Hyper-V virtual lab tips and tricks

Brien M. Posey

It has been said that network administrators have to have nerves of steel. Administrative actions such as operating system (OS) upgrades, patch management and application deployments can have catastrophic consequences if not properly planned and tested. Even so, it has traditionally been difficult for administrators to know with 100% certainty what the outcome of such actions will be, even if testing has been done.

Thankfully, server virtualization has the potential to take most of the risks out of administrative actions. Virtual labs make it possible to rehearse upgrades, migrations and other high-risk administrative actions safely within an isolated environment. This article discusses some tips and tricks for virtual lab testing in a Hyper-V environment.

Why lab testing is so important

Before I delve into a discussion of Hyper-V and virtual lab testing, I want to set the tone for this article by talking about my worst ever experience as a network administrator. Back in the early 1990s, I worked as a network administrator for a large insurance company. I was in my freshman year of college at the time, but had been lucky enough to land a network administration job even though I had very little experience.

At that time, all of our servers were running on Novell NetWare 3.X. The problem with NetWare 3.X was that it was not very scalable. Each server contained an independent bindery, which was basically an authentication and authorization database. Because of the way that the software worked, each user had to have a separate account on every server containing resources that they needed access to. These accounts were not automatically synchronized, nor was there any sort of centralized administration.

All of that changed with NetWare 4.X. Novell introduced their own version of the directory services, which were very similar to the Windows Active Directory that was released several years later. At first, NetWare 4.X seemed like the perfect solution to our organization's scalability problems. My manager immediately purchased the necessary licenses, and sent the entire IT staff to some training classes.

Once the training was complete, my manager insisted on jumping right into the migration process. The problem was, however, that he had put very little thought into the actual migration. He simply bought some server hardware and a utility that was supposed to automate the migration, and that was it. There was no pre-migration testing, and there was very little planning. There was also no contingency plan in case anything went wrong.

My manager was the hands-on type and wanted to perform the migration himself, but insisted on having the entire IT staff on hand in case he needed help with anything. He began the migration process at about 6 o'clock on a Friday evening. We spent about twenty minutes watching the migration software dutifully move content from an old server. Since everything appeared to be going smoothly, my manager made the decision to start parallel migrations for the remaining servers. Once all the migrations were in process, we all took a break to go have some dinner.

My manager had estimated that the migrations would probably be complete by about ten or eleven o'clock that night. It was actually several hours later when the migration of the first server completed. It was only then that we discovered that something had gone horribly wrong.

There were actually several problems that occurred as a result of the migration, not the least of which was extremely poor performance. The big problem, however, was that every bit of security information had been lost during the migration process. User passwords were completely removed, so anyone could log into any account without specifying a password. Furthermore, all but the most basic permissions were also gone. We had completely lost all security group memberships, as well as any record of which security groups had access to which files and folders. Even the individual user accounts had lost permission to access their home directories.

What made this problem even worse, however, was that in his impatience, my manager had made the decision to do parallel migrations of all of our servers. We were left in a state where every server in our entire organization had lost all security information and we also had introduced some serious performance problems. There was no turning back. Our migration software had purged the contents of the old servers and my manager had not bothered to make backups of the old servers just before the migration because he didn't want to waste time waiting for the backups to complete.

It took us four days to manually reimplement all of the necessary permissions and to correct the performance problems. Some of the missing permissions we were able to figure out on our own, but in a lot of cases we had to wait for the phone to start ringing on Monday. Over the span of four days, nobody in the IT staff went home and nobody slept. When we finally did have everything fixed, I remember being so tired that I could barely keep the car on the road during the short drive home. Once I got home, however, it took me a while to fall asleep because I was so wired from being awake for the last four days.

This was by far the worst IT experience of my entire life (and I hope that nothing ever happens to top it). As horrible as the experience was however, it taught me some extremely valuable lessons. First, I learned that there is a price to pay when IT managers make bad decisions. That was something that I always kept in mind later in life when I eventually became a manager, and later a CIO. More importantly however, the experience taught me in no uncertain terms the importance of backups and the importance of testing potentially destructive procedures before performing them in a production environment.

The practicality of lab testing

If lab testing had been performed, the situation described above could have been completely avoided. Those involved could have discovered that the migration technique was flawed before any damage could have been done to production systems. Of course major OS migrations are not the only administrative action for which lab-based testing could prove to be beneficial. Lab testing can be helpful for patch management, application upgrades and all types of configuration changes.

All too often, organizations such as the one previously described, do not engage in any sort of extensive lab testing. Lab testing has traditionally been regarded as expensive, cumbersome, and time-consuming. Although there was once a time when lab testing tended to be impractical, server virtualization now makes lab testing much easier and dramatically cheaper than it was in the past.

Why not just use virtual machine snapshots?

Although lab testing can be undeniably beneficial, some have suggested that it's an outdated concept. The reasoning behind this idea is that hypervisors such as VMware vSphere and Microsoft Hyper-V include a virtual machine (VM) snapshot feature that allows a VM to be rolled back to a previous state in the event that something goes wrong. With that being the case, it seems fair to consider whether an organization is better off deploying a virtual lab and testing administrative actions in a lab environment, or simply attempting an administrative action and using snapshots (or checkpoints as Microsoft calls them in Windows Server 2012 R2) as a mechanism for instantly recovering when something goes wrong. The other critical consideration is that a snapshot (or SCVMM checkpoint) will keep the VM on the production network, which can interfere with Active Directory Update Sequence Numbers (USNs) or exchange data between distributed systems.

Although Hyper-V snapshots have their place, snapshots are a poor substitute for comprehensive lab testing. After all, if an administrator relies solely on snapshots, the consequences of an administrative action that has gone wrong will impact the production environment for at least for a short period of time.

Another good reason for not relying on snapshots is that sometimes an administrative action can impact multiple VMs. Consider a scenario in which a PowerShell script is accidentally run against all VMs instead of a specific VM. It is naïve to think that in such a situation an up-to-date snapshot would exist for every VM in the entire organization. Even if up-to-date snapshots were available for every VM, the rollback process could potentially involve a lot of work simply due to the number of VMs that are involved.

It is also a good idea to avoid the use of snapshots whenever possible because snapshots impact VM read performance. When you create a snapshot of a VM, Hyper-V creates a differencing disk and redirects all write traffic to the differencing disk. This slows down read performance because Hyper-V must first check the differencing disk to see if the requested data resides on that differencing disk. If Hyper-V does not locate the requested data then it must retry the read operation against the original virtual hard disk (VHD). And since each snapshot that's created for a VM causes an additional differencing disk to be created, the impact on performance is compounded.

Perhaps the best reason of all for not relying on snapshots is that some application servers do not support snapshots. For example, Microsoft does not support VM snapshots for use with applications such as Exchange Server or SharePoint Server.

The ideal solution

Before I begin discussing any Hyper-V specifics, I want to first talk about what I perceive to be the ideal solution to virtual lab testing. Backup applications exist that include built-in virtual lab functionality. Such applications are able to provision a sandboxed virtual lab environment with a few mouse clicks. The virtual lab environment is typically created from the most recent backup, but is done in such a way that a restoration of the backup is not required and the backup's integrity is maintained.

There are two main advantages to using this approach. First, this approach is easy to use. A virtual lab environment can be created using minimal effort and minimal system resources.

The second advantage is that the virtual lab environment perfectly mimics your production environment. This means that any testing that you do in your virtual lab environment should yield the same results as the procedure would if performed in your production environment.

A home-grown virtual lab

Although backup software with a built-in virtual lab feature can take almost all of the work out of setting up a virtual lab, many organizations use backup applications that lack these capabilities. In such environments, it is still possible (and highly recommended) to perform virtual lab testing, but the virtual lab will have to be manually created. The remainder of this article focuses on issues to keep in mind when you create a virtual lab.

Backup software can even provide a virtual lab that doesn't require additional storage, such as reading the backup files directly to present the VMs to the Hyper-V host without restoring the whole data footprint.

The network infrastructure

One of the biggest logistical challenges with regard to virtual lab testing is network connectivity. The virtual lab environment must be configured in a way that provides isolation from the production network. You can't risk having the activities that you are performing in your lab tests impact the production environment.

The easiest way to achieve this isolation is to use a physically isolated network segment. In other words, you can use dedicated lab hardware in a way that ensures your lab VMs and your production VMs never cross paths. At the same time, however, physical separation of the two environments presents challenges of its own.

One of the issues that you will encounter if you choose to use physical separation relates to Internet connectivity. Today it is more or less assumed that most servers will have some form of Internet access (even if that access is highly restricted). Preventing your lab servers from being able to access the Internet may cause them to behave slightly differently in a lab environment than they would in a production environment.

If you choose this approach, the more important issue that you may encounter involves making various resources accessible to the lab environment. Think about it for a moment. The only way to truly make sure that the machines in your virtual lab perform in the same way as your production VMs is to restore a backup of your production VMs to the lab environment. You could try to manually configure the lab machines to match the configurations that are in use in your production environment, but doing so leaves a lot of room for error. From an accuracy standpoint, you are much better off restoring your backups to the lab environment. The problem with this is if your lab environment is physically isolated from the production environment, it may cause logistical problems with restoring your backups.

Another approach to keeping your lab environment and your production environment separate is to use software-defined networking. In the world of Hyper-V, software-defined networking is relatively new. In fact, Microsoft didn't introduce a true software-defined networking mechanism until Windows Server 2012 R2.

For those who are unfamiliar with software-defined networking, it is a form of logical networking that allows multiple logical network segments to exist on top of a Hyper-V virtual network. Microsoft created this feature as a way of improving Hyper-V support for multi-tenancy. Multi-tenant networks can exist on a common virtual network without the administrator having to worry about the tenant networks conflicting with one another.

The thing that makes this approach especially helpful for lab testing is that software-defined networking does not care about IP address overlaps. The VMs on your lab network could use the same IP addresses as their production counterparts without causing any problems.

Resource requirements

Another issue that must be taken into account when developing a virtual lab relates to the resources that are going to be needed within the virtual lab. The resource requirements will differ depending upon what it is that you are trying to test with the virtual lab. Typically, however, there are going to be some infrastructure server requirements.

Suppose, for example, that you are constructing a virtual lab as a way of testing a migration from Exchange Server 2010 to Exchange Server 2013. In this type of situation, you would obviously need to include instances of your Exchange Server 2010 VMs in the virtual lab. However, Exchange Server also has a dependency upon the Active Directory. Therefore, you would need a domain controller in your virtual lab as well.

While this might seem fairly obvious, this is only the first of the dependencies that must be brought into the virtual lab. For instance, the Active Directory depends on a DNS server. Therefore, your virtual lab will require at least one DNS server, and you will probably need a DHCP server as well.

Furthermore, Exchange Server depends on having a global catalog server available. As such, you will need to determine which of your domain controllers are acting as global catalog servers so that you know which domain controllers to include in your virtual lab. Similarly, some domain controllers in your organization hold certain Flexible Single Master Operations (FSMO) roles. You will need to identify which domain controllers hold these roles so that you can make sure to include those domain controllers in your virtual lab. The FSMO roles are required because Exchange Server 2013 updates the Active Directory schema, and those updated can only be applied to the domain controller that is acting as the schema master.

Because of these considerations and more, before you engage on a virtual lab strategy for Hyper-V, it's important to ensure that all requirements are met in terms of the VMs and networking available in the virtual lab. This is important in the era of distributed applications.

Hardware considerations

Another issue that you will have to take into account when developing your virtual labs is the hardware that the virtual lab servers will run on. If your production Hyper-V deployment has sufficient free CPU and memory resources, you might be able to use production hardware to temporarily accommodate the virtual lab. If you do decide to use this approach, it is important to make sure that you leave sufficient resources on each Hyper-V server to accommodate a failover in case one of your Hyper-V hosts should drop offline. The last thing that you want to do is to choke out production VMs in a failover situation by putting lab VMs on production hardware.

If you are concerned that your hardware may be inadequate to host lab VMs and still be able to absorb a failover, you can use the Windows Server 2012 VM prioritization feature to prioritize your VMs. By doing so, you can ensure that high-priority VMs (which would be your production VMs in this case) continue to run, while low-priority VMs such as the lab VMs are shut down if necessary so that Hyper-V can accommodate more important VMs.

Of course you might find that your production environment simply does not have sufficient hardware resources available to accommodate the virtual lab environment. If this happens, you can usually get away with running your virtual lab machines on relatively low-end hardware. Keep in mind that the virtual lab does not actually have to host a production workload. With that being the case, you can usually get away with assigning virtual lab machines fewer CPU and memory resources than would ordinarily be used in a production environment.

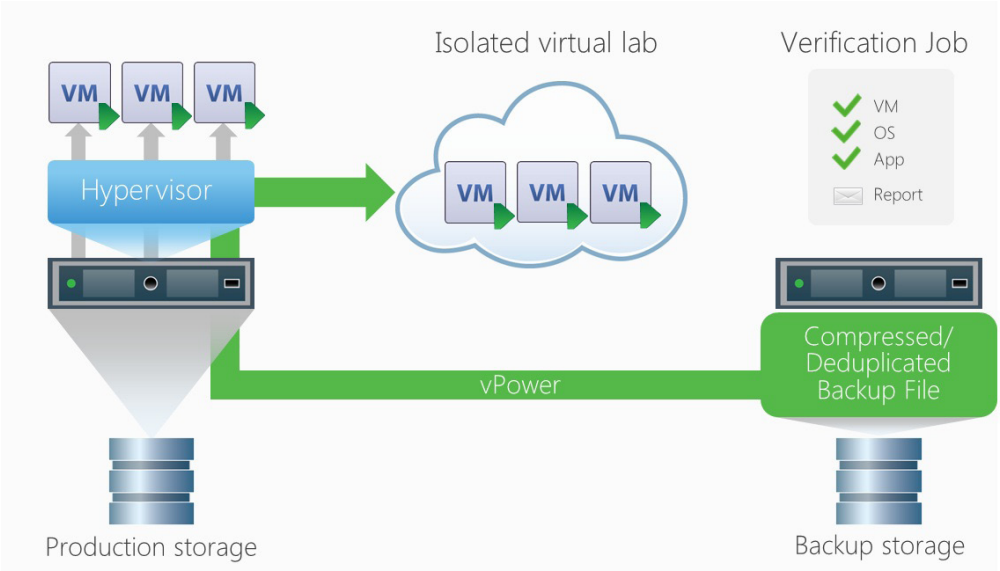
One resource that does dramatically impact a virtual lab environment, however, is disk I/O. Although your virtual labs will not need to carry a workload as I/O intensive as a production environment might, there are going to be certain minimum I/O requirements. Even if these requirements aren't necessarily set in stone, the disk subsystem will need to deliver enough IOPS to be able to provide an acceptable experience.

To give you a more concrete example of what I'm talking about, consider this. I work as a freelance technology writer, and I speak at dozens of technology events all over the world every year. Because of my hectic travel schedule, I sometimes have to work on articles while I am on the go. With that being the case, I purchased a high-end laptop with two hard drives and lots of memory so that I could create a number of VMs on my laptop. The idea was to be able to run the applications that I write about. The only problem with doing so, however, is that once I get more than a couple of VMs running, all of my VMs slow to a crawl. This occurs because multiple VMs are all sharing a single hard disk and are limited by the I/O bottleneck of this one disk.

The same principle also applies to running a virtual lab on hardware that isn't production grade. There is nothing wrong with using low-end hardware for your lab environment. However, you do have to make sure that the storage hardware provides sufficient performance to be able to at least keep pace with the virtual lab machines' demands.

Of course storage I/O is not the only thing that must be considered, you will also have to consider capacity. Since it is possible to go into any office supply store and purchase a multi-terabyte hard drive for just over $100, capacity isn't usually a huge problem. Even so, it does occasionally become an issue if you need to perform a test involving extremely large databases.

In the case of Veeam's implementation of the virtual lab, you can see in the diagram below that the VMs are presented from the backup storage and run in an isolated network. From a compute and memory perspective, the VMs run on the production Hyper-V hosts:



Hyper-V replicas as a lab testing resource

One of the best features in Windows Server 2012 Hyper-V is the VM replica feature. This feature allows you to replicate a VM to a separate Hyper-V server. Although replication does not provide the level of fault tolerance that a true failover cluster would, it provides an easy way of having a somewhat current secondary copy of a VM.

For organizations that use the Hyper-V replica feature, it is actually possible to use replica VMs as a lab testing resource. Under normal circumstances, a replica VM remains powered off. However, Hyper-V contains a mechanism that you can use to test your ability to fail over to the replica. To do so, you would open the Hyper-V Manager, right-click on the replica and choose the Replication | Test Failover commands from the shortcut menu.

Doing so causes a brand-new VM to be created based on the selected replica's latest recovery point. This VM can be powered up and you can interact with it in any way necessary without having to worry about harming the primary VM or its replica. All of your changes occur in a sandbox environment, safely away from the production environment.

The most important thing to know about Hyper-V replica failover testing is that replica VMs are not connected to the network by default. Normally if you perform lab testing, there are going to be multiple VMs involved in the test. With that being the case, it is a good idea to either create a dedicated virtual switch for the test machines or place all of the lab VMs on a dedicated VLAN. Whatever solution you choose to use, the lab virtual network must not come into contact with the production virtual network.

The lab testing process

So far I have talked at length about the advantages of creating a virtual lab and about some of the things you need to watch out for during the virtual lab construction process. Now, I want to turn my attention to the lab testing process.

Obviously your testing methodologies are going to vary depending upon what it is that you are trying to test. That's a given. However, there are certain best practices that you should adhere to during the testing process.

The first thing that I recommend doing once the virtual lab is in place and properly configured is to take time to make sure that the virtual lab environment is functioning properly. In order for your virtual lab tests to be valid, the virtual lab environment needs to mimic the production environment as closely as possible. It is critical to the validity of your tests to take the time to make sure that your virtual lab is configured properly. For example, it is a good idea to make sure that all of the VMs in the lab environment can communicate with one another, that they all have the proper IP addresses and that the DNS records for use in a virtual lab environment are correct. While you are at it, you should verify that all applications are working correctly and that any required services on your lab VMs have started.

I also recommend taking the time to verify the time zones and the clocks for all of the machines in your virtual lab environment. Windows uses Kerberos-based authentication. Kerberos is a time-sensitive protocol. If your clocks are out of sync, Kerberos can break down. When this happens it may cause authentication errors, and it can lead to all sorts of other unpredictable behavior.

Once you have taken the time to verify that everything in the virtual lab is configured properly and is working correctly, I recommend shutting down all of the VMs in the lab, and making a Hyper-V snapshot of each VM.

One example of backup software with a built-in virtual lab is Veeam Backup & Replication v7. The user in *Figure A* posted a scenario on Twitter in which he was able to test a number of VMs (Active Directory Domain Controller, SQL Server and a vCenter Server VM) in a virtual lab before an upgrade.

Back in school, I had a friend in math class who liked to come up with his own ridiculous mathematical theorems as a way of making a mockery of the subject matter that we were studying. One of the mathematical theorems that he came up with was a statement that "if something works once it's law." Of course we all knew that that was a big joke. In math it is very possible for something to work one time, but not hold up to further testing. The same concept applies to VMs in lab testing. Just because something works once does not necessarily mean that it will work again.

Your lab testing needs to be done in a way that it is repeatable. Of course this involves a major time commitment given the amount of effort that goes into setting up your virtual lab environment. By creating a Hyper-V snapshot of each VM before you begin testing, you have an opportunity to reset the lab environment so that you can repeat your testing without the hassle of setting everything up from scratch.

As you may recall, I previously stated that there were certain server applications that were not supported for use with Hyper-V snapshots. When it comes to this type of testing, this is not an issue. There are two reasons for this. First, supportability isn't a problem because you are dealing with a lab environment. It doesn't matter if the lab environment is supported or not.

More importantly, however, the reason why making Hyper-V snapshots of things like Exchange Server and SharePoint Server is not supported is because of the way the snapshot and process deals with VM state data. By shutting down the VM before you create the snapshot, you take the state data out of the equation completely. It therefore becomes perfectly safe to create a snapshot, and to use that snapshot if necessary. Keep in mind, however, that when it comes time to reset your virtual lab, you should power down all of the VMs and rollback all of the snapshots at once. Don't try to roll back the snapshots while the VMs are running or to roll back one VM and not another. The only way to ensure a consistent experience is to power down all of the VMs and roll them all back at once.

Conclusion

As you can see, using virtual labs to test administrative actions can go a long way toward preventing problems in your production environment. The best way to construct a virtual lab is to use backup software that has built-in virtual lab functionality. If you do not have this type of software, it is possible to create a perfectly adequate virtual lab using the mechanisms that are built into Hyper-V. Doing so simply involves more work than would be required if you had backup software with virtual lab functionality.

About the Author

Brien Posey is a freelance technical writer who has received Microsoft's MVP award 11 times for his work with Exchange Server, Windows Server, IIS, and File Systems Storage.

Brien has written or contributed to about three dozen books, and has written well over 4,000 technical articles and white papers for a variety of printed publications and Web sites.

In addition to his writing, Brien routinely speaks at IT conferences and is involved in a wide variety of other technology related projects.

About Veeam Software

Veeam® is Protection for the Modern Data Center™ - providing powerful, easy-to-use and affordable solutions that are Built for Virtualization™ and the Cloud. Veeam Backup & Replication™ delivers VMware backup, Hyper-V backup, rrecovery and replication. This #1 VM Backup™ solution helps organizations meet RPOs and RTOs, save time, eliminate risks and dramatically reduce capital and operational costs. Veeam Backup Management Suite™ provides all the benefits and features of Veeam Backup & Replication along with advanced monitoring, reporting and capacity planning for the backup infrastructure. Veeam Management Pack™ MP) extends enterprise monitoring to vSphere through Microsoft System Center and also offers monitoring and reporting for the Veeam Backup & Replication infrastructure. The Veeam Cloud Provider Program (VCP) offers flexible monthly and perpetual licensing to meet the needs of hosting, managed service and cloud service providers. VCP currently has over 4,000 service provider participants worldwide. Monthly rental is available in more than 70 countries from more than 50 Veeam aggregators.

Founded in 2006, Veeam currently has 23,000 ProPartners and more than 91,500 customers worldwide. Veeam's global headquarters are located in Baar, Switzerland and the company has offices throughout the world. To learn more, visit http://www.veeam.com.

VEEAM

IT JUST WORKS!™

Protection for the **Modern** Data Center

To learn more, visit **http://www.veeam.com/backup**