenServer, and Microsoft Hyper-V.
- **Virtual Desktop Integration Agents** - used to integrate with external virtual desktop systems such as VMware Horizon or XenDesktop.
- **External Provisioning Integration Agents** - allows vRA to integrate external systems into the provisioning process utilizing PowerShell
- **Windows Management Instrumentation (WMI) Agent** - enables vRA to enhance its ability to control, monitor, and gather data from managed Microsoft Windows Servers.

Integration with vRealize Orchestrator

The workflow engine sitting beneath the vRA interface is vRealize Orchestrator (vRO). For many years, vRO has been packaged as a completely separate product than vRA. In fact, even to this day, vRO can be used to automate and orchestrate workflows without the existence of vRA at all. That said, pairing the self service and approval functionality within vRA with the customization and workflow execution capabilities of vRO make them a perfect combination to deploy a complete end-to-end provisioning and management solution.

Although it's possible to run vRO completely separate from vRA it's now recommended for customers to utilize the embedded vRO instance included within vRA. The vRO instance included with vRA is fully functional, and contains no differences as the standalone deployment.

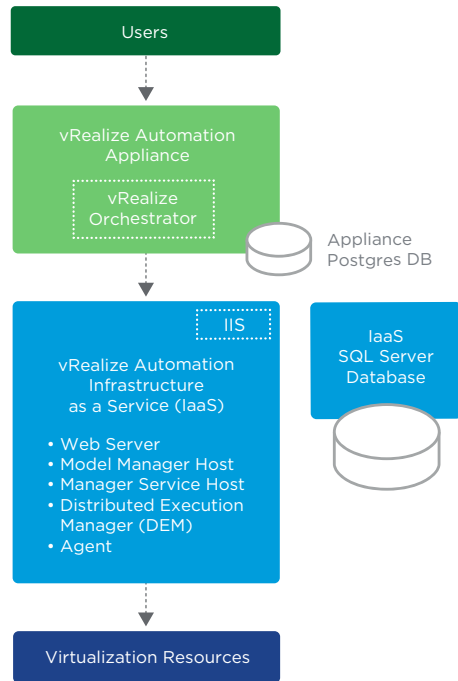
vRO is the main workflow execution engine for vRA, hosting integrated and third-party workflows and actions allowing vRA to broaden its provisioning capabilities, integrating with many different technologies across on-premises datacenter and cloud.

vRA Deployment Types

vRA can be installed to meet the smallest or largest of environments, including everything from a small, proof-of-concept deployment to a large distributed, enterprise environment.

Minimal Deployment

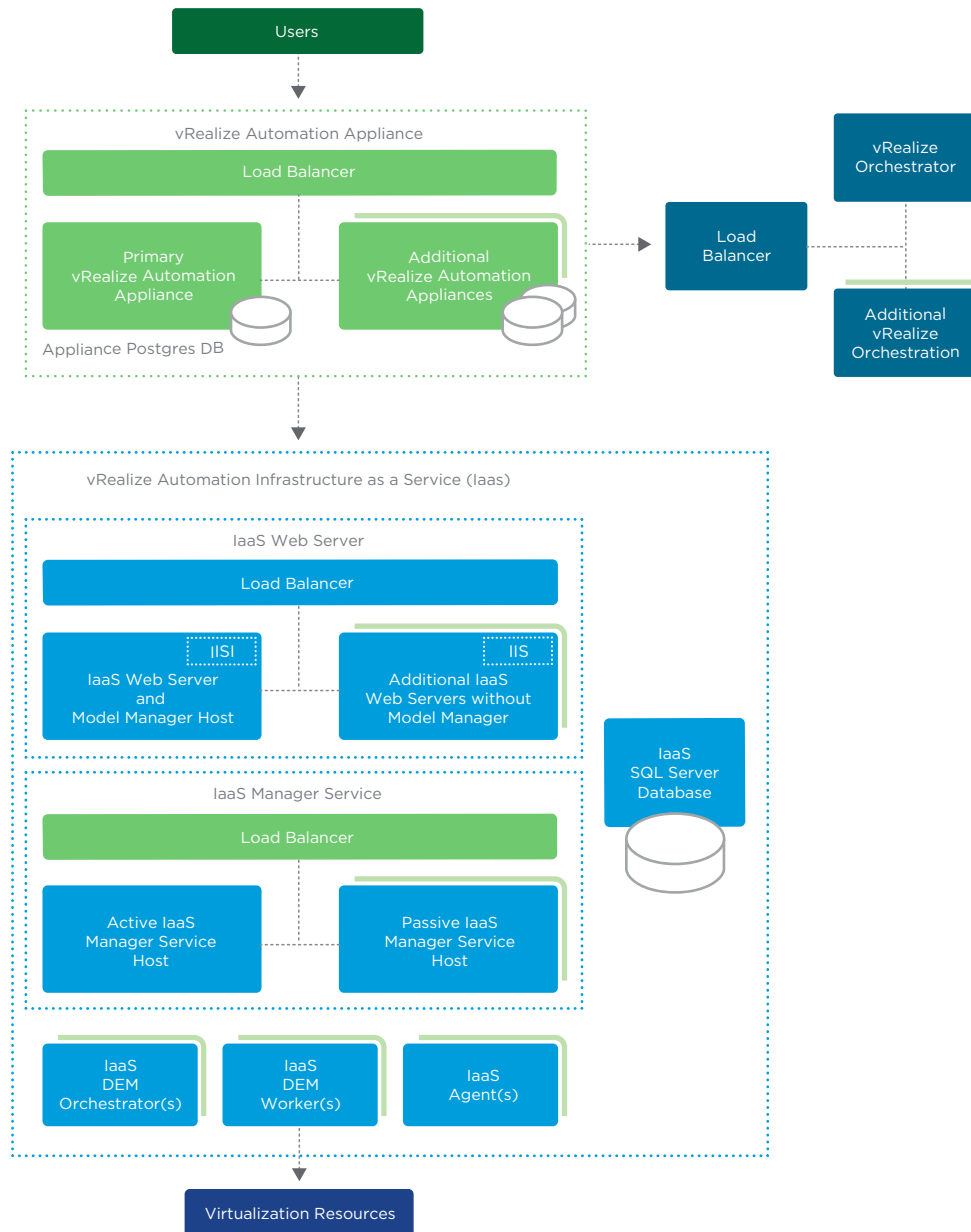
A vRA minimal deployment consists of two components: one vRA Appliance and one Microsoft Windows Server which hosts all of the IaaS components. During the installation of a vRO minimal deployment the SQL Server Database for the IaaS components can be installed either on the same server as the IaaS services, or on an external server of your choosing.



It should be noted that a minimal deployment is not able to be converted or upgraded to an enterprise distributed deployment. For these reasons, minimal deployments should only be used for small proof of concept designs which will remain static in nature and require no scaling. If wishing to start small and scale, it is recommended to begin with a small distributed deployment and add components as necessary.

Distributed Deployment

Distributed enterprise deployments of vRA come in various sizes. For instance, a basic model may contain one vRA appliance paired with many Windows servers hosting the different IaaS services. That said, any enterprise deployment needs to consider availability and scale - with vRA this is as simple as deploying more vRA appliances and splitting the load of IaaS services across many different Windows instances.



The image above illustrates a highly available, scalable distributed enterprise deployment of vRA. Multiple vRA appliances work together through a load balancer in a fault tolerant manner, along with multiple IaaS services spread across many Windows servers.

For the purposes of this paper, a small distributed deployment of vRA is utilized containing the following components:

- 1 - vRA Appliance with embedded vRO and Postgres database
- 1 - Windows 2012 R2 VM hosting all IaaS components
- 1 - external Microsoft SQL Server database instance hosting IaaS database

RUBRIK INTEGRATION WITH VRA

It is easy to integrate Rubrik CDM functionality and features into a vRA service catalog. Simply import the Rubrik Plugin for vRealize into vRO. The Rubrik API-first architecture enables ease of automation and integration. Once installed, custom vRA Property Definitions and Groups can be created and assigned to existing or newly created blueprints. Leveraging Rubrik with vRA brings many benefits to organizations such as:

- **Data Loss Prevention** - Ensure data protection is a key stakeholder in your VM Lifecycle management and provisioning processes by requiring newly created VMs to be protected.
- **Self-Service Instant Recovery** - One click to restore files to any point-in-time and the ability to resume operations within minutes of an attack with self-service file-level recovery.
- **Streamline Operations** - Eliminates lengthy wait times at the help desk with self-service access to backup and restore operations.
- **Align with Business SLAs** - Automate compliance policies in just a few clicks to adhere to strict business requirements.

Rubrik Plugin for vRealize

The Rubrik Plugin for vRealize contains a number of pre-built workflows, actions, and configuration elements which may be imported into vRO. Once imported, vRA administrators are able to easily integrate data management processes into self-service catalogs such as:

- Assigning a VM to an SLA Domain
- Performing On-Demand Backups of a VM
- Instantly Recovering VMs to a specified point-in-time
- Live Mounting point-in-time copies of VMs for testing and development

The Rubrik Plugin for vRealize is shipped as a standalone [download](#), freely available for anyone to utilize. Integrating Rubrik and vRA relies solely on one prerequisite: the presence and enablement of vRO's built-in HTTP-REST plugin. Other than ensuring the HTTP-REST plugin is enabled there is no other physical interaction or configuration required, as the Rubrik Plugin for vRealize contains all of the needed workflows to install and configure itself, including the creation of a HTTP-REST inventory object. This gives administrators the ability to quickly move into "Day 2" operations integrating Rubrik with their vRA environment.

Note: In addition to the Rubrik Plugin for vRealize, a number of pre-built vRA blueprints showcasing Rubrik integration may also be imported. For additional details around utilizing the pre-built Rubrik vRA blueprints, see the Rubrik Plugin for vRealize Quickstart located on the [Rubrik vRealize GitHub Repository](#).

Rubrik APIs and Added Extensibility

The Rubrik Plugin for vRealize contains a number of pre-built workflows and actions built solely around VMware vSphere technology, however, that's not to say that it isn't able to be extended to other areas within the data center. Over the years, vRA has expanded its data center footprint, supporting deployment and management of individual applications both on-premises and in the cloud - why not enhance these deployments by integrating Rubrik protection as well?

Although there is no pre-built workflows around items like SQL Protection or EC2 Protection, the Rubrik functionality is exposed through the extensible and feature-rich APIs Rubrik provides. Utilizing the HTTP-REST plugin and the created Rubrik inventory objects, administrators are able to take advantage of any of the Rubrik functionality by simply scripting around and consuming the available APIs. For instance, vRA administrators can build and offer custom blueprints around the deployment of EC2 instances or SQL databases, all the while offering automated Rubrik protection.

For the purposes of this paper the following Rubrik architecture was utilized:

- Rubrik Cluster consisting of 4 nodes running 4.2.x
- Rubrik Plugin for vRealize

vSphere Integration

Due to Rubrik's consumption of the vSphere Storage APIs for Data Protection (VADP), no special considerations are required to integrate Rubrik with a vSphere environment. That said there are a number of built-in VMware features and Rubrik configurations which can be used in order to provide optimal performance:

- **Change Block Tracking (CBT)** - CBT is a VMware feature which tracks changes to storage blocks on a per-VM basis, allowing Rubrik to gain efficiency when performing its incremental forever backup approach.
- **Rubrik Backup Service (RBS)** - RBS can be installed on virtual servers to provide higher performance for file level recoveries and VSS quiescing.
- **Adaptive Data Consistency** - Rubrik takes an automated, tiered approach as it pertains to flushing I/O and data consistency. Using a tiered approach allows for the most consistent backup to always be taken without any additional configuration.
- **VM Linking** - VM Linking is a process that maintains VM Identities within the Rubrik cluster, even if they are migrated to and away from protected vCenter Server instances. This provides the most efficient use of capacity while streamlining the operations of data protection management, supporting the Rubrik design center around simplicity.

To learn more about these and many other features and how they integrate with Rubrik CDM see the [Rubrik and VMware vSphere Reference Architecture](#).

For the purposes of this paper the following was put in place in terms of vSphere architecture:

- vCenter Server Appliance with embedded PSC running VMware vCenter Server 6.5
- Management Cluster hosting the vRA virtual appliance, IaaS server, MS SQL Server, and vCSA
 - 2x ESXi 6.5 with both local and external storage running on a variety of platforms
- Test Cluster hosting provisioned vRA workloads
 - 4x ESXi 6.5 hosts running all-flash vSAN

OPERATIONAL OVERVIEW

Rubrik and VMware vRealize Automation 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 lightweight the Rubrik and VMware vRealize Automation joint solution is.

PROTECTING AND RESTORING VRA

All components of vRA can and should be protected. In the event of data corruption or loss, the vRA virtual appliance along with the IaaS Servers hosting the individual services are restorable from the backup copies.

It is recommended that regular backups are scheduled for all of the deployed vRA virtual appliances and IaaS Servers within the vRealize architecture.

Aside from taking regular backups of the vRA components, it is recommended to take backups prior to and after any upgrades, updating component certificates, or performing any major maintenance of any of the system components. The following table outlines the order in which backups should be performed, along with the respective backup and restore methods supported.

VSPHERE COMPONENTS		
Component	Backup Method	Restore Method
Platform Services Controller	Image level	Image level
vCenter Server Appliance	Image level	Image level
VRO COMPONENTS (IF STANDALONE)		
Component	Backup Method	Restore Method
vRO Virtual Appliance	Image level	Image level
IAAS COMPONENTS		
Component	Backup Method	Restore Method
Proxy Agents	Image level	Image level
DEM Workers/Orchestrator	Image level	Image level
Manager Services	Image level	Image level
Web Server(s)	Image level	Image level
VRA COMPONENTS		
Component	Backup Method	Restore Method
vRA Database	Database level	Database level
vRA Virtual Appliance	Image level	Image level
SUPPORTING DATABASE(S)		
Component	Backup Method	Restore Method
IaaS MS SQL Database	Database level	Database level
vRO Postgres Database (optional, if applicable)	Database level	Database level
vRA Postgres Database (optional)	Database level	Database level

vSphere Recoverability

The Rubrik cluster accesses VM data through a connection to the VMware vCenter Server that manages the hypervisor from where the VM is running. 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 (with proper permissions; details can found at Appendix B)
- 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, populating any newly created VMs in the Rubrik inventory. 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 anytime the Rubrik **create snapshot** task is initiated. A full refresh can be manually initiated at any time.

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

vRO Recoverability

It is recommended to utilize the embedded vRO instance within the vRA virtual appliance. If the embedded instance is utilized, there is no need to perform additional steps to backup and recover vRO as it is all processed during the vRA recovery operations.

However, customers may wish to keep their vRO instances separate due to sizing constraints, upgrade paths, or overall preference. When utilizing a separate instance, backup and restore processes of vRO should be processed before recovering the vRA IaaS components.

Rubrik may be used to provide automated image level protection and restoration of a standalone vRO virtual appliance or when embedded as a part of the vRA virtual appliance.

IaaS Recoverability

The IaaS services within a vRA environment should be backed up in the following order:

- Proxy Agents
- DEM Workers
- DEM Orchestrator
- Passive Manager Service (if applicable)
- Active Manager Service
- Web Nodes

If using the minimal deployment or hosting multiple IaaS components on the same server, the order should be selected considering the server hosting the latter component on the list.

When running IaaS components as a VM, Rubrik may be used to provide automated image and file level protection and restoration. For IaaS components running on a physical Windows workload, Rubrik may be used to provide automated volume and file level protection and restoration.

vRA Recoverability

If running vRA in a highly available architecture, secondary vRA virtual appliance nodes should be backed up first before the master node.

Rubrik may be used to provide automated image level protection and restoration of the master vRA virtual appliance node as well as any deployed secondary vRA virtual appliance node(s).

Supporting Database(s) Recoverability

Depending on the chosen deployment method a number of databases need to be protected along with the components, services and appliances supporting vRA. These database may include one or all of the following:

- IaaS MS SQL database
- Standalone vRO database(s)
- Embedded or external vRA/vRO Postgres database

Note: It is recommended to process the backups of all the supporting databases as nearly simultaneously as possible to prevent or minimize data loss. Also, when restoring, it's recommended to restore all supporting databases to a point-in-time which are consistent with each other.

IaaS MS SQL Database

The IaaS component of vRA utilizes a Microsoft SQL Server database to store and maintain information about the machines it manages, as well as some of its own policies and configurations.

As a guideline, backups of the IaaS database should be performed during a time when activity is minimized, and, when possible, ensure all IaaS workflows are complete and the IaaS services have been stopped.

Rubrik may be used to provide automated database level protection and recovery of the IaaS database, as well as image level protection of the server the database is installed on. Protecting the IaaS database with database level protection allows for a quick recovery in the event only the database has been lost or corrupted.

vRO Database

For environments leveraging the embedded vRA database, there is no further action required to ensure the database elements of vRO are protected because of the inclusion within the vRA database.

For environments leveraging a standalone instance of vRO, follow the in-house procedures for backing up and restoring the vRO database. Depending on the type of database used, MS SQL or Postgres, Rubrik may be used to provide automated database-level, file-level, or image-level backup and recovery of the vRO database.

vRA Postgres Database

While the recommended approach to backing up the vRA Postgres database is to create image level backups of the complete appliance it is also possible to create backups of just the database itself. The following script exports a backup of the embedded vRA Postgres database to the `/tmp` directory:

```
service vcac-server stop
su -m -c "/opt/vmware/vpostgres/current/bin/pg_dumpall -c -f
/tmp/vcadb.sql" postgres
service vcac-server start
```

When using an external vRA Postgres database to support legacy systems, follow the in-house procedures for backing up a Postgres database.

Once exported, database backup files can then be placed in a location which can take advantage of Rubrik's fileset backup capabilities, or be scripted to utilize Rubrik Elastic App Services, stored on Rubrik volumes optimized for performance and capacity as it relates to Postgres backup data.

Protecting and Restoring vRA Provisioned Resources

Empowering end-users with a self service catalog brings many benefits to organizations. No longer are employees or customers waiting on lengthy approval requests to be handled before resources can be provisioned. The days of administrators and organizations wasting resources having to manage the lifecycle of the resources which have been provisioned are over. Self-service can also bring many challenges to organizations as well, especially in the data protection realm.

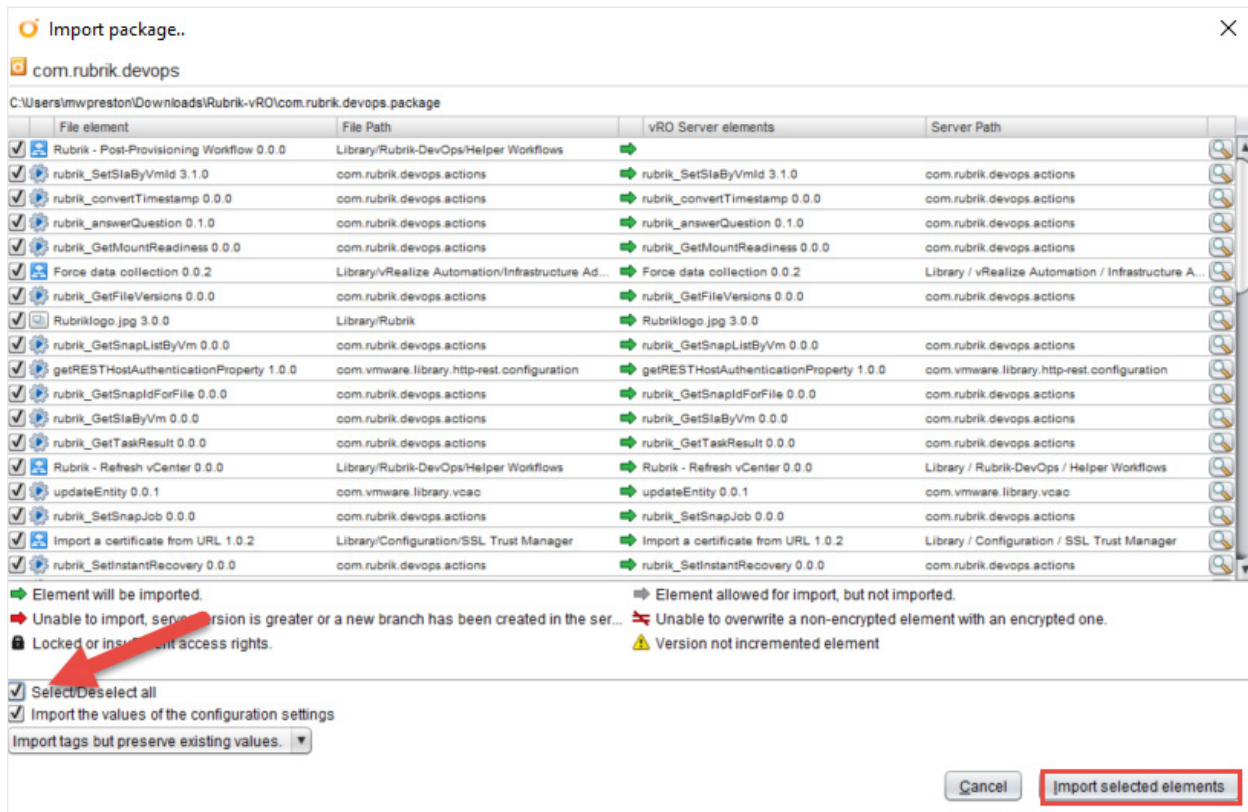
With the power in the end-users hands, often data protection is either pushed to the background, or not thought of at all, which makes it easy for things to get out of control. Rubrik provides the ability to tame this "wild west" by applying a data protection policy on a broad level, such as a vCenter Server, folder, host, or cluster and have those define RPOs inherited by child workloads. Often, a broad, less restrictive policy is applied as a catch-all to all current and new workloads (See [Automated Data Protection Use Case](#)), while allowing or requiring end-users to override the default SLA Policy with a stricter, more protective policy (See [Provision and Protect Use Case](#)). Furthermore, as deployed VMs are expired or destroyed, a Retention SLA is able to ensure backups are maintained for a specified number of days after, giving organizations a second chance to restore workloads in the event of a misconfigured or misunderstood vRA lifecycle policy.

This automatic protection mechanism allows end-users to get the agility and efficiencies that self-service provides, all the while giving the organization the power and piece of mind that their data is protected and abiding by any compliance or governance restrictions.

Integrating Rubrik CDM with vRA

As previously mentioned, the vRA integration with Rubrik CDM leverages a custom-built plugin for vRealize containing a number of pre-built workflow and configuration items.

After [downloading](#) the Rubrik Plugin for vRealize, it is simply imported into vRO. During this process a dialog displaying all of the included workflows and configuration items is displayed. It is recommended to ensure that each item is selected, allowing all items to be imported to vRO.

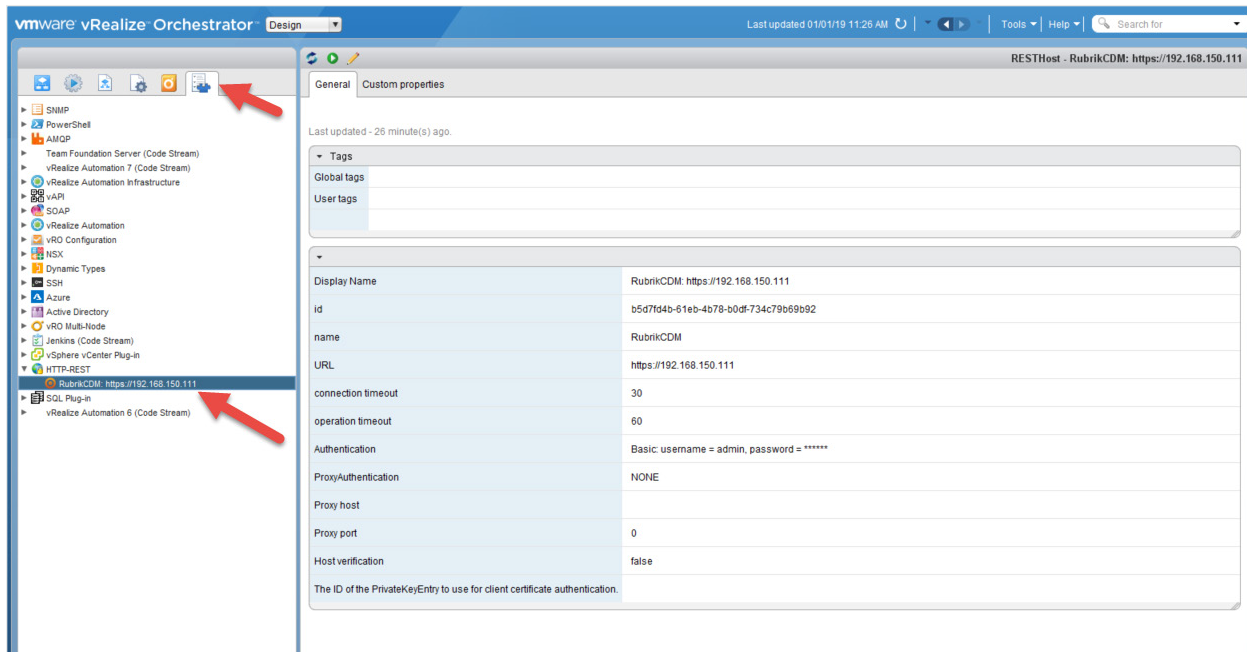


Once imported, the next step is to execute the Rubrik - Add Cluster Instance workflow. This workflow receives cluster and credential information as input and outputs an HTTP-REST inventory object pointing to the specified Rubrik cluster.

The screenshot displays the VMware vRealize Orchestrator interface. The top navigation bar shows the user 'administrator @ mpreston-via.rubrik.us' and the current workflow 'Rubrik - Add Cluster Instance'. The left sidebar contains a library of workflows, with 'Rubrik - Add Cluster Instance' selected. The main workspace shows a workflow diagram with steps: 'Get host from base', 'Custom decision', 'Import a certifi...', 'Add a REST host', and 'Import a certifi...'. The 'Logs' tab is active, showing a series of messages:

```
./Messages
[2019-01-01 11:44:19.002] [https://192.168.150.111]
[2019-01-01 11:44:19.498] [REST host: DynamicWrapper (instance): [RESTHost]-[class com.vmware.o11n.plugin.rest.RESTHost] -- VALUE: RESTHost [name=RubrikCDM, id=null, uri=https://192.168.150.111, connectionTimeout=30, operationTimeout=60, hostVerification=false, proxyHost=, proxyPort=0, schemaRepo=null, resourceId=null, privateKeyId=null]
[2019-01-01 11:44:19.500] [REST host authentication: DynamicWrapper (instance): [HTTPBasicAuthentication]-[class com.vmware.o11n.plugin.rest.HTTPBasicAuthentication] -- VALUE: Basic: username = admin, password = *****
[2019-01-01 11:44:19.716] [REST host added: DynamicWrapper (instance): [RESTHost]-[class com.vmware.o11n.plugin.rest.RESTHost] -- VALUE: RESTHost [name=RubrikCDM, id=2334b99c-410a-4552-b0c2-ce8197b463e, uri=https://192.168.150.111, connectionTimeout=30, operationTimeout=60, hostVerification=false, proxyHost=, proxyPort=0, schemaRepo=com.vmware.o11n.plugin.rest.schema.SchemaRepository@1fb34ee2, resourceId=null, privateKeyId=null]
```


Upon successful completion the workflow will add the desired Rubrik cluster instance as a RESTHost within vRO. This **RESTHost** can then be used as input for the various Rubrik related workflows.



The Rubrik Plugin for vRealize is now installed, configured, and ready to be consumed through vRA catalog items. For more detailed information around installing and configuring the Rubrik Plugin for vRealize see the [Rubrik Plugin for vRealize Quickstart](#) on GitHub.

USE CASES AND SAMPLE WORKFLOWS

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

Provision and Protect

vRA provides complete application lifecycle management, automating everything from the provisioning and initial configuration to the decommissioning stage. Coupling vRA with Rubrik allows IT organizations to inject automated routines focusing on data management into these lifecycle management operations. The most common use case in pairing vRA with Rubrik focuses around requiring the designation of an SLA Domain during the provisioning request. This use case, often called Provision and Protect, ensures that data protection policies are not an afterthought as new workloads are created. Instead, new workload requests require the selection of a Rubrik SLA Domain, giving IT Operations the guarantee that any data provisioned through vRA's self service catalog is protected and available.

Applying the Provision and Protect use case to new or existing blueprints can be broken down into three main sections; Creating Rubrik SLA Domain Property Definitions and Groups, Configuring blueprints to require the SLA property definition, and creating an associated workflow subscription. The following will outline each step in more detail.

Creating Rubrik SLA Domain Property Definitions and Groups

Property definitions within vRA allow for the input of custom information from an end-user during a self-service request. Then this data can be utilized throughout the provisioning workflow as variables for specific actions or processes. To support a Provision and Protect use case with Rubrik, new Property Definitions named rubrik.cluster and rubrik.sla_name are created.

The screenshot shows the 'Create Property Definition' form in vRealize Automation. The form is titled 'Create Property Definition' and is located in the 'Administration' section. The left sidebar shows a navigation menu with 'Property Dictionary' expanded to 'Property Definitions'. The form fields are as follows:

- Name:** rubrik.cluster (Text input)
- Label:** Rubrik Cluster (Text input)
- Visibility:** All tenants, This tenant
- Description:** (Empty text area)
- Display order:** 1 (Dropdown menu)
- Data type:** String (Dropdown menu)
- Required:** No (Dropdown menu)
- Display as:** Dropdown (Dropdown menu)
- Values:** Static list, External values
- Enable custom value entry:**
- Script action:** SELECT... (Dropdown menu)

Buttons for 'OK' and 'CANCEL' are located at the bottom of the form.

The screenshot shows the 'Create Property Definition' form in vRealize Automation. The form is titled 'Create Property Definition' and is located in the 'Administration' section. The left sidebar shows a navigation menu with 'Property Dictionary' expanded to 'Property Definitions'. The form fields are as follows:

- Name:** rubrik.sla_name (Text input)
- Label:** SLA Domain Name (Text input)
- Visibility:** All tenants, This tenant
- Description:** (Empty text area)
- Display order:** 2 (Dropdown menu)
- Data type:** String (Dropdown menu)
- Required:** No (Dropdown menu)
- Display as:** Dropdown (Dropdown menu)
- Values:** Static list, External values
- Enable custom value entry:**
- Script action:** SELECT... (Dropdown menu)

Buttons for 'OK' and 'CANCEL' are located at the bottom of the form.

During the creation of the property definition the available cluster names and SLA Domain names may be manually entered, or subsequently, a vRO action can be utilized to retrieve a real-time list of SLA Domains available. For more information on creating vRO actions to perform SLA Domain retrieval, see the [Provision and Protect Use Case on GitHub](#).

A new property group is now created with the `rubrik.cluster` and `rubrik.sla_name` property definitions added as a member properties. In addition, the built-in vRA `Extensibility.Lifecycle.Properties.VMPSMasterWorkflow32.MachineProvisioned` property is added with a value of “`__*,*`” in order to define at which point in the provisioning process the properties are passed to vRO.

The screenshot shows the VMware vRealize Automation Administration console. The main content area is titled "Edit Property Group: CustomVMRequest". The "Name" field is set to "CustomVMRequest", the "ID" is "VMRequest", and the "Visibility" is "This tenant". The "Description" field is empty. Below the description is a table of properties:

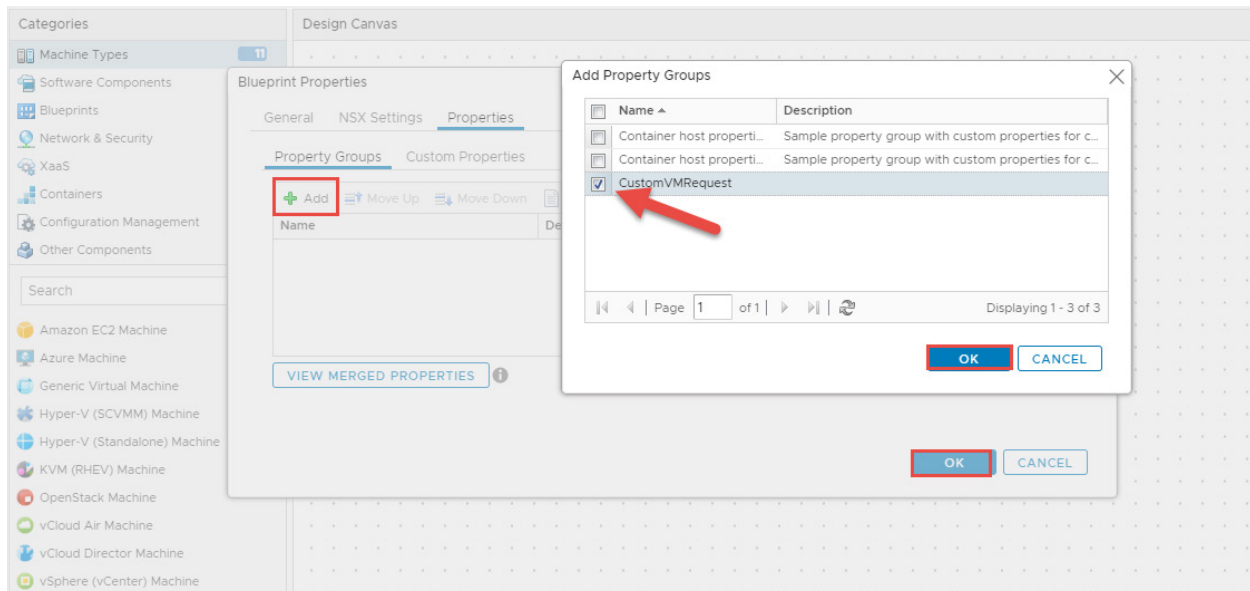
Name	Value	Encrypted	Show in Request
Extensibility.Lifecycle.Properties.VMPSMasterWorkfl...	__*,*	No	No
rubrik.cluster		No	Yes
rubrik.sla_name		No	Yes

The "rubrik.sla_name" row is highlighted with a red box. Below the table, there is a note: "To avoid conflict with vRealize Automation properties, use a prefix such as a company or feature name followed by a dot for all custom property names." At the bottom of the form, there are "OK" and "CANCEL" buttons.

The property definition and groups are now configured and can be assigned to a new or existing blueprint.

Assigning Property Definitions to New or Existing Blueprints

In order to have vRA process the newly created properties during a catalog item request the property group must be associated with a blueprint. These can be applied to either a newly created or existing blueprint. The CustomVMRequest property group is now assigned to a blueprint within vRA.



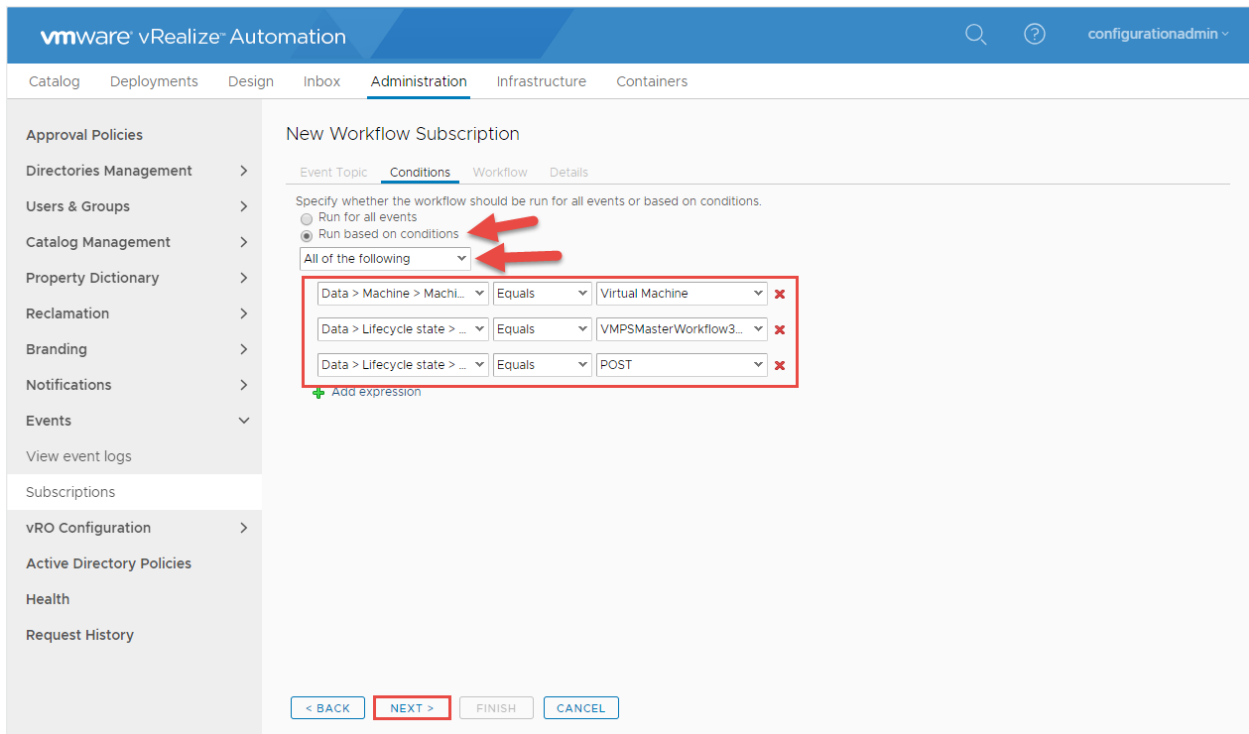
The property definition and groups are now configured and can be assigned to a new or existing blueprint.

Creating a Workflow Subscription

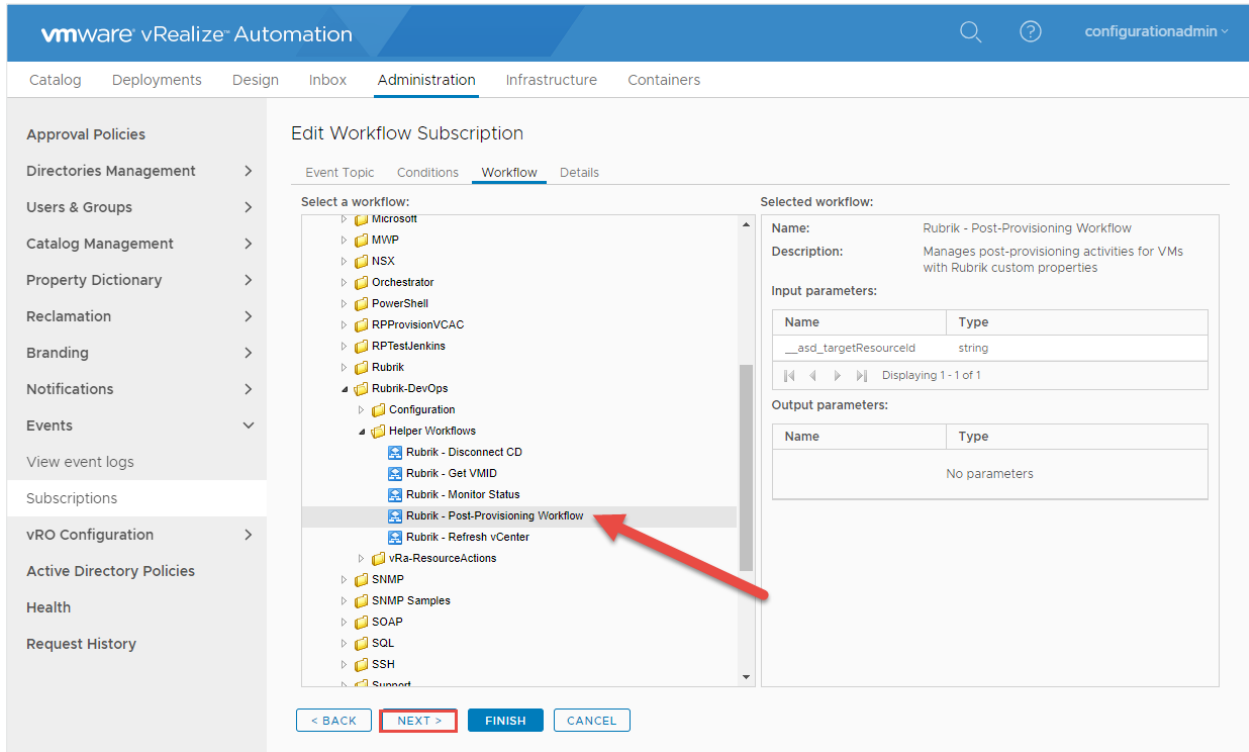
A Workflow Subscription is used to instruct vRA to execute a specified vRO workflow during a specific state of the request. The Provision and Protect use-case requires the request to execute the **Rubrik Post Provision State** workflow after the VM has been provisioned. The **Rubrik Post Provision State** workflow then receives the selected SLA Domain property and makes the appropriate API calls to Rubrik CDM to assign it to an SLA Domain.

The workflow subscription requires the following three clauses be defined to run based off all conditions.

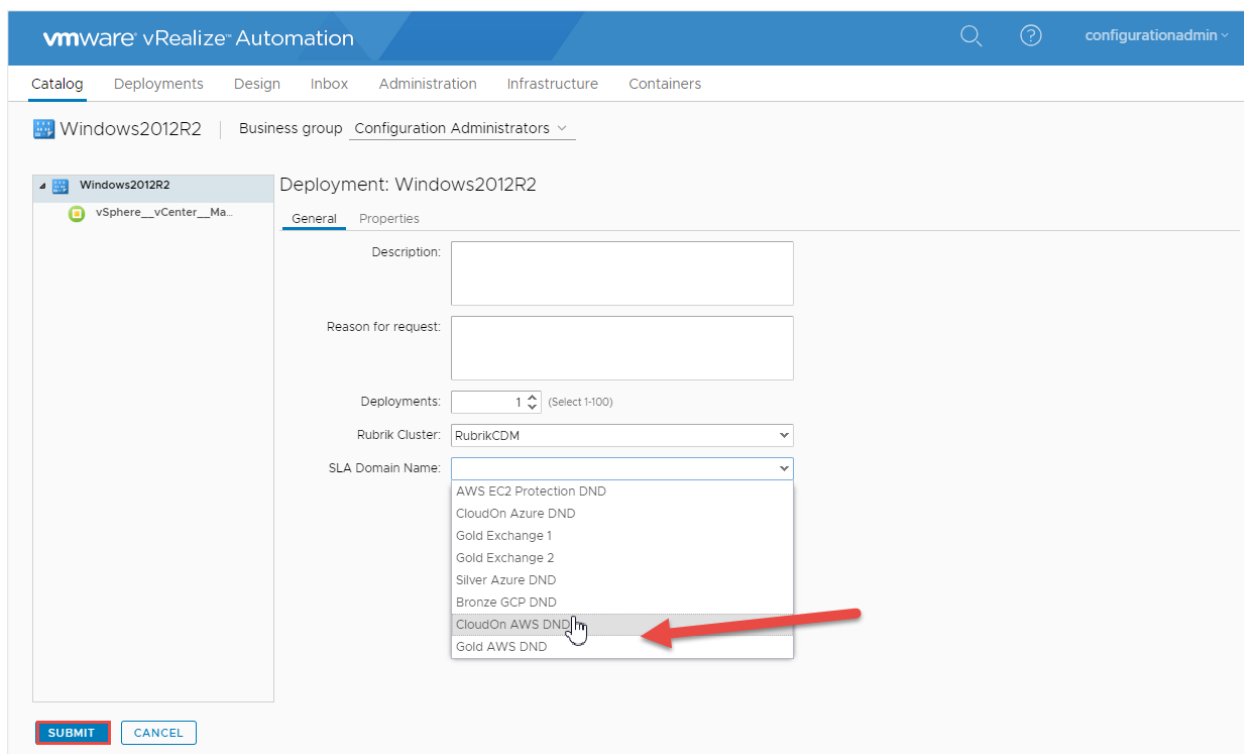
Clause	Operator	Value
Data > Machine > Machine Type	Equals	Constant > Virtual Machine
Data > Lifecycle state > Lifecycle state name	Equals	Constant > VMPSMasterWorkflow32.MachineProvisioned
Data > Lifecycle state > State phase	Equals	Constant > POST



The Rubrik – Post Provisioning Workflow workflow is then selected to execute based off the above defined conditions.



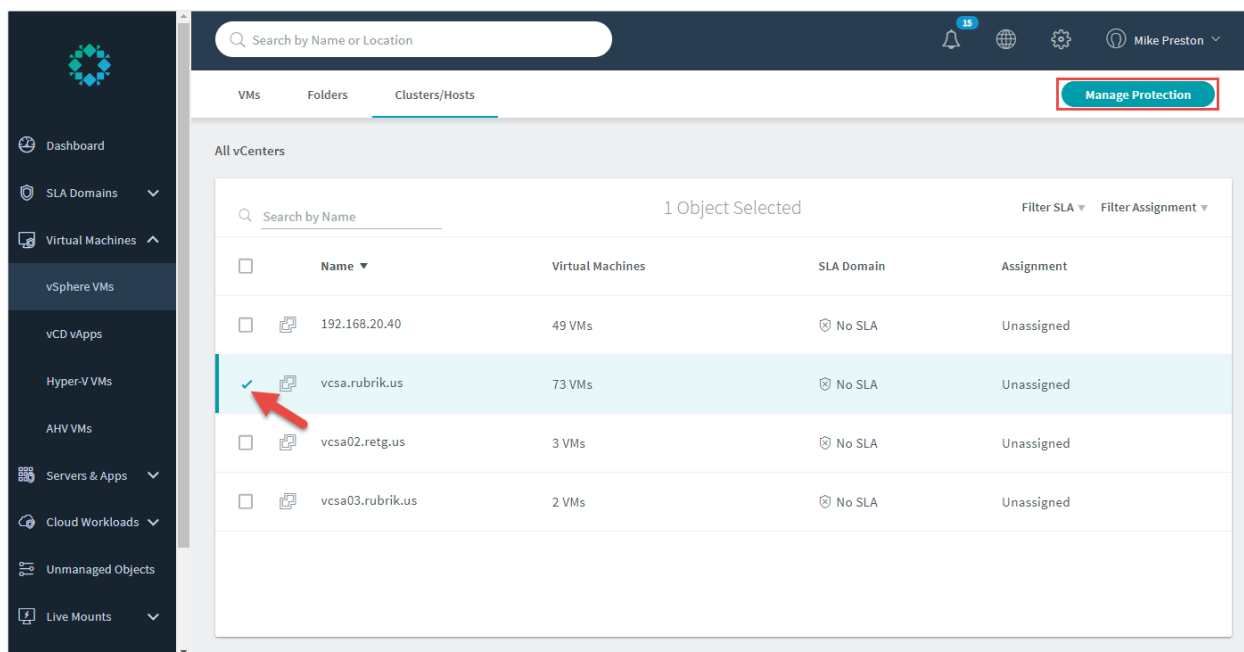
The Provision and Protect Use Case is now configured. End-users are now required to select a Rubrik SLA Domain to assign to the soon-to-be provisioned VM during the item request process.



Automated Data Protection

vRA brings many benefits to organizations in terms of self service and VM provisioning, however all too often the provisioning process is narrowly focused and does not include data protection. While it is possible to require end users to select a Rubrik SLA upon provisioning, it is not always ideal or even permitted within organizations with strict compliance policies. This results in many deployed resources not belonging to any SLA Domain and going unprotected until backup administrators manually intervene.

Rubrik solves this dilemma by allowing backup administrators to broadly assign policies and enable inheritance by applying an SLA Domain on a higher level object being used by vRA for VM deployments, such as the vCenter Server, folder, host, or cluster levels. This means that VMs will automatically inherit their respective data protection policy by being provisioned into these targeted areas protected by an SLA Domain. Rubrik's data control plane detects any newly added or provisioned VMs and automatically applies the policy assigned to its parent, eliminating the need for any manual administrator interaction.



The process to integrate Automated Data Protection in vRA is as follows:

1. Rubrik SLA Domain is assigned to the cluster into which vRA is deploying managed resources.
2. Upon deploying a new VM, the Rubrik SLA Domain assigned to the parent cluster is automatically inherited by the newly provisioned VM.
3. The newly provisioned VM will adhere to the user-defined policies and configurations within the parent SLA Domain.

Note: In the event that a Rubrik SLA policy has been assigned directly to an individual VM that also auto-inherits a policy from a higher level object, a conflict resolution will occur allowing administrators to resolve the conflict to suit their data protection needs.

Once the policy has been inherited and assigned to the newly provisioned VM, Rubrik will ensure adherence to the user-defined configuration and settings such as frequency, retention, archival and replication. Rubrik's Automated Data Protection strategy to protect vRA workloads completely eliminates any manual configuration, ensuring data protection is not an afterthought, even if ignored.

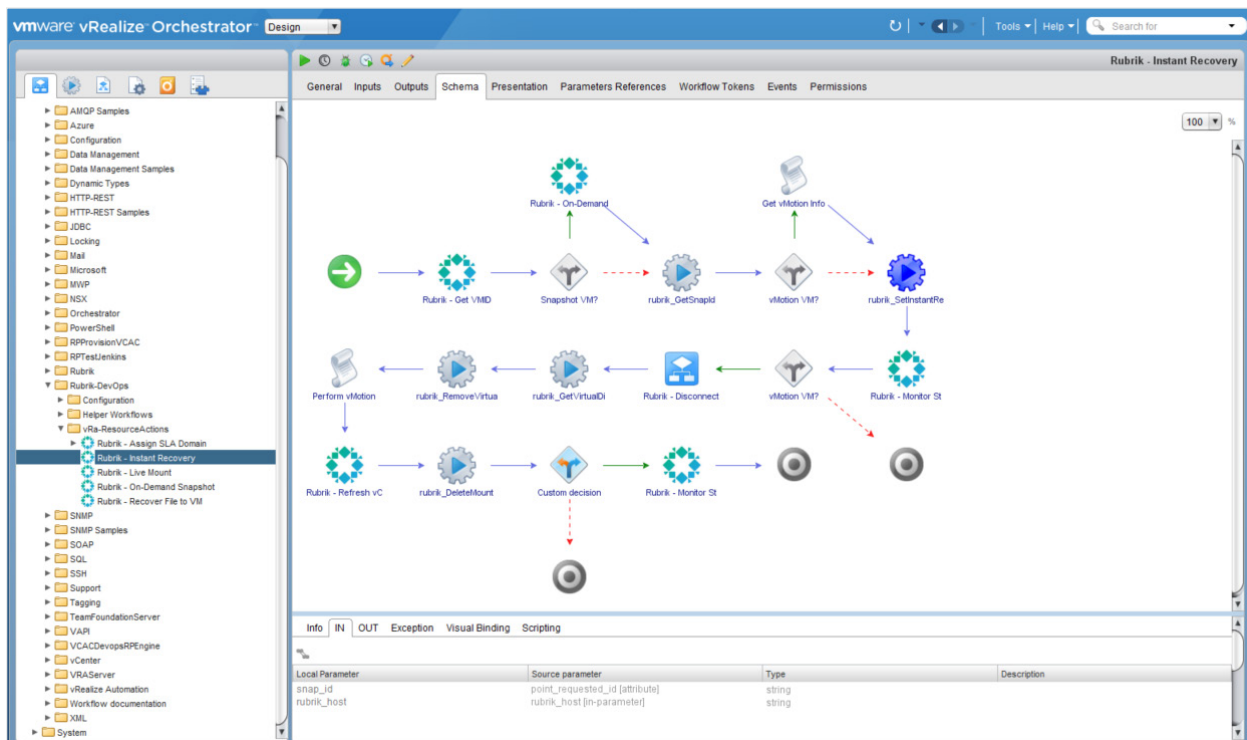
Self-Serve Recovery

Imagine not having to wait for backup administrators to get around to restore requests when looking to restore data to production or for testing and development purposes. The Rubrik Plugin for vRealize contains a number of workflows that can be integrated into vRA's XaaS designs in order to provide a variety of options pertaining to self-serve recovery. This section will outline a couple of workflows included within the Rubrik Plugin for vRealize to support self-serve recovery scenarios; Self-Serve Instant Recovery and Self-Serve Live Mount.

Self-Serve Instant Recovery

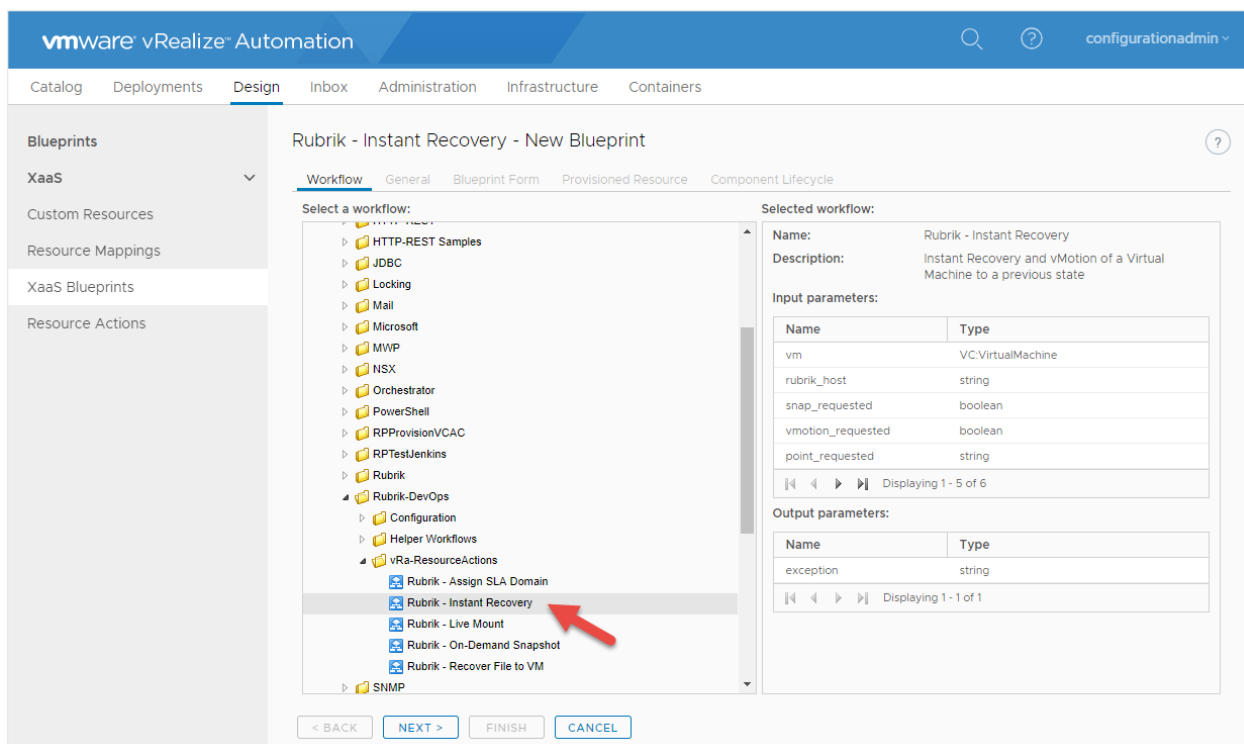
The built-in workflow, **Rubrik – Instantly Recover**, may be used to provide end users the ability to request that a VM be instantly recovered within Rubrik CDM. The process execution works as follows:

1. Workflow receives a VM, date (point-in-time for recovery), target host, whether to back the VM up before processing, and whether to vMotion the VM afterwards as input.
2. Workflow authenticates to the Rubrik cluster.
3. Appropriate API calls are made to perform optional on-demand snapshots, an Instant Recovery and an optional Storage vMotion of the VM.
4. Workflow waits for the Instant Recovery job to fully complete before reporting success.

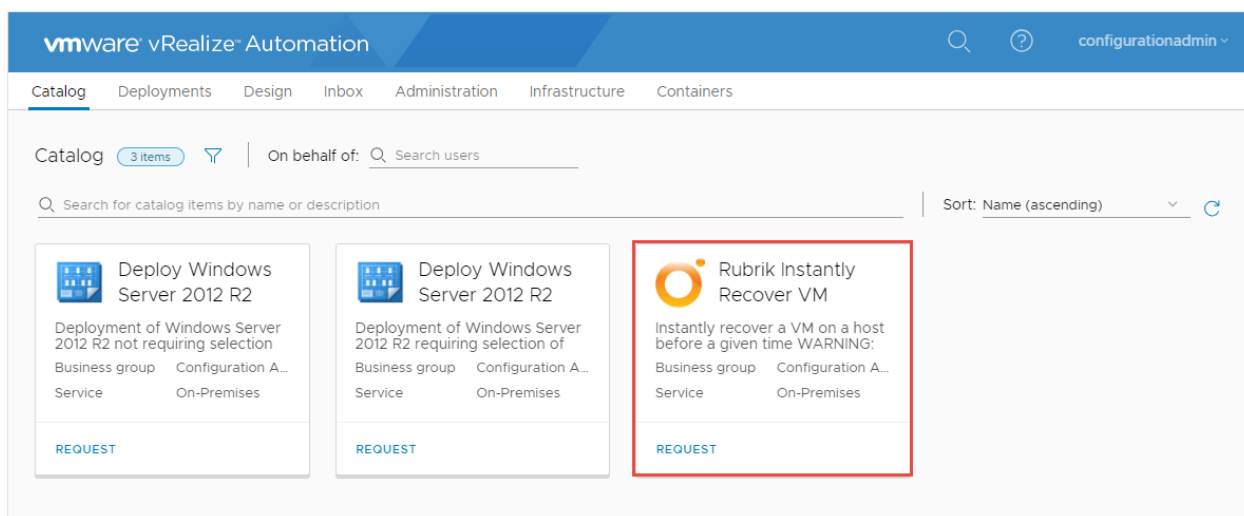


A new XaaS blueprint can be easily configured within vRA by selecting the **Rubrik – Instantly Recover** and ensuring the blueprint has a status of Published.

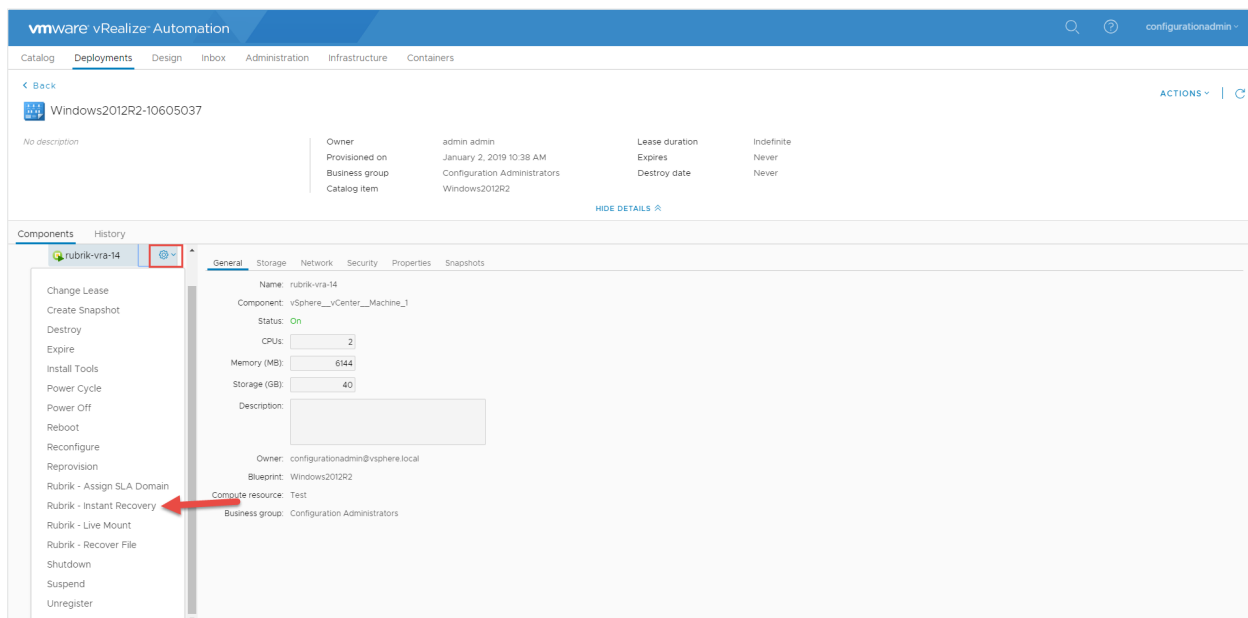
Note: Instant Recovery will proceed to power off the production VM and mark it as deprecated, while powering up a point-in-time copy of the VM running directly on the Rubrik cluster.



Once the proper entitlements or service assignments have been applied, end users are now able to perform a self-serve instant recovery of their VMs directly from within the vRA Catalog, without having to log into the Rubrik CDM cluster.



Optionally, the Rubrik – Instantly Recover workflow may be used as a Resource Action within vRA as well.

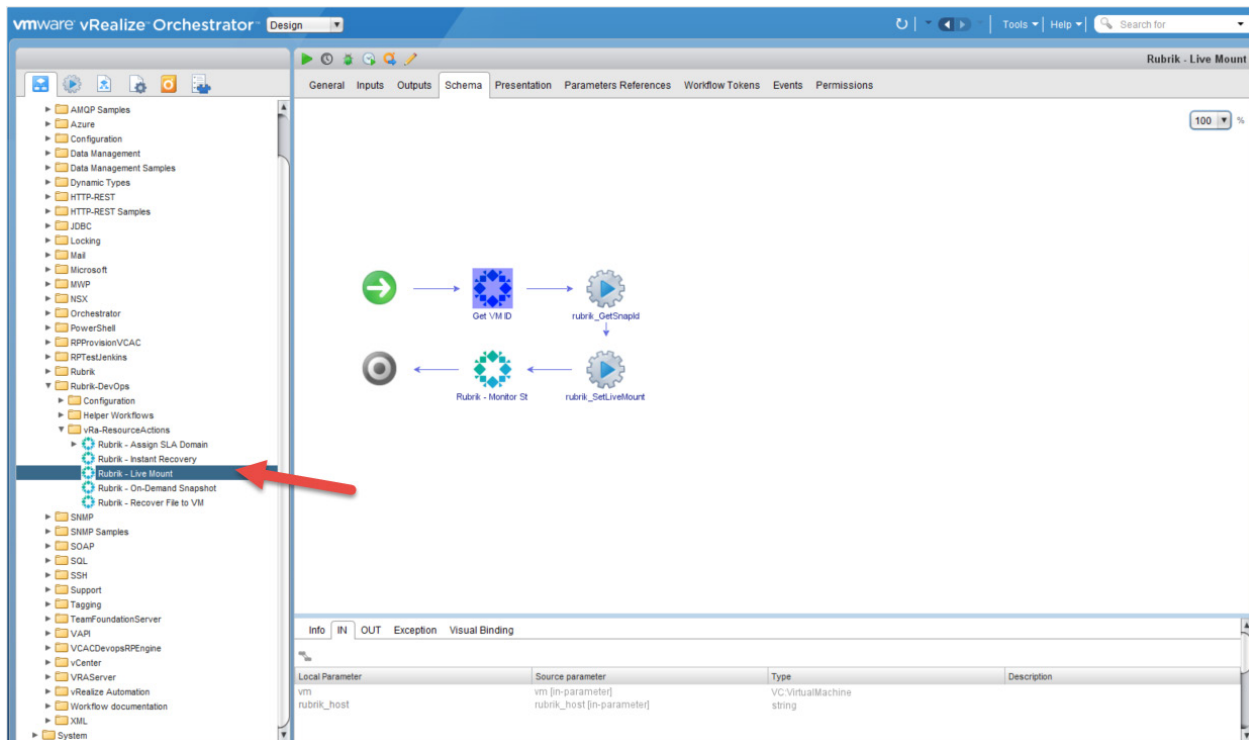


Self-Serve Instant Recovery enables end users to perform recovery of their own VMs, completely eliminating the need to wait on lengthy approvals or for backup administrator's time to free up. A perfect option for those organizations looking to place the power in their customers or end-users hands. For organizations with tighter compliance and processes, vRA catalog requests can also be configured to require approvals from different departments or individuals - providing the perfect mix of self-service and governance.

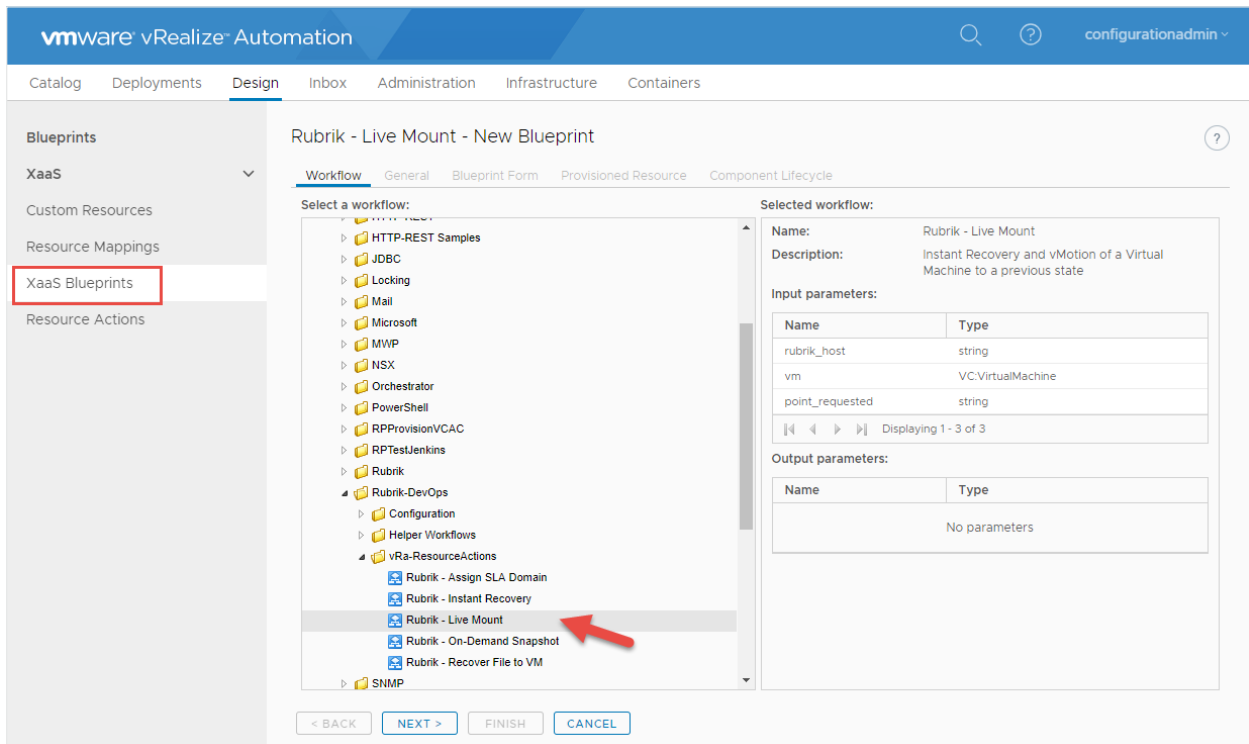
Self-Serve Live Mount

The built-in workflow, **Rubrik – Live Mount**, may be used to provide end-users with the ability to provision duplicate point-in-time copies of production VMs that can be used for testing and development purposes. Imagine an end user being able to duplicate a complex application to test upgrades or patch installs, or having the ability to view many point-in-time copies of VMs to compare change to a file or application over time for troubleshooting purposes - all the while not having to involve a backup administrator. The process execution works as follows:

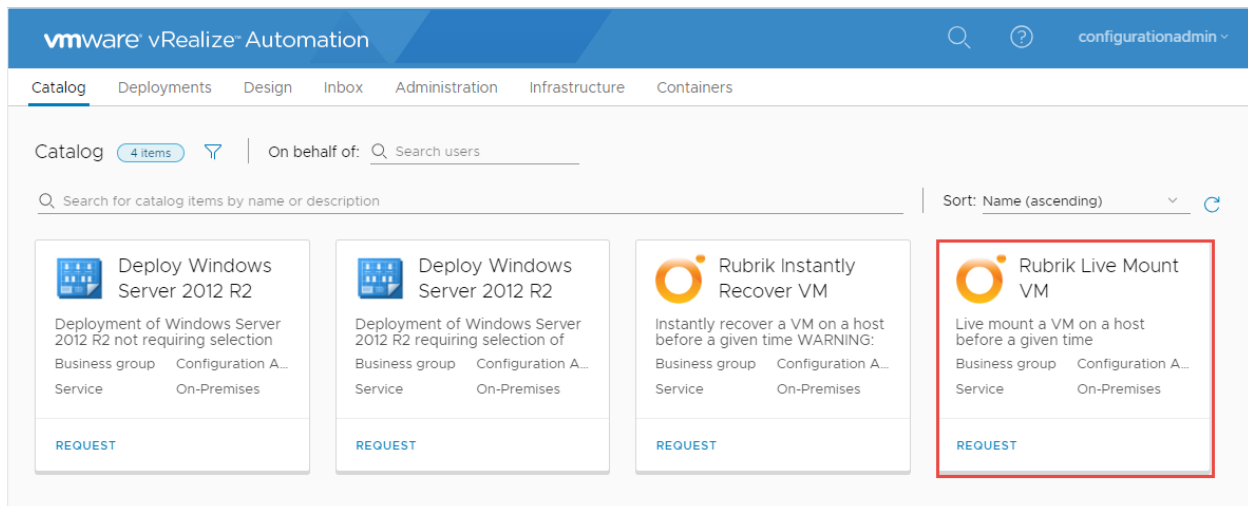
1. The workflow receives a VM, date (point-in-time to mount), and target (host) as input.
2. The workflow authenticates to the Rubrik cluster.
3. The appropriate API calls are made to Live Mount the VM.
4. The workflow waits for the Live Mount to fully complete before reporting success.



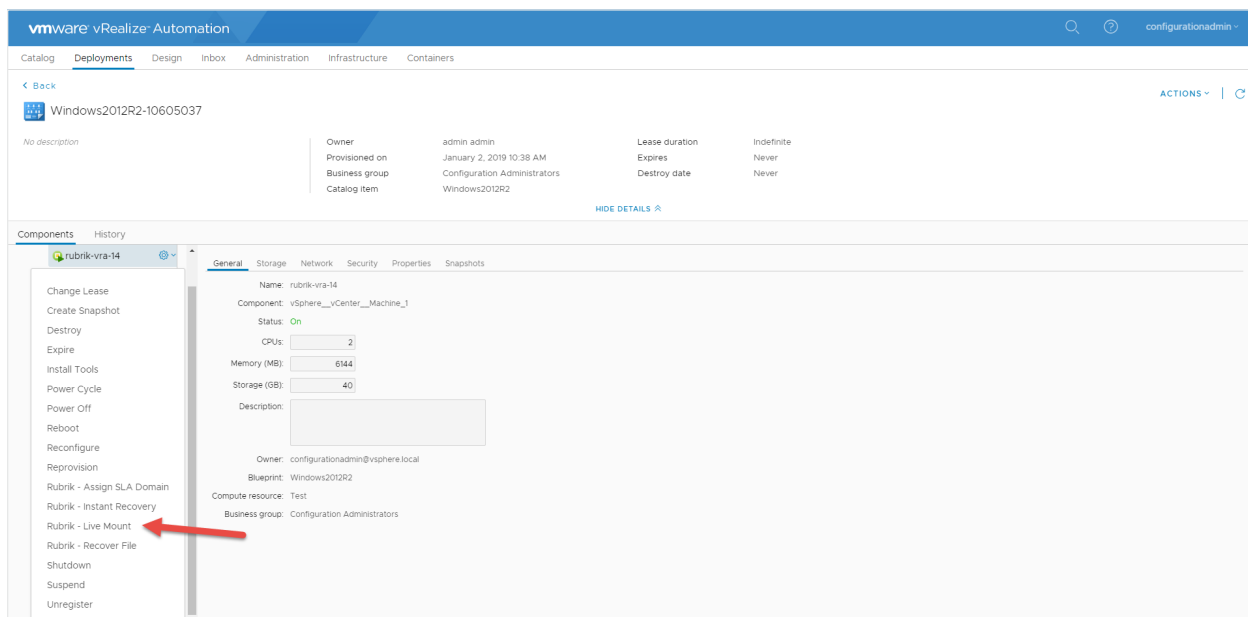
A new XaaS blueprint can be easily configured within vRA by selecting the **Rubrik – Live Mount** and ensuring the blueprint has a status of Published.



Once the proper entitlements or service assignments have been applied, end users are now able to perform a self-serve Live Mount of their VMs directly from within the vRA Catalog, without having to log into the Rubrik UI.



Optionally, the Rubrik – Instantly Recover workflow may be used as a Resource Action within vRA as well.



Self-Serve Live Mount enables end users to provision point-in-time copies of their production VMs for testing and development purposes without the intervention of IT Operations. It is a perfect option for those organizations looking to place the power into the hands of customers or end users. For organizations with tighter compliance and processes, vRA catalog requests can also be configured to require approvals from different departments or individuals, providing the perfect mix of self-service and governance.

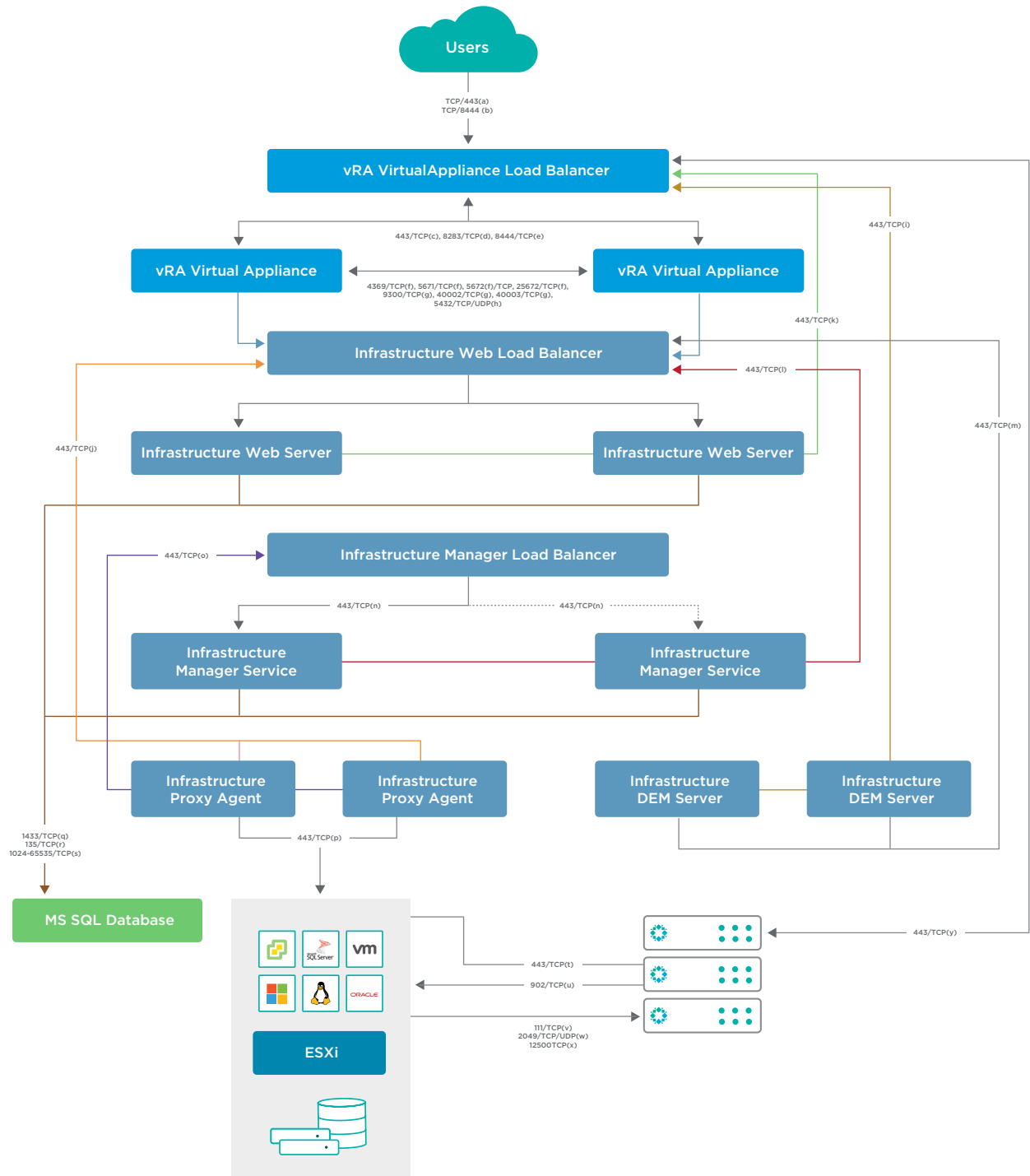
CONCLUSION

Pairing Rubrik with VMware vRealize Automation allows for an automated data protection approach to be applied to the existing flexible, self-serve architecture. The vendor agnostic, API-first approach allows Rubrik and vRA to work in harmony to provide complete automation around data management. Integrating Rubrik and vRA delivers various self-serve options around backup and recovery to end-users, while allowing IT teams to maintain the governance and control they require.

ABOUT THE AUTHOR

Mike Preston is a Technical Marketing Engineer 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](#).

APPENDIX A: PORT DIAGRAM



ID	Port	Protocol	Source	Destination	Description
a	443	TCP	External Clients	vRA Virtual Appliance Load Balancer	Allows for connection into vRA user interfaces
b	8444	TCP	External Clients	vRA Virtual Appliance Load Balancer	Allows console proxy communication for vSphere VMware Remote Console connections
c	443	TCP	vRA Virtual Appliance Load Balancer vRA Virtual Appliance(s)	vRA Virtual Appliance(s) vRA Virtual Appliance Load Balancer	Allows for load balancing of connections to embedded vRO instances
d	443	TCP	vRA Virtual Appliance Load Balancer	vRA Virtual Appliance(s)	Allows console proxy communication for vSphere VMware Remote Console connections
e	8444	TCP	vRA Virtual Appliance Load Balancer	vRA Virtual Appliance(s)	Allows for load balancing of connections to vSphere VMware Remote Console
f	4369 5671 5672 25672	TCP TCP TCP TCP	vRA Virtual Appliance	vRA Virtual Appliance	Allows for synchronization of RabbitMQ Messaging between vRA Virtual Appliances
g	9300 40002 40003	TCP TCP TCP	vRA Virtual Appliance	vRA Virtual Appliance	Allows for synchronization of vIDM cluster between vRA Virtual Appliances
h	5432	TCP UDP	vRA Virtual Appliance	vRA Virtual Appliance	Allows for connection to PostgreSQL database between vRA Virtual Appliances
i	443	TCP	Infrastructure DEM Servers	vRA Virtual Appliance Load Balancer	Allows for communication from Infrastructure DEM Servers to the vRA Virtual Appliance Load Balancer
j	443	TCP	Infrastructure Proxy Agents	Infrastructure Web Load Balancer	Allows for communication from Infrastructure Proxy Agents to the Infrastructure Web Load Balancer
k	443	TCP	Infrastructure Web Servers	vRA Virtual Appliance Load Balancer	Allows for communication from the Infrastructure Web Servers to the vRA Virtual Appliance Load Balancer
l	443	TCP	Infrastructure Manager Services	Infrastructure Web Load Balancer	Allows for communication from Infrastructure Manager Services to the Infrastructure Web Load Balancer

ID	Port	Protocol	Source	Destination	Description
m	443	TCP	Infrastructure DEM Servers	Infrastructure Web Load Balancer	Allows for communication from Infrastructure DEM Servers to Infrastructure Web Load Balancer
n	443	TCP	Infrastructure Manage Loadr Balancer	Infrastructure Manager Services	Allows for communication from Infrastructure Manager Load Balancer to the Infrastructure Manager Services in an active/passive configuration
o	443	TCP	Infrastructure Proxy Agents	Infrastructure Manager Load Balancer	Allows for communication from Infrastructure Proxy Agents to the Infrastructure Manager Load Balancer
p	443	TCP	Infrastructure Proxy Agents	vCenter Server	Allows for communication from the Infrastructure Proxy Agents to the vCenter Server
q	1433	TCP	Infrastructure Web Servers Infrastructure Manager Services	Infrastructure Web Servers Infrastructure Manager Services	Allows for communication from the Infrastructure Web Servers and the Infrastructure Management Services to an external MS SQL database
r	135	TCP	Infrastructure Web Servers Infrastructure Manager Services	Microsoft SQL Server Database	Allows for communication utilizing the Microsoft Distributed Transaction Coordinator (MSDTC) from the Infrastructure Web Servers and Infrastructure Manager Services to an external MS SQL database
s	1024-65535	TCP	Infrastructure Web Servers Infrastructure Manager Services	Microsoft SQL Server Database	Allows for the assignment of a required random port for MSDTC communications
t	443	TCP	Rubrik Cluster	vCenter Server	Allows for informational queries from the Rubrik cluster to the vCenter Server
u	902	TCP	Rubrik Cluster	ESXi	Allows for Network Block Device (NBD) data transfers
v	111	TCP	ESXi	Rubrik Cluster	Allows for an NFS datastore to be mounted from the Rubrik cluster to an ESXi host

ID	Port	Protocol	Source	Destination	Description
w	2049	TCP/ UDP	ESXi	Rubrik Cluster	Allows for connection from the ESXi hosts to the NFS daemon running on the Rubrik cluster
x	12500	TCP	ESXi	Rubrik Cluster	Allows an ESXi host to perform an NFS mount to acquire a Virtual Machine Live Mount from the Rubrik cluster
y	443	TCP	Rubrik Cluster	vRA Virtual Appliances	Allows vRA to consume Rubrik API endpoints

APPENDIX B: VREALIZE AUTOMATION PRIVILEGES

The following table describes the minimum privileges on the vCenter Server that are required by the vCenter Server Proxy Agent. The table uses an asterisk (*) to indicate a privilege that is required specifically for Rubrik related operations.

Privilege Category	Privilege	Description
Datastore	Allocate space	Used by Rubrik to create virtual machines for export. Also used by Rubrik to provide space for delta files on the datastore when creating a snapshot. Used by vRA to create virtual machines.
Datastore	Browse datastore	Permits Rubrik to find and download the vmware.log file for a virtual machine after a failed snapshot and to send the vmware.log file out for support. Used by vRA to browse the contents of datastores in order to direct the placement of virtual machines during provisioning.
Datastore	Configure datastore*	Allows Rubrik to connect the datastore on a Rubrik cluster to the vCenter Server for Live Mount and Instant Recovery.
Datastore	Low level file operations*	Permits Rubrik to ingest and to export the contents of snapshot VMDKs.
Datastore	Move datastore*	Allows Rubrik to place a Live Mount datastore into a vCenter Server folder to enhance manageability.
Datastore	Remove datastore*	Used by Rubrik to detach a Live Mount datastore that is no longer in use.
Folder	Create Folder	Used by vRA during VM provisioning operations.
Folder	Delete Folder	Used by vRA during VM decommissioning operations.
Global	Manage Custom Attributes	Used by vRA to manage custom attributes belonging to various items within the vCenter Server inventory.
Global	Set Custom Attributes	Used by vRA to set and configure custom attributes belonging to various items within the vCenter Server inventory.
Global	Enable methods*	Permits the Rubrik cluster to provide vSphere APIs for Data Protection license information to the vCenter Server. Required when using vSphere APIs for Data Protection to transfer VMDK contents.

Global	Licenses*	Permits the Rubrik cluster to provide vSphere APIs for Data Protection license information to the vCenter Server. Required when using vSphere APIs for Data Protection to transfer VMDK contents.
Host	Configuration: A. Storage partition Configuration	Configuration privileges: A. Used by Rubrik for storage partition configuration when attaching Live Mount datastores to ESXi hosts.
Network	Assign network	Permits Rubrik to connect Instant Recovery virtual machines to a network when powering on the virtual machines. Used by vRA to attach provisioned virtual machines to desired networks.
Permissions	Modify Permission	Allows vRA to modify and create permissions on provisioned virtual machines.
Resource	Assign virtual machine to resource pool	Allows Rubrik to allocate resources on an ESXi host for powering on virtual machines that are created through the Export, Live Mount, and Instant Recovery features. Used by vRA to allocate resources for powering on provisioned virtual machines.
Resource	Migrate Powered Off Virtual Machines	Allows vRA to migrate powered off virtual machines to another host or resource pool.
Resource	Migrate Powered On Virtual Machines	Allows vRA to migrate powered on virtual machines to another host or resource pool.
Sessions	Validate session*	Used by Rubrik to discover, cache, and reuse previous vCenter Server sessions.
Sessions	Validate and stop sessions*	Used by Rubrik to discover, cache, and reuse previous vCenter Server sessions.

<p>Virtual Machine</p>	<p>Configuration:</p> <ul style="list-style-type: none"> A. Add existing disk B. Add new disk C. Advanced D. Change resource E. Disk change tracking F. Disklease* G. Rename H. Settings I. Swapfile placement J. Add or Remove Device K. Remove Disk L. Change CPU Count M. Device Extend Virtual Disk Settings N. Memory O. Modify Device Settings P. Set Annotation 	<p>Configuration privileges:</p> <ul style="list-style-type: none"> A. Used by Rubrik when creating virtual machines through the Export, Live Mount, and Instant Recovery features. Used by vRA when provisioning virtual machines. B. Used by Rubrik when creating virtual machines for the Export, Live Mount, and Instant Recovery features. Used by vRA when provisioning virtual machines. C. Required for Live Mount, Instant Recovery, and Export. Also permits creation of the proxy virtual machine that is required for storage array integration. Used by vRA when provisioning virtual machines. D. Permits Rubrik to configure virtual machine resources that are created in resource pools. Used by vRA to manage virtual machines within various resource pools. E. Used by Rubrik to enable incremental snapshots, and to reset CBT when required. F. Allows Rubrik to acquire leases to permit using vSphere APIs for Data Protection for transferring VMDK contents. G. Permits Rubrik to rename the Live Mount datastore to enhance manageability. Used by vRA to rename various aspects of a virtual machine. H. Used by Rubrik to configure virtual machines that are created through the Export, Live Mount, and Instant Recovery features. Used by vRA to configure virtual machines during the provisioning process. I. Allows Rubrik to power on virtual machines that are created through the Export, Live Mount, and Instant Recovery features. Used by vRA to create swap files in appropriate datastores during the virtual machine power up process. J. Allows vRA to add and remove non-virtual disk devices to virtual machines. K. Allows vRA to remove virtual disk devices from virtual machines. L. Allows vRA to modify the number of virtual CPUs assigned to a provisioned virtual machine. M. Allows vRA to increase the size of a virtual disk assigned to a virtual machine. N. Allows vRA to change the amount of assigned memory to a virtual machine.
------------------------	--	--

		<ul style="list-style-type: none"> O. Allows vRA to change various properties of existing devices assigned to a provisioned virtual machine. P. Allows vRA to add or edit the annotation property of provisioned virtual machines.
Virtual Machine	<p>Guest Operations:</p> <ul style="list-style-type: none"> A. Guest Operation Modifications* B. Guest Operation Program Execution* C. Guest Operation Queries* 	<p>Guest Operations Privileges:</p> <ul style="list-style-type: none"> A. Permits Rubrik to deploy the Rubrik VSS agent into guest operating systems when creating application consistent snapshots. B. Permits Rubrik to start the Rubrik VSS agent on guest operating systems when creating application consistent snapshots. C. Allows Rubrik to monitor and manage the Rubrik VSS agent while the agent is running on guest operating systems
Virtual Machine	<p>Interaction:</p> <ul style="list-style-type: none"> A. Answer question* B. Backup operation on virtual machine* C. Device connection D. Guest operating system management by VIX API* E. Power Off F. Power On G. Reset H. Suspend I. VMware Tools install J. Configure CD Media K. Console Interaction 	<p>Interaction Privileges:</p> <ul style="list-style-type: none"> A. Permits Rubrik to automatically handle situations where a virtual machine is in a stuck state waiting for a question to be answered. B. Used by Rubrik to perform backup operations on virtual machines. C. Used by Rubrik to connect and disconnect devices which are attached to virtual machines that are created through the Export, Live Mount, and Instant Recovery features. Used by vRA to connect and disconnect devices on provisioned virtual machines. D. Permits Rubrik to manage a guest operating system along with the Rubrik VSS agent when creating application consistent snapshots. E. Allows Rubrik to power off Live Mount virtual machines and Instant Recovery virtual machines before deleting the virtual machine. Allows vRA to manage the power state of provisioned virtual machines. F. Allows Rubrik to power on Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines after creating the virtual machine. Allows vRA to manage the power state of virtual machines.

		<ul style="list-style-type: none"> G. Permits Rubrik to manage Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines after creating the virtual machine. Allows vRA to perform resets on provisioned virtual machines. H. Permits Rubrik to manage Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines after creating the virtual machine. Allows vRA to perform suspend operations on provisioned virtual machines. I. Allows Rubrik to upgrade VMware Tools on a guest OS as needed to prevent the guest OS from hanging or crashing when quiescing for a snapshot. Allows vRA to install and upgrade VMware tools on provisioned virtual machines. J. Allows vRA to perform configuration of a virtual CD or DVD device. K. Allows vRA to interact with a virtual machines virtual mouse, keyboard, and screen.
Virtual Machine	<p>Inventory:</p> <ul style="list-style-type: none"> A. Create new B. Move C. Register* D. Remove E. Unregister* F. Create from existing 	<p>Inventory Privileges:</p> <ul style="list-style-type: none"> A. Used by Rubrik to create Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines. Used by vRA to create new virtual machines. B. Permits Rubrik to move an original virtual machine into a “deprecated” folder before replacing the original with an Instant Recovery virtual machine. Used by vRA to move provisioned or decommissioned virtual machines in the vCenter folder hierarchy. C. Used by Rubrik to create Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines. D. Allows Rubrik to remove Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines. Used by vRA to remove decommissioned expired or destroyed virtual machines. E. Allows Rubrik to remove Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines. F. Allows vRA to provision virtual machines based off existing virtual machines or templates.

<p>Virtual Machine</p>	<p>Provisioning:</p> <ul style="list-style-type: none"> A. Allow disk access* B. Allow read-only disk access* C. Allow virtual machine download* D. Allow virtual machine files upload* E. Customize F. Clone Template G. Clone Virtual H. Deploy Template I. Read Customization Specs 	<p>Provisioning Privileges:</p> <ul style="list-style-type: none"> A. Permits Rubrik to write the VMDK contents of Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines. B. Permits Rubrik to read the VMDK contents of Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines when backing up the virtual machines. C. Allows Rubrik to download the non-VMDK files of protected source virtual machines, including configuration files and support logs. D. Allows Rubrik to upload non-VMDK files of Export virtual machines, Live Mount virtual machines and Instant Recovery virtual machines, when creating and configuring the virtual machines. E. Allows vRA to customize a provisioned virtual machine's guest operating system. F. Allows vRA to clone templates.. G. Allows vRA to provision virtual machines by cloning existing virtual machines. H. Allows vRA to provision virtual machines by deploying from an existing template. I. Allows vRA to read existing customization specifications for presentation during the provisioning process.
<p>Virtual Machine</p>	<p>Guest Operations:</p> <ul style="list-style-type: none"> A. Create snapshot B. Remove snapshot C. Rename snapshot* D. Revert to snapshot 	<p>Snapshot Management Privileges:</p> <ul style="list-style-type: none"> A. Permits Rubrik to create temporary snapshots of virtual machines for ingest into Rubrik cluster storage. Allows vRA to create snapshots of provisioned virtual machines. B. Permits Rubrik to remove temporary snapshots of virtual machines that were created for ingest into Rubrik cluster storage. Allows vRA to remove snapshots of provisioned virtual machines. C. Allows Rubrik to manage the temporary snapshots of virtual machines that were created for ingest into Rubrik cluster storage. D. Used by Rubrik to prepare an Export virtual machine with data from a Rubrik snapshot. Allows vRA to revert to various point-in-time snapshots created on provisioned virtual machines.



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

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

Rubrik delivers a single platform to manage and protect data in the cloud, at the edge, and on-premises. Enterprises choose Rubrik's Cloud Data Management software to simplify backup and recovery, accelerate cloud adoption, and enable automation at scale. Rubrik's run-anywhere, scale-out architecture is built to empower IT departments today and in the future, reducing total cost of ownership while enabling infrastructure flexibility for a multi-cloud world. For more information, visit www.rubrik.com and follow [@rubrikinc](https://twitter.com/rubrikinc) on Twitter.

20190204_v1