BEST PRACTICES

Managing VMware VirtualCenter Roles and Permissions

vmware®

Table of Contents

Managing VMware VirtualCenter Roles and Permissions

Introduction

One key management task in a VMware Infrastructure environment is determining who can use VMware VirtualCenter and what tasks those users are authorized to perform. The person who has the role of administrator for the system is authorized to assign the rights needed by other users. Generally, only a limited set of people should be given the administrator role. If you are the administrator, you should then use VirtualCenter roles, described in the sections that follow, to delegate management of ESX Server hosts and virtual machines to others.

This paper introduces you to the way Virtual Infrastructure 3 controls access to resources and describes techniques you can use to assign appropriate access rights efficiently. It explains the concept of roles, provides information to help in the design of custom roles, and gives recommendations for how to work with roles and privileges in VirtualCenter.

VirtualCenter Objects and Permissions

The authorization to perform tasks in VMware Infrastructure is governed by an access control system. This system allows the VirtualCenter administrator — using the Virtual Infrastructure Client — to specify in great detail which users or groups can perform which tasks on which objects. It is defined using three key concepts:

- **Privilege** — The ability to perform a specific action or read a specific property. Examples include powering on a virtual machine and creating an alarm.
- **Role** — A collection of privileges. Roles provide a way to aggregate all the individual privileges that are required to perform a higher-level task, such as administer a virtual machine.
- **Object** — An entity upon which actions are performed. VirtualCenter objects are datacenters, folders, resource pools, clusters, hosts, and virtual machines.

Figure 1 shows the hierarchy of objects you can manage in the Virtual Infrastructure Client.

In addition, VirtualCenter depends upon the users and groups defined in your Active Directory environment or on the local Windows server on which VirtualCenter runs. One key point to note is that an ESX Server host can have its own set of users and groups that is independent of the Active Directory users and groups. If you are using VirtualCenter, you should avoid defining any users on the ESX Server host beyond those that are created by default. This approach provides better manageability, because there is no need to synchronize the two lists if a user or group is added or updated on one of the systems. It also improves security, because it makes it possible for all permissions to be managed in one place. For a full description of the way ESX Server and Virtual Infrastructure Client recognize and manage users and groups, see the sections "Users" and "Groups" in Chapter 15 of the manual *Basic System Administration* in your VMware Infrastructure documentation.

Figure 2 shows the relationship between roles, objects, and users. Together they define a permission. The role defines the actions that can be performed. Users and group indicate who can perform the action, and the object is the target of the action. Each combination of user or group, role, and object must be specified. In other words, the administrator first selects an object from the overall VirtualCenter inventory, then selects

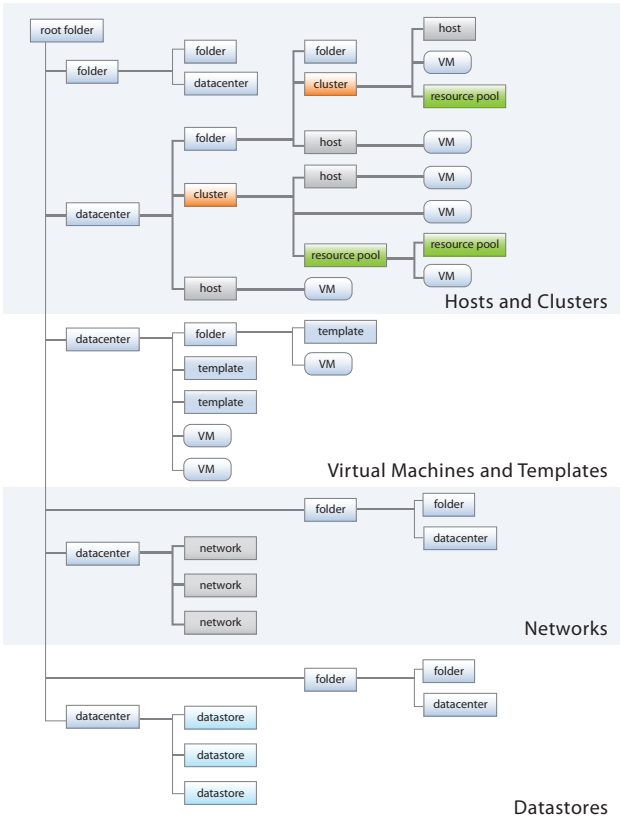

Figure 1 — The Virtual Infrastructure Client object hierarchy

a role to be assigned to that object, then selects the user or group to which this permission pertains. For detailed instructions, see the section "Assigning Access Permissions" in chapter 15 of the *Basic System Administration* guide.

There are more than 100 privileges, which roughly correspond to individual actions a VirtualCenter user can take. They are grouped hierarchically in the Virtual Infrastructure Client for convenience. Appendix A of the manual *Basic System Administration* in your VMware Infrastructure documentation describes all of the privileges.

For each permission, you can decide whether the permission propagates down the object hierarchy to all subobjects, or if it applies only to that immediate object. For example, you can have a role called Datacenter Administrator, which gives a user privileges to manage hosts, network, and datastores, but then choose for that role not to grant that user administrative privileges for virtual machines on those hosts. In a contrasting case, you can grant a user very limited permissions (for example, read-only) from the datacenter level on downward, then grant more permissive roles on certain subobjects, for example, a folder of virtual machines.

In addition to specifying whether permissions generally propagate downward, you can override permissions set at a higher level by explicitly setting different permissions for a lower-level object. For example, you might give a user read-only permission at the datacenter level and administrator permission for a particular folder. If you set the administrator permission to propagate, that permission also applies to all branches below that particular folder. If you set the administrator permission but do not set it to propagate, the user has no rights at all on branches below that particular folder — not even read-only.

Note: There is a known issue in VirtualCenter 2.0.1 and lower that causes a misleading display indicating read-only permission at lower levels even when propagation is not set. This issue affects only the display in the user interface. The actual permissions are set as described in this paper.

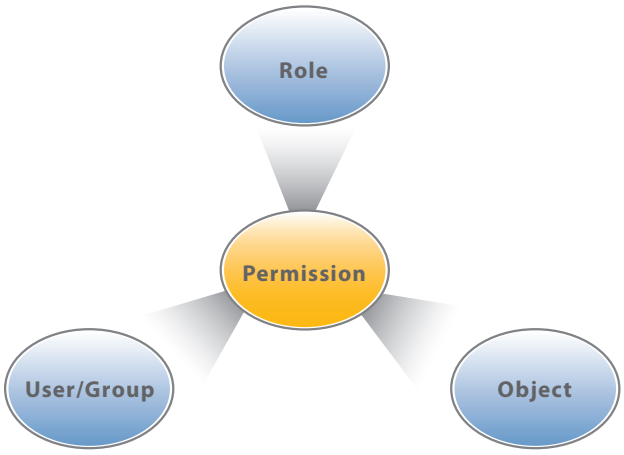

Figure 2 — The conceptual structure of a permission

The normal process of setting up users, groups, and permissions can grant a user differing permissions on the same object. This can happen easily if, for example, the user belongs to two different groups and the two groups have different permissions on the object. In this case, the user is granted permissions that are a union of the groups' permissions. For example, if one group is allowed to power on virtual machines and the other is allowed to take snapshots, then a user who is a member of both groups can do both. If an individual user has an explicit permission set on the object, however, this individual permission overrides all implied group permissions. For example, if a role that does not permit powering on virtual machines or taking snapshots is granted to a user explicitly on that object, the user cannot perform either action.

Built-in and Custom Roles

VirtualCenter and ESX Server hosts provide default roles:

- **System roles** – System roles are permanent and the privileges associated with these roles cannot be changed. The three system roles are: No Access, Read-Only, and Administrator. The latter two also exist in VirtualCenter 1.x
- **Sample roles** – Sample roles are provided for convenience as guidelines and suggestions. Table 1 lists the sample roles in VirtualCenter 2.x. Note that two of these roles are meant to emulate the roles with the same names in VirtualCenter 1.x

The Administrator role is the most powerful one in VirtualCenter. It essentially allows the user to perform every available action in VirtualCenter. You should grant this role to as few users as possible. The Read-Only role allows the user to view the state and configuration of objects without modifying them. The No Access role prevents a user from seeing any objects. It is equivalent to assigning no role to a user for a particular object. The No Access role is useful in conjunction with other roles to limit their scope, as shown in an example later in this paper.

The built-in roles provide a way to get started with VirtualCenter permissions management. By studying Table 1, then examining the privileges of each role in the VI Client, you can determine which roles are appropriate for the personnel in your environment. Bear in mind that a role must be applied to an object for a specified user or group in order to create a permission. You should decide which object in the inventory hierarchy is the appropriate one to which to apply the role. For example, instead of granting the Virtual Machine Administrator role to someone on individual virtual machines, you can group selected virtual machines in a folder, then apply this role to the folder, with propagation enabled.

Role	User Capabilities
Virtual Machine User (equivalent to the role with the same name in VirtualCenter 1.x)	Perform actions on virtual machines only. Interact with virtual machines, but not change the virtual machine configuration. This includes: • All privileges for the scheduled tasks privileges group. • Selected privileges for the global items and virtual machine privileges groups. • No privileges for the folder, datacenter, datastore, network, host, resource, alarms, sessions, performance, and permissions privileges groups.
Virtual Machine Power User	Perform actions on the virtual machine and resource objects. Interact and change most virtual machine configuration settings, take snapshots, and schedule tasks. This includes: • All privileges for scheduled task privileges group. • Selected privileges for global items, datastore, and virtual machine privileges groups. • No privileges for folder, datacenter, network, host, resource, alarms, sessions, performance, and permissions privileges groups.
Resource Pool Administrator	Perform actions on datastores, hosts, virtual machines, resources, and alarms. Provides resource delegation and is assigned to resource pool inventory objects. This includes: • All privileges for folder, virtual machine, alarms, and scheduled task privileges groups. • Selected privileges for global items, datastore, resource, and permissions privileges groups. • No privileges for datacenter, network, host, sessions, or performance privileges groups.
Datacenter Administrator	Perform actions on global items, folders, datacenters, datastores, hosts, virtual machines, resources, and alarms. Set up datacenters, but with limited ability to interact with virtual machines. This includes: • All privileges for folder, datacenter, datastore, network, resource, alarms, and scheduled task privileges groups. • Selected privileges for global items, host, and virtual machine privileges groups. • No privileges for session, performance, and permission privileges groups.
Virtual Machine Administrator (equivalent to the role with the same name in VirtualCenter 1.x)	Perform actions on global items, folders, datacenters, datastores, hosts, virtual machines, resources, alarms, and sessions. This includes: • All privileges for all privilege groups, except permissions.

Table 1 — Sample roles included in VirtualCenter 2.x

In most cases, you should enable propagation when assigning a role. This prevents confusion when a new object is inserted into the inventory hierarchy; if propagation is not set, it might not be clear why a user has no permissions on the new object. Instead of disabling propagation, you can explicitly limit the extent of a role by using the No Access role. For example, you can grant the VirtualCenter Power User role on a folder of virtual machines to a group. With propagation enabled, the same role is granted on all virtual machines in that folder at any given time, and you do not need to add and remove these privileges on virtual machines as they come and go from the folder. However, if you want to have a subfolder of virtual machines that should not be usable by this group, you can assign the No Access role on that specific subfolder. The virtual machines in the specified subfolder are now effectively invisible to users in this group; the setting effectively masks this folder for the group.

The role-editing facilities in the Virtual Infrastructure Client allow you to create privilege sets that match your user needs. You can create custom roles either in VirtualCenter or directly on an ESX Server host. However, the roles you create directly on an ESX Server host are not accessible within VirtualCenter. You can work with these roles only if you log in to the host directly from the Virtual Infrastructure Client.

One convenient way to create a custom role is to start with an existing role, then make modifications to it. In the Virtual Infrastructure Client, right-clicking on a role and selecting **Clone** produces a copy of the role. You can then rename the role and modify the privileges appropriately

Note: If any role is granted on an object for a user, that user is able to view all the information for that object. In other words, the user might have privileges to perform only certain tasks but also has the equivalent of read-only privileges for everything else that pertains to that object. Therefore, you must be careful about granting visibility to users on certain parts of the infrastructure that might not be intended.

Task-based Privilege Assignment

Roles gather together certain privileges, making it simpler to assign those privileges to users or groups. In most cases, the name of the privilege indicates the task that it allows a user to perform. However, there are some tasks that require a coordinated set of privileges to be enabled. This section presents some examples of such tasks, and what privileges are required to enable them to be performed in their entirety.

Creating a Virtual Machine

Table 2 presents all the privileges related to creating a new virtual machine and the objects to which they should be applied.

Several points deserve special attention.

- **Read-Only role** — As shown in Table 2, for the user in this example, you must apply the Read-Only role for the datacenter that contains the datastore on which the virtual machine will reside or on a folder containing the datacenter. This setting allows the provisioning operation to determine where the virtual machine should be placed. Because datastores themselves cannot be assigned roles, you manage privileges for datastores indirectly through the parent datacenter.

  You do not need to assign the Read-Only role if you have assigned any of the other privileges at the datacenter level. Whenever any role other than No Access is assigned for the datacenter, the user automatically gets read-only permissions on the datacenter object. So, for example, if you assign **Virtual Machine** > **Inventory** > **Create** at the datacenter level, the additional Read-Only role assignment would be redundant.

- **Propagation** — If you explicitly assigned the Read-Only role to a datacenter, that does not need to propagate beyond the datacenter (down to folders, hosts, clusters, resource pools, or virtual machines). However, if you apply the Read-Only role to a folder containing the datacenter, you must enable propagation for that role to reach the datacenter object. Because the depth of propagation cannot be specified, this setting gives the user read-only privileges on every object in the datacenter. In this case, you can use the No Access role to mask objects that should not be visible to the user.

- **Disk management** — The privilege **Virtual Machine** > **Configuration** > **Add New Dis**k allows the user to create a new virtual disk file on a datastore contained in the specified datacenter or in the datacenter in which the specified virtual machine folder is located. The privilege **Virtual Machine** > **Configuration** > **Raw Device** is necessary only if an RDM volume will be used to store the internal disk for the virtual machine. Similarly, the privilege **Virtual Machine** > **Configuration** > **Add Existing Disk** is normally not necessary, because in most cases a new virtual disk is created when someone creates a virtual machine. However, this privilege is needed in the following situations:

  - Using a virtual disk from another VMware product (such as VMware Workstation).

Privilege	Object
Virtual Machine > **Inventory** > **Create**	A destination folder of virtual machines in the datacenter, a folder containing a datacenter, or the datacenter itself if you do not use folder-based organization. Required for any virtual machine creation.
Virtual Machine > **Configuration** > **Add New Disk**	A destination folder of virtual machines in the datacenter; a folder containing a datacenter, or the datacenter itself if you do not use folder-based organization. Only if including a virtual disk device that creates a new virtual disk file (not RDM). **Note:** This privilege or Add Existing Disk required for any virtual machine creation.
Virtual Machine > **Configuration** > **Add Existing Disk**	A destination folder of virtual machines in the datacenter; a folder containing a datacenter, or the datacenter itself if you do not use folder-based organization. Only if including a virtual disk device that refers to an existing virtual disk file (not RDM). **Note:** This privilege or Add New Disk required for any virtual machine creation.
Virtual Machine > **Configuration** > **Raw Device**	A destination folder of virtual machines in the datacenter; a folder containing a datacenter, or the datacenter itself if you do not use folder-based organization. Only if including a raw device mapping (RDM) or SCSI pass-through device for use by the virtual machine.
Resource > **Assign VM to Resource Pool**	A destination resource pool, host, or cluster.
Read-Only role	The datacenter that contains the datastore on which the virtual machine will reside or a folder containing the datacenter. Propagation does not have to be enabled for the datacenter, but it must be enabled for a folder.

Table 2 — Privileges needed for creating a virtual machine

- Using a virtual disk created by a third-party product that imports a physical machine configuration into a virtual machine.
- Using a virtual disk that was manually copied from another datacenter or datastore.

In all these cases, the existing disk file should be on a datastore contained in the datacenter where the virtual machine will be created.

- **Resource pools** — When you apply the privilege **Resource** > **Assign VM to Resource Pool**, be aware that the object model of VMware Infrastructure 3 uses resource pools as objects that partition compute resources, such as memory and CPU. Normally, a resource pool is defined explicitly as some portion of the resources available on one host or a cluster of hosts. However, if no explicit resource pools are defined, each host or cluster is considered to have its own implicit resource pool that groups the resources of that host or cluster. The root resource pool is not displayed because the resources of the host (or cluster) and the root resource pool are always the same. Therefore, if there is no named resource pool into which the virtual machine is to be deployed, you must assign this privilege for the destination host or cluster. If a user does not hold this privilege (by virtue of this or some other role) on any named resource pool, host, or cluster, that user cannot create a virtual machine.
- **Operating system deployment** — The example presented here lists the minimum privileges needed to create a new virtual machine. The next task a user is likely to perform is to deploy an operating system onto the new, blank virtual machine. You need to grand appropriate privileges for this task. The specific privileges depend on how your users deploy operating systems. For example, if they deploy the operating system from an ISO image on shared storage, assign **Datastore** > **Browse Datastore** for the datacenter. If they deploy the operating system from an ISO image on a local disk on the host, assign **Datastore** > **Browse Datastore** for both the datacenter and the host. If they deploy the operating system from a physical CD, datastore privileges are not needed. In all of these cases, most of the privileges in **Virtual Machine** > **Configuration** and **Virtual Machine** > **Interaction** are needed to deploy and configure the operating system on the virtual machine.

Inventory Manipulation

Table 3 shows examples of tasks that affect the organization of compute resources in the overall VMware Infrastructure 3 inventory and privileges required for each one.

Networking, Storage, and Host Maintenance

There are certain privileges that pertain specifically to the configuration of networking and storage virtualization. In both cases, VMware Infrastructure 3 maintains a host-centric view of the resources, and the privileges are defined on a per-host basis. If you take advantage of privilege propagation, these privileges can be assigned at a higher level, such as cluster or folder, and they then apply to all contained hosts.

Because some of these privileges actually enable a large number of tasks, it is important to understand exactly what actions are permitted for a user holding a role that contains these privileges. Table 4 on page 8 provides a list of networking- and storage-related privileges and the specific capabilities that they allow. Within an individual privilege, it is not possible to disallow some of these tasks while allowing others; the level of granularity in VMware Infrastructure 3 allows you to associate either all or none of them with a role. Therefore, you must be

Task	Required Privileges
Migrate a virtual machine	**Resource** > **Migrate** if the virtual machine is powered on or **Resource** > **Relocate** if the virtual machine is powered off. Also requires **Resource** > **Assign Virtual Machine to Resource Pool** if destination is a different resource pool from the source.
Move a host into a folder	**Host** > **Inventory** > **Modify Cluster** on the source cluster, **Host** > **Inventory** > **Move Host** on the host, and **Host** > **Inventory** > **Add Standalone Host** on the target Folder.
Move a virtual machine, stand-alone host, folder, cluster or datacenter into a folder	**Folder** > **Move** if the object is a folder, **Datacenter** > **Move** if the object is a datacenter, **Host** > **Inventory** > **Move Cluster/Standalone Host** if the object is a cluster or standalone host, **Virtual Machine** > **Inventory** > **Move** if the object is a virtual machine or virtual machine template. These privileges are checked against the source, destination, and object being moved.
Move a set of resource pools or virtual machines into a resource pool	If the object being moved is a resource pool, **Resource** > **Move Pool** must be held on the pool being moved, its former parent pool, and the target pool. If the object is a virtual machine, **Resource** > **Assign Virtual Machine to Resource Pool** must be held on the target pool and the virtual machine.
Remove all child resource pools	The **Resource** > **Remove Pool** privilege must be held on the parent and each of its immedi-ate children to be removed. The **Resource** > **Assign Virtual Machine to Resource Pool** privilege must be held on the parent resource pool as well as on the virtual machine.

Table 3 — Tasks that required coordinated privileges on multiple objects

comfortable with granting every one of those abilities to any potential holder of a role that contains the privilege.

Table 4 also includes some privileges related to the configuration and maintenance of the ESX Server host. As with the networking and storage privileges, make sure that users or groups assigned to a role containing one of these privileges are authorized to perform all the actions the privilege enables them to perform.

Creating Custom Roles

The use cases described in this section illustrates the process of selecting and defining the privileges required to complete a task from start to finish.

Example: Allowing Template Deployment to a Resource Pool

Suppose that you want to enable some users to create new virtual machines from existing templates and deploy those virtual machines into a specific resource pool. You might want to do this, for example, in a development environment where you want developers to be able to work with virtual machines of a fixed type and want to enable them to create as many as needed for their development work. If you allow these virtual machines to run only in a specified resource pool, you can exercise finer-grained control over the server resources used by the developers. For example, you can use limits to cap the amount of CPU or memory used by all the developer virtual machines, or you can use shares to ensure that resources used by these virtual machines are returned to other, more mission-critical resource pools when needed.

Privilege	Allowed actions
Host > **Configuration** > **Network Configuration**	• Add, remove, or update the following: port groups, virtual Ethernet adapters, virtual switches, and service console virtual Ethernet adapters • Update the following: IP routing for the host, IP routing for the service console, DNS configuration for the host, link speed and duplex settings for the physical Ethernet adapters • Restart the service console virtual network adapter interface.
Host > **Configuration** > **Storage Partition Configuration**	• Enable, disable, or configure policies for multipathing on a LUN • Rescan some or all HBAs on virtual machines for new or removed storage devices • Rescan for new or removed VMFS volumes • Extend a VMFS volume by attaching a disk partition as an extent • Format a new VMFS volume on a LUN or disk partition • Change the partitions on the disk • Add and remove send target entries and static target entries to the host bus adapter discovery list • Enable or disable the iSCSI software initiator • Update the following on an iSCSI host bus adapter: name, alias, authentication properties, IP properties, discovery properties
Datastore > **Browse Datastore**	• Browse the files on a datastore, for example, to search for a virtual machine (vmx) file or ISO image file. Must be granted at the datacenter level for a shared datastore, and at both the datacenter and host level for a local disk datastore.
Datastore > **Rename File**	• Rename a datastore (note inconsistency of naming)
Datastore > **Remove File**	• Delete a file from a datastore. If a valid virtual disk file is specified, all the components of the virtual disk are deleted.
Host > **Configuration** > **Maintenance**	• Put a host into or out of maintenance mode • Reboot a host • Shut down a host
Host > **Configuration** > **Security Profile and Firewall**	• Enable and disable network services on a host (by opening or closing the corresponding port in the firewall) • Configure the startup policy for the services • Manually start or stop the services

Table 4 — Actions enabled by networking, storage, and host maintenance privileges

One way to approach this is to create a new user-defined role called Developer and set the minimum privileges necessary for a user with that role to accomplish these tasks. Table 5 shows which privileges you must enable for this use case.

Privilege	Object
Virtual Machine > **Inventory** > **Create**	A destination folder in the datacenter, or the datacenter itself if you do not use folder-based organization. If not applied on the datacenter, you must also grant the user **Read-Only** on the datacenter separately.
Virtual Machine > **Configuration** > **Add New Disk**	A destination folder in the datacenter, or the datacenter itself if you do not use folder-based organization. Although this privilege is required if using the Virtual Infrastructure Client, it is not necessary of the same custom role is being used by an SDK client.
Virtual Machine > **Provisioning** > **Deploy Template**	A template or folder of templates in the datacenter.
Resource > **Assign VM to Resource Pool**	A destination resource pool, host, or cluster
Virtual Machine > **Interaction**	A destination resource pool, host or cluster

Table 5 — Privileges used in creating a Developer role

Although the privilege **Virtual Machine** > **Configuration** > **Add New Disk** is always required when creating a new virtual machine, the VI Client also requires this privilege for deploying a virtual machine from a template and for cloning a virtual machine. This requirement is unique to the VI Client; the privilege is not required for an SDK client that tries to deploy a template or clone a virtual machine.

Example: Network Administrator

A custom role would also be useful for an organization in which separate groups are responsible for managing servers and networks. The networking team has traditionally managed a discrete set of physical networking equipment. In a VMware Infrastructure environment, however, they may need to take responsibility for the virtual networking that runs in software on the ESX Server hosts.

A role for network administrators might give them the privilege needed to add, remove, and configure virtual switches on an ESX Server host — or a group of hosts, either in a folder or in a datacenter. Table 6 shows the privilege needed for this role. If you apply this privilege at the cluster, folder, or datacenter level, make sure that propagation is enabled.

Privilege	Object
Host > **Configuration** > **Network**	All hosts whose networks are to be managed by the network administrator, or the folder or datacenter containing these hosts, with propagation enabled.

Table 6 — Privilege required for Network Administrator role

Although users assigned this role are able to view configurations for resources other than network switches, they do not have permissions to change anything except network settings. This role thus corresponds roughly to the activities that are normally handled by a network administrator.

Example: VMware Consolidated Backup User

VMware Consolidated Backup is a product that helps to perform backups of virtual machines in a Virtual Infrastructure 3 environment from a dedicated proxy host using the VMware snapshot technique and industry-standard backup software. The proxy host connects to VirtualCenter using a special user account in order to perform the snapshots and other related tasks. You can create a role that contains only the privileges necessary for this purpose and assign it to the special user account. Table 7 contains the list of privileges and the objects to which they should be applied.

Privilege	Object
Virtual Machine > **Configuration** > **Disk Lease**	The virtual machines to be backed up, a folder of virtual machines, or the datacenter containing the virtual machines.
Virtual Machine > **State** > **Create Snapshot**	The virtual machines to be backed up, a folder of virtual machines, or the datacenter containing the virtual machines.
Virtual Machine > **State** > **Remove Snapshot**	The virtual machines to be backed up, a folder of virtual machines, or the datacenter containing the virtual machines.
Virtual Machine > **Provisioning** > **Allow Virtual Machine Download**	The virtual machines to be backed up, a folder of virtual machines, or the datacenter containing the virtual machines.

Table 7 — Privilege required for VMware Consolidated Backup user

Recommendations for VirtualCenter Roles

To make most effective use of roles in VirtualCenter, follow these guidelines:

- Design the roles with the notion that VirtualCenter should be treated as an administration tool, not a general-purpose means of gaining access to virtual machines. In particular:
  - By default, all users who are not assigned to a role and do not belong to group assigned to a role have the equivalent of **No Access** at the top-level Hosts and Clusters folder. This prevents unauthorized users from logging in to VirtualCenter, enhancing security and avoiding increased load on VirtualCenter caused by an excessive number of VI Client sessions. You should assign to roles only those specific users and groups that must perform administrative tasks for VMware Infrastructure, and you should assign those roles only for relevant objects in the inventory.
  - Ordinary users should not use the VMware virtual machine console to access virtual machines. Instead, they should use a standard remote access tool, such as Remote Desktop, RAdmin, or SSH. Even for users who might want to manage parts of the virtual infrastructure, remote console access should be strictly controlled, for both security and auditing purposes. This is analogous to controlling access to the integrated lights-out console on a physical server. You can disable virtual machine console access by removing the privilege **Virtual Machine** > **Interaction** > **Console Interaction** for a role.
- VirtualCenter runs as a user that requires local administrator privilege and must be installed by a local administrative user. However, to limit the scope of administrative access, avoid using the Windows Administrator user to operate VirtualCenter after you install it. Instead, use a dedicated VirtualCenter administrator account. To do so, take the following steps:
  1. Create an ordinary user account that will be used to manage VirtualCenter, for example, the VI Admin user. Make sure that this user does not belong to any local groups, such as Users or Administrators. This precaution ensures that any future role assignments involving a local group does not inadvertently affect this account.
  2. In VirtualCenter, log on as the Windows Administrator, then grant the role of Administrator (that is, the global VirtualCenter administrator) to the newly created account on the top-level Hosts and Clusters folder.
  3. Log out of VirtualCenter, then make sure you can log in to VirtualCenter as the new user and that this user is able to perform all tasks available to a VirtualCenter administrator.
  4. Remove the permissions in VirtualCenter for the local Administrators group.

  By configuring accounts in this way, you avoid automatically giving administrative access to domain administrators, who typically belong to the local Administrators group. You also provide a way of getting into VirtualCenter when the domain controller is down, because the local VirtualCenter administrator account does not require remote authentication.
- Although it is possible to edit the built-in sample roles (not the system roles), do not modify them. Instead, clone new roles from them, then modify the cloned roles. This approach allows you to refer to the original sample roles if you want to roll back changes you have made to them..

- Try to define a role using the smallest number of privileges possible, so that security and control over your environment can be maximized. In the virtual machine creation example, the minimum number of privileges required to enable virtual machine creation is three:
  - **Virtual Machine** > **Inventory** > **Create**
  - **Virtual Machine** > **Configuration** > **Add New Disk**
  - **Resource** > **Assign VM to Resource Pool**
- Because the same role can be applied to any VMware Infrastructure 3 object, one way to ensure that the fewest privileges are granted is to create multiple roles, each of which is targeted at a specific set of tasks, then grant each user or group the appropriate role on the appropriate object. For example, in the case of the custom Developer role, you can choose to split this across three roles:
  - **Deploy Template** — One role allows only deployment from a template.
  - **Create Virtual Machine** — Another role allows creation of a virtual machine and virtual disk in a datacenter or folder.
  - **Interact with Virtual Machine** — The third role allows assigning a virtual machine to a resource pool and interaction with a virtual machine.

  Then you can grant a user the Deploy Template role on template folder BuildA, the Create Virtual Machine role on datacenter East, and the Interact with Virtual Machine role on resource pool Dev.
- As a corollary to the previous guideline, note that you can grant only one explicit role to a user on a virtual machine, but two different roles might apply implicitly through propagation. In the example given above, you can apply the Create Virtual Machine role to a folder and also apply the Interact with Virtual Machine role to a resource pool. The user then has a union of these privileges on any virtual machine that is in the folder as well as in the resource pool. This means, for example, that if you want to allow a user to both create a virtual machine and interact with it, without depending on indirect privileges through propagation, you must use a role that combines the two sets of privileges.
- Try to give the roles names that explicitly indicate what each role allows, to make their purposes clear. The examples above illustrate this point.
- Use folders to contain the scope of permissions. For example, if you want to limit the templates from which users can deploy new virtual machines, you can put the allowed templates into a folder, then apply the Deploy Template role on this folder for the users.
- Because of membership in different groups, and the union of privileges inherited from them, it might not always be obvious what privileges are granted to a user on an object. However, Active Directory does not allow the inspection of a user's group memberships unless the user is logged in. One way around this restriction is to inspect all the role assignments on all objects, then cross-reference them with a known list of group memberships of users. The VI Client allows you to see the roles assigned for objects individually, but by using the VMware Infrastructure SDK, you can obtain this information for all objects at once in a more straightforward manner. Appendix A shows an example of a Perl script that uses the VI Perl Toolkit and generates a list of objects with the roles assignments associated with each one. You can use this script as a starting point and modify it to suit your needs.
- Any user who has the ability to generate a virtual machine or template can potentially initiate a denial-of-service attack by completely filling up a datastore with virtual disk files, whether purposefully or inadvertently. The specified privileges that allow this are:
  - **Virtual Machine** > **Configuration** > **Add New Disk**
  - **Virtual Machine** > **Provisioning** > **Deploy Template**
  - **Virtual Machine** > **Provisioning** > **Create Template from Virtual Machine**
  - **Virtual Machine** > **Provisioning** > **Clone Template**

  If you are uncomfortable granting a user this ability to fill a datastore, you must enable a more trusted individual to generate virtual machines or templates on behalf of this user..

Appendix: Perl Script for Listing All Role Assignments

The Perl script shown in Listing 1 makes use of the VMware Infrastructure Perl Toolkit to query VirtualCenter for a list of all the roles assigned to every object in the inventory. The resulting list should be cross-referenced with the known set of group memberships in your Active Directory environment. An example of the output is shown in Listing 2.

You can save the output as a CSV file, then open it in a spreadsheet or other program for additional processing or analysis. In order to run this script, you must have the VMware Infrastructure Perl Toolkit installed on a system that also has Perl installed. You can find the toolkit at *http://sourceforge.net/projects/viperltoolkit/*.

Listing 1: Script to Query VirtualCenter for Roles

```
#!/usr/bin/perl -w

#  Permission Export Utility v1.0
#  Contribution by: Karl Rumelhart  (krumelhart@vmware.com)
#
#  For each for each type of managed entity, HostSystem, VirtualMachine, Datacenter,
# Folder, ComputeResource (i.e. host or cluster), and ResourcePool, this script
# retrieves all objects of that type and then all permissions that are set on the
# objects. It prints out the Object Type, Object Name, User/Group, and Role in comma
# separated value format. This can be piped to a file ("> foo.csv" in windows) and
# opened with Excel.
#  Version History:
#  V1.00 - (22 Dec 2006)

use strict;

use Getopt::Long;
use VMware::VIRuntime;

my %opts = (service_url => undef,
            userid      => undef,
            password    => undef);

GetOptions (\%opts,
            "service_url=s",
            "userid=s",
            "password=s");


if( !defined ($opts{service_url} && $opts{userid} && $opts{password} ) ) {
   help();
   exit (1);
}
```

```
# login
Vim::login(service_url => $opts{service_url}, \
    user_name => $opts{userid}, password => $opts{password});

# the authorization manager is the key to getting permission info
my $auth_mgr = Vim::get_view(mo_ref => Vim::get_service_content()->authorizationManager);

# Get all roles and put them in a hash so we can easily get the name corresponding to
# a roleId
my %role_hash;
my $role_list = $auth_mgr->roleList;
foreach (@$role_list) {
  $role_hash{$_->roleId} = $_->name;
}

# Heading for csv columns
print "Object Type, Object Name, User/Group, Role" . "\n";

# for each type of managed entity run through all objects of that type and all
# permissions defined on that object and print out the corresponding Object Type,
# Object Name, User/Group, Role
my @obj_types = ('HostSystem', 'VirtualMachine', 'Datacenter', 'Folder', \
    'ComputeResource', 'ResourcePool');
foreach my $this_type (@obj_types){
    my $obj_views = Vim::find_entity_views(view_type => $this_type);
    foreach (@$obj_views) {
        my $obj_name = $_->name;
        my $perm_array = $auth_mgr->RetrieveEntityPermissions(entity => $_, inherited => 1);
        foreach(@$perm_array) {
        # print object type and name
        print $this_type . ", " . $obj_name . ", ";
        # print user/group and role
        print $_->principal . ", " . $role_hash{$_->roleId} . "\n";
        }
    }
}


# logout
Vim::logout();
```

```
sub help {
   my $help_text = <<'END';

USAGE:
   printperms.pl --service_url <SDK service URL> --userid <VC user login> --password
<VC password>

Example:
   perl printperms.pl --service_url https://localhost/sdk/vimService --userid
administrator --password mypassword

The output will be in csv format.  Pipe to a file to open with Excel.

END
   print $help_text;
}
```

Listing 2: Sample Output Listing Roles

```
'Object Type','Object Name','User/Group','Role'
HostSystem,hostA.vmware.com,Administrators,Admin
HostSystem,hostB.eng.vmware.com,Administrators,Admin
VirtualMachine,CRM Server,Administrators,Admin
VirtualMachine,CRM Server,VCUser,VirtualMachinePowerUser
VirtualMachine,Webserver2,Administrators,Admin
VirtualMachine,Webserver2,VCUser,VirtualMachinePowerUser
```

About the Author

Charu Chaubal is technical marketing manager at VMware, where he specializes in enterprise datacenter management. Previously, he worked at Sun Microsystems, where he had more than seven years' experience designing and developing distributed resource management and grid infrastructure software solutions. He has also developed and delivered training courses on grid computing to a variety of customers and partners in the United States and abroad. Chaubal received a Bachelor of Science in Engineering from the University of Pennsylvania and a Ph.D. from the University of California at Santa Barbara, where he studied the numerical modeling of complex fluids. He is the author of numerous publications and several patents in the fields of datacenter automation and numerical price optimization.

Acknowledgments

The author would like to thank the following for their valuable input: Doug Clark, Karl Rummelhart

Revision: 20070404 Item: BP-017-PRD-01-01

VMware, Inc. 3145 Porter Drive Palo Alto CA 94304 USA Tel 650-475-5000 Fax 650-475-5001 www.vmware.com

vmware®