Global Knowledge®

Expert Reference Series of White Papers

Hyper-V Replicas in Windows Server 2012

Hyper-V Replicas in Windows Server 2012

Glenn Weadock, Global Knowledge Instructor, MCITP, MCSE, MCT, A+

Introduction

Hyper-V Replicas make it possible to copy Hyper-V virtual machines across a LAN or WAN even if you don't have a failover cluster or shared storage between the virtual machines. Hyper-V fault tolerance and disaster recovery are therefore cheaper and simpler than ever before.

I. The Hyper-V Replica Concept

One of the most significant new features in Windows Server 2012 is the Hyper-V Replica (HVR) capability. Whether you are considering this for your own organization, or just prepping for your Windows Server 2012 MCSA, this white paper presents the essentials of deploying this disaster recovery feature.

A. How Replicas Work

HVR allows you to designate a remote host to which a Hyper-V virtual machine is replicated. Once the replication relationship has been established and the initial replication has occurred, writes to the source virtual machine (VM) are collected in a log file and sent asynchronously to the destination VM on a designated schedule. Because the writes are "deltas" (i.e., differences), the bandwidth may not be prohibitive, even across a WAN link, although some initial monitoring is probably advisable. Because the replication is asynchronous, a link with variable latency does not present problems.

B. How HVR is Different

HVRs serve a different purpose than either Hyper-V live migration or storage migration. The latter operations are intended for use in the normal course of business, and presume availability of source VMs. HVRs, however, are meant for use during a disaster or other unplanned failure, when the source VMs are unavailable.

Comparing HVR to failover clusters, the requirements for HVR are less onerous. Basically all you need is Server 2012 or Server 2012 R2 on the Hyper-V hosts, and hardware that supports Hardware-Assisted Virtualization (Intel VT or AMD-V) as well as Data Execution Prevention (DEP)—that is to say, just about any server built in the last few years. Note that although Second Level Address Translation (SLAT) is required for Client Hyper-V on Windows 8.x, this CPU technology (included for example in Intel i3, i5, and i7 processors) is *not* required for the server flavor of Hyper-V, even though it should improve performance. Note also that you do not need identical hardware for the primary and replica servers.

HVR technology does not guarantee that there will be zero data loss in all circumstances. We'll look at the different types of failover scenarios shortly, but for now just note that there could be some data loss in the "unplanned failover" situation. (No data loss occurs in the "test failover" and "planned failover" scenarios.)

HVR host machines do not need to be in the same domain, and they can even be workgroup machines. As of mid-2014, they can also be in the Microsoft cloud. Initially you had to have System Center Virtual Machine Manager (SCVMM), but at the very end of 2014, Microsoft opened up Azure-based replication to non-System-Center customers, calling it "Disaster Recovery to Azure for Branch offices." For more, see the links at the end of this paper.

C. Scenarios where Replicas Make Sense

HVRs make sense in various scenarios, but perhaps the most common is when you need to provide the ability to quickly "fail over" a virtual machine to an alternate site across a WAN link in the event the VM becomes unavailable in a primary site. For example, you could use replicas to provide fault tolerance for branch office VM deployments by replicating branch VMs to the central headquarters. A service provider could also provide VM replication hosting as a service. Such hosting can be cloud-based using Azure.

It is true that Server 2012 provides a multi-site failover cluster capability, but this requires a pre-existing SAN-to-SAN replication capability, as well as Microsoft logo-certification of cluster hardware and the passing of cluster validation tests. Also, a multi-site failover cluster is significantly more complex to implement than HVR.

Another scenario is when you need speedy VM disaster recoverability on a LAN, but your VMs are not configured with the shared storage that would permit setup of a failover cluster. (Note that Server 2012 permits shared storage in the form of SMB 3.0 file shares, so you don't have to use attached storage anymore.) Of course, another possibility is that your LAN VMs may have shared storage, but you have not set up failover clustering because of the logo-certification or validation issues mentioned earlier.

D. Daisy-Chaining Replicas in Server 2012 R2

The "R2" release of Server 2012 provides additional flexibility in that it permits you to "daisy-chain" the replica path, meaning, you can replicate a VM from Server 1 to Server 2, and then replicate it again from Server 2 to Server 3. (Microsoft calls this "extended replication.") This provides fault tolerance in the event the primary server *and* the first replica server should fail.

In Hyper-V Manager on the replica server, right-click the replica VM and choose "Extend Replication," then proceed through the wizard. Note that there is no option to configure the primary server to replicate to multiple destinations.

II. Setting up the Replica Server

The first step in implementing Hyper-V replication is to set up the "replica server," that is, the target or destination system for the virtual machines you wish to replicate. For example, a replica server may live on a remote site that you can use for disaster recovery. (The discussion in this section presumes that a single physical Hyper-V host acts as the replica server; the last section discusses the scenario in which the replica server is actually a failover cluster.) Note that you can also accomplish the setup steps listed here with PowerShell; see the link at the end of this white paper for details.

The Hyper-V host must be running Server 2012 or Server 2012 R2. In Hyper-V Manager, either run locally or from a workstation outfitted with the Remote Server Administrative Tools (RSAT), open the Hyper-V settings window for the host that you want to be the replica server (it's one of those non-resizable windows from the Jurassic era of GUI design), click "Replication Configuration" in the left pane, and select "Enable this computer as a Replica server" in the right pane (see Figure 1). The various options will then appear blacked-in beneath the checkbox.

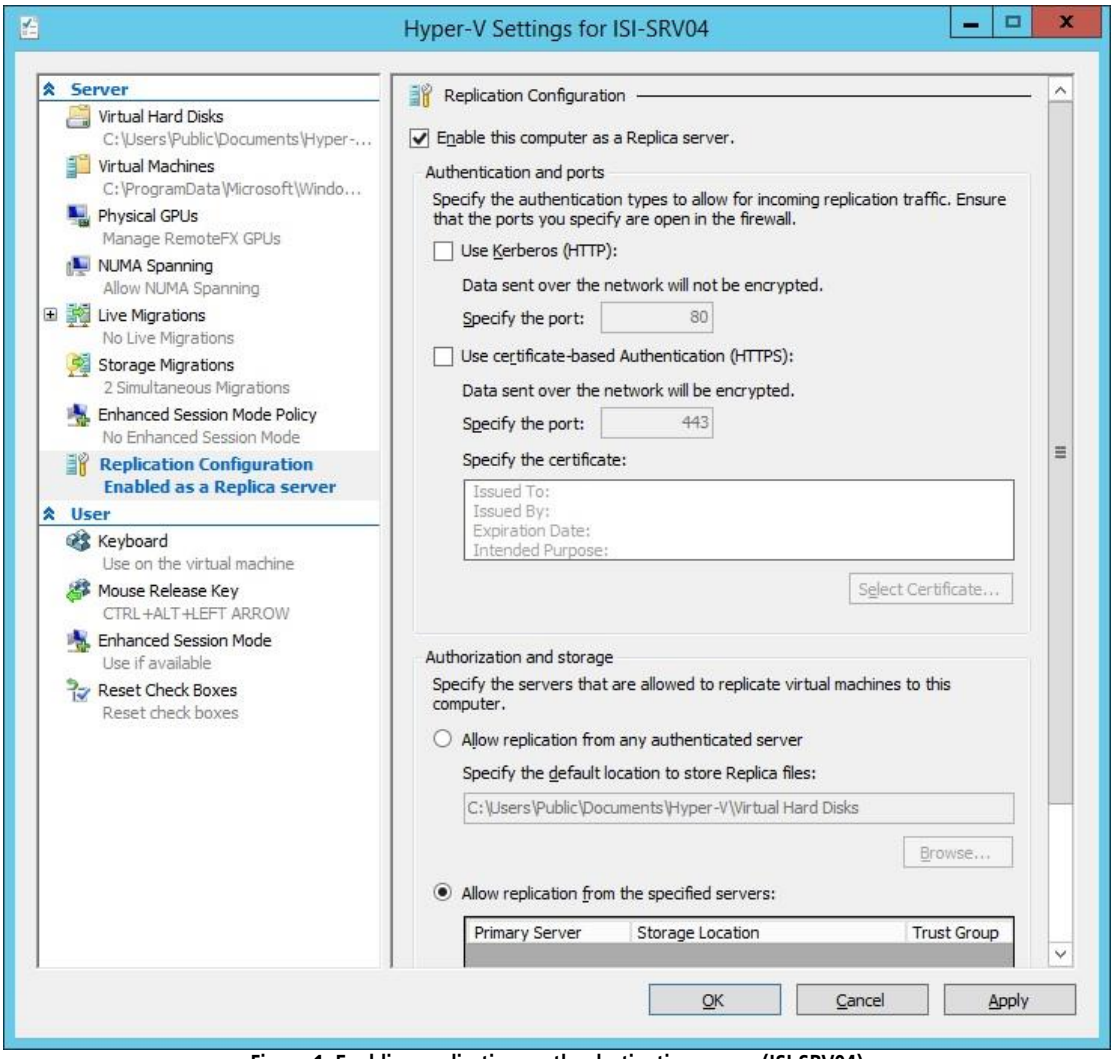

Figure 1: Enabling replication on the destination server (ISI-SRV04)

A. Kerberos *vs.* Certificate Authentication

Hyper-V replication requires some form of authentication. If the server you are configuring is not domain-joined, certificate-based authentication via HTTPS is your only option. The certificate can be issued in the context of a Public Key Infrastructure (PKI) or it can be self-signed; in either case, it must support both client authentication and server authentication, and be installed on both primary and replica servers. The standard port is 443. Advantageously, certificate authentication permits the replication traffic to be encrypted.

Kerberos authentication via HTTP is an additional option if both the primary and replica servers are trusted domain members. The default port is 80. No further configuration is required. Kerberos authentication does not provide for encryption; however, you can at least perform the initial replication using external media, which may be manually encrypted (e.g., with BitLocker-to-Go).

B. Firewall Rules

You have to enable the appropriate firewall rules manually, as the wizard is apparently a bit lazy; but Windows at least tells you which rules they are, depending on the authentication decisions you made (see Fig. 2 below). For Kerberos, enable the Hyper-V HTTP listener, and for certificates, enable the HTTPS listener.

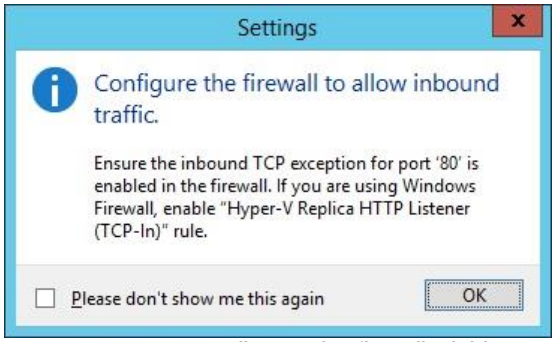

Figure 2: Hyper-V Manager tells you what firewall rule(s) to enable.

C. Restricting Replication to Certain VMs

Hyper-V replication can be bandwidth-intensive, so it makes sense to restrict replication sources when configuring a replication target (see Fig. 3 below). Also, for security purposes, it is generally beneficial to limit any kind of replication traffic (e.g., DNS updates) to known servers. Here, you don't name specific VMs, but rather specific VM hosts. Note also that when you specify a server DNS name, you must also provide a "trust group" name—if you only have one source server, call the trust group whatever you like. Note also that it's fine to specify multiple servers; a single replica server can host replicas from multiple primary VMs, from one or more primary VM hosts.

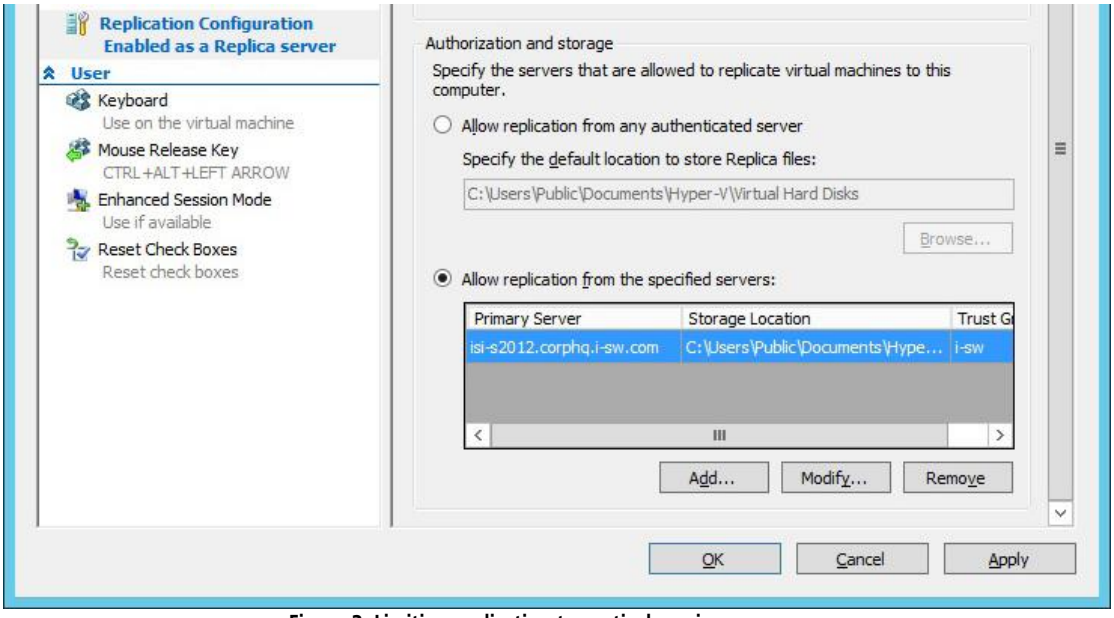

Figure 3: Limiting replication to particular primary servers.

III. Setting up the Primary Server

After you have the replica server set up, you can proceed to configure the virtual machine you wish to replicate, which lives on a so-called "primary server" (the physical host).

TIP: If you wish to perform a planned failover, you should configure the primary server's "Hyper-V settings" as you did on the replica server, because during a planned failover, the replication direction is reversed. This makes sense when you realize that any changes or updates that occur when the destination VM is active must be replicated back to the source VM when it comes back online. (The reversal does not automatically occur during an unplanned failover.)

A. Enabling Replication

Enable replication for the VM using a wizard that you initiate by right-clicking the VM in Hyper-V Manager and selecting "Enable Replication" (see Figure 4). You can also highlight the VM and choose the same command from the Actions pane.

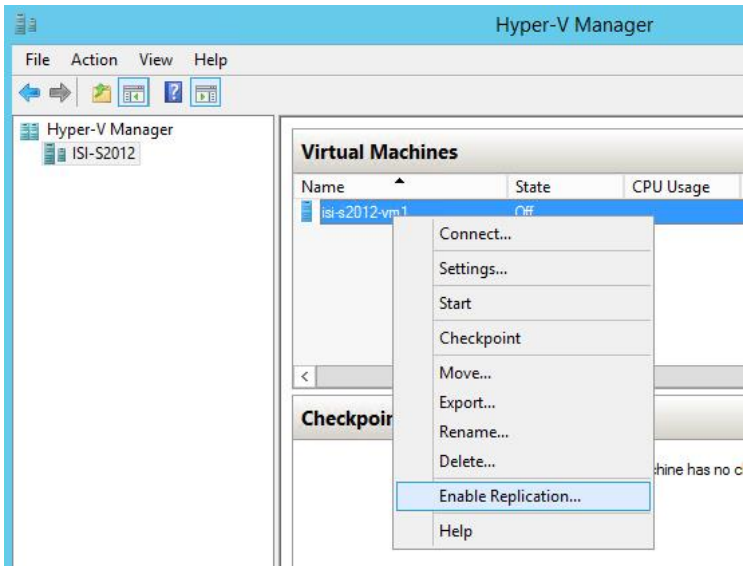

Figure 4: Starting the replication wizard for ISI-S2012-VM1.

To begin, you specify the name of the replica server you intend to be the target host for the selected virtual machine. Then you specify whether you want the data sent over the network to be compressed; you may wish to depart from the default (on) if you have a more efficient site-to-site traffic optimization technology already in place.

The "Choose Replication VHDs" step of the wizard lets you clear the checkbox next to any VHDs you do not wish to replicate, such as a VHD whose only use might be for system files (e.g., a page file) that your replica VM could rebuild when activated.

B. Frequency

In Windows Server 2012 you can't specify the replication period, but Microsoft advises that it's about five minutes. In Server 2012 R2, you can choose among thirty seconds, five minutes, or fifteen minutes.

The best replication interval will depend on available network bandwidth, VM activity, and how important it is to you not to lose data during an unplanned failover. If your network can support it, a thirty-second interval means that you will lose very little data under any circumstances.

Tip: You can configure the replication frequency separately for different primary VMs.

C. Recovery Points

By default, the replica server maintains only the latest (most recent) recovery point (snapshot, or in Server 2012 lingo, "checkpoint"—perhaps to distinguish from shadow copy snapshots or Active Directory snapshots) for the replica VM, but you can opt to keep multiple recovery points as long as you have sufficient storage (see Figure 5). Windows creates the recovery points hourly. You will have the option to specify which recovery point you wish to use when you perform a failover. This might be useful if you suspect that the failure of the primary VM may have occurred over a period of hours and you want to go back to a known-good state. Microsoft provides a VSS snapshot capability for non-Windows guest operating systems.

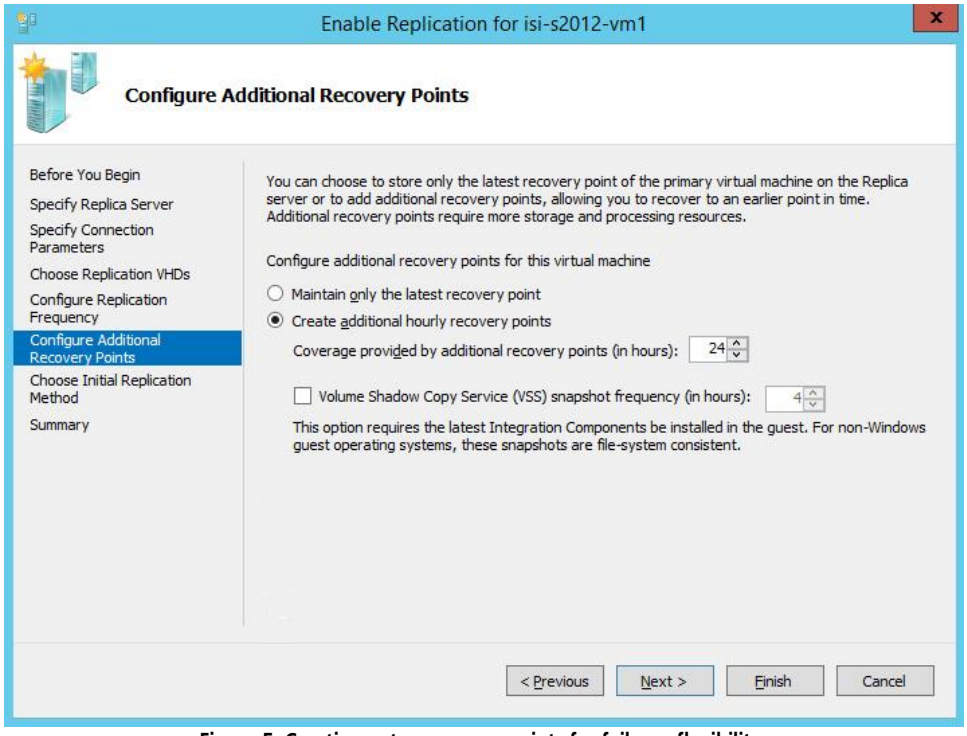

Figure 5: Creating extra recovery points for failover flexibility.

Note that Server 2012 only lets you store up to fifteen hours' worth of recovery points, whereas Server 2012 R2 lets you store up to twenty-four hours' worth.

D. Initial Replication Method

The initial copy of your primary virtual hard drives can be resource intensive. You may wish to avoid using the network to perform the initial replication if your source VM is large (the one in Figure 6 below is more than 8GB) or if your network is already performing at or near capacity. You may also wish to encrypt the initial replication even though you will be using Kerberos authentication for subsequent "delta" replications.

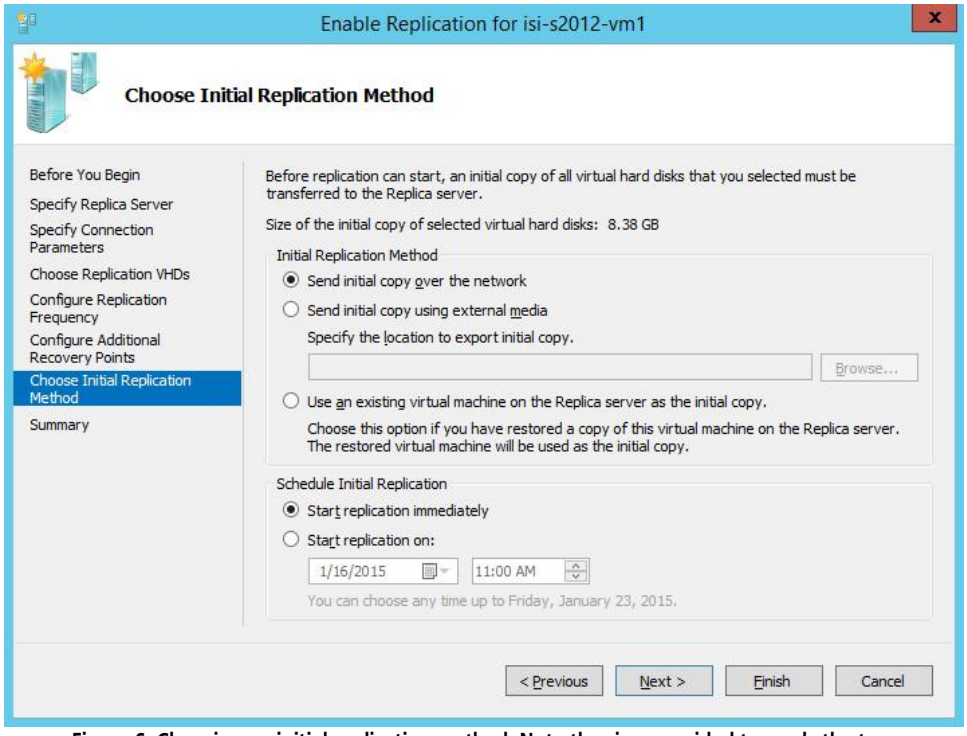

Figure 6: Choosing an initial replication method. Note the size, provided towards the top.

You can use external media, such as a flash drive, in such a scenario. (If the drive doesn't have enough space and the initial replication fails, you can retry it by right-clicking the primary VM and choosing "Replication" > "Start Initial Replication.") Windows doesn't supply either a progress bar or a completion message. If you click the "Replication" tab in the details pane for the primary VM in Hyper-V Manager, you can see the replication state listed as "Initial replication in progress" (see Figure 7), but that message remains even after the external media has been prepared; so right-click the VM occasionally until the option to pause or cancel the initial replication disappears. (The initial-replication checkpoint also disappears in the central pane when it's done.)

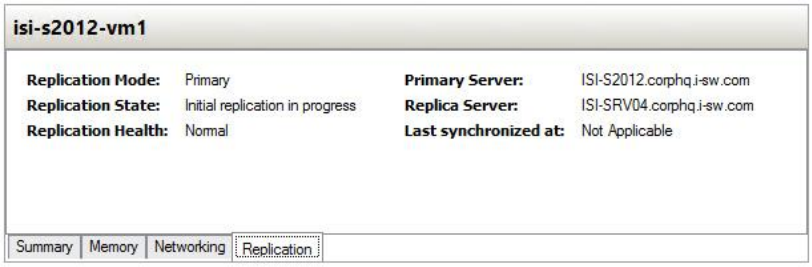

Figure 7: Monitoring the initial replication to external media

The next step is to bring the external device to the replica server, where you can import the VM in Hyper-V Manager, as shown below in Figure 8. The import goes more quickly than the export, and this time you do get a progress bar!

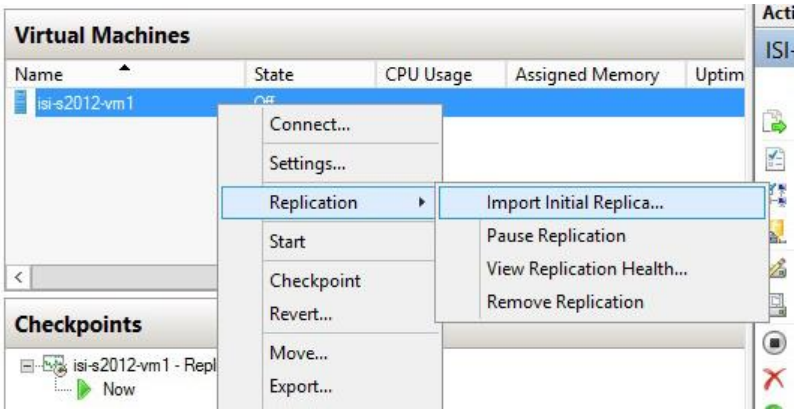

Figure 8: Completing the initial replication on the replica server

As an alternative to using external media, you can also use a backup of the primary VM that you have restored on the replica server, in which case only operations that have occurred since the backup will be replicated across the network.

Tip: If you do choose to perform the initial replication over the network, you can schedule the procedure for up to a week into the future, preferably for a day and time when the network is relatively quiet.

E. Completion

After the initial replication occurs, the "delta-based" replications begin; the first one may fail with a timeout error, but subsequent attempts should succeed. You can monitor replication via the "Replication" tab as shown in Figure 7, or for more complete information, choose "Replication" and "View Replication Health" from the VM's context menu. Clear any old warning or error conditions with the "Reset Statistics" button. (You can also add the Replication Status column to the top-center pane of Hyper-V Manager.) Finally, check the event log under "Applications and Services Logs > Microsoft > Windows > Hyper-V-VMMS > Admin."

IV. Types of Replica Failover

A. Test Failover

The "test failover" is the only type of replica failover that involves zero downtime for the virtual machine. You initiate a test failover on the replica server via the VM's context menu, and choose a recovery point (see Figure 9). A new virtual machine with "Test" appended to the name is created during this procedure. Not only does the primary VM remain up, replication activity to the original replica VM continues uninterrupted. Once the test VM is built, you can fire it up to make sure it works.

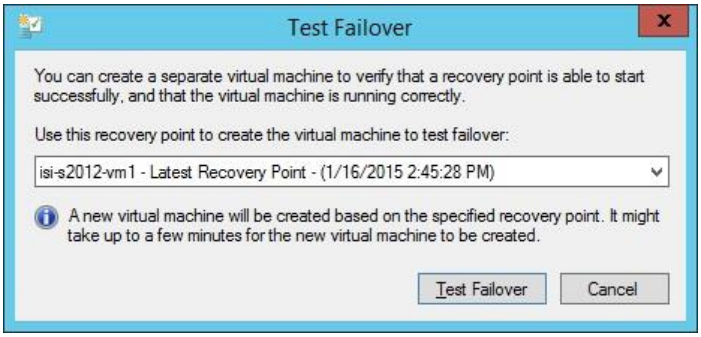

Figure 9: Choosing a recovery point for a test failover

B. Planned Failover

The "planned failover" involves some downtime for the virtual machine because the first step is to shut down the source VM. The planned failover, therefore, differs from the test failover in that you initiate it at the primary server. Right-click the VM and choose "Planned Failover" from the "Replication" submenu. Make the appropriate selections in the dialog box (see Figure 10).

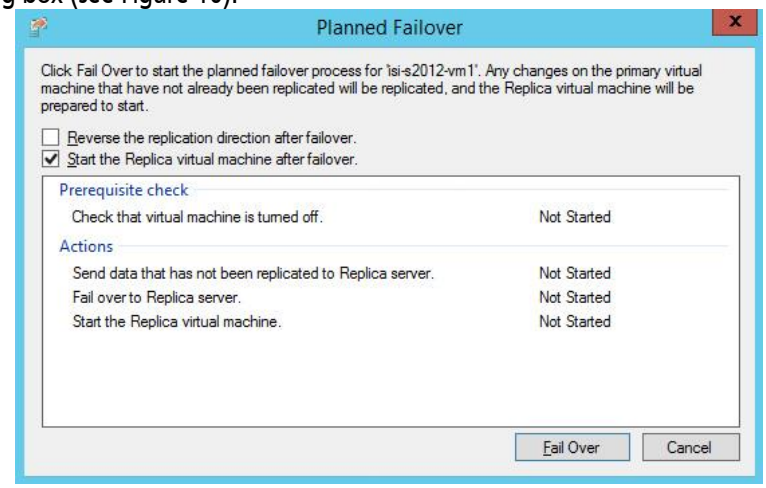

Figure 10: Initiating a planned failover

Any data on the source VM that had not already been replicated to the target VM is replicated; the failover is performed; and (if you choose the checkbox) the replication direction is reversed, so that changes to the VM can be replicated back when the source VM comes back online. (This reversal is why you need to configure the primary server, as well as the replica server, to host Hyper-V replicas.) You can perform another planned failover in the reverse direction after the source VM is once again ready to go "live."

The planned failover exercises more aspects of the replica architecture than the test failover does, and is therefore a more thorough test, but should be scheduled for a time when the primary VM does not need to be continuously available.

C. Unplanned Failover

An unplanned failover is a response to the unexpected failure of the primary server, for example, from flood, earthquake, fire, or someone forgetting to pay the electric bill. If you specified the option to create multiple recovery points when configuring the primary server, you can choose which recovery point you want to "fail over" to when performing the unplanned failover. Initiate an unplanned failover at the replica machine by choosing "Failover" from the VM's context menu.

V. Using Replicas with Failover Clusters

A failover cluster can function as a HVR site. The key thing to remember in this scenario is that you use the Failover Cluster Manager tool rather than the Hyper-V Manager.

In order to replicate to a failover cluster, install the "Hyper-V Replica Broker" role on the target system using the Failover Cluster Manager. Then, when configuring the primary VM in the "Enable Replication" wizard, specify the name of the Hyper-V Replica Broker instead of the name of the replica server.

Conclusion

Fault tolerance for Hyper-V has been available for a few years now in the form of failover clusters, but Server 2012 brings a simpler option. HVRs don't require shared storage, they don't require logo-certified or matching hardware, and they work well across sites. When your traditional VM backup and restore procedures can't provide fast enough recovery for specific VMs, and you'd rather not deal with the requirements of clustering, consider HVRs. Just keep an eye on that network traffic, perform occasional test and planned failovers, and periodically check replication health. And finally, be aware that when you make a configuration change (such as memory allocation) on the primary VM, that change does *not* propagate to the replica VM.

For Further Reading

- PowerShell cmdlets for Hyper-V, including replica functions: http://technet.microsoft.com/hh848559.aspx
- Hyper-V component architecture posters for Server 2012 R2: http://www.microsoft.com/en-us/download/details.aspx?id=40732
- Hyper-V component architecture posters for Server 2012:
- http://www.microsoft.com/en-us/download/details.aspx?id=29189
- Azure Site Recovery:
  http://azure.microsoft.com/en-us/services/site-recovery/
  ...and the December 2014 policy update is here:
  https://weblogs.asp.net/scottgu/azure-premium-storage-remoteapp-sql-database-update-live-media-streaming-search-and-more

Learn More

Learn more about how you can improve productivity, enhance efficiency, and sharpen your competitive edge through training.

Administering Windows Server 2012 (M20411)

MCSA: Windows Server 2012 Boot Camp

Implementing and Managing Microsoft Server Virtualization (M10215)

Visit **www.globalknowledge.com** or call **1-800-COURSES (1-800-268-7737)** to speak with a Global Knowledge training advisor.

About the Author

Glenn Weadock is a longtime instructor for Global Knowledge and teaches Windows 8, Windows 7, Vista, Server 2012, Server 2008, and Active Directory. He co-developed with Mark Wilkins two advanced Server 2008 classes in the Microsoft Official Curriculum. Glenn also consults and provides expert witness services in patent infringement cases through his Colorado-based company Independent Software, Inc. (www.i-sw.com).