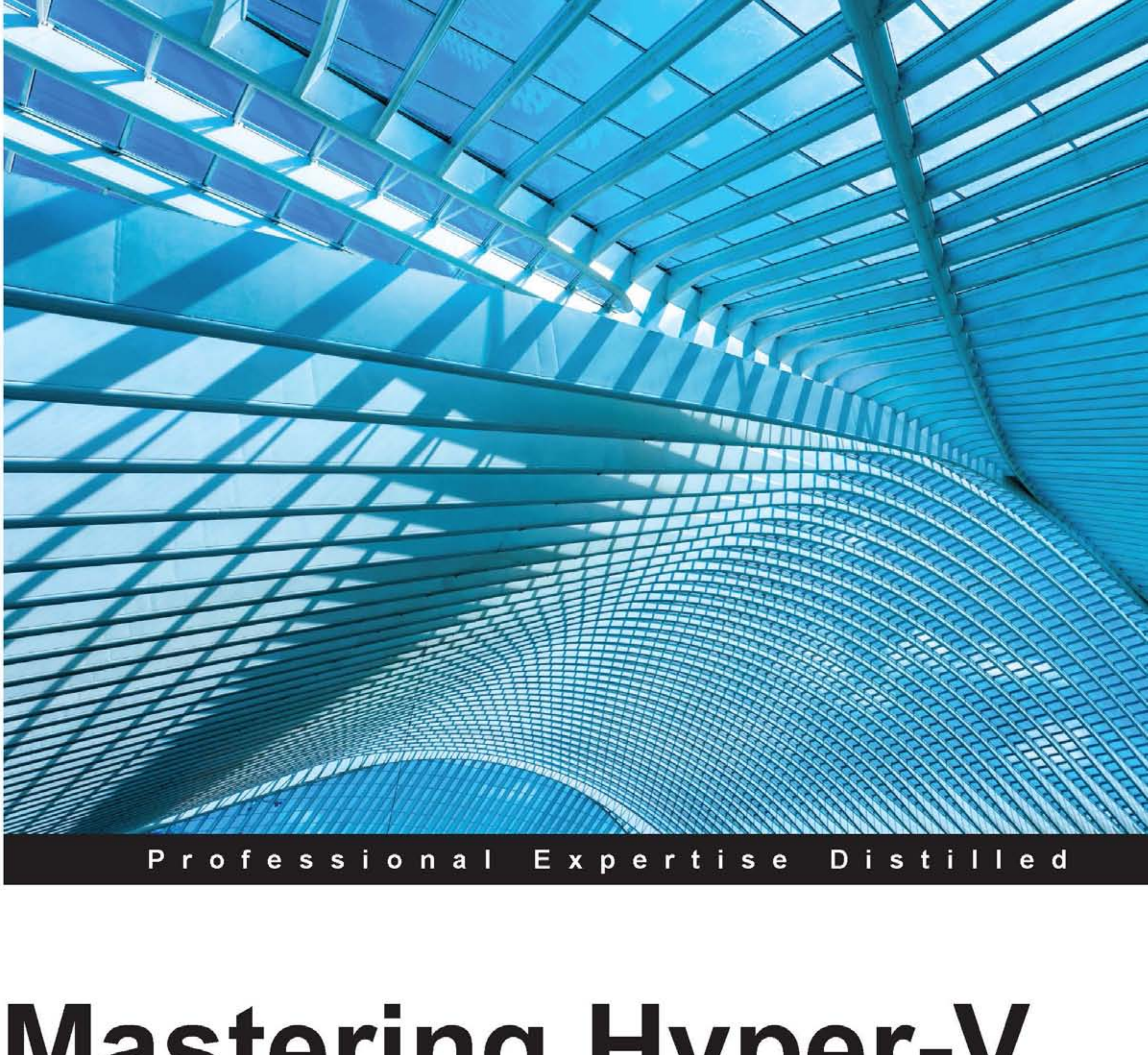


Professional Expertise Distilled

Mastering Hyper-V

Learn to design, build, and manage a virtualized data center using Microsoft Hyper-V

Peter De Tender

[PACKT] PUBLISHING enterprise professional expertise distilled

Mastering Hyper-V

Learn to design, build, and manage a virtualized data center using Microsoft Hyper-V

Peter De Tender

[PACKT] PUBLISHING enterprise professional expertise distilled

BIRMINGHAM - MUMBAI

Mastering Hyper-V

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book has been sold without warranty, either expressed or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: January 2015

Production reference: 1220115

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-78217-607-7

www.packtpub.com

Credits

Author
Peter De Tender

Reviewers
Fredrik Nilsson
Tom Webster
Matt Wharton

Acquisition Editor
Sonali Vernekar

Content Development Editor
Sumeet Sawant

Technical Editor
Taabish Khan

Copy Editors
Hiral Bhat
Neha Karnani
Veena Mukundan
Nithya P

Project Coordinator
Purav Motiwalla

Proofreaders
Ting Baker
Ameesha Green

Indexer
Priya Sane

Graphics
Sheetal Aute

Production Coordinator
Shantanu N. Zagade

Cover Work
Shantanu N. Zagade

About the Author

Peter De Tender has a strong focus on Microsoft Infrastructure technologies as an IT professional with over 16 years of experience. Having a past expertise in Exchange Server since version 4.0 back in 1995 all the way up to Exchange 2013, he has shifted to the Microsoft private and public cloud solutions in the last 18 months. He has worked on numerous design and implementation projects in Belgium and internationally for companies ranging from between 250 and 50,000 users. He has frequently worked on general Microsoft Core IO platform integration and consultancy projects as an infrastructure architect, mainly working with Windows Server, Hyper-V, and System Center Operations Manager.

In addition to being a consultant, Peter is also a recognized Microsoft Certified Trainer. Because of his expertise, he is both a country lead for Belgium and the European Chairman of IAMCT, the International Association of Microsoft Certified Trainers (`http://www.iamct.org`). He is also a Microsoft Springboard Series member.

For the past few years, Peter has been regularly traveling around the world to speak at international conferences on Microsoft technologies such as MCT Summits NA and EU, TechFuse Minneapolis, Community Day, and so on or to work as a staff member at Microsoft TechEds NA and EU, MMS, and so on.

Peter's passion for the IT communities is shown by his involvement in TheKrewe (`http://www.IamKrewe.org`), and he is the founder of TechEdYellowPantsTeam (`http://www.techedyellowpantsteam.com`). Mainly for his community contributions, he was rewarded with the MVP title in October 2013 for the first time.

As a technical writer for TrainSignal (`http://www.trainsignal.com`) and Petri Knowledgebase (`http://www.petri.co.il`), Peter writes many technical how-to articles on a multitude of Microsoft products, always with a twist from his own experience.

Besides this book, Peter was the lead author of *Exchange 2013 Cookbook*, *Packt Publishing*. He was also the co-author of *Upgrading Skills to Exchange Server 2013*, a courseware training guide published by MVP-Press.

Acknowledgments

After finishing my *Exchange 2013 Cookbook* for Packt Publishing in October 2013, I promised my wife and kids to never do it again—until Sonali contacted me asking if I was interested in writing a practical overview of Microsoft virtualization solution, with a main focus on Hyper-V Server 2012 R2, Virtual Machine Manager 2012 R2, and Windows Azure IaaS. As this book wasn't going to be a in the Cookbook format, there was a bit more flexibility on the writing style and chapter contents. As the topic was close to my heart, it didn't take much to convince me.

I would like to specially thank the Packt Publishing team for their support and patience along the line. Also, I thank the technical review team, as they gave marvelous feedback and forced me to update the content to improve the quality of the book.

In general, I would also like to thank all the people who have ever attended one of my trainings, workshop sessions, or public speaking gigs in all those different places in the world. It is your feedback and stories that gave me enough inspiration for writing this book, pulling real-life examples out of the discussions and conversations we had.

Last but not least, a more than special thanks goes once more to my wonderful family for accepting I'm always busy on my PC, helping customers, being away to speak at conferences, or writing another book. My dear wife, Els, I thank you so much for supporting me in all that I do. Without your continuous help, all that I'm busy with would just not be possible at all. You are an amazing woman and the best mom our girls can have. Kaylee and Kitana, our wonderful teenage daughters, I see so much energy and dedication in everything the two of you are doing, and it's just fun watching.

I would also like to thank my mom and sister who have supported me the last few months. A lot happened in our lives and I could always count on you sharing my frustrations about the things going on, and my book-writing experiences. Dad, although you have been gone for about 2 years now, you are in my memories forever. It is with pride and respect for you that I'm making the best of my life as you always taught me.

About the Reviewers

Fredrik Nilsson is a highly qualified and innovative Principal Consultant with a broad view of the different aspects of service delivery.

Fredrik has an extensive background in the outsourcing industry from within corporations such as HP, EDS, and IBM. He has worked in different roles and functions within the service delivery organizations as a capability architect and SME for Windows, Citrix, and VMware delivery units.

Fredrik's focus in combination with his skills and delivery focus makes him ever so successful in building effective solutions on Microsoft technologies.

Tom Webster is an IT and security professional who has been in the business of helping businesses achieve their goals for over a decade. His main occupation is leading the charge at Jaunty Hat LLC, the tech consulting and solutions company he co-founded. He's constantly writing and developing new projects and open source software at `http://SamuraiLink3.com`.

> I'd like to thank the love of my life, dark roast coffee, for helping me finish this project.

Matt Wharton is a systems engineer who specializes in designing and building server virtualization infrastructure solutions with many years of experience in both Hyper-V and VMware. Educated at the University of Bath, he holds a degree in electronic and communication engineering and is MCSE: Server Infrastructure and CCNA certified. He blogs regularly at `http://mattwharton.co.uk` and can be found on Twitter via `@electricinca`.

> I thank my wife, Amy, and my daughter, Lyra, for their support and encouragement.

Table of Contents

Preface

Hyper-V, Microsoft's virtualization technology, has matured into an enterprise-ready platform. Unfortunately, if you are new to virtualization or have an extensive background in VMware, for example, often you find yourself stranded in a pool of unfamiliar terms and concepts or even get lost in the combination of Hyper-V with System Center, without even talking about the questions that might arise around adopting cloud technology. The good news is that these questions are common, both in Server Message Blocks (SMBs) and in enterprise environments. By writing this book, I hope you as a reader can gather together information on some of the most common stumbling blocks to help you chart a clear path to a successful deployment, whether you are using Hyper-V in a standalone or in a clustered setup, integrating with System Center or not, or optimizing toward a private or a hybrid cloud architecture.

At the time of writing, Windows Server 2012 R2 has been released for about 6 months, and the adoption in the market is huge. Besides all the other interesting features of this latest version of Microsoft's server operating system, one of the key reasons to actually migrate is the updated virtualization component, Hyper-V.

When I'm talking to customers myself about virtualization and more specifically the Microsoft solutions that enable virtualization, I always tell them to forget everything they know and have heard about Hyper-V before. Not that the previous version was bad, actually not at all, but its feature set is so much different than the initial version in Windows Server 2008 close to 6 years ago. At that time, the market adoption of Hyper-V was rather low. Other virtualization platforms owned the market.

But not anymore…

And this is exactly the key reason why we have worked out this book, to take you on an interesting journey, where you will learn everything there is to know about Server 2012 R2 Hyper-V.

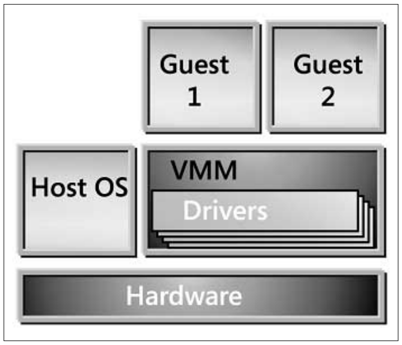
The book starts from the planning phase, during which we highlight some interesting tools that can help in assessing your current physical and/or virtual environment, whether it is running Microsoft or Linux operating systems, regardless of what hypervisor you are already using in your data center. This is followed by explaining what the new Hyper-V features of Server 2012 R2 are. A very important aspect of adopting Hyper-V is its management layer, made possible by System Center Virtual Machine Manager 2012 R2, as well as by using other System Center components such as Server App Controller to name just one.

The second part of this book will guide you in the realization of a "private cloud", starting from describing what private cloud actually means, what tools you need for running such a private cloud, how to perform full management of this on premises with a mixed virtualized infrastructure, ending with a chapter dedicated to Microsoft Azure, which allows you to not only run virtual machines "in-the-cloud", but also actually provide migration mechanisms back and forth.

Okay, now you know what you will learn from reading this book. And we know your expectations are high. You are a technical expert in operating and administering your current environment, and you want to dive into the technology right away. But before we do this, let's make sure everybody is on the right track here. That's why we would like to start with explaining some of the basics around server virtualization and what it takes to run a virtualized infrastructure. And if you are already familiar with these terminologies, feel free to jump right away to *Chapter 1*, *Introduction to Microsoft Server 2012 R2 Hyper-V*.

What is virtualization

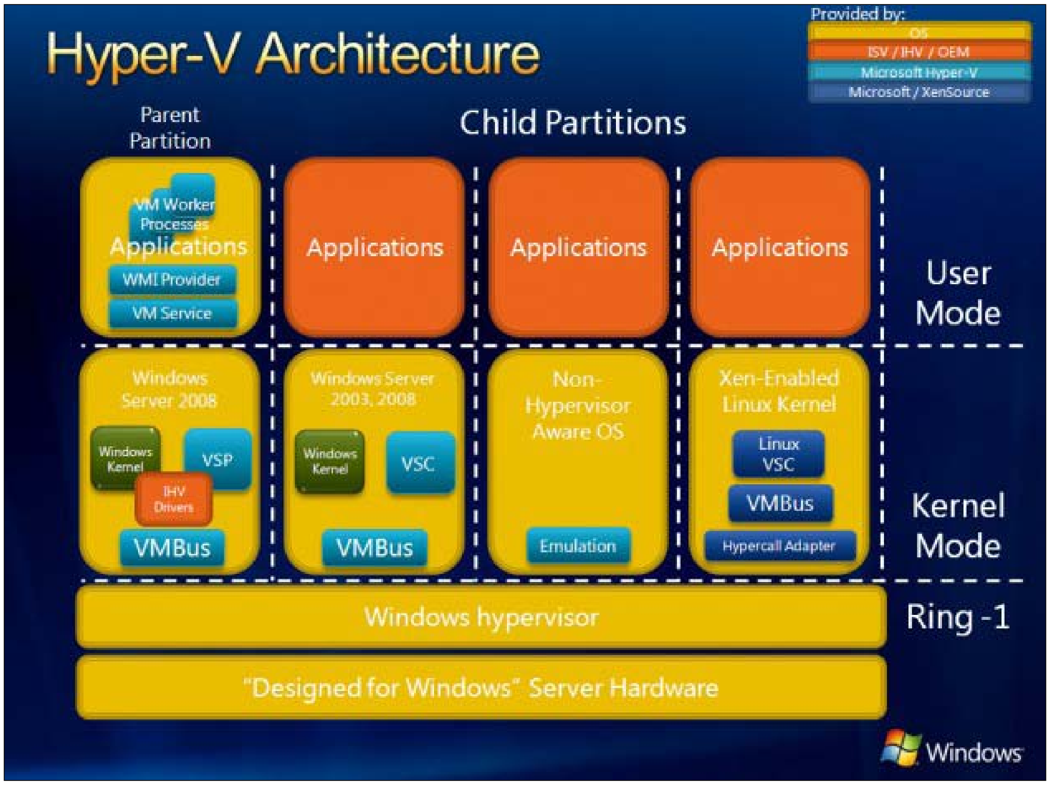
When talking about virtualization in computing, it refers to the act of virtualizing (instead of using the actual) computing resources such as memory, disk space, and CPU power. The virtualization architecture is based on a physical resource, for example, a server, known as the host, which shares its physical resources with multiple virtual instances, known as guests. This is shown in the following diagram:



While it was not really described as virtualization, the earliest computing resources that fit our preceding description were mainframes in the 1960s. These enormous physical machines were running multiple instances of multiple applications, all in their own bubble, on the same hardware. One application wasn't aware of what the other application was doing, what resources it was using, and so on.

Bringing this description back to the present in relation to Hyper-V, it could be understood as follows:

Imagine you have a physical machine having 64 GB of RAM, 2 TB of local disk space, and four CPUs. Instead of installing and running a single operating system on the physical layer of this machine, virtualization allows you to share these physical resources with multiple virtual machines. One machine could be a virtual Windows Server 2012 using 8 GB of RAM, 500 GB of disk space, and one CPU. Next to this, another virtual Windows Server 2008 could be installed using 16 GB of RAM, 1 TB of disk space, and two CPUs. Lastly, a third virtual machine is created running a Linux OS consuming 16 GB of RAM, 300 GB of disk space, and another CPU.



Benefits of using virtualization

As should be clear from the preceding description, the main benefit of using virtualization is sharing machine resources. Think of any application you are running in our organization, where it is running on a dedicated physical machine. In most cases, the physical resources of this machine will be under-utilized. But that server hardware is consuming rack space, power/electricity, requires cooling, maintenance, and so on.

So, consolidating multiple workloads from different physical servers into virtual servers, running on less physical machines, is already a cost benefit for your organization.

Or, imagine you are running outdated physical machines, where the hardware maintenance support cost is high. By the migration of this physical instance to a virtual instance (known as P2V—physical-to-virtual), your organization can again save money. Or maybe you are required to run legacy operating systems, which don't always run anymore on the newest hardware. Maybe using virtualization is another benefit in this specific scenario.

To continue describing some of the benefits, we can refer to our own examples being IT consultants and trainers. How often do we need to "try out" new things, such as new application versions, service pack upgrades, and just perform testing? How costly and inefficient would it be to perform these on physical machines? Even on a recent laptop having Windows 8 and enough resources available, it is possible to install the Hyper-V client and run virtual machines. This is actually what I'm doing on a day-to-day basis. Today, I will run a demo VM for a presentation on Exchange 2013; tomorrow, I will show a System Center Virtual Machine Manager to another customer. Without using virtualization, this would be a very hard nut to crack.

> While we briefly mentioned that the Windows 8/8.1 client operating system also supports running a Hyper-V client, there are certain differences in features between this version and the server version of Hyper-V. For a good overview of what is supported in a Hyper-V client, have a look at the Microsoft Knowledgebase article at `http://technet.microsoft.com/en-us/library/hh857623.aspx`.

While we could easily extend this list of benefits with another five pages of real-life examples, we are pretty sure you understand what we are trying to say. Just keep in mind not all workloads can be virtualized (think of specific interfaces being required to operate certain machines, which are not available in a virtual infrastructure).

Different kinds of virtualization

When talking about virtualization in the previous paragraphs, we mainly focused on server virtualization. But that description is far from complete nowadays. At present, we can determine the following "kinds of virtualization":

- Server virtualization
- Storage virtualization
- Network virtualization
- Desktop virtualization
- Application virtualization

Let us briefly explain what each kind of virtualization refers to.

Server virtualization

Server virtualization is what we described previously, having a physical server that runs on virtualization software (called the hypervisor), allowing you to deploy and run multiple virtual machines on top of this. Each virtual machine is configured with virtual machine resources such as memory, disk space, and CPU. These resources are allocated to each individual virtual machine and cannot be shared with other virtual machines. Other virtual machines rely on the shared physical resources, however.

Storage virtualization

Storage virtualization refers to the abstraction of the complexity and possibilities within a storage solution. A storage system is also known as a storage array or a disk array, which is actually a combination of physical hard drives that are split in different disk sets and that can be allocated to servers and/or applications.

Within the hypervisor as such, storage virtualization could also refer to the virtual hard disks that are linked to a virtual machine. Hyper-V supports the VHD and VHDX disk format, whereas VMware uses the VMDK format for its virtual disks.

Network virtualization

Where server virtualization allows you to consolidate and share the physical resources of server hardware between multiple virtual machines running on top of it, network virtualization provides about the same capability. On the same physical network infrastructure, you are running on multiple virtual network interfaces, where each virtual network interface operates as if it is the only network running.

We will dive more into the concepts and possibilities of Hyper-V specific network virtualization later on in *Chapter 2*, *Unwrapping Hyper-V 2012 R2 Components*.

Desktop virtualization

Desktop virtualization is very similar to server virtualization, except it is client operating system oriented. Think of running virtual Windows 7/8/8.1 clients on top of your hypervisor. Within desktop virtualization, you can detect three different flavors:

- **Session virtualization**: This refers to the end user connecting to the virtual desktop infrastructure (VDI) backend by using a session (remote desktop services or Citrix, for example).
- **Personal VM or personal VDI**: In this scenario, the end user is always connected to the same virtual desktop client. From a hypervisor perspective, the virtual machine is using its own virtual disk.
- **Pooled VM or pooled VDI**: In this setup, the end user connects to an individual virtual desktop, which is part of a pool of similarly and identically configured VMs. When starting up the session, it connects to pooled VM 123, whereas the next time it might occur that the user is connecting to pooled VM 345, for example. From a hypervisor perspective, the virtual machine is actually created out of a so-called base virtual disk and a differentiating disk, which owns the specific characteristics for a given session.

Although the VDI can be configured on top of a Hyper-V-based platform, we decided to not talk about this feature in this book, as it is a rather specific subject.

Application virtualization

This last virtualization possibility talks specifically about running an application in an isolated virtualized way, irrelevant of the virtual operating system or the physical layer it runs on. This mechanism is often described as "running the application in a bubble". Instead of installing the application bits and bytes directly within the operating system (could be a client or a server OS), it is isolated.

The advantages of using application virtualization are easier upgrades, solving compatibility issues between operating system and/or similar application dependencies to name just a few, and so on.

From Microsoft, there are two possible solutions available here: one is called App-V, which is part of the MDOP add-on on a client level. The second one is called Server App-V, which is managed through a System Center Virtual Machine Manager, a Server Application Sequencer, and an App-V Server agent.

> For more details on Microsoft Desktop Optimization Pack (MDOP) and more specifically, about its App-V client component, have a look at the Microsoft resources at `http://www.microsoft.com/en-us/windows/enterprise/products-and-technologies/mdop/app-v.aspx`.
>
> For more details on Server App-V, we would like to refer to the Microsoft TechNet library documentation at `http://technet.microsoft.com/en-us/library/hh397409.aspx`.

We hope with this little introduction and positioning of virtualization from one side, and slightly lifting the blanket on Microsoft Server 2012 R2 Hyper-V, we've inspired you to dive into the subject right away.

Happy reading!

What this book covers

Chapter 1, Introduction to Microsoft Server 2012 R2 Hyper-V, helps the reader in finding out if their existing environment is "ready" for Hyper-V; we will position a Microsoft tool here that can help in doing the assessment (MAP Toolkit).

Chapter 2, Unwrapping Hyper-V 2012 R2 Components, handles the different components of a Hyper-V virtual infrastructure, such as virtual machine, virtual disks, virtual network configurations, and alike.

Chapter 3, Live Migration, Storage Migration, and Hyper-V Replica, talks about virtual machine migration, including what mechanisms are available from within Hyper-V itself and the difference between live migration and quick migration. The last topic will detail the Hyper-V Replica feature.

Chapter 4, Building a High Available Hyper-V Cluster, describes how to implement a high available cluster, the different components you need, and how to execute failover testing.

Chapter 5, Hyper-V Management Using System Center Virtual Machine Manager, will not only position a System Center VMM as a Hyper-V management and administration solution, but also explain the benefits of using it compared to the standalone Hyper-V manager.

Chapter 6, Integrating System Center VMM 2012 R2 with Your VMware Environment, details how the integration works, how to configure the interaction between SCVMM and VMware vCenter, and explains the possibilities of the interaction between both platforms.

Chapter 7, Operating Your Private Cloud Using SCVMM and App Controller, tells you that once your private cloud is operational from a technical perspective, it's time to start looking at optimizing your service delivery; maybe providing self-service functionalities to your users. By using System Center App Controller, we will show you how this can become possible in your environment. We will cover how to install it, how to configure it, and what could be a possible roadmap for integrating it in your environment.

Chapter 8, The Road to a Public Cloud Data Center Infrastructure Using Microsoft Azure, starts with the features of Microsoft Azure, related to running a virtualized infrastructure, and describes how this platform can be integrated in your existing on-premise private cloud strategy.

Chapter 9, Hyper-V and System Center Licensing, talks about the different aspects of Hyper-V licensing and System Center licensing, as well as guiding you in making a decision, based on some real-life scenarios.

What you need for this book

We'll need the following software for our book:

- Windows Server 2012 R2 Standard or Data center edition
- Microsoft System Center 2012 R2
- Microsoft Azure subscription
- VMware ESX 5.1
- VMware vCenter Management tools

Who this book is for

This book is mainly targeted at the common network and system administrator who has to deal with server virtualization in their day-to-day job, primarily using Microsoft Hyper-V and System Center. It assumes you have some practical experience with the previous versions of Hyper-V, although this is not a requirement, without being a subject matter expert.

Conventions

In this book, you will find a number of text styles that distinguish between different kinds of information. Here are some examples of these styles and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: "The parameter `-Restart` forces a reboot if required."

Any command-line input or output is written as follows:

```
Bcdedit /enum
```

New terms and **important words** are shown in bold. Words that you see on the screen, for example, in menus or dialog boxes, appear in the text like this: "On the **Select server roles** page, select **Hyper-V**."

> Warnings or important notes appear in a box like this.

> Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or disliked. Reader feedback is important for us as it helps us develop titles that you will really get the most out of.

To send us general feedback, simply e-mail `feedback@packtpub.com`, and mention the book's title in the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide at `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt Publishing book, we have a number of things to help you to get the most from your purchase.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you could report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting `http://www.packtpub.com/submit-errata`, selecting your book, clicking on the **Errata Submission Form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded to our website or added to any list of the existing errata under the Errata section of that title.

To view the previously submitted errata, go to `https://www.packtpub.com/books/content/support` and enter the name of the book in the search field. The required information will appear under the **Errata** section.

Piracy

Piracy of copyrighted material on the Internet is an ongoing problem across all media. At Packt Publishing, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works in any form on the Internet, please provide us with the location address or the website name immediately so that we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors and our ability to bring you valuable content.

Questions

If you have a problem with any aspect of this book, you can contact us at `questions@packtpub.com`, and we will do our best to address the problem.

1
Introduction to Microsoft Server 2012 R2 Hyper-V

Microsoft offers different technologies that can help organizations in building a virtual infrastructure platform. We briefly described some of those technologies in the *Preface*. In this chapter, we'll continue on the same path, but detailing a bit more what Server 2012 R2 features allow for virtualization, followed by the relationship between a virtual infrastructure and System Center Suite R2. Before we dive into the deep technical aspects of such Hyper-V and System Center platforms, we will guide you on how to do a technical assessment of an existing environment in the context of a Microsoft virtualized platform implementation or migration.

Once all that is clear, we will jump into the Hyper-V building blocks and the Hyper-V server role installation process and provide you with a quick overview of the Hyper-V management console.

Overview of Microsoft virtualization solutions

While we have only touched on the surface of different kinds of virtualization in the *Preface*, and mentioned already some of the Microsoft virtualization solutions there to make certain things more clear, we want to make sure you get a good understanding of what is possible with Hyper-V, running on Windows Server 2012 R2.

We will start off with Server Hyper-V 2012 R2, moving over to mentioning some System Center 2012 R2 components, and how they are of importance in a virtualized environment.

Windows Server Hyper-V 2012 R2

The primary solution we will talk about during the remainder of this book is Hyper-V 2012 R2. This is Microsoft's server hypervisor, running as a server role of Windows Server 2012 R2. Just like some of the other well-known hypervisor solutions such as VMware ESX, Citrix XenServer and RedHat Enterprise Virtualization, Hyper-V is also a **type 1 hypervisor**.

A type 1 hypervisor refers to a hypervisor technology that runs directly on the bare metal components of the physical server. We are mentioning this here about Hyper-V because it is not always easy to understand. After all, you rely on Windows Server 2012 R2, where you install the Hyper-V Server Role "on top of the OS".

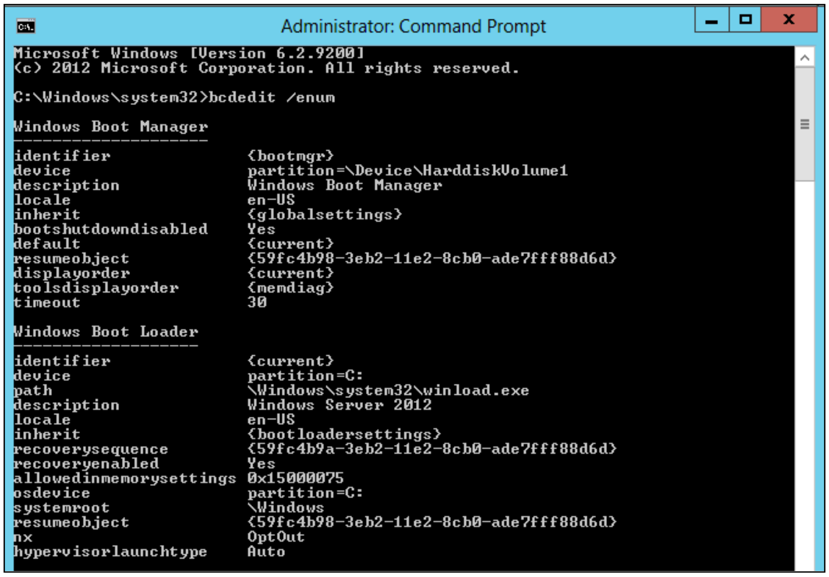
This is where most people get confused, and with a good reason.

During the installation of the Hyper-V server role, changes occur to the boot configuration of the physical installation, which actually puts the Windows Boot Loader mode's `hypervisorlaunchType` setting to `Auto`. This means it automatically loads from the hypervisor. And thus, your physical Windows OS actually runs in a virtualized mode.

To check the Windows Boot Loader details yourself, run the following instruction from a command prompt:

```
Bcdedit /enum
```

You can see that the parameter `hypervisorlaunchtype` is set to `Auto` as shown in the following screenshot from our test environment:



```
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.

C:\Windows\system32>bcdedit /enum

Windows Boot Manager
--------------------
identifier              {bootmgr}
device                  partition=\Device\HarddiskVolume1
description             Windows Boot Manager
locale                  en-US
inherit                 {globalsettings}
bootshutdowndisabled    Yes
default                 {current}
resumeobject            {59fc4b98-3eb2-11e2-8cb0-ade7fff88d6d}
displayorder            {current}
toolsdisplayorder       {memdiag}
timeout                 30

Windows Boot Loader
-------------------
identifier              {current}
device                  partition=C:
path                    \Windows\system32\winload.exe
description             Windows Server 2012
locale                  en-US
inherit                 {bootloadersettings}
recoverysequence        {59fc4b9a-3eb2-11e2-8cb0-ade7fff88d6d}
recoveryenabled         Yes
allowedinmemorysettings 0x15000075
osdevice                partition=C:
systemroot              \Windows
resumeobject            {59fc4b98-3eb2-11e2-8cb0-ade7fff88d6d}
nx                      OptOut
hypervisorlaunchtype    Auto
```

As already mentioned before, a lot has changed in the Hyper-V component in Server 2012 R2 compared to its predecessor in Windows Server 2008. While there are again specific updates and new features added in Server 2012 R2, most of them are actually new in Windows Server 2012:

- **Multiple VM live migration**: Live migration already existed in Hyper-V 2008, but was limited to migrating a single VM between hosts. As of Server 2012, it allows for multiple VM migrations at once.
- **Virtual machine storage migration**: In Hyper-V 2008 R2, moving a running instance of a VM using live migration is supported, but it is not possible to move the VM's storage while the machine is running. However, with Hyper-V on Server 2012 and R2, it is now possible to move the VM storage, while the virtual machine is running.
- **Using SMB 3.0 file shares as virtual machine file locations**: SMB 3.0 can be configured on Windows Server 2012 file servers, or on supported SAN or NAS solutions. By using SMB 3.0, your storage configuration becomes a lot easier. If the VMs are stored within an SMB 3.0 file share, one can also make use of SMB live migration, which allows you to migrate VMs between different servers, even if they are not part of the same Hyper-V cluster.
- **Shared Nothing Live Migration**: This type of VM migration was also born in Server 2012 timeframe. It allows you to migrate a VM between any host in the data center, as well as to migrate the virtual storage between storage solutions in the data center. This is done without any downtime for the virtual machine itself.
- **Hyper-V Replica**: In this scenario, an exact copy/clone of a running VM is being replicated to another host; this host doesn't need to be part of a cluster. In the case of downtime of the production VM, a failover can be initiated to the replica server, which will have an up-to-date state of the original VM.

While this list is already impressive, the following are the specific new updates and features within Hyper-V on Server 2012 R2:

- **Generation 2 VMs**: A VM is built using virtualized emulated hardware such as CPU, NIC, BIOS, and so on. A Gen 2 VM is not using this emulated hardware anymore; several new features that are specific to Generation 2 virtual machines are PXE boot, boot from SCSI VHD, UEFI firmware support, and secure boot.
- **Dynamically resize VHDX**: In Hyper-V on Server 2012 R2, it is now possible to dynamically resize (extend and shrink) the disk space size of a running VM.

- **Compressed live migration**: While nothing has changed on the live migration feature itself, the data to be migrated is compressed. This results in faster migrations and less data packages going through the wire.
- **Hyper-V Replica time interval**: This interval time can now be set as low as 30 seconds, meaning that VM updates on a host will be synchronized to the replica host every 30 seconds.
- **Better support for Linux VM**: Server 2012 has already supported running certain Linux VMs; the list of these supported Linux versions and distributions has been extended.
- **Automatic virtual machine activation**: If the physical host is running Server 2012 R2 data center and has been activated, the same activation process will be applied to any running Windows Server 2012 R2 VM on that host. No need of KMS for these machines anymore.

> For a full overview of all new features within Hyper-V on Server 2012 R2, have a look at the Microsoft Knowledgebase article at `http://technet.microsoft.com/en-us/library/dn282278.aspx`.

Of course, discussing all details about all the preceding mentioned features, including a lot more, is the main goal of this book.

Introduction to Microsoft System Center 2012 R2 in relation to Hyper-V

Whenever you are planning to build a large virtualization environment on Hyper-V, you should also consider implementing (parts) of Microsoft System Center 2012 R2. While it is perfectly possible to manage and administer your Hyper-V platform from within the Hyper-V management console or by using PowerShell, at some point in time you might lack features that allow for thorough IT service management, automation, detailed monitoring and alike. And that's where System Center comes into play. System Center is actually a combination of different products, one can run totally independent from any other, but licensed as a bundle, as of version 2012. Before that, a customer had the choice to license one or more System Center components on an individual basis. If they were only using Operations Manager, they only needed to acquire the licenses for Operations Manager. As of System Center version 2012, the customer acquires a management license for the full suite, where one can decide which components to install.

Let us give you a short description of the key usage scenarios and possibilities of each System Center component.

System Center Virtual Machine Manager – SCVMM

This component allows for full virtual platform management. Where most larger environments will use this solution for primarily managing Hyper-V hosts, storage, networks and VMs, you can also use it to administer and manage your VMware environment as well as Citrix XenServer environment. However, to manage your VMware environment out of System Center Virtual Machine Manager, you still need to have a running VMware vCenter available, as well as having the relevant license for it. Key functionality here is overall, centralized management of your full virtual infrastructure, whether it is virtual machines, virtual storage, virtual networks, or any other required configuration.

Chapter 5, *Hyper-V Management Using System Center Virtual Machine Manager*, is dedicated to all details around Virtual Machine Manager 2012 R2, including the following:

- How to install it
- How to use it to deploy or administer your virtual storage environment
- How to configure the virtual network fabric
- How to create, remove, and configure virtual machines and much more.

System Center App Controller

Where Virtual Machine Manager is the management portal of your overall virtual infrastructure, App Controller allows for self-service to give an answer to both private cloud and public cloud scenarios. So basically it is the IT admin department managing the technical resources of the virtual platform using VMM, providing self-service features to the end users from within a web portal, which is App Controller. (Actually, VMM initially also had a self-service portal, but it got removed from the SP1 version of VMM 2012.)

Unlike other System Center components, which need to be installed on a separate server, App Controller can run on the same server as VMM, as well as on a single or multiple dedicated server(s).

Chapter 7, Operating Your Private Cloud by Using SCVMM and App Controller, will be your main resource throughout this book, which discusses the installation and integration of App Controller in your environment, as well as how to plan and prepare the configuration that needs to be done on your virtual environment prior to implementing App Controller.

System Center Operations Manager – SCOM

SCOM is the overall IT infrastructure management and monitoring solution. SCOM collects detailed information about any possible asset in your IT environment mainly by the power of management packs. This can be Microsoft servers, Microsoft server applications, Linux-based servers, network switches, firewalls, server hardware, server applications such as Oracle and SAP, Windows Server components, and so on.

Besides monitoring and alerting you about the health of your platform, SCOM helps keep track of solutions for given alerts, both company-specific and public information, by using its knowledgebase feature set. SCOM can be extended with dashboards and detailed reporting.

> Head over to the Microsoft TechNet Library article at `http://go.microsoft.com/fwlink/p/?LinkID=231676` for all details about SCOM.

System Center Configuration Manager – SCCM

SCCM or Config Manager is used for client and server management and deployment, has integrations with WSUS, and allows for deploying applications to these clients as well. It can also be extended with Microsoft Endpoint Protection, a very powerful antivirus/antimalware solution.

The intelligence of the product lies in the logic of collections one can create on a custom basis. A PC or any other client device that can be managed through SCCM gets applications and Windows updates installed, will report its inventory to a centralized CMDB, and much more.

Version 2007, which was mainly used for managing client environments, got extended intensively in version 2012 to also allow server OS deployment. Version 2012 R2 allows a strong integration (both technical and license-wise) with the Microsoft Intune cloud product, thus providing a full management platform solution for mobile device management.

> The Microsoft TechNet Library article providing information about SCCM is reachable at `http://go.microsoft.com/fwlink/p/?LinkID=231675`.

System Center Service Manager – SCSM

Service Manager is the part of System Center that allows an organization to streamline its service management best practices through automation. The built-in logic and processes are based on those found in the **Microsoft Operations Framework** (**MOF**) and ITIL, which are incident management, change management, problem management, and asset life cycle management.

The true power lies in the open framework, which allows you to customize forms, tasks, and processes up to a very high extent. The other true power comes from integration with the other System Center suite components, of course. Think of a possible situation as follows:

1. Your application server is generating several alerts (being reported in SCOM); out of these alerts, an automatic Service Manager incident problem ticket gets created.
2. This incident will be handled by an IT admin person. As the incident requires more investigation, the incident problem gets transferred into a change request ticket, which gets picked up by an external IT partner.
3. After investigating and solving the root cause of the issue, the problem ticket gets closed as resolved.
4. On its turn, the linked incident ticket gets closed as well; all alerts in SCOM will be marked as resolved too.

By using scripts, System Center Orchestrator, or any other mechanism (depending on the application), this imaginary workflow can be partly, almost, or completely automated.

> All details about SCSM on Microsoft TechNet can be found at `http://go.microsoft.com/fwlink/p/?LinkID=231678`.

System Center Data Protection Manager – SCDPM

Data Protection Manager is the backup solution out of System Center. It allows for disk-based or tape-based backups of Windows Server, Exchange Server, SharePoint servers, SQL servers, virtual machines (both running Windows OS and supported Linux flavors), as well as desktop or laptop clients.

By using snapshot technology in combination with the VSS engine of Windows Server, it allows for backups to be taken every few minutes. Restoration is similar to backups, a no-brainer. Detailed reporting is provided by default.

> Read about SCDPM 2012 R2 at `http://go.microsoft.com/fwlink/p/?LinkId=239439`.

System Center Orchestrator

Orchestrator is the workflow automation engine of System Center, allowing you to automate any task you can think of in your data center, both on-premises and in the cloud. It can not only interact with other System Center products, but also with any product you can think of. By default, Orchestrator has around 40 preconfigured workflow activities, but if these are not enough for your environment, you can extend the functionality and feature set of Orchestrator by using the so-called integration packs. Besides Microsoft technologies integration packs for Active Directory, Exchange, or SharePoint, there are a lot of others that allow for integration with third-party components such as HP iLo, HP Operations Manager, and VMware vSphere integration packs to name just a few.

> A full list of the currently available Orchestrator integration packs is available at `http://technet.microsoft.com/en-us/library/hh295851.aspx`.

The intelligence and possibilities of the automation engine is configured by using the so-called runbooks. It's the runbook (or set of runbooks) that contains the different tasks or processes that provide automation. As you can understand from the previous paragraph, these runbooks don't stop in the Microsoft environment, but allow for interaction with a lot of other IT components in your data center.

The previous sections should have made clear how powerful System Center and its different components have become in the latest release of the product (suite). Though it is not required to use System Center in your environment, we can say from our experience that it offers a lot of benefits in an enterprise environment.

Besides highlighting System Center, we briefly introduced you to the key features of Windows Server Hyper-V 2012 R2, which will be handled in a lot more detail in the next chapters.

Keeping both technologies in mind, it's about time we start helping you in thinking about your future virtualized environment. What about physical server and storage sizing, what about licensing, how to migrate your existing virtualized workloads—running already on Microsoft Hyper-V or other platforms—to Hyper-V 2012 R2. The keyword here is planning and assessing your existing platform. And that's exactly what we will discuss in the next topic.

Executing a technical assessment of your as-is environment by using the MAP Toolkit

Knowing the technical details of a hypervisor solution such as Microsoft Hyper-V 2012 R2 is just one part of the complete story. In our own experience as consultants, we have seen a lot of customers that are using virtualization, and everything seems to run fine. But when looking a bit closer into the environment, we find that there is a lot of room for optimization. This can be the case in the server and/or storage sizing, which can be both over- and undersized; with some other enterprise customers, we see still that a lot of manual interactions are needed to deploy virtual machines, or IT admins do not know at all what different workloads there are actually running in their environment.

A lot of this information can get answered by a simple, free but very useful tool, the MAP Toolkit or Microsoft Assessment and Planning Toolkit in full. This tool helps organizations in preparing a migration from existing servers and clients' infrastructure to new versions. Without talking about "how" to do the migration, this tool helps in making a preparation assessment. It mainly answers questions such as:

- Is my server hardware capable of hosting a virtualization platform?
- Which of my client devices need to be upgraded with more RAM and more CPU power in order to run Windows 7 / 8.1?

- Which PCs should be replaced?
- Which server loads can be migrated to a virtualized platform?
- What software am I running on my clients, and will it work with Windows 7 / 8.1?

So after all these years, we are still surprised at times by the fact that it is a rather unknown tool. However, we are also happy to see how interesting the results are for our customers.

Customer satisfaction is the best reason why we have mentioned this tool here as part of this chapter. Without describing all the details of the tool, we will focus on the installation part and configuration steps required to get a detailed assessment report, which will help you or your customer in planning the future virtualized environment.

Installing the tool

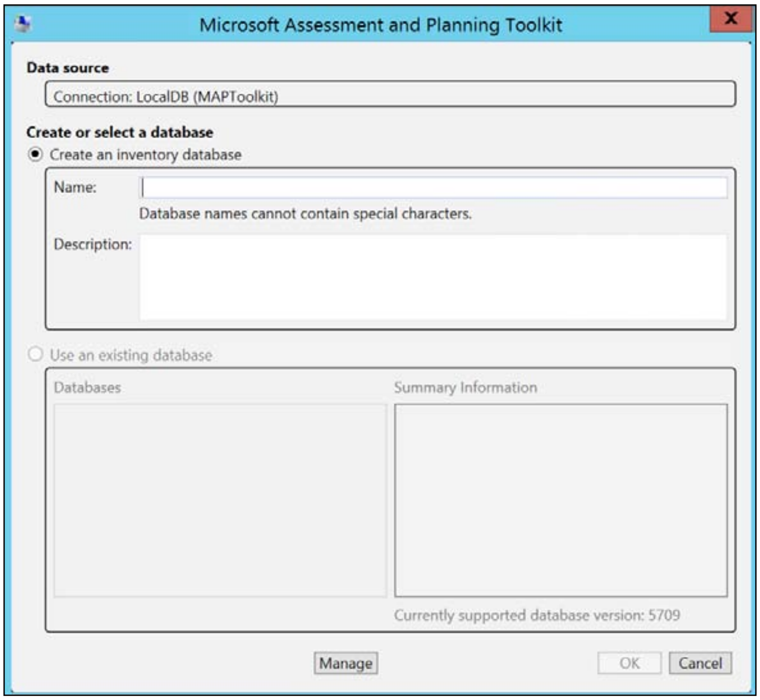
It all starts with downloading the installation bits from the Microsoft website. The latest version 9.0 dates from Feb 10 2014, so is rather new, and can be downloaded from `http://www.microsoft.com/en-us/download/details.aspx?id=7826`.

While I don't expect any difficulties in the installation itself, here are some guidelines and prerequisites that can be of assistance:

- Don't install this tool on a Domain Controller, it's not supported and won't work.
- You don't need a powerful machine to run this tool on; I often install this on the management station of the IT team, or on a lightweight server such as File & Print or any other application server you have available.
- The tool can be installed on any OS version Windows 7 SP1 or higher, Windows 2008 R2 or higher and even Windows Server 2012 R2 if you have that already in your environment.
- Make sure you have the latest version of .Net Framework 4.5 installed (download can be found at `http://go.microsoft.com/fwlink/?LinkId=389161`). Mind you—this installation requires a reboot, so make sure you pick any non-business critical machine to install this tool on.
- By default, the MAP Toolkit will install a local version of SQL Server 2012 Express edition; this is mainly my default scenario. However, you could integrate this with an already present SQL Server installation having version 2008 or higher.

Running the tool – creating your inventory

Once MAP Toolkit is installed, you can launch it from your Start screen or Start menu by going to **Microsoft Assessment and Planning Toolkit**. Your default screen should look like this, in which we will create the SQL database:



Microsoft Assessment and Planning Toolkit

Data source

Connection: LocalDB (MAPToolkit)

Create or select a database

● Create an inventory database

Name:

Database names cannot contain special characters.

Description:

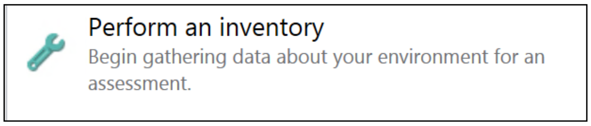
○ Use an existing database

Databases

Summary Information

Currently supported database version: 5709

Once your database has been created in the background, we can start the inventory; obviously, you need to click on **Perform an inventory** from the main page, which launches the Inventory and Assessment wizard:



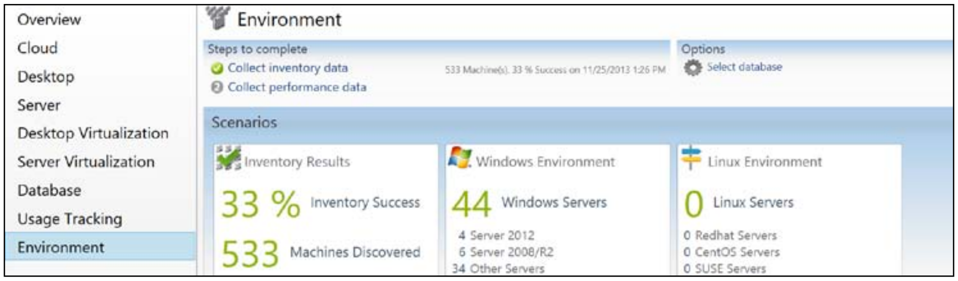
Perform an inventory

Begin gathering data about your environment for an assessment.

Depending on what systems you want to integrate in your inventory, completing this step might be easy or a bit more difficult. It could be difficult in a sense that you have to know your administrative user accounts of the different systems (such as VMware, Linux, Oracle Database admin account, and so on). Some technologies also have certain plugin dependencies (for example, the Oracle database client).

Completing this wizard shouldn't be much of a problem. So, I allowed myself to skip writing down the different steps from this wizard.

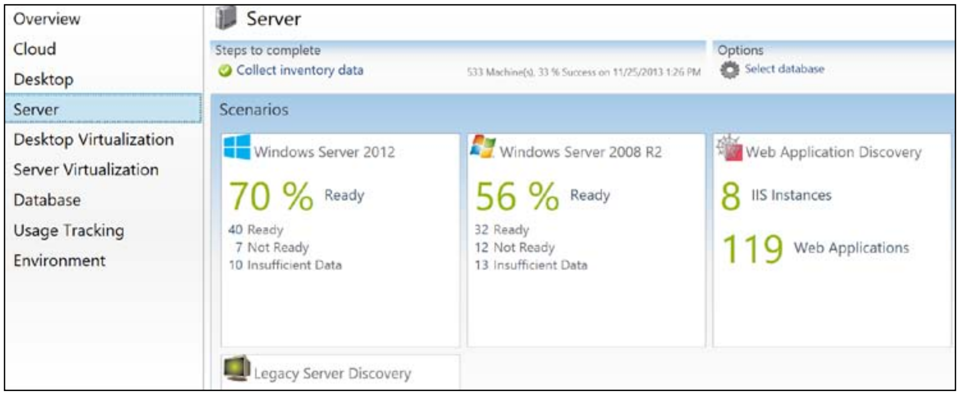
So I'll basically continue with the assumption that your inventory ran fine, and you should now see a console with data in it, looking similar to my screenshot:



Now, depending on your environment analysis and assessment scenario, you may want to go through the process of migration (for example, client migration, server migration, cloud migration, and so on). You have different options and results you can use from within the assessment tool. In our example, we will focus on the server migration and cloud migration scenarios.

Server assessment

Server inventory features within the MAP Toolkit are actually straightforward. By selecting the **Server** section on the left-hand side of the screen, you'll come across a graphical overview of your current assessed platform. You will see which servers are ready to run Windows Server 2012 (and 2012 R2) or 2008 R2, what web server instances and web applications you are running, and what your legacy server platform (Windows 2003 and 2000) looks like.



By selecting one of the scenarios, you are transferred to a more detailed view, which will have the options available to generate a Windows Server report and proposal and/or customize assessment properties.

Cloud assessment

When selecting the cloud assessment section, we are presented with the following scenarios:

- **Azure VM readiness**: This is an overview of machines that can be migrated to Azure.
- **Azure VM capacity**: It gathers performance data to guide you in Azure capacity requirements.
- **Office 365 readiness**: Is your current environment ready for Office365? (For example, you need to check if your Exchange 2010 on-premise instance can be migrated to Exchange Online.)
- **Private Cloud Fast Track**: This is explained more in detail in the next section.

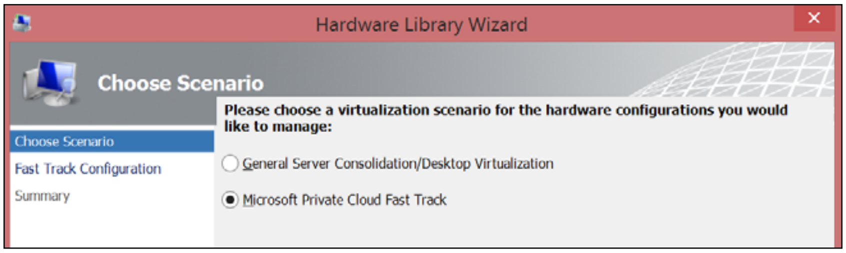
And again, for each scenario, you can get more detailed dashboards and export the data into reports in both Word and Excel file format.

Private Cloud Fast Track

One of the more popular scenarios for why we are using the MAP Toolkit a lot with our customers is because of the built-in Private Cloud Fast Track. This refers to a predefined set of hardware specifications, which have been tested by both Microsoft and well-known server and storage partners such as HP, EMC, NetApp, Dell and many others, which could be used as a guideline for your own in-house infrastructure.

By using this Fast Track, organizations can benefit from the flexibility of solutions from Microsoft partners offering core capabilities of the Windows Server OS, Hyper-V, and System Center, which are the true building blocks of a Microsoft Private Cloud.

From within the MAP Toolkit menu, select **Tools** | **Configure Hardware Library**, which opens the following wizard:

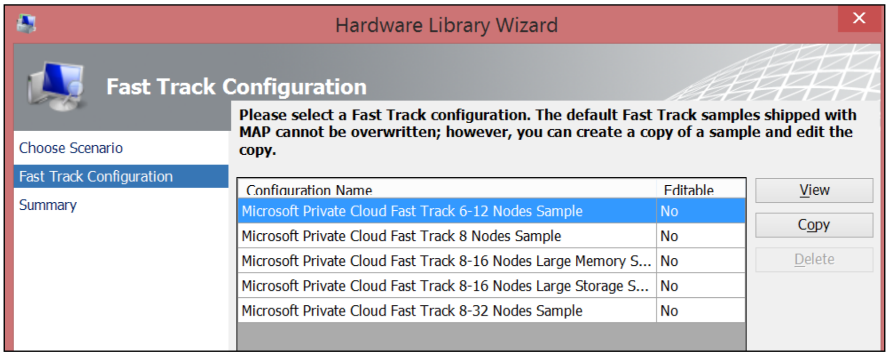


Please choose a virtualization scenario for the hardware configurations you would like to manage:

○ General Server Consolidation/Desktop Virtualization

● Microsoft Private Cloud Fast Track

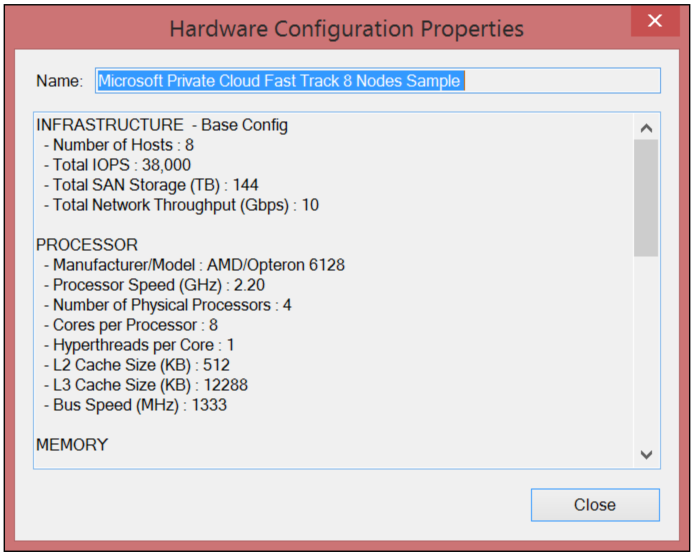
Select the **Microsoft Private Cloud Fast Track** scenario here. This brings us to the **Fast Track Configuration** step.



Please select a Fast Track configuration. The default Fast Track samples shipped with MAP cannot be overwritten; however, you can create a copy of a sample and edit the copy.

Configuration Name	Editable
Microsoft Private Cloud Fast Track 6-12 Nodes Sample	No
Microsoft Private Cloud Fast Track 8 Nodes Sample	No
Microsoft Private Cloud Fast Track 8-16 Nodes Large Memory S...	No
Microsoft Private Cloud Fast Track 8-16 Nodes Large Storage S...	No
Microsoft Private Cloud Fast Track 8-32 Nodes Sample	No

By default, we are offered five sample architectures, of which we can see the detailed specifications by clicking on the **View** button; let me take the example of the Fast Track 8 nodes sample, which is a common scenario for a lot of my customers:



Name: Microsoft Private Cloud Fast Track 8 Nodes Sample

```
INFRASTRUCTURE  - Base Config
 - Number of Hosts : 8
 - Total IOPS : 38,000
 - Total SAN Storage (TB) : 144
 - Total Network Throughput (Gbps) : 10

PROCESSOR
 - Manufacturer/Model : AMD/Opteron 6128
 - Processor Speed (GHz) : 2.20
 - Number of Physical Processors : 4
 - Cores per Processor : 8
 - Hyperthreads per Core : 1
 - L2 Cache Size (KB) : 512
 - L3 Cache Size (KB) : 12288
 - Bus Speed (MHz) : 1333

MEMORY
```

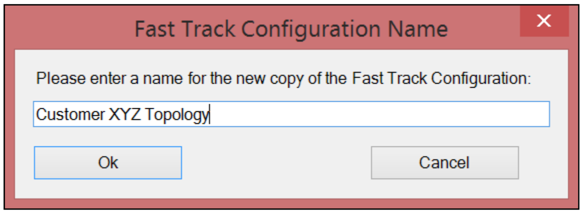
As you can see from this sample, this infrastructure offers the following specifications:

Number of nodes	8
Total IOPS	38,000
Total SAN storage	144 TB to 420 TB
Network throughput	10 GB

CPU	2.2 GHz / 4 physical CPUs per node / 8 cores per CPU
Memory	125GB to 500GB

Imagine the prebuild configurations are not valid for your specific future environment. No worries! You can copy one of the configs and create your own customized versions:

1. Select any scenario that matches your specific config the most; click on **Copy**.
2. Give your scenario a descriptive name.

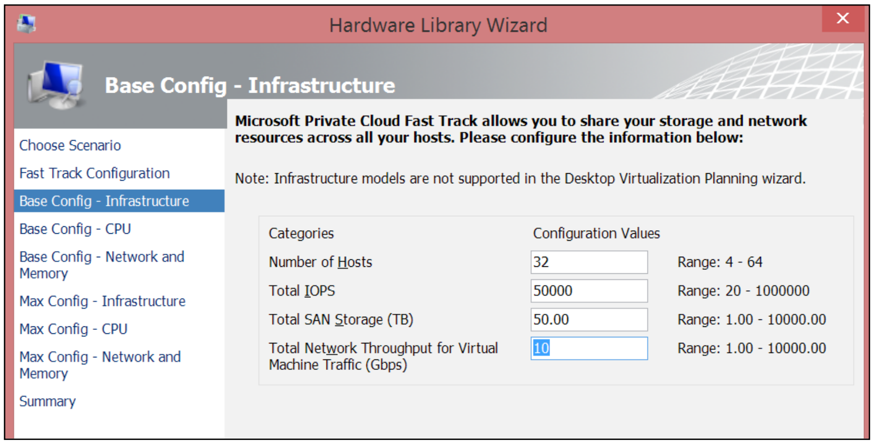


Fast Track Configuration Name

Please enter a name for the new copy of the Fast Track Configuration:

Customer XYZ Topology

3. In the following steps of the wizard, you can customize any settings of the specifications to meet your needs:



Hardware Library Wizard

Base Config - Infrastructure

Microsoft Private Cloud Fast Track allows you to share your storage and network resources across all your hosts. Please configure the information below:

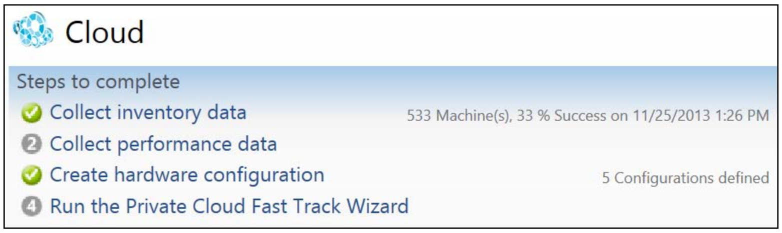
Note: Infrastructure models are not supported in the Desktop Virtualization Planning wizard.

Categories	Configuration Values	
Number of Hosts	32	Range: 4 - 64
Total IOPS	50000	Range: 20 - 1000000
Total SAN Storage (TB)	50.00	Range: 1.00 - 10000.00
Total Network Throughput for Virtual Machine Traffic (Gbps)	10	Range: 1.00 - 10000.00

Choose Scenario
Fast Track Configuration
Base Config - Infrastructure
Base Config - CPU
Base Config - Network and Memory
Max Config - Infrastructure
Max Config - CPU
Max Config - Network and Memory
Summary

You can modify any numbers you want by updating the values. These will be saved to your specific Fast Track architecture.

4. After the numbers have been updated, we go back into our **Cloud** scenario, and select **4 Run the Private Cloud Fast Track Wizard**.



Cloud

Steps to complete

- ✓ Collect inventory data — 533 Machine(s), 33 % Success on 11/25/2013 1:26 PM
- 2 Collect performance data
- ✓ Create hardware configuration — 5 Configurations defined
- 4 Run the Private Cloud Fast Track Wizard

Once these steps are completed, we can again have a look at the resulting report, that is the Microsoft Private Cloud Fast Track Consolidation Report named `<date & time>.xlsx` from within the `C:\Program Files\Microsoft Assessment and Planning Toolkit\Reports` folder.

What you get out of this is again a very detailed report in Excel format, consisting of different tab sheets, containing (not limited to) the following information:

Worksheet	Description
`Summary`	A summarized overview of your assessed Windows Server environment.
`Utilization Setting`	This sheet contains data about the maximum utilization that the selected Microsoft Private Cloud Track can provide.
`ConsolidationOnBasicConfig`	Recommended host / VM scaling of your current infrastructure, when migrating everything to the base set of the Microsoft Private Cloud Fast Track configuration parameters.
`ConsolidationOnMaxConfig`	Recommended host / VM scaling of your current infrastructure, when migrating everything to the maximized set of the Microsoft Private Cloud Fast Track configuration parameters.
`UnplacedMachinesReport`	Gives a list of those currently existing machines that cannot be migrated into the Private Cloud Fast Track topology, because of technical limitations, constraints in OS, or not having the supported configuration for allowing a P2V migration to Hyper-V.

Worksheet	Description
`UtilizationBeforeConsolidation`	This report provides details of the current utilization of machines in your network based on performance details collected earlier. The values indicate the utilization before each machine is virtualized for consolidation.

This concludes the usage of the Microsoft Assessment and Planning Toolkit.

Hyper-V building blocks for creating your Microsoft virtualization platform

After having gone through the previously described MAP Toolkit, we now have a good overview and understanding of our current physical and/or virtual environment. The gathered information allows us to clearly define a detailed list of required server hardware, storage hardware, physical and virtual machine operating systems, and anything else we need to be able to build our future virtualization platform.

These components are known as the Hyper-V building blocks, and we describe each one of them in the following sections.

Physical server hardware

One of the first important components when building a virtualization platform is the physical server hardware.

One of the key elements to check is the Microsoft certified hardware and software supportability and compatibility list. This list gives a detailed overview of all tested and certified server brands, server types, and their corresponding configuration components. While it is not a requirement to use this kind of machine, we can only recommend it, based on our own experience. Imagine you have a performance issue with one of your applications running inside a VM, being hosted on non-supported hardware, using non-supported physical NICs, and you're not getting decent support from your IT partner or Microsoft on that specific platform, as the hardware is not supported.

> The landing page for this compatibility list is `http://www.windowsservercatalog.com`.

After checking the compatibility of the server hardware and software, you need to find out which system resources are available for Hyper-V. The following table shows the maximum scaling possibilities for different components of the Hyper-V platform (the original source is Microsoft TechNet Library article at `http://technet.microsoft.com/en-us/library/jj680093.aspx`.)

System	**Resource**	**Maximum number**	
		Windows 2008 R2	**Windows Server 2012 (R2)**
Host	Logical processors on hardware	64	320
	Physical memory	1 TB	4 TB
	Virtual processors per host	512	1,024
Virtual machine	Virtual processors per virtual machine	4	64
	Memory per virtual machine	64 GB	1 TB
	Active virtual machines	384	1,024
	Virtual disk size	2 TB	64 TB
Cluster	Nodes	16	64
	Virtual machines	1,000	4,000

Physical storage hardware

Next to the physical server component, another vital part of the virtualization environment is the storage hardware. In the Hyper-V platform, multiple kinds of storage are supported, that is DAS, NAS, and/or SAN:

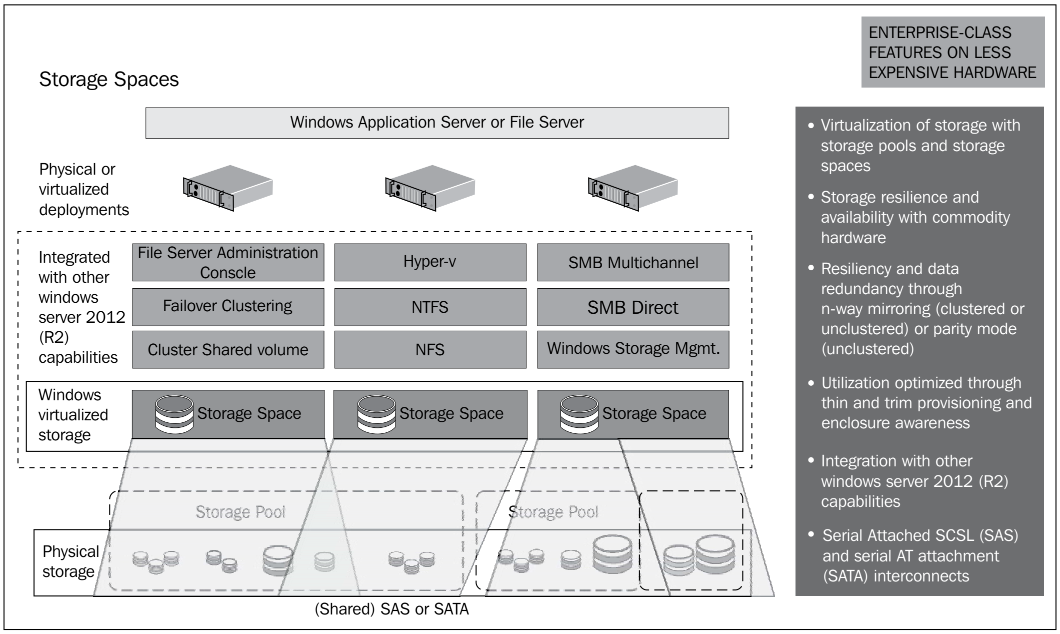
- **Direct Attached Storage** (**DAS**): This is directly connected to the server (think of disk which is located inside the server chassis).
- **Network Attached Storage** (**NAS**): This is the storage provided via the network and presented to the Hyper-V server or virtual machines as file shares. This disk type is file-based access. Server 2012 and 2012 R2 make use of SMB 3.0 as file-sharing protocol, which allows us to use plain file shares as virtual machine storage location.
- **Storage Area Network** (**SAN**): This is also network-based storage, but relies on block-based access. The volumes are presented as local disks to the host. Popular protocols within SAN environments are iSCSI and Fibre Channel.

The key point of consideration when sizing your disk infrastructure is providing enough storage, at the best performance available, and preferably high availability as well. Depending on the virtual machine's required resources, the disk subsystem can be based on high-performant / expensive SSD disks (solid-state drives), performant / medium-priced SAS disks (serial attached SCSI), or slower but cheaper SATA (serial ATA) disks. Or it could even be a combination of all these types.

Although a bit outside of Hyper-V as such, one technology that is configured and used a lot in combination with Hyper-V Server 2012 R2, is Storage Spaces. Storage Spaces is new as of Server 2012, and can be considered as a storage virtualization subsystem. Storage Spaces are disk volumes built on top of physical storage pools, which is in fact **just a bunch of physical disks** (**JBOD**).

> A very important point to note is that the aforementioned network-based SAN and NAS storage solutions cannot be a part of Storage Spaces, as it is only configurable for DAS storage.

The following schema diagram provides a good overview of the Storage Spaces topology, possibilities, and features:


Storage Spaces
ENTERPRISE-CLASS FEATURES ON LESS EXPENSIVE HARDWARE
Windows Application Server or File Server
Physical or virtualized deployments
Integrated with other windows server 2012 (R2) capabilities
File Server Administration Console
Hyper-v
SMB Multichannel
Failover Clustering
NTFS
SMB Direct
Cluster Shared volume
NFS
Windows Storage Mgmt.
Windows virtualized storage
Storage Space
Storage Space
Storage Space
Storage Pool
Storage Pool
Physical storage
(Shared) SAS or SATA
• Virtualization of storage with storage pools and storage spaces
• Storage resilience and availability with commodity hardware
• Resiliency and data redundancy through n-way mirroring (clustered or unclustered) or parity mode (unclustered)
• Utilization optimized through thin and trim provisioning and enclosure awareness
• Integration with other windows server 2012 (R2) capabilities
• Serial Attached SCSL (SAS) and serial AT attachment (SATA) interconnects

Physical network devices

It's easy to understand that your virtual platform is dependent on your physical network devices such as physical (core) switches and physical NICs in the Hyper-V hosts. When configuring Hyper-V, there are a few configurations to keep into consideration.

NIC Teaming

NIC Teaming is the configuration of multiple physical network interface cards into a single team, mainly used for high availability or higher bandwidth purposes. NIC Teaming as such is no technology of Hyper-V, but Hyper-V can make good use of this operating system feature. When configuring a NIC team, the physical network cards are bundled and presented to the host OS as one or more virtual network adapter(s). Within Hyper-V, two basic sets of algorithms exist where you can choose from during the configuration of Hyper-V networking:

- **Switch-independent mode**: In this configuration, the teaming is configured regardless of the switches to which the host is connected. The main advantage in this configuration is the fact the teaming can be configured to use multiple switches (for example, two NICs in the host are connected to switch 1 and 2 NICs are configured to use switch 2).
- **Switch-dependent mode**: In this configuration, the underlying switch is part of the teaming configuration; this automatically requires all NICs in the team to be connected to the same switch.

> NIC Teaming is managed through the Server Manager / NIC Teaming interface or by using PowerShell cmdlets. Depending on your server hardware and brand, the vendor might provide you with specific configuration software to achieve the same. For example, the HP Proliant series of servers allows for HP Team configuration, which is managed by using a specific HP Team tool.

Network virtualization

While we will talk more in detail about network virtualization in *Chapter 2*, *Unwrapping Hyper-V 2012 R2 Components* (configuring networks in Hyper-V), and *Chapter 5*, *Hyper-V Management Using System Center Virtual Machine Manager* (configuring the network fabric in Virtual Machine Manager), we want to mention it only briefly here. Within Hyper-V 2012 R2, network virtualization not only refers to the virtual networking connections that are used by the virtual machines but also refers to the technology that allows for true network isolation to the different networks in which virtual machines operate. This feature set is very important for hosting providers, who run different virtual machines for their customers in an isolated network. You have to make sure that there is no connection possible between the virtual machines from customer A and the virtual machines from customer B. That's exactly the main purpose of network virtualization.

Another possible way of configuring network segmentation is by using VLANs. However, this also requires VLAN configuration to be done on the physical switches, where the described network virtualization completely runs inside the virtual network switch of Hyper-V.

Server editions and licensing

The last component that comprises the Hyper-V building blocks is the server editions and licensing of the physical and virtual machines operating system.

As we do have a chapter dedicated to this (*Chapter 9*, *Hyper-V and System Center Licensing*), we would like to refer you to this chapter for all details around this subject.

Walk-through of the Hyper-V Server role installation process

This section describes how to actually install the Hyper-V Server role on Windows Server 2012 R2, by using both GUI and PowerShell as a way to get the component installed.

We want to inform you about the fact that the installation process is identical on all versions of Windows Server 2012 R2, whether it is being installed on the Windows 2012 R2 Standard edition, on the Windows 2012 R2 Datacenter edition, or on the free Windows Hyper-V Server 2012 R2.

> If you want to know more about functionality and licensing differences between the Windows Server 2012 R2 editions, we would like to refer you to *Chapter 9*, *Hyper-V and System Center Licensing*, which talks about this topic more in-depth.

Installing the Hyper-V Server role using the GUI

Log on to your Windows Server 2012 R2 machine using administrative credentials, and perform the following steps:

1. In Server Manager, from the **Manage** menu or by selecting **Dashboard** on the left-hand side, click on **Add Roles and Features**.
2. On the **Before you begin** page, go through the verification steps. If your machine is compatible, click on **Next**.
3. On the **Select installation type** page, select **Role-based or feature-based installation** and then click on **Next**.
4. On the **Select destination server** page, select a server from the server pool (in most cases, this will be your local server; if you are connected using remote admin tools, you can select more than one server to install this server role on and then click on **Next** again).
5. On the **Select server roles** page, select **Hyper-V**.
6. To add the tools that you use to create and manage virtual machines, click on **Add features**. On the **Features** page, click on **Next**.
7. On the **Create Virtual Switches** page, **Virtual Machine Migration** page, and **Default Stores** page, select the appropriate options for your environment.

> If you use this host from within a Virtual Machine Manager environment, we suggest you skip the virtual switches configuration at this stage, and configure the virtual network fabric from within VMM itself later on to avoid conflicts in the configuration.

8. On the **Confirm installation selections** page, verify the **Reboot this server if required** option is flagged according to your organization's policies, and click on **Finish**.
9. Once the installation is finished, start the Hyper-V Manager console from the Start screen to verify it is up and running.

This completes the installation of the Hyper-V Server role on Windows Server 2012 R2, by using the GUI.

Installing the Hyper-V Server role using PowerShell

Log on to your Windows Server 2012 R2 machine using administrative credentials, and perform the following steps:

1. Start PowerShell from the taskbar or Start screen; make sure it is started in Administrator mode (if not by default, right click on the PowerShell icon in the taskbar or Start screen and select **Run as administrator**).
2. Type the following PowerShell cmdlet:

   ```
   Install-WindowsFeature –Name Hyper-V -Computername<your host's
   Windows computer name> -IncludeManagementTools –Restart
   ```

   This will install the Windows Hyper-V server role on the given host, as well as the Hyper-V Manager console. The parameter `–Restart` forces a reboot if required.

3. When the installation is finished, use the following PowerShell cmdlet to verify that the Hyper-V Server role is indeed installed:

   ```
   Get-WindowsFeature –computername<your host's Windows computer
   name>
   ```

4. Once the installation is finished, start the Hyper-V Manager console from the Start screen to verify it is up and running.

This completes the installation of the Hyper-V Server role on Windows Server 2012 R2, by using PowerShell.

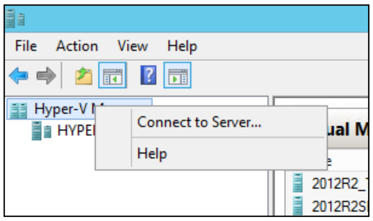
Rapid-eye view of the Hyper-V management console

As the Hyper-V Manager console is installed, let us finish this first chapter with a short walkthrough of this console.

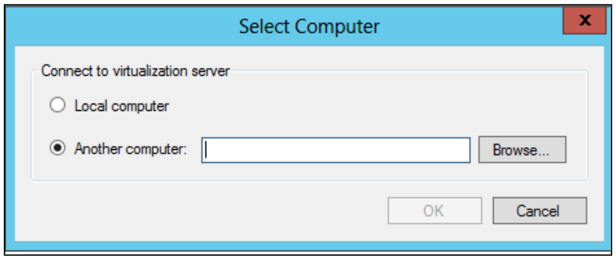
Connecting to a Hyper-V host

As mentioned in our introduction, the Hyper-V Manager is the recommended tool to manage only a handful of Hyper-V hosts, whereas the System Center Virtual Machine Manager is recommended in an enterprise environment. That said, nothing stops you from using the Hyper-V Manager to connect remotely to different Hyper-V hosts, all from the same console. This can be done as follows:

1. On the left-hand side of the console, right-click on **Hyper-V Manager** and select **Connect to Server...**.



2. Enter the Hyper-V host name or IP address, or click on **Browse** to get a list of Hyper-V hosts to which you can connect using your administrative credentials.


Select Computer

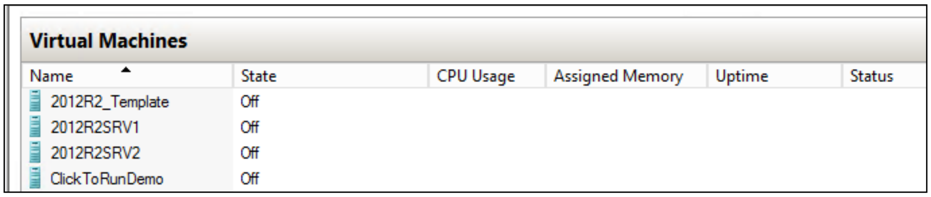
Connect to virtualization server

○ Local computer

● Another computer: |

Managing virtual machines

In the middle pane of the Hyper-V Manager, you will see a list of all the virtual machines that are configured on the selected Hyper-V host, and in what state (running, paused, or shutdown) they are. It also shows you some basic resource consumption such as CPU, memory and uptime.



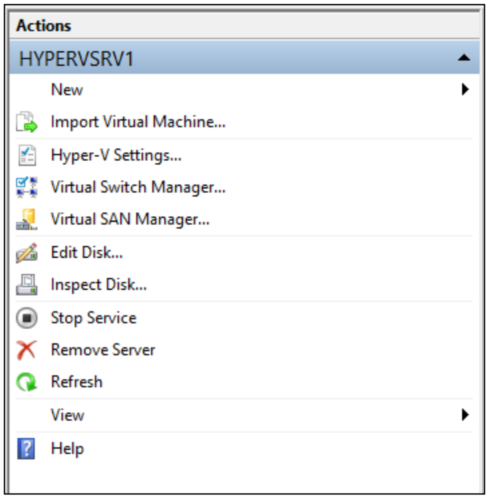
Virtual Machines

Name	State	CPU Usage	Assigned Memory	Uptime	Status
2012R2_Template	Off				
2012R2SRV1	Off				
2012R2SRV2	Off				
ClickToRunDemo	Off				

Right clicking on one or more of the virtual machines brings you to a context menu from which you can connect to a virtual machine, delete the virtual machine from the list, and update the virtual machine's settings and much more. We will talk about all these in the next chapter.

Managing your Hyper-V environment

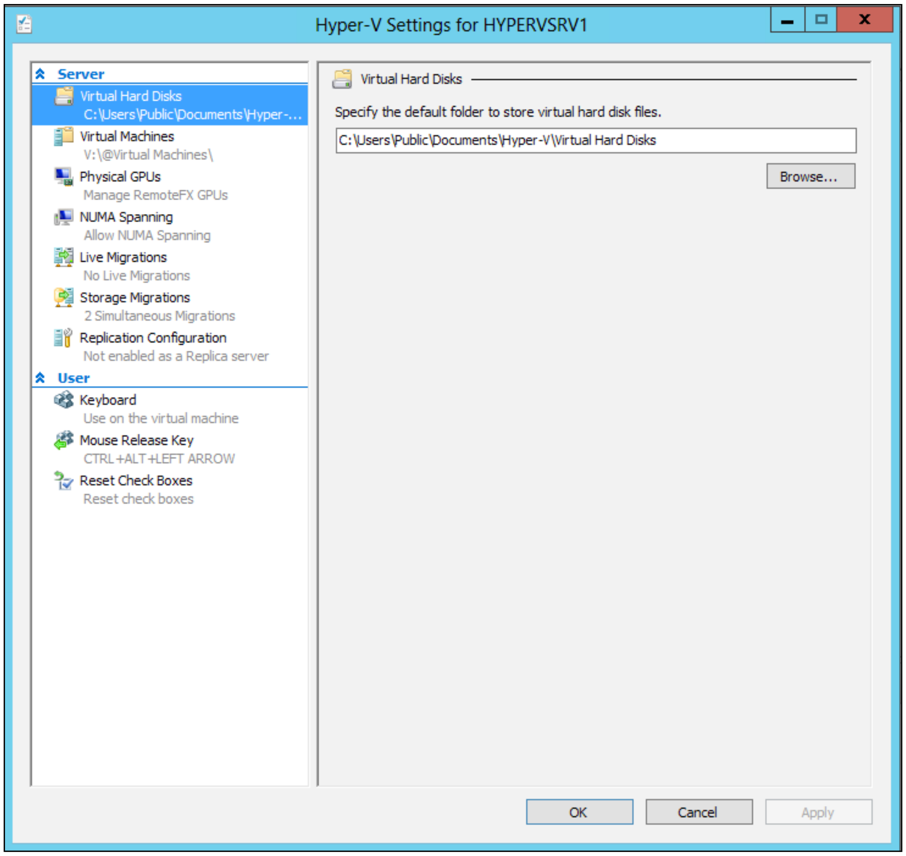
On the right side of the Hyper-V Manager, we find the **Actions** pane; from here, we can perform multiple actions on our Hyper-V environment, Hyper-V host, and virtual machines.



The import actions that you can launch from the Hyper-V platform level are as follows:

- Create a new virtual machine or virtual disk
- Import a virtual machine
- Modify general Hyper-V settings such as default folder paths for virtual machines and virtual hard disks
- Configure live migration and storage migration settings
- Enable or Disable the Hyper-V host as the replica server

This is shown in the following screenshot:



Hyper-V Settings for HYPERVSRV1

Server

Virtual Hard Disks
C:\Users\Public\Documents\Hyper-...

Virtual Machines
V:\@Virtual Machines\

Physical GPUs
Manage RemoteFX GPUs

NUMA Spanning
Allow NUMA Spanning

Live Migrations
No Live Migrations

Storage Migrations
2 Simultaneous Migrations

Replication Configuration
Not enabled as a Replica server

User

Keyboard
Use on the virtual machine

Mouse Release Key
CTRL+ALT+LEFT ARROW

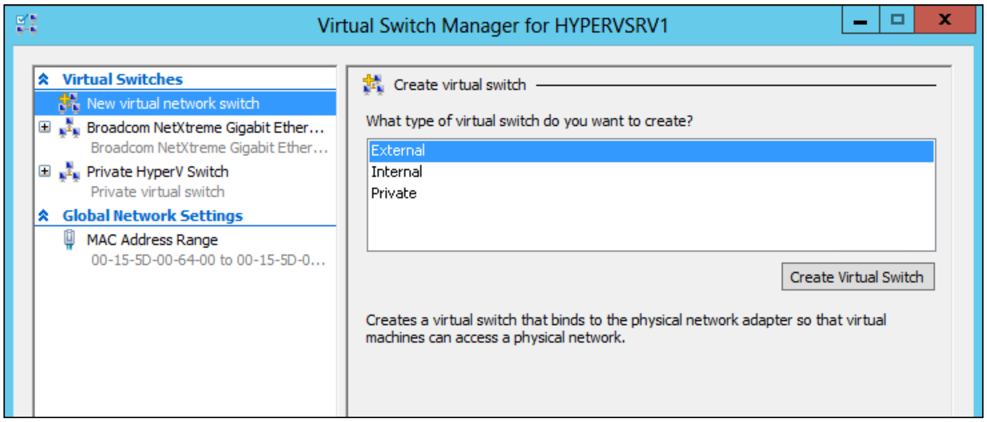
Reset Check Boxes
Reset check boxes

Virtual Hard Disks

Specify the default folder to store virtual hard disk files.

C:\Users\Public\Documents\Hyper-V\Virtual Hard Disks

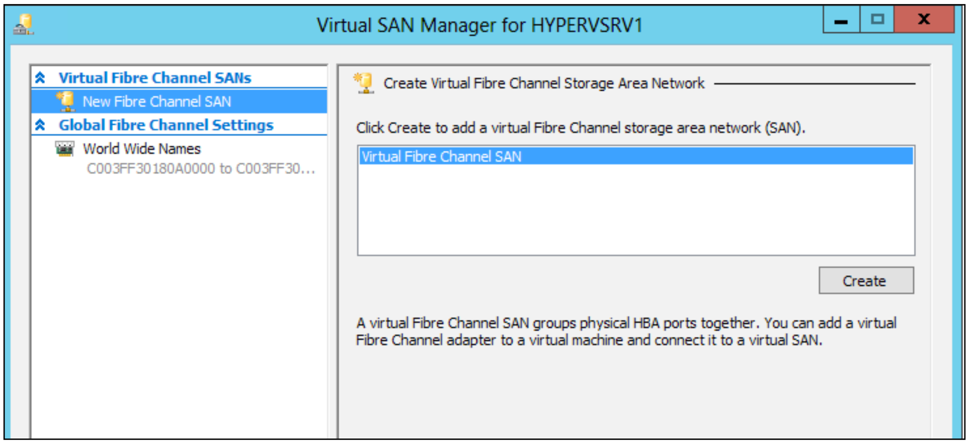
Next to these Hyper-V platform settings, one can also configure Hyper-V Virtual Switch Manager settings here:


Virtual Switch Manager for HYPERVSRV1
Virtual Switches
New virtual network switch
Broadcom NetXtreme Gigabit Ether...
Broadcom NetXtreme Gigabit Ether...
Private HyperV Switch
Private virtual switch
Global Network Settings
MAC Address Range
00-15-5D-00-64-00 to 00-15-5D-0...
Create virtual switch
External
Internal
Private
Create Virtual Switch

What type of virtual switch do you want to create?

Creates a virtual switch that binds to the physical network adapter so that virtual machines can access a physical network.

Lastly, you can configure the Hyper-V SAN Manager settings here:


Virtual SAN Manager for HYPERVSRV1
Virtual Fibre Channel SANs
New Fibre Channel SAN
Global Fibre Channel Settings
World Wide Names
C003FF30180A0000 to C003FF30...
Create Virtual Fibre Channel Storage Area Network
Virtual Fibre Channel SAN
Create

Click Create to add a virtual Fibre Channel storage area network (SAN).

A virtual Fibre Channel SAN groups physical HBA ports together. You can add a virtual Fibre Channel adapter to a virtual machine and connect it to a virtual SAN.

For more details on the actual configuration of the Hyper-V Virtual Switch and Virtual SAN Manager, refer to *Chapter 2, Unwrapping Hyper-V 2012 R2 Components*.

Summary

In this chapter, we started from an explanation of the Microsoft virtualization platform and learned what components are available for building such a platform. The Windows Server 2012 R2 Hyper-V server role and the System Center suite of products are the key technologies here.

After this introduction, we guided you through the MAP Toolkit, a free tool from Microsoft that allows you to plan and assess your current environment. The outcome of this tool is a series of detailed documents around sizing, compatibility, and platform resources.

In the last section of this chapter, we explained the different ways of installing the Hyper-V server role on a physical host, by using both GUI and PowerShell, on a local machine or by using remote admin tools. We closed the chapter with a quick walk-through of the Hyper-V Manager console.

2

Unwrapping Hyper-V 2012 R2 Components

If you are reading this book chapter by chapter, you should have a running Hyper-V server 2012 R2. If you already had a Hyper-V server available and therefore skipped whole or parts of the previous chapter, you're welcome here.

This chapter is completely focused on the Hyper-V 2012 R2 components, such as virtual networking, virtual storage, virtual machines, and more. As the Hyper-V layer forms the baseline for creating and configuring your virtual platform and private cloud later on, it is very important that you make yourself familiar with all aspects of the hypervisor.

The key components we will talk about in this chapter are as follows:

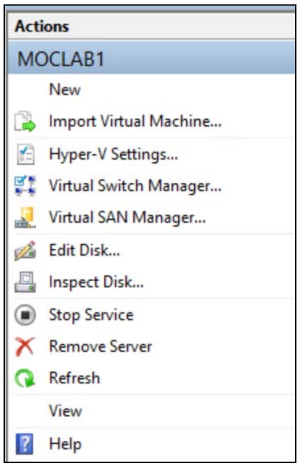
- Base Hyper-V settings
- Resource pools
- Hyper-V virtual storage configuration
- Hyper-V virtual network configuration
- Virtual machine creation and configuration settings

Next to these, we will also describe NUMA hardware, RemoteFX, and Enhanced Session Mode as part of the overall Hyper-V configuration settings.

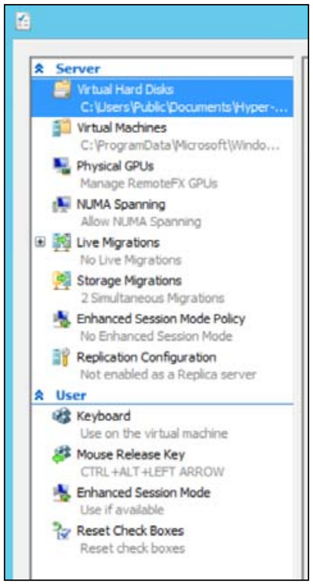
Base Hyper-V settings

After installing the Hyper-V Server role on your physical server, one of the first components that needs configuration is actually the "base Hyper-V settings". You can reach these in three different ways, as follows:

1. From within the **Actions** pane at the right-hand side of the Hyper-V manager console:



2. By right-clicking on your Hyper-V host's **Hyper-V Settings** option:



Server

Virtual Hard Disks
C:\Users\Public\Documents\Hyper-...

Virtual Machines
C:\ProgramData\Microsoft\Windo...

Physical GPUs
Manage RemoteFX GPUs

NUMA Spanning
Allow NUMA Spanning

Live Migrations
No Live Migrations

Storage Migrations
2 Simultaneous Migrations

Enhanced Session Mode Policy
No Enhanced Session Mode

Replication Configuration
Not enabled as a Replica server

User

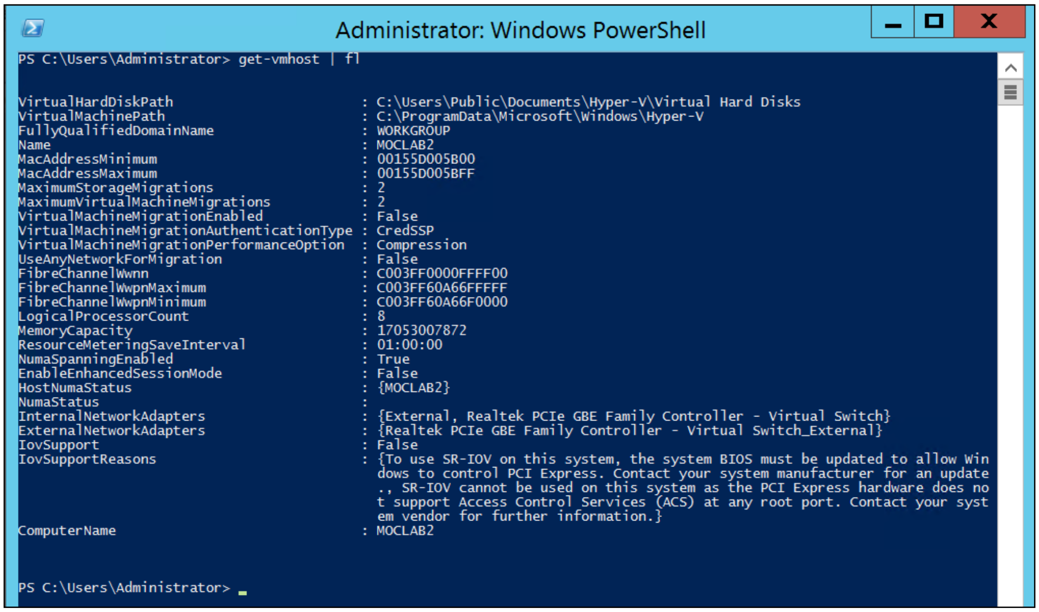
Keyboard
Use on the virtual machine

Mouse Release Key
CTRL+ALT+LEFT ARROW

Enhanced Session Mode
Use if available

Reset Check Boxes
Reset check boxes

3. By using PowerShell, more specifically the `Get-VMHost` and `Set-VMHost` cmdlets:



```
PS C:\Users\Administrator> get-vmhost | fl


VirtualHardDiskPath                       : C:\Users\Public\Documents\Hyper-V\Virtual Hard Disks
VirtualMachinePath                        : C:\ProgramData\Microsoft\Windows\Hyper-V
FullyQualifiedDomainName                  : WORKGROUP
Name                                      : MOCLAB2
MacAddressMinimum                         : 00155D005B00
MacAddressMaximum                         : 00155D005BFF
MaximumStorageMigrations                  : 2
MaximumVirtualMachineMigrations           : 2
VirtualMachineMigrationEnabled            : False
VirtualMachineMigrationAuthenticationType : CredSSP
VirtualMachineMigrationPerformanceOption  : Compression
UseAnyNetworkForMigration                 : False
FibreChannelWwnn                          : C003FF0000FFFF00
FibreChannelWwpnMaximum                   : C003FF60A66FFFFF
FibreChannelWwpnMinimum                   : C003FF60A66F0000
LogicalProcessorCount                     : 8
MemoryCapacity                            : 17053007872
ResourceMeteringSaveInterval              : 01:00:00
NumaSpanningEnabled                       : True
EnableEnhancedSessionMode                 : False
HostNumaStatus                            : {MOCLAB2}
NumaStatus                                :
InternalNetworkAdapters                   : {External, Realtek PCIe GBE Family Controller - Virtual Switch}
ExternalNetworkAdapters                   : {Realtek PCIe GBE Family Controller - Virtual Switch_External}
IovSupport                                : False
IovSupportReasons                         : {To use SR-IOV on this system, the system BIOS must be updated to allow Win
                                            dows to control PCI Express. Contact your system manufacturer for an update
                                            ., SR-IOV cannot be used on this system as the PCI Express hardware does no
                                            t support Access Control Services (ACS) at any root port. Contact your syst
                                            em vendor for further information.}
ComputerName                              : MOCLAB2



PS C:\Users\Administrator> _
```

Virtual hard disks' setting

The virtual hard disks setting defines the default location of the virtual hard disks that will be created on this local Hyper-V host. Aside from the default path, when deploying new VMs or creating new virtual hard disks, you always have the option to modify the default path to a specific one as per that step in the wizard.

From our own practical experience, we suggest modifying this public folder path reference on the Hyper-V host's system disk to another folder path on a nominated virtual hard disk storage location. (This could be local data partition on the Hyper-V host, a storage volume on a SAN, or SMB 3.0 shared folder.)

Virtual machines' setting

The virtual machines setting refers to the default folder location of the virtual machine configuration files. When configuring a new virtual machine by using the wizard, it is this path that will be suggested as default. However, it is always possible to select a different virtual machine folder path during the wizard steps.

Just as with the virtual hard disks setting, this folder path could be on a local data disk on the Hyper-V host, a SAN storage volume/LUN, or the SMB 3.0 shared folder.

Physical GPUs' setting

The GPU setting refers to the Graphical Processing Unit, a video card let's say, which can be configured here to be used in conjunction with RemoteFX. RemoteFX is a Hyper-V feature that allows for rich video capabilities inside the virtual machines or in VDI (virtual desktop infrastructure) scenarios.

> We will describe the details of RemoteFX later on in this chapter.

The NUMA spanning setting

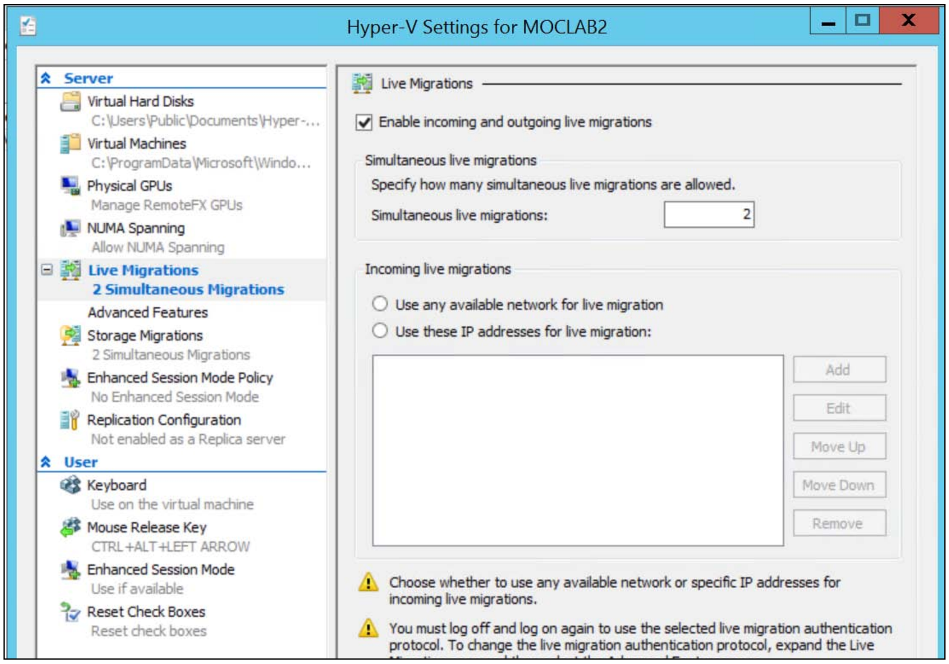
The **Non Uniform Memory Access** (**NUMA**) hardware architecture allows for serious performance increase on physical machines with many CPUs, by pairing memory with these CPUs. More and more server applications are also becoming NUMA aware (for example, SQL Server 2012, Internet Information Server 8.5, and others). The NUMA spanning setting within the Hyper-V settings allows you to span the memory settings between multiple NUMA capable machines.

> For a well-written blog post on NUMA hardware and NUMA spanning, we would like to refer you to an article by `@WorkingHardinIT` available at `http://workinghardinit.wordpress.com/tag/numa-node-spanning/`.

Live Migrations setting

The **Live Migrations** setting is where you configure live migration options for that specific Hyper-V host.

While we will talk more about live migration in the next chapter, we will describe the key live migration settings here. Have a look at the following screenshot:



Hyper-V Settings for MOCLAB2

Server

- Virtual Hard Disks — C:\Users\Public\Documents\Hyper-...
- Virtual Machines — C:\ProgramData\Microsoft\Windo...
- Physical GPUs — Manage RemoteFX GPUs
- NUMA Spanning — Allow NUMA Spanning
- Live Migrations — 2 Simultaneous Migrations
  - Advanced Features
- Storage Migrations — 2 Simultaneous Migrations
- Enhanced Session Mode Policy — No Enhanced Session Mode
- Replication Configuration — Not enabled as a Replica server

User

- Keyboard — Use on the virtual machine
- Mouse Release Key — CTRL+ALT+LEFT ARROW
- Enhanced Session Mode — Use if available
- Reset Check Boxes — Reset check boxes

Live Migrations

☑ Enable incoming and outgoing live migrations

Simultaneous live migrations

Specify how many simultaneous live migrations are allowed.

Simultaneous live migrations: 2

Incoming live migrations

○ Use any available network for live migration

○ Use these IP addresses for live migration:

⚠ Choose whether to use any available network or specific IP addresses for incoming live migrations.

⚠ You must log off and log on again to use the selected live migration authentication protocol. To change the live migration authentication protocol, expand the Live

First of all, by default, the live migration option is disabled. You have to enable this option to allow for setting configuration changes. Next to that, you will notice the option to define the number of simultaneous live migrations that can happen. In Hyper-V 2008 and 2008 R2, this was limited to a single VM. Of course, you have to take your overall Hyper-V hosts and network performance into account when configuring this to a higher number of virtual machines.

In the second part of the window, you can configure whether to use a dedicated IP address (NIC) for performing live migration or use any available network.

Under the **Advanced Features** topic within **Live Migrations**, you can specify which authentication protocol one can use, as well as configuring some performance optimization settings.

> We will describe all configuration details of Hyper-V live migration in the next chapter.

The storage migration setting

The storage migration setting is where you define the number of simultaneously happening storage migration actions.

We will describe all configuration details of Hyper-V storage migration in the next chapter.

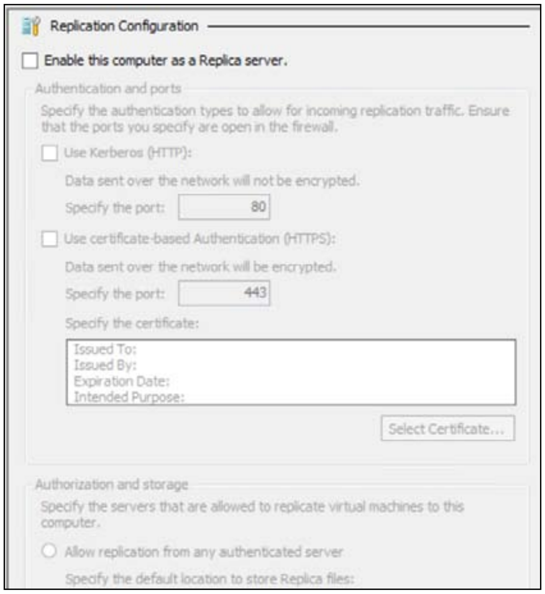
The enhanced session mode policy setting

With enhanced session mode enabled, one has the benefit of having the local devices and resources redirected to the virtual machine connection, if the OS running inside the virtual machine is supporting this feature. (Windows Server 2012, 2012 R2, and Windows 8 and 8.1 are supported.)

By using the enhanced session mode, the user can connect to the virtual machine by using the virtual machine connection tool, which relies on the remote desktop service of the VM. Features such as folder redirection, shared clipboard, and copying files by simple drag and drop are all possible in this mode. This is possible even when the VM doesn't have a working network connection, or when the network connection of the VM is on a different IP subnet than the physical host or remote management client.

The replication configuration setting

Here, you can configure Hyper-V Replica settings, such as specifying if that specific Hyper-V host has replication enabled (which by default, is not). While we will talk about Hyper-V Replica later on in this book (in the next chapter), I would like to describe the key configuration settings that are available here:



Replication Configuration

□ Enable this computer as a Replica server.

Authentication and ports

Specify the authentication types to allow for incoming replication traffic. Ensure that the ports you specify are open in the firewall.

□ Use Kerberos (HTTP):

Data sent over the network will not be encrypted.

Specify the port: 80

□ Use certificate-based Authentication (HTTPS):

Data sent over the network will be encrypted.

Specify the port: 443

Specify the certificate:

Issued To:
Issued By:
Expiration Date:
Intended Purpose:

Authorization and storage

Specify the servers that are allowed to replicate virtual machines to this computer.

○ Allow replication from any authenticated server

Specify the default location to store Replica files:

First of all, you have to enable the host for it to become a Hyper-V Replica server. Once this setting is enabled, you have access to different additional parameters that need to be configured before Hyper-V Replica will actually work. The important parameters here are authentication and IP ports to be used for the VM replication, as well as the default location where to save the Hyper-V Replica VMs and other hosts that are Hyper-V Replica partners.

[We will describe Hyper-V Replica more in detail in the next chapter.]

User-specific session settings

In the second part of the **Hyper-V Settings** window, you will detect a few user-related settings that can be configured, such as what happens with keyboard shortcuts using the Windows Key, and if those keystrokes should run inside the virtual machine's session or directly on the physical host. It also allows you the mouse release key (which by default is set to *Ctrl* + *Alt* + the left arrow key).

Lastly, there is also an option to reset all configured settings to Hyper-V default configuration settings right after the installation of the server role.

That's all about the base Hyper-V settings we can configure for a specific host.

Resource pools

If you are familiar with other hypervisor technologies, you are probably wondering whether Hyper-V is supporting the idea of resource pools, and if it is the same as with the other hypervisors.

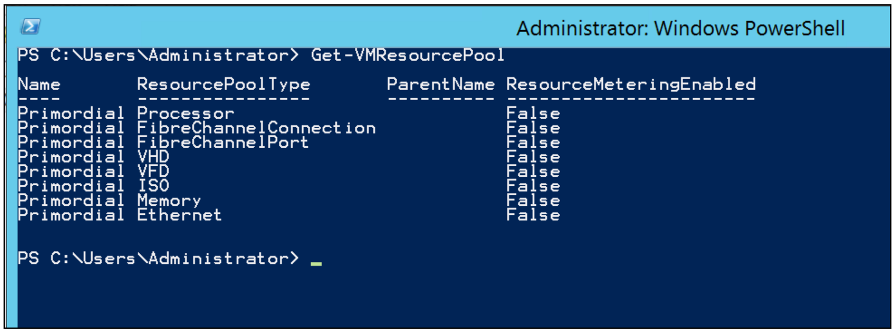
A resource pool is a combination of physical computer resources on host level (CPU, memory, network connections, storage, and so on) that are bound to VMs. Within VMware for example, these are managed from within the **Resource Allocation** tab in the VMware vCenter console.

Where you might expect to find a similar configuration setting within Hyper-V, it is not that easy to find. Actually, it is not visible as long as you don't create a resource pool first, which is only possible from within PowerShell.

Let's start by verifying the already existing resource pools, if any. This is done by using the following PowerShell cmdlet:

```
Get-VMResourcePool
```

The output is as follows:



```
PS C:\Users\Administrator> Get-VMResourcePool

Name       ResourcePoolType       ParentName ResourceMeteringEnabled
----       ----------------       ---------- -----------------------
Primordial Processor                         False
Primordial FibreChannelConnection            False
Primordial FibreChannelPort                  False
Primordial VHD                               False
Primordial VFD                               False
Primordial ISO                               False
Primordial Memory                            False
Primordial Ethernet                          False

PS C:\Users\Administrator> _
```

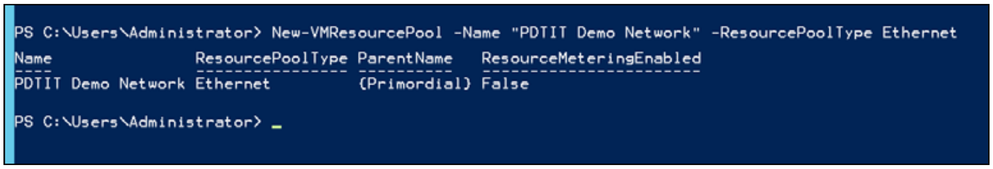
As you can see, there are already a few resource pools available, one for each default resource within Hyper-V.

Now let's create two new resource pools, one having a `ResourcePoolType` of Ethernet (networking connections) and a second one having a `ResourcePoolType` of VHD (virtual hard disks).

Creating a new resource pool for our network can be done as follows:

```
New-VMResourcePool -Name "PDTIT Demo Network" -ResourcePoolType Ethernet
```

The output is as follows:

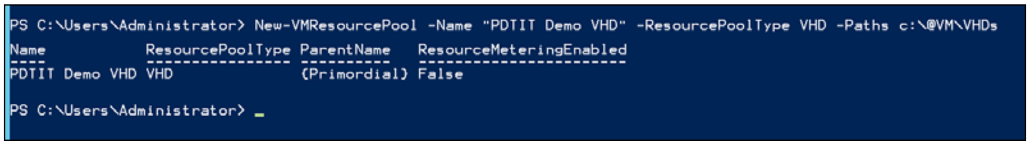


```
PS C:\Users\Administrator> New-VMResourcePool -Name "PDTIT Demo Network" -ResourcePoolType Ethernet
Name               ResourcePoolType ParentName   ResourceMeteringEnabled
----               ---------------- ----------   -----------------------
PDTIT Demo Network Ethernet         {Primordial} False

PS C:\Users\Administrator> _
```

Creating a new resource pool for our VHDs can be done as follows:

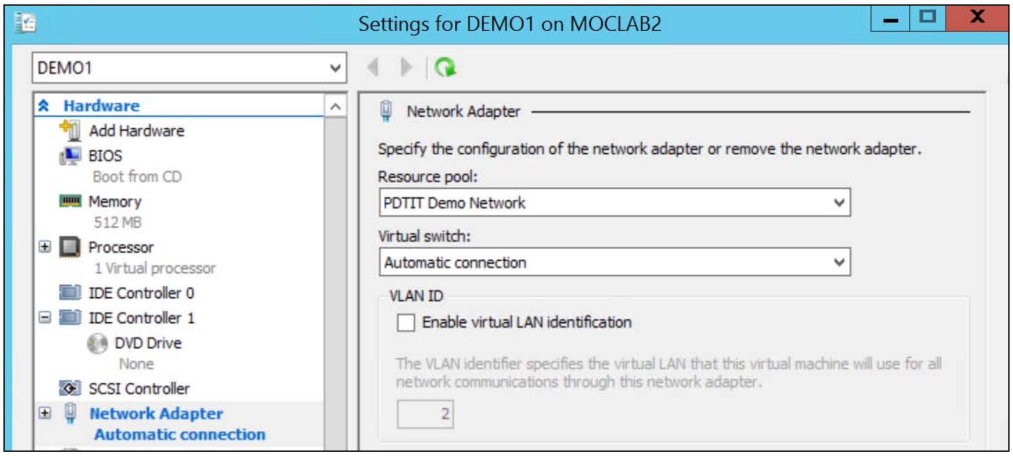
```
New-VMResourcePool -Name "PDTIT Demo VHD" -ResourcePoolType VHD -Paths
c:\@VM\VHDs
```

The output is as follows:



```
PS C:\Users\Administrator> New-VMResourcePool -Name "PDTIT Demo VHD" -ResourcePoolType VHD -Paths c:\@VM\VHDs
Name           ResourcePoolType ParentName   ResourceMeteringEnabled
----           ---------------- ----------   -----------------------
PDTIT Demo VHD VHD              {Primordial} False

PS C:\Users\Administrator> _
```

If we now go back into our Hyper-V console, we can see the **Resource pool** setting showing up in the virtual machine settings window under the **Network Adapter** configuration:



Settings for DEMO1 on MOCLAB2

DEMO1

Network Adapter

Specify the configuration of the network adapter or remove the network adapter.

Resource pool: PDTIT Demo Network

Virtual switch: Automatic connection

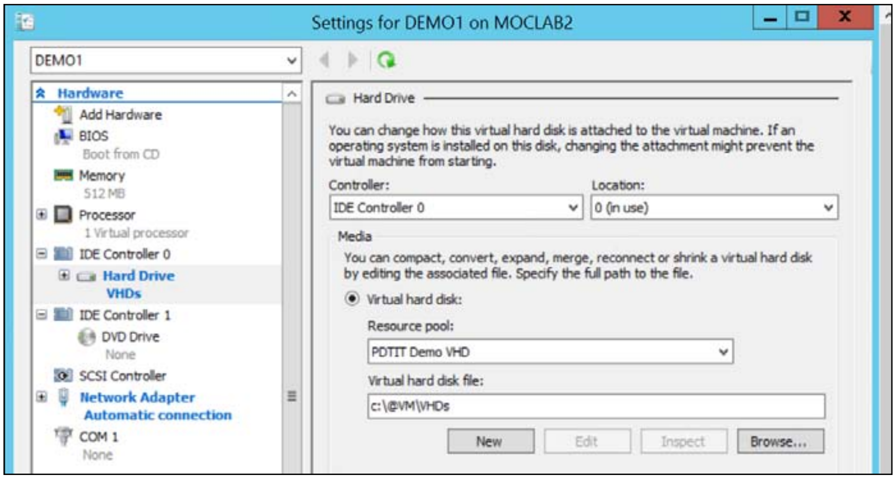
VLAN ID

□ Enable virtual LAN identification

The VLAN identifier specifies the virtual LAN that this virtual machine will use for all network communications through this network adapter.

2

Similarly, the same **Resource pool** settings are now visible when trying to add a virtual hard disk to the VM:



Settings for DEMO1 on MOCLAB2

DEMO1

Hard Drive

You can change how this virtual hard disk is attached to the virtual machine. If an operating system is installed on this disk, changing the attachment might prevent the virtual machine from starting.

Controller: IDE Controller 0

Location: 0 (in use)

Media

You can compact, convert, expand, merge, reconnect or shrink a virtual hard disk by editing the associated file. Specify the full path to the file.

● Virtual hard disk:

Resource pool: PDTIT Demo VHD

Virtual hard disk file: c:\@VM\VHDs

As should be clear now from the preceding screenshots, each VM resource can be bound to a resource from within a configured resource pool. Now, that's so far so good.

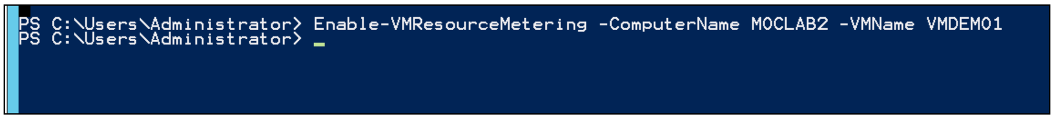
But the key question still unanswered here is, "Why in the world should I care about resource pools?"

One of the main functions behind the concept of resource pools is resource metering. Resource metering is used by data center administrators to monitor and track the usage of that specific resource. This could be due to performance reasons, but also could be for billing reasons, where a department pays more money for a high-performant server on high-performant disks.

Just as with the configuration of the resource pool itself, the metering is also only configurable through PowerShell. Enable metering as follows:

```
Enable-VMResourceMetering -ComputerName MOCLAB2 -VMName VMDEMO1
```

This cmdlet will enable the metering for the physical host `MOCLAB2`, as well as for the specific virtual machine called `VMDEMO1`. This is shown in the following screenshot:

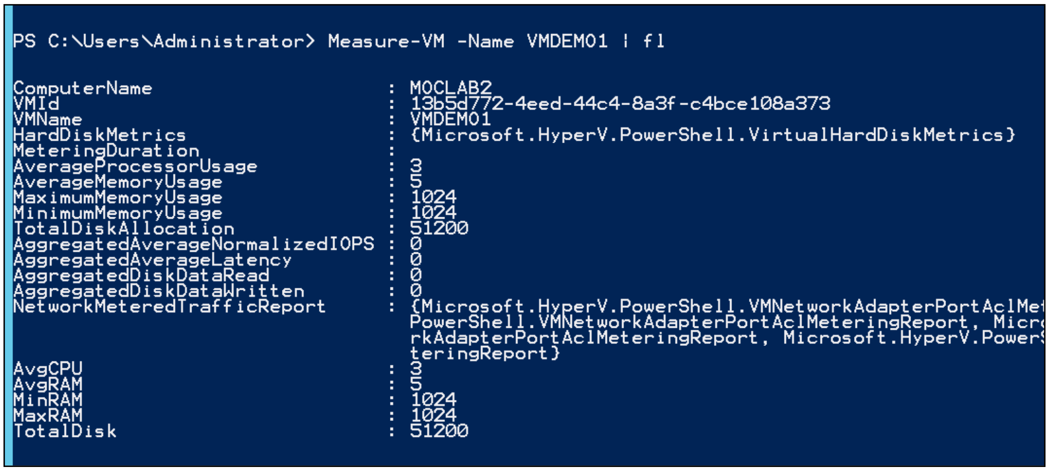


```
PS C:\Users\Administrator> Enable-VMResourceMetering -ComputerName MOCLAB2 -VMName VMDEMO1
PS C:\Users\Administrator> _
```

Lastly, gathering the effective metering information for the VM is done as follows:

```
Measure-VM -Name VMDEMO1 | fl
```

This gives the following result for a running machine:



```
PS C:\Users\Administrator> Measure-VM -Name VMDEMO1 | fl


ComputerName                    : MOCLAB2
VMId                            : 13b5d772-4eed-44c4-8a3f-c4bce108a373
VMName                          : VMDEMO1
HardDiskMetrics                 : {Microsoft.HyperV.PowerShell.VirtualHardDiskMetrics}
MeteringDuration                :
AverageProcessorUsage           : 3
AverageMemoryUsage              : 5
MaximumMemoryUsage              : 1024
MinimumMemoryUsage              : 1024
TotalDiskAllocation             : 51200
AggregatedAverageNormalizedIOPS : 0
AggregatedAverageLatency        : 0
AggregatedDiskDataRead          : 0
AggregatedDiskDataWritten       : 0
NetworkMeteredTrafficReport     : {Microsoft.HyperV.PowerShell.VMNetworkAdapterPortAclMet
                                  PowerShell.VMNetworkAdapterPortAclMeteringReport, Micro
                                  rkAdapterPortAclMeteringReport, Microsoft.HyperV.PowerS
                                  teringReport}
AvgCPU                          : 3
AvgRAM                          : 5
MinRAM                          : 1024
MaxRAM                          : 1024
TotalDisk                       : 51200
```

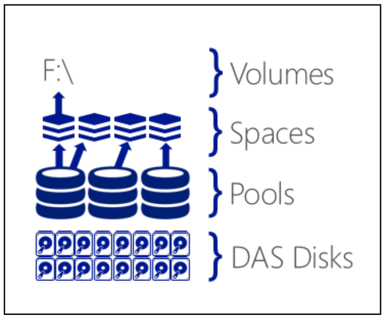
Hyper-V virtual storage configuration

Besides the more traditional iSCSI and Fibre Channel storage, Hyper-V also integrates and supports a number of other key Windows Server 2012 R2 storage features and capabilities, such as the following:

- Storage Spaces
- Storage tiering
- Deduplication
- Hyper-V over SMB 3.0

Storage Spaces

Storage Spaces is a technology in Windows Server 2012 and 2012 R2 that allows you to virtualize storage by combining DAS (Direct Attached Storage—local to the physical server) connected physical disks, which enables you to aggregate disks, expand capacity in a flexible manner, and delegate administration. These pools are split-up into spaces. Storage spaces are virtual disks created from free space in a storage pool. Storage spaces have such attributes as resiliency level, storage tiers, fixed provisioning, and precise administrative control. Storage Spaces is manageable through the Windows Storage Management API in **Windows Management Instrumentation** (**WMI**) and Windows PowerShell, and through the File and Storage Services role in Server Manager. Storage Spaces is completely integrated with failover clustering for high availability, and it is integrated with CSV for scale-out deployments.



Storage tiering

To benefit from the value of storage hardware, an administrator can combine hard disks and solid-state drives (SSDs) in the same pool, using storage tiers to move frequently accessed portions of files to SSD storage, and using write-back caches to buffer small random writes to SSD storage.

Pools can be expanded dynamically by simply adding additional drives, thereby seamlessly scaling to provide an answer to unforeseen data growth.

> Storage Spaces and Storage tiering, as described here, are new to Windows Server 2012 R2. For a good overview of how to configure Storage Spaces and Storage tiering in your lab or production environment, we would like you to refer to the Microsoft TechNet blog post by Jose Barreto available at `http://blogs.technet.com/b/josebda/archive/2013/08/28/step-by-step-for-storage-spaces-tiering-in-windows-server-2012-r2.aspx`.

Deduplication

Windows Server 2012 R2 also provides built-in deduplication capabilities, which utilize compression to considerably reduce storage consumption for files and folders hosted on deduplicated Windows Server volumes.

With Windows Server 2012 R2, support has been added for VDI deployments. Deduplication rates for VDI deployments can range as high as 95 percent savings, and that includes VDI deployments that utilize differencing disks for rapid provisioning. This is partly because a lot of data blocks of such VDI virtual disks are identical.

SMB 3.0 for Hyper-V

SMB is a file share protocol, built into Windows machines for a long time already. However, SMB 3.0 is totally new to Windows Server 2012. By leveraging the benefits of SMB 3.0 such as ease of configuration and being configurable using the already existing networking hardware, it has become a very popular storage mechanism for storing your virtual machines on.

Taking high availability and redundancy into account, as well as optimizing performance, SMB 3.0 introduced the concept of SMBMultiChannel, which increased the network performance and availability for file servers (this is a lot similar to MPIO—the multipathing option within iSCSI or Fibre Channel networks).

In addition, Windows Server 2012 and R2 can harness the power of RDMA capable NICs, through a feature known as SMB Direct, to drive even higher levels of performance. SMB Direct, which supports the use of network adapters that have **Remote Direct Memory Access** (**RDMA**) capability, enables NICs to function at full speed with very low latency, while using very little CPU. For workloads such as Hyper-V or Microsoft SQL Server, this enables a remote file server to resemble local storage.

> RDMA capable NICs can reach speeds of up to 56 GBps, enabling significant performance advantages over the costly Fibre Channel or iSCSI alternatives.

Hyper-V virtual network configuration

Just as with the virtual storage configuration, the virtual network configuration of the Hyper-V platform is equally important.

If you are working on a Hyper-V environment without Virtual Machine Manager, the entire virtual network configuration is done from within the Hyper-V console or through PowerShell.

If, however, you are (planning to) running Virtual Machine Manager, it is advised to configure the virtual network settings from within VMM directly, as it provides a few more options, specific to VMM.

The intelligence of the virtual network comes from the Hyper-V virtual switch. Think about a typical network with multiple computers and other networking resources; they are connected to each other by using a network switch. In a virtualized environment, the role of the virtual switch is no different; it allows you to interconnect different virtual machines with each other, as well as allowing for pass-through connection to the physical network, if needed.

Hyper-V virtual switch configuration

One of the first things you need to perform after installing the Hyper-V Server role is the virtual network switch configuration.

When creating or configuring a Hyper-V virtual switch, we can decide between three different types: Internal, External, or Private.

- **Internal**: This switch type allows you to network connections between all virtual machines on the specific hosts and the physical host itself. Connecting to other physical machines on the physical network or virtual machines on a different host is not possible.
- **External**: This switch type allows you to network connections between all virtual machines in the full network topology, as well as interconnecting between physical hosts and virtual machines on these different hosts.
- **Private**: The private switch type allows only networking communication between virtual machines on a specific host, but not with the physical host itself.

Besides limiting the scope of the virtual machine's networking connections by specifying any of the different virtual switch types, a second networking isolation can be achieved by using VLAN tagging.

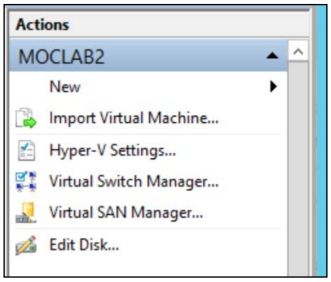
If your physical network interface cards do support SR-IOV and/or Dynamic Virtual Machine Queue, you can configure these options as well to optimize the networking throughput even more.

When compared to the Hyper-V virtual switch in previous OS versions, Windows Server 2012 and 2012 R2 provide Hyper-V virtual switch extensions. These extensions allow for monitoring, filtering, or forwarding specific network traffic. These extensions are applied by specifying the NDIS filter drivers and **Windows Filtering Platform** (**WFP**) drivers.

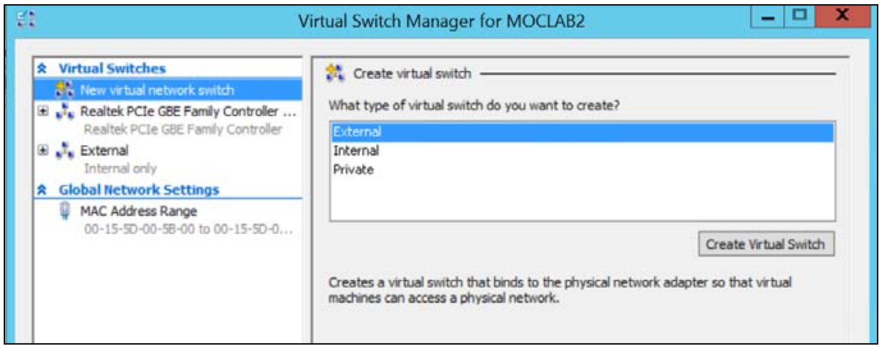
Creating or configuring a Hyper-V virtual switch using GUI

To create or configure a Hyper-V virtual switch, use the following steps:

1. From within the Hyper-V management console, select **Virtual Switch Manager** from the **Actions** pane:



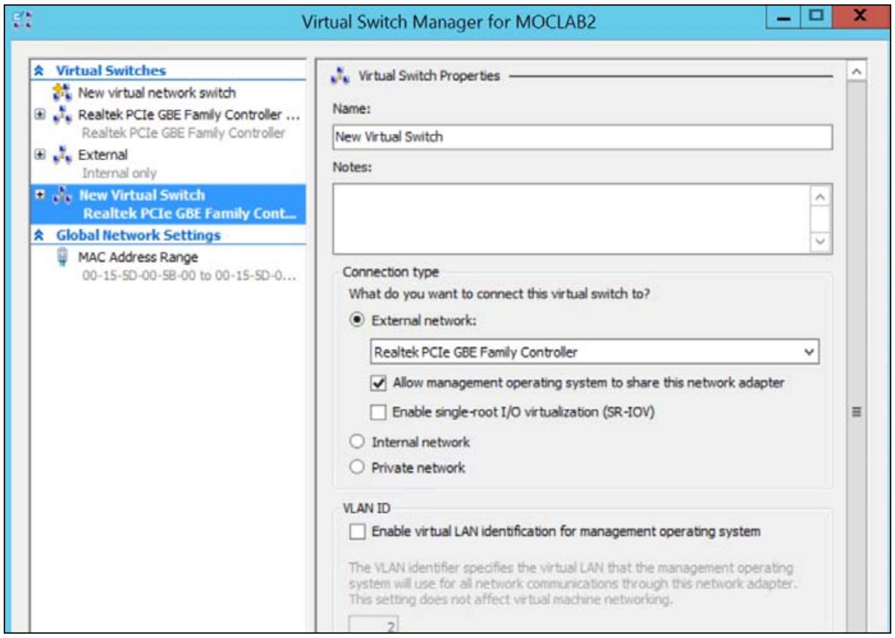
2. Select the **New virtual network switch** option from the left-hand side menu:


Virtual Switch Manager for MOCLAB2
Virtual Switches
New virtual network switch
Realtek PCIe GBE Family Controller ...
Realtek PCIe GBE Family Controller
External
Internal only
Global Network Settings
MAC Address Range
00-15-5D-00-5B-00 to 00-15-5D-0...
Create virtual switch
External
Internal
Private
Create Virtual Switch

What type of virtual switch do you want to create?

Creates a virtual switch that binds to the physical network adapter so that virtual machines can access a physical network.

3. Specify the type of virtual switch you want to create, for example **External** as shown in the preceding screenshot, and click on the **Create Virtual Switch** button.



Virtual Switch Manager for MOCLAB2

Virtual Switches
New virtual network switch
Realtek PCIe GBE Family Controller ...
Realtek PCIe GBE Family Controller
External
Internal only
New Virtual Switch
Realtek PCIe GBE Family Cont...
Global Network Settings
MAC Address Range
00-15-5D-00-5B-00 to 00-15-5D-0...
Virtual Switch Properties

Name: New Virtual Switch

Notes:

Connection type

What do you want to connect this virtual switch to?

● External network:

Realtek PCIe GBE Family Controller

☑ Allow management operating system to share this network adapter

□ Enable single-root I/O virtualization (SR-IOV)

○ Internal network

○ Private network

VLAN ID

□ Enable virtual LAN identification for management operating system

The VLAN identifier specifies the virtual LAN that the management operating system will use for all network communications through this network adapter. This setting does not affect virtual machine networking.

2

4. Provide a descriptive name for this virtual switch connection, for example `Company LAN`.
5. As we did select the External switch type, we have to select the physical network interface card from the host to which this external network will be bound.
6. You may select the option to allow SR-IOV, if your physical NIC supports this. (We will explain more about SR-IOV in the next couple of sections.)
7. You may enable the option for VLAN tagging, and specify the number of the VLAN you want to bind this virtual switch to.
8. Finish the configuration by clicking on **Apply**. Confirm the pop up warning you about the possible loss of network communication during reconfiguration of the NIC and creation of the virtual switch.

This concludes the configuration of your virtual switch from within the Hyper-V management console. Repeat these steps if you want to create additional virtual switches.

Configuring a Hyper-V virtual switch using PowerShell

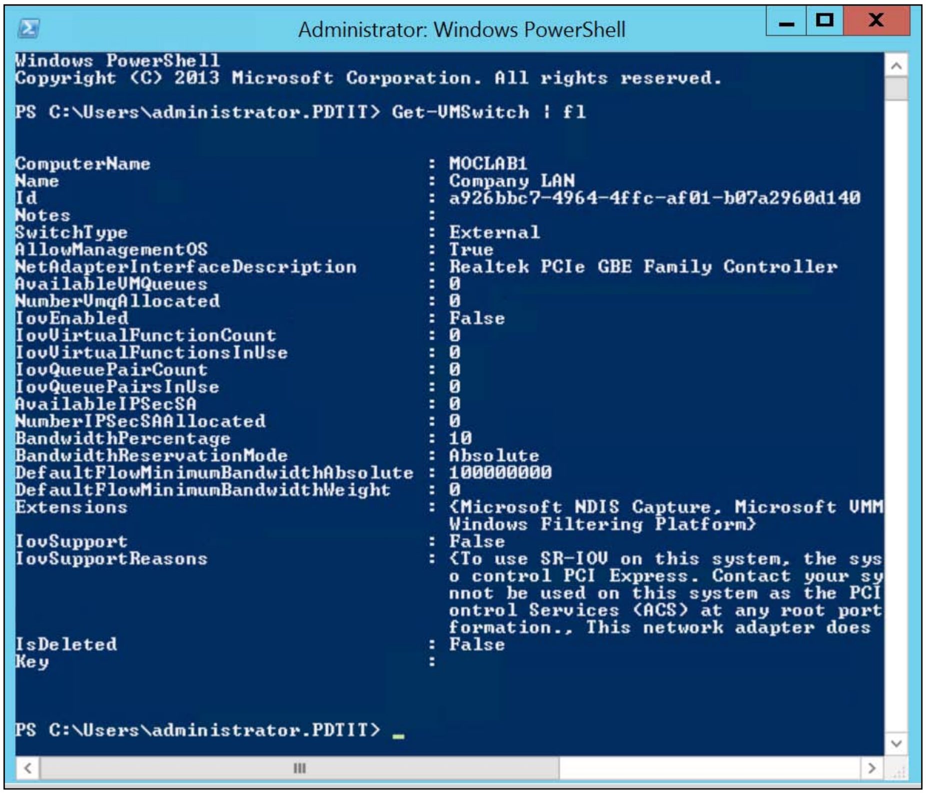
In the same way as the configuration process using GUI described in the previous section, you can create or configure Hyper-V virtual switch settings by using PowerShell, or get the current configuration displayed.

Viewing current Hyper-V virtual switch details

The following cmdlet will show you all the details of all currently existing Hyper-V switches:

```
Get-VMSwitch | fl
```

The output is as follows:



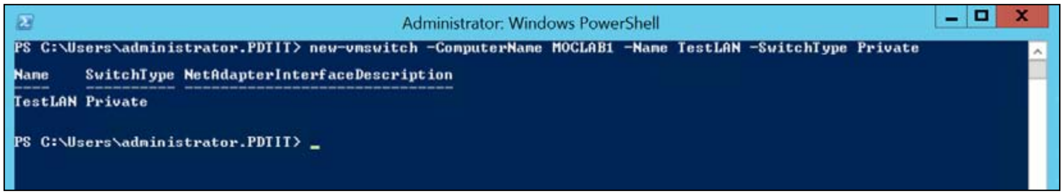
```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

PS C:\Users\administrator.PDTIT> Get-VMSwitch | fl


ComputerName                        : MOCLAB1
Name                                : Company LAN
Id                                  : a926bbc7-4964-4ffc-af01-b07a2960d140
Notes                               :
SwitchType                          : External
AllowManagementOS                   : True
NetAdapterInterfaceDescription      : Realtek PCIe GBE Family Controller
AvailableVMQueues                   : 0
NumberVmqAllocated                  : 0
IovEnabled                          : False
IovVirtualFunctionCount             : 0
IovVirtualFunctionsInUse            : 0
IovQueuePairCount                   : 0
IovQueuePairsInUse                  : 0
AvailableIPSecSA                    : 0
NumberIPSecSAAllocated              : 0
BandwidthPercentage                 : 10
BandwidthReservationMode            : Absolute
DefaultFlowMinimumBandwidthAbsolute : 100000000
DefaultFlowMinimumBandwidthWeight   : 0
Extensions                          : {Microsoft NDIS Capture, Microsoft VMM
                                      Windows Filtering Platform}
IovSupport                          : False
IovSupportReasons                   : {To use SR-IOV on this system, the sys
                                      o control PCI Express. Contact your sy
                                      nnot be used on this system as the PCI
                                      ontrol Services (ACS) at any root port
                                      formation., This network adapter does
IsDeleted                           : False
Key                                 :



PS C:\Users\administrator.PDTIT> _
```

Creating a new Hyper-V virtual switch of Private type

The following cmdlet creates a new virtual switch named `TestLAN` on the host `MOCLAB1`:

```
new-vmswitch -ComputerName MOCLAB1-Name TestLAN -SwitchType Private
```

The output is as follows:



```
PS C:\Users\administrator.PDTIT> new-vmswitch -ComputerName MOCLAB1 -Name TestLAN -SwitchType Private

Name    SwitchType NetAdapterInterfaceDescription
----    ---------- ------------------------------
TestLAN Private

PS C:\Users\administrator.PDTIT> _
```

Advanced Hyper-V virtual network card settings

Besides the already mentioned features and base settings of the Hyper-V virtual switch, a virtual machine's network card also has a few more advanced settings. Some of these are dependent on the capabilities of the physical NICs in the Hyper-V host. Additionally, some of these advanced features can only be configured by using PowerShell.

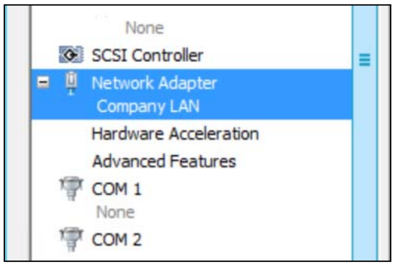
We will go through some of these advanced features now, starting by listing them up:

- **MAC address spoofing**: This allows virtual machines to change MAC address settings in outgoing IP packets
- **DHCP guard protection**: This drops DHCP server messages from virtual machines pretending to be DHCP servers
- **Router guard**: This drops routing protocol messages from virtual machines pretending to be routers
- **Protected network**: This allows you to move the virtual machine within the Hyper-V cluster if the network card of the VM becomes disconnected
- **Port mirroring**: This allows you to monitor the incoming and outgoing network packets from a virtual machine
- **NIC teaming**: This allows you to bundle multiple network cards of that specific VM, and present them as a single network connection

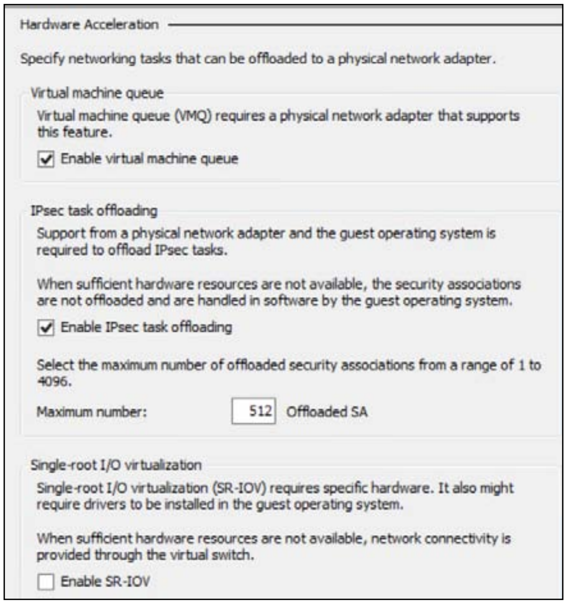
Specifically regarding the performance, the following advanced settings are also available for a virtual machine's network card, if the physical NIC in the host supports these features:

- Dynamic Virtual Machine Queue provides optimized network throughput where network traffic is shared over multiple CPUs
- IPSec task offloading
- Single Root I/O virtualization (SR-IOV)

These virtual network adapter settings can be viewed/configured by selecting a virtual machine, clicking on **Settings**, and then clicking on the **+** sign next to **Network Adapter**. This will show the **Hardware Acceleration** setting as well as the **Advanced Features** setting.



The following screenshot shows the **Hardware Acceleration** settings:



Hardware Acceleration

Specify networking tasks that can be offloaded to a physical network adapter.

Virtual machine queue

Virtual machine queue (VMQ) requires a physical network adapter that supports this feature.

☑ Enable virtual machine queue

IPsec task offloading

Support from a physical network adapter and the guest operating system is required to offload IPsec tasks.

When sufficient hardware resources are not available, the security associations are not offloaded and are handled in software by the guest operating system.

☑ Enable IPsec task offloading

Select the maximum number of offloaded security associations from a range of 1 to 4096.

Maximum number: 512 Offloaded SA

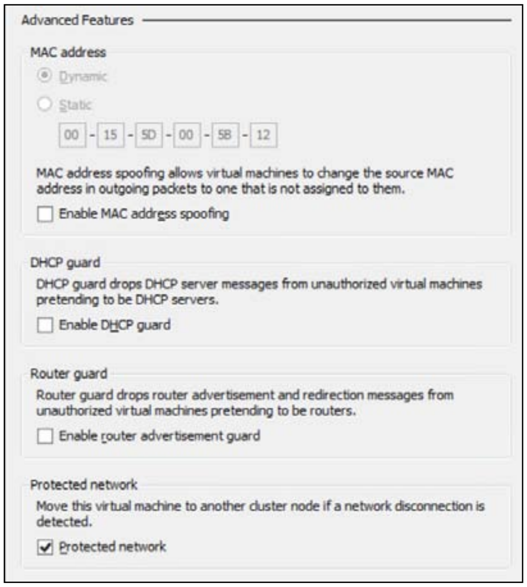
Single-root I/O virtualization

Single-root I/O virtualization (SR-IOV) requires specific hardware. It also might require drivers to be installed in the guest operating system.

When sufficient hardware resources are not available, network connectivity is provided through the virtual switch.

□ Enable SR-IOV

The following screenshot shows the **Advanced Features** settings:



Advanced Features

MAC address

● Dynamic

○ Static

00 - 15 - 5D - 00 - 5B - 12

MAC address spoofing allows virtual machines to change the source MAC address in outgoing packets to one that is not assigned to them.

□ Enable MAC address spoofing

DHCP guard

DHCP guard drops DHCP server messages from unauthorized virtual machines pretending to be DHCP servers.

□ Enable DHCP guard

Router guard

Router guard drops router advertisement and redirection messages from unauthorized virtual machines pretending to be routers.

□ Enable router advertisement guard

Protected network

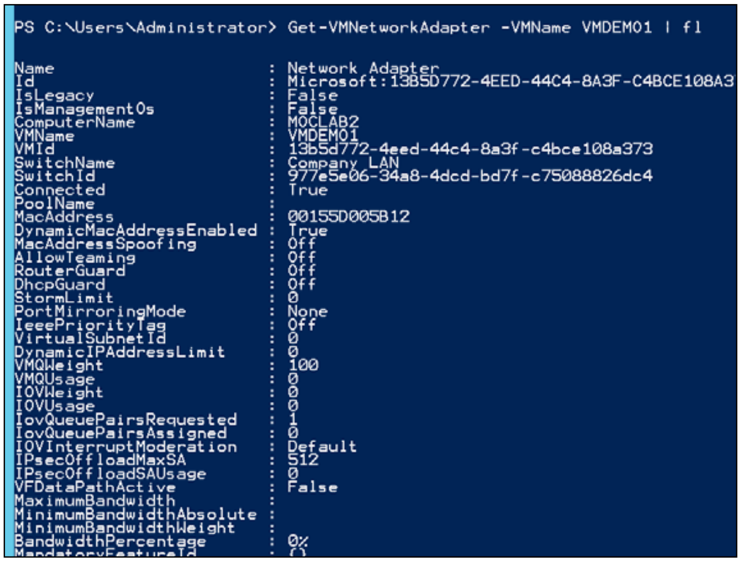
Move this virtual machine to another cluster node if a network disconnection is detected.

☑ Protected network

In the same way, the aforementioned settings can also be viewed/customized by using PowerShell. This is achieved using the following cmdlet:

```
Get-VMNetworkAdapter -VMName VMDEMO1 | fl
```

This results in the following details:



```
PS C:\Users\Administrator> Get-VMNetworkAdapter -VMName VMDEM01 | fl

Name                          : Network Adapter
Id                            : Microsoft:13B5D772-4EED-44C4-8A3F-C4BCE108A3
IsLegacy                      : False
IsManagementOs                : False
ComputerName                  : MOCLAB2
VMName                        : VMDEM01
VMId                          : 13b5d772-4eed-44c4-8a3f-c4bce108a373
SwitchName                    : Company LAN
SwitchId                      : 977e5e06-34a8-4dcd-bd7f-c75088826dc4
Connected                     : True
PoolName                      :
MacAddress                    : 00155D005B12
DynamicMacAddressEnabled      : True
MacAddressSpoofing            : Off
AllowTeaming                  : Off
RouterGuard                   : Off
DhcpGuard                     : Off
StormLimit                    : 0
PortMirroringMode             : None
IeeePriorityTag               : Off
VirtualSubnetId               : 0
DynamicIPAddressLimit         : 0
VMQWeight                     : 100
VMQUsage                      : 0
IOVWeight                     : 0
IOVUsage                      : 0
IovQueuePairsRequested        : 1
IovQueuePairsAssigned         : 0
IOVInterruptModeration        : Default
IPsecOffloadMaxSA             : 512
IPsecOffloadSAUsage           : 0
VFDataPathActive              : False
MaximumBandwidth              :
MinimumBandwidthAbsolute      :
MinimumBandwidthWeight        :
BandwidthPercentage           : 0%
MandatoryFeatureId            : {}
```

If you want to modify any of these advanced settings through PowerShell, use the `Set-VMNetworkAdapter` cmdlet for this. The following cmdlet will activate the DHCP guard setting for the virtual NIC in my `VMDEMO1` virtual machine:

```
Get-VMNetworkAdapter -VMNameVMDEMO1 | Set-VMNetworkAdapter -DhcpGuard on
```

SR-IOV explained

SR-IOV or Single Root I/O Virtualization is a standard that allows for better network throughput performance on a virtual layer. As this feature is natively dependent on the capability within a physical NIC, it can only be configured on the **External** Hyper-V virtual switch type.

By using this direct connection between a physical NIC and a virtual machine's NIC, the network adapter bypasses the virtual switch, whereby the networking throughput increases, and the CPU performance and network latency decreases.

> If you want to know more about SR-IOV, please refer to the Microsoft TechNet article at `http://go.microsoft.com/fwlink/?LinkID=386698`.

Dynamic Virtual Machine Queue explained

Virtual Machine Queue (**VMQ**) is similar to Receive Side Scaling, in such a way that it enables NICs to distribute the network load across multiple processors, but on a virtual layer. This results in better network throughput, as it can handle more network bandwidth than a single CPU could process.

VMQ is enabled by default within Hyper-V, but it depends on the physical NIC to actually work. So if your physical NIC doesn't support it, it won't give any benefit in the Hyper-V environment either.

> Have a look at the Microsoft article at `http://msdn.microsoft.com/en-us/library/windows/hardware/ff571034(v=vs.85).aspx` to get a better and more detailed understanding of VMQ.

Creating and configuring virtual machines

After having the physical Hyper-V hosts up-and-running, having configured the virtual storage locations and completing the virtual networking configuration, we can move to the last module of this chapter, creating and configuring virtual machines.

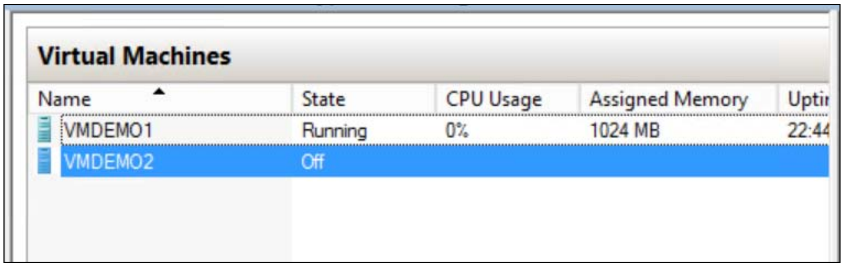
Creating virtual machines is done from within the Hyper-V management console or by using PowerShell; we will show you both ways to achieve this. After explaining the creation process, we will walk you through the base virtual machine settings, explaining the difference between a Generation 1 and Generation 2 virtual machine, as well as what the idea behind the Hyper-V Integration Services is.

Creating a new virtual machine using GUI

Without talking too much about the detailed configuration settings yet, let us guide you in the process of creating a new virtual machine. (As we don't expect you to have too many issues in creating a virtual machine in this way, we decided to not publish screenshots of the different steps and settings here. This should not be a problem for you to understand the topic.) The steps are as follows:

1. From within the Hyper-V console, select **New… | Virtual Machine…** from the **Actions** pane; click on **Next** in the first window of the wizard.
2. Provide a descriptive name for the virtual machine and also select the folder path where the virtual machine configuration settings will be saved; click on **Next** to continue.
3. Choose the **Generation of the VM** to be created (for more details on VM Generations, see *Chapter 1*, *Introduction to Microsoft Server 2012 R2 Hyper-V*); click on **Next** to continue.
4. Select the **Virtual Network** to be used for this virtual machine; if your VM requires additional network adapters, add them once the VM is configured. Click on **Next** to continue.
5. In the next step, select how to connect the virtual machine to a virtual hard disk. Here you have the **Create a virtual hard disk** option, which can be done by specifying the folder path to your Hyper-V storage location and defining the size of the virtual hard disk. Besides creating a new virtual disk, you can also choose to select an already existing disk or create the virtual machine without a virtual hard disk connected yet.

6. In the following step, you can define how to install the operating system on the virtual machine. The installation files can be mounted from an ISO file, from a bootable CD/DVD-ROM, from a shared folder, or you can decide to install the OS later on. Click on **Next** to continue.
7. In the last step, a summary is provided. After clicking on **Finish**, the virtual machine will be created. This should only take a few seconds, depending on your physical hardware.
8. Once the virtual machine is created, it will show up in the virtual machines list in the Hyper-V console. By default, the machine is turned off.



Virtual Machines

Name	State	CPU Usage	Assigned Memory	Upti
VMDEMO1	Running	0%	1024 MB	22:44
VMDEMO2	Off			

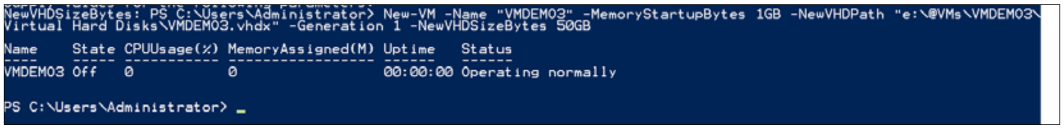
Now let us create a similar virtual machine by using PowerShell.

Creating a virtual machine using PowerShell

From within the PowerShell window, the key cmdlet to be used is `New-VM`. This cmdlet has all the parameters available to specify the same settings that were used to create the virtual machine using the GUI.

```
New-VM -Name VMDEMO3 -MemoryStartupBytes 1GB -NewVHDPath "e:\@VMs\
VMDEMO3\Virtual Hard Disks\VMDEMO3.vhdx" -Generation 1 -NewVHDSizeBytes
50GB
```

The preceding cmdlet creates a new virtual machine named `VMDEMO3`, having 1 GB of memory allocated to it. The virtual hard disk will be created in the given folder path, having a size of 50 GB. By using the `-Generation` parameter, we specify a Generation 1 virtual machine. The result of the cmdlet is shown in the following screenshot:



```
NewVHDSizeBytes: PS C:\Users\Administrator> New-VM -Name "VMDEMO3" -MemoryStartupBytes 1GB -NewVHDPath "e:\@VMs\VMDEMO3\
Virtual Hard Disks\VMDEMO3.vhdx" -Generation 1 -NewVHDSizeBytes 50GB

Name    State CPUUsage(%) MemoryAssigned(M) Uptime   Status
----    ----- ----------- ----------------- ------   ------
VMDEMO3 Off   0           0                 00:00:00 Operating normally

PS C:\Users\Administrator> _
```

Once the virtual machine has been created, you can validate its settings from within the Hyper-V console again if you want to.

> For a full list of all parameters that are available for the `New-VM` cmdlet, have a look at the Microsoft TechNet article at `http://technet.microsoft.com/en-us/library/hh848537.aspx`.

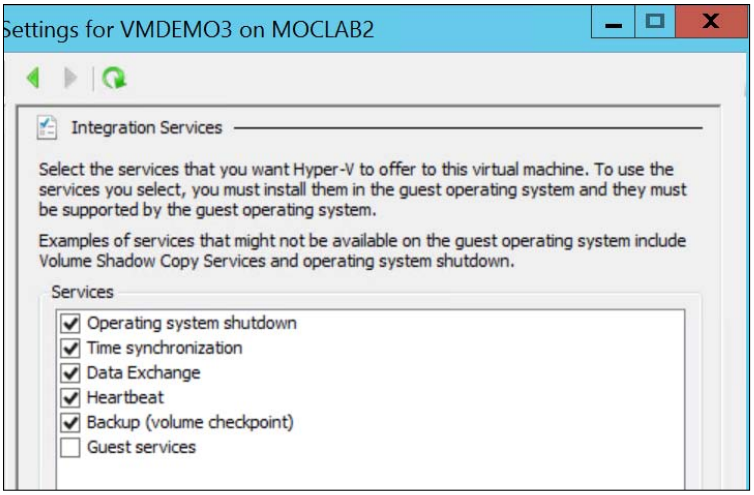
Hyper-V Integration Services

When an operating system is installed on a virtual machine, it does not always recognize the emulated virtual layer between the VM and the hypervisor. Also, certain features of the hypervisor might not work, as again the virtual machine is not aware of the virtual layer. To allow full functionality and provide full support between the hypervisor and the virtual machines running on top of it, one could install **Hyper-V Integration Services**.

Hyper-V Integration Services provides the following feature set:

- **Hyper-V Guest Shutdown Service**: This allows you to initiate a proper VM shutdown from within the Hyper-V host directly, without initiating it from within the virtual machine's operating system itself.
- **Hyper-V Time Sync Service**: This allows for time synchronization between the VM and the physical host, where the VM syncs its clock with the physical host.
- **Hyper-V Heartbeat Service**: This component allows the host to verify if the virtual machine responds to requests.
- **Backup**: This allows integration with the VSS engine to enable virtual machine synchronization and backups of a running virtual machine.
- **Guest Services**: This option is new to Hyper-V Server 2012 R2; it allows for an enhanced session mode (which we talked about in the previous chapter), device redirection, shared clipboard, as well as drag and drop of files between the VM and the Hyper-V host.

These services are shown in the following screenshot:



Integration Services

Select the services that you want Hyper-V to offer to this virtual machine. To use the services you select, you must install them in the guest operating system and they must be supported by the guest operating system.

Examples of services that might not be available on the guest operating system include Volume Shadow Copy Services and operating system shutdown.

Services

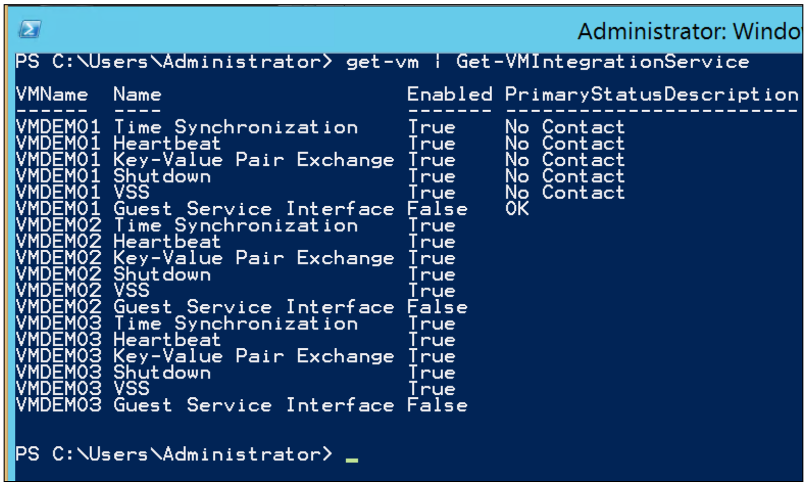
- ☑ Operating system shutdown
- ☑ Time synchronization
- ☑ Data Exchange
- ☑ Heartbeat
- ☑ Backup (volume checkpoint)
- □ Guest services

Depending on your specific configuration and environment, enable or disable some of these integration services, by selecting or deselecting them.

The same can be achieved by using PowerShell's `Get-VMIntegrationService` cmdlet:

```
get-vm | Get-VMIntegrationService
```

This provides a list of all virtual machines on the Hyper-V host, with their corresponding Integration Services options and their status:



```
PS C:\Users\Administrator> get-vm | Get-VMIntegrationService

VMName  Name                     Enabled PrimaryStatusDescription
------  ----                     ------- ------------------------
VMDEMO1 Time Synchronization     True    No Contact
VMDEMO1 Heartbeat                True    No Contact
VMDEMO1 Key-Value Pair Exchange  True    No Contact
VMDEMO1 Shutdown                 True    No Contact
VMDEMO1 VSS                      True    No Contact
VMDEMO1 Guest Service Interface  False   OK
VMDEMO2 Time Synchronization     True
VMDEMO2 Heartbeat                True
VMDEMO2 Key-Value Pair Exchange  True
VMDEMO2 Shutdown                 True
VMDEMO2 VSS                      True
VMDEMO2 Guest Service Interface  False
VMDEMO3 Time Synchronization     True
VMDEMO3 Heartbeat                True
VMDEMO3 Key-Value Pair Exchange  True
VMDEMO3 Shutdown                 True
VMDEMO3 VSS                      True
VMDEMO3 Guest Service Interface  False

PS C:\Users\Administrator> _
```

Summary

In this chapter, we started by explaining the base Hyper-V settings, followed by detailing the concept of Hyper-V resource pools. After that, we explained the details of virtual storage and virtual network settings that are available in the Hyper-V environment.

In the second module of this chapter, we showed you how to create and configure virtual machines, both from within the Hyper-V console and by using PowerShell.

In the last part, we provided you with more detailed information on Hyper-V Integration Services, what they are used for, and how you can enable/disable them.

Having the Hyper-V environment available now, as well as having a few virtual machines running inside it, we move over to the next chapter where we will guide you on the cool features of live migration, Shared Nothing Live Migration, and Hyper-V Replica.

3

Live Migration, Storage Migration, and Hyper-V Replica

When you have a Hyper-V platform up and running, one of the important tasks you as an admin have to fulfill, is most probably, providing redundancy and high availability to the virtualization platform.

Within Hyper-V, this can be achieved in multiple ways, by using different configurations.

Basically, it always comes back to moving your virtual machines and their configurations and data (for example virtual hard disks) between different Hyper-V hosts. In the previous versions of Hyper-V, this worked best by using Hyper-V clusters. While Hyper-V clustering is still a viable option (see the next chapter for all details), thanks to the updated and new features of Hyper-V 2012 and 2012 R2, almost the same result can be achieved by using any of the following:

- Storage migration
- Live migration
- Hyper-V Replica

In this chapter, we will start by explaining the difference between these three features and why you should use them. After that, we will dive deeper into each feature, guiding you on the setup and configuration by using both the Hyper-V manager console and PowerShell.

> No matter what feature you use, the virtual machines will always remain active and live, without any downtime.

Why moving virtual machines is important

Before we move on to the technical details of the different features available, let's guide you on the subject of "why it is important to be able to move virtual machines".

As mentioned before, redundancy and high availability are the key triggers here. No matter what size your organization or data center itself is, you always want to make sure your applications stay up and running, your virtual machines remain reachable, and your end users can continue to work.

By using any of the already mentioned features of Hyper-V 2012 (and 2012 R2), redundancy can be provided on the following different layers in your data center:

- **Physical machine failover**: In any situation of planned or unplanned downtime of a physical machine, having the possibility to actually move the virtual machine to other available physical machine(s) is a great benefit. Imagine a component such as a RAM module in the physical machine has failed. When you as an administrator detect this problem in time, you can leverage the power of live migration to move the virtual machines to any other Hyper-V host in your data center. Again, by any host we mean any host, and not only Hyper-V hosts that are part of the same cluster.
- **Storage failover**: Not only the Hyper-V hosts, but also the storage needs to be redundant, which is a vital part in the overall platform configuration. Imagine you want to expand your storage system with new hard disks, which might require you to shut down the storage solution (depending on vendor and storage type). By using storage migration, you are not only moving the Hyper-V virtual machine, but also the virtual hard disks that are linked to it, to a different storage solution. It is also possible to perform a full storage migration by using this scenario. Imagine you are switching between storage solutions (from the same or another vendor). Instead of taking offline backups and doing full restores on the new storage platform, you can move your entire Hyper-V platform, most importantly, without any downtime for the virtual machines and linked virtual hard disks.

- **Location failover**: By using live migration, virtual machines can be moved between two physical Hyper-V hosts, online. It's a great feature as long as the network connectivity between both hosts is reliable, stable, and of adequate bandwidth. While this is most certainly the case within the same data center, it is not always the case between data centers in different locations. By using the new Hyper-V Replica feature of Windows Server 2012, virtual machines can be moved in an online state, between Hyper-V hosts in stretched data centers in different locations.

> A good Microsoft TechNet document describing the details around network requirements and providing bandwidth samples can be found at `http://technet.microsoft.com/en-us/library/ff428137(v=ws.10).aspx`.

Live migration without a Hyper-V cluster

With Hyper-V running on Windows Server 2008 R2, migrating virtual machines between hosts was only possible within the same Hyper-V failover cluster and only if the data was stored on the shared storage within that cluster.

Though you can still migrate virtual machines in the same way on a Windows Server 2012 R2 Hyper-V cluster, this edition of Hyper-V brings a few other scenarios that allow for moving machines between hosts, without requiring the Hyper-V cluster dependency, as follows:

- Live migration between hosts, when the virtual machine data is stored on a local storage to the host; this is known as Shared Nothing Live Migration
- Live migration between hosts, when the virtual machine data is stored on a SMB 3.0 shared location

The main difference between both the solutions is that with Shared Nothing Live Migration, both virtual machine configuration and data are actually being moved, where in the case of the SMB 3.0 share, only the virtual machine configuration is moved, but the virtual machine data itself remains on the shared folder.

> It is important to note, however, that—in both solutions—there is no downtime at all for the running virtual machine(s) that are being moved.

Before moving on to the technical details on how to configure live migration, it is important to understand the different steps that are being performed while executing a live migration process. As already mentioned, there is only a small difference between both solutions, where the virtual machines' data needs to be moved between hosts or remains on the SMB 3.0 share. All other steps in the move sequence are similar in both.

The process of live migration

In short, the following steps are taken:

1. The virtual machine configuration information is copied to the destination Hyper-V host. This is actually based on a fresh virtual machine, which gets the configuration files imported. There is no data disk available yet for the new virtual machine. Here, the memory is in the process of being allocated to the destination virtual machine.
2. In the second step, the live memory content is copied over from source to destination virtual machine. This is known as the *working set* of the VM, which copies the in-memory state in chunks of 4 KB. It's the Hyper-V host that monitors this migration and tracks updates of changes in memory in between starting live migration and the actual failover.
3. After the initial memory copy, a continuous update of memory changes is synchronized between the source and target virtual machine/host. At some point in time, the working set will be identical between both machines.
4. If the move is part of the Shared Nothing Live Migration process, an initial synchronization happens from the source virtual machine data files to the target host. As long as the initial synchronization is not completed, all read/write operations to the virtual machine are being performed on the source side.
5. After the initial copy is completed, write operations to the virtual machine are being mirrored to both source and destination VM.
6. Once all data between both source and target are aligned, the virtual machine switches over to the disk on the destination host.
7. As the last step, in case of Shared Nothing Live Migration, the data disk on the source server is deleted. In case of SMB 3.0 shared storage, no data disk needs to be moved physically.

By default, two migrations can be executed simultaneously, but you can configure this parameter from within the Hyper-V console or by using PowerShell.

Although we've already talked about this before, we want you to remember that this kind of storage migration does not work from within a Fibre Channel connection or when you use a SAN or pass-through disk subsystem. If that is the case, the live migration will fail with an error.

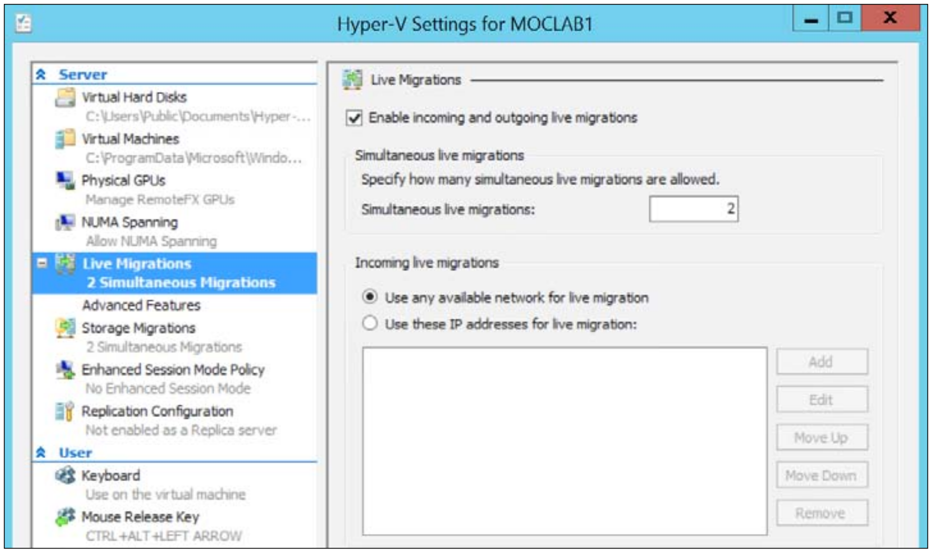
Configuring your Hyper-V host for live migration

In this topic, we will guide you through how to prepare your Hyper-V hosts for making live migration possible, as this feature is disabled by default.

Configuration done using the Hyper-V console

The steps are as follows:

1. From within the Hyper-V management console, select your Hyper-V host.
2. Click on **Hyper-V Settings** from within the **Actions** pane.
3. Click on **Live Migrations** from within the popup window.
4. Make the following selections:
    1. Select the **Enable incoming and outgoing live migrations** option.
    2. Specify the number for the **Simultaneous live migrations** option.
    3. Select the **Use any available network for live migration** option, or specify IP addresses (mostly used in scenarios where you have configured a dedicated live migration network using a dedicated NIC).



Hyper-V Settings for MOCLAB1

Live Migrations

☑ Enable incoming and outgoing live migrations

Simultaneous live migrations

Specify how many simultaneous live migrations are allowed.

Simultaneous live migrations: 2

Incoming live migrations

● Use any available network for live migration

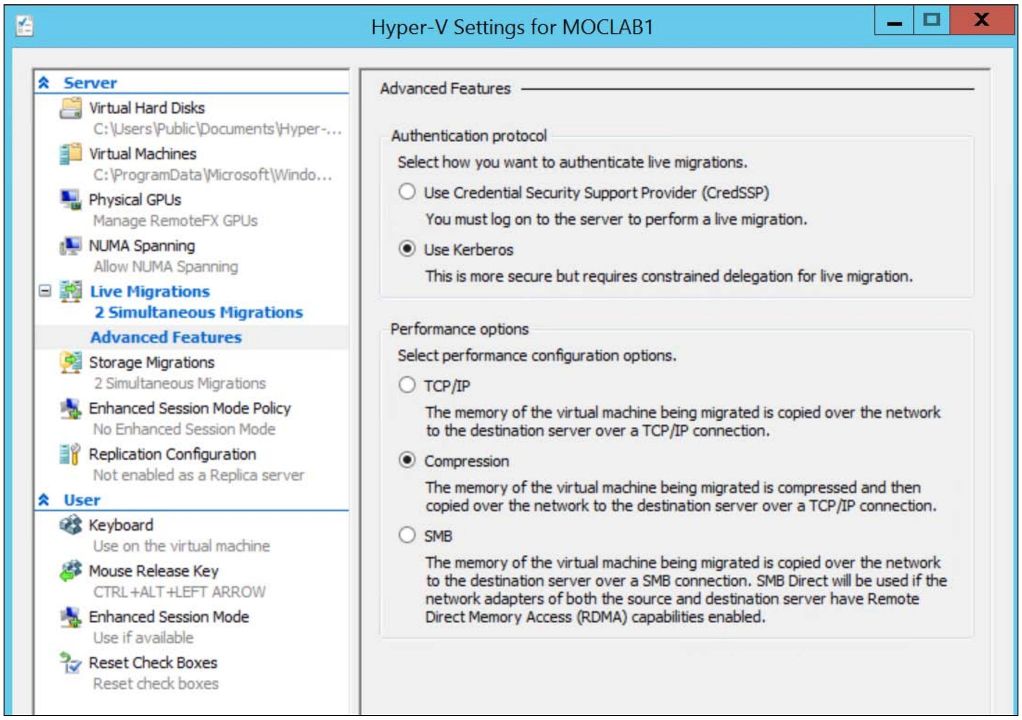
○ Use these IP addresses for live migration:

As mentioned in the first chapter, live migration is performed in a secure way, which relies on Active Directory (Kerberos) authentication (if the Hyper-V hosts are configured as domain members), or you can also make use of **Credential Security Support Provider** (**CredSSP**).

> You can learn more about the different authentication mechanisms later on in this chapter.

To configure the authentication, the steps are as follows:

1. To configure the authentication you want to use, select **Advanced Features** under the **Live Migrations** option (by clicking on the **+** sign), which will show the following:



Hyper-V Settings for MOCLAB1

Server
- Virtual Hard Disks
  C:\Users\Public\Documents\Hyper-...
- Virtual Machines
  C:\ProgramData\Microsoft\Windo...
- Physical GPUs
  Manage RemoteFX GPUs
- NUMA Spanning
  Allow NUMA Spanning
- Live Migrations
  2 Simultaneous Migrations
  - Advanced Features
- Storage Migrations
  2 Simultaneous Migrations
- Enhanced Session Mode Policy
  No Enhanced Session Mode
- Replication Configuration
  Not enabled as a Replica server

User
- Keyboard
  Use on the virtual machine
- Mouse Release Key
  CTRL+ALT+LEFT ARROW
- Enhanced Session Mode
  Use if available
- Reset Check Boxes
  Reset check boxes

Advanced Features

Authentication protocol

Select how you want to authenticate live migrations.

○ Use Credential Security Support Provider (CredSSP)

You must log on to the server to perform a live migration.

● Use Kerberos

This is more secure but requires constrained delegation for live migration.

Performance options

Select performance configuration options.

○ TCP/IP

The memory of the virtual machine being migrated is copied over the network to the destination server over a TCP/IP connection.

● Compression

The memory of the virtual machine being migrated is compressed and then copied over the network to the destination server over a TCP/IP connection.

○ SMB

The memory of the virtual machine being migrated is copied over the network to the destination server over a SMB connection. SMB Direct will be used if the network adapters of both the source and destination server have Remote Direct Memory Access (RDMA) capabilities enabled.

2. For this first scenario, we select the **Use Kerberos** option to use Kerberos as the authentication protocol.
3. Leave the default option (the **Compression** option) active to use compression on the memory synchronization for the moment.

That's all you need to do in order to enable live migration!

Configuration done using PowerShell

In about the same easy way, you can configure the Hyper-V host for live migration by using PowerShell. Key cmdlets here are `Enable-VMMigration` and `Set-VMHost`.

We will use PowerShell to configure our second host, `MOCLAB2`. The steps are as follows:

1. Open the PowerShell window using administrative rights.
2. Enter the following cmdlet to enable live migration:

   ```
   enable-VMMigration
   ```

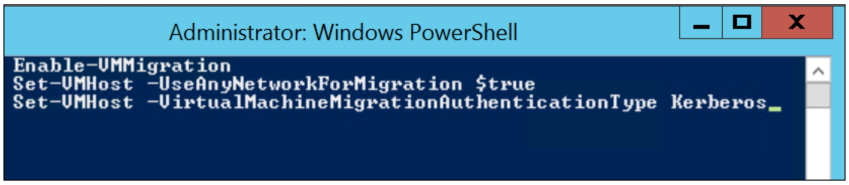
3. Specify the available network to be used for live migration using the following cmdlet:

   ```
   Set-VMHost -UseAnyNetworkForMigration $true
   ```

4. Specify the live migration authentication type to use (Kerberos) using the following cmdlet:

   ```
   Set-VMHost -VirtualMachineMigrationAuthenticationType Kerberos
   ```

These cmdlets are shown in the following screenshot:



```
Enable-VMMigration
Set-VMHost -UseAnyNetworkForMigration $true
Set-VMHost -VirtualMachineMigrationAuthenticationType Kerberos_
```

This concludes the configuration of live migration settings by using PowerShell. It's indeed no more than just these four steps to complete the configuration. That's the great side of PowerShell.

Performing the live migration

Once our Hyper-V hosts are enabled for live migration, it's a no-brainer to actually move a running VM from one host to another. As you can already imagine, this action can be performed from within the Hyper-V console as well as by using PowerShell.

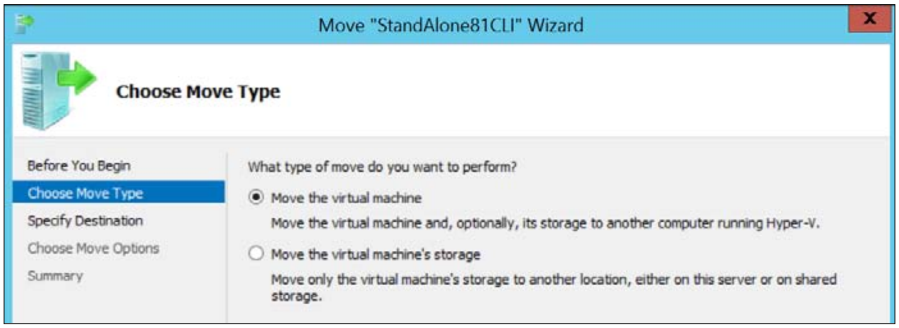
In this example we will move a VM called `StandAlone81CLI` from host `MOCLAB1` to `MOCLAB2`.

Here we go.

Executing live migration using the Hyper-V console

The steps are as follows:

1. Select the virtual machine from the Hyper-V source host, right-click on it, and choose **Move...**, which will open the **Move Wizard** window.
2. In the first step, select the **Move the virtual machine** option and click on **Next**:



Move "StandAlone81CLI" Wizard

Choose Move Type

What type of move do you want to perform?

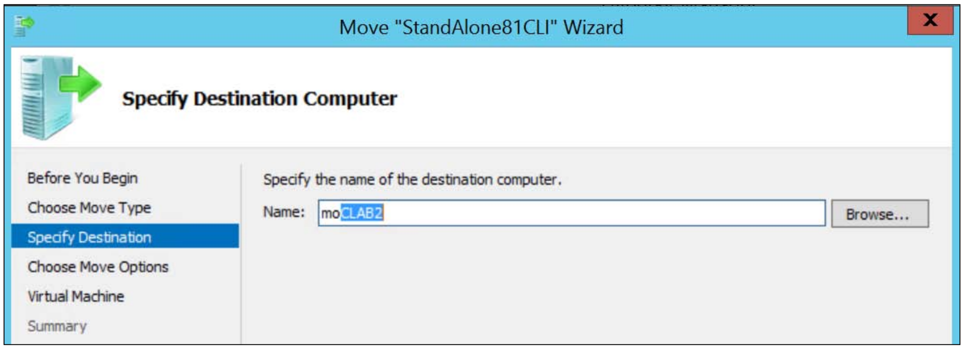
● Move the virtual machine

Move the virtual machine and, optionally, its storage to another computer running Hyper-V.

○ Move the virtual machine's storage

Move only the virtual machine's storage to another location, either on this server or on shared storage.

3. In the next step, enter the name of the target Hyper-V Host as `MOCLAB2` and click on **Next**:



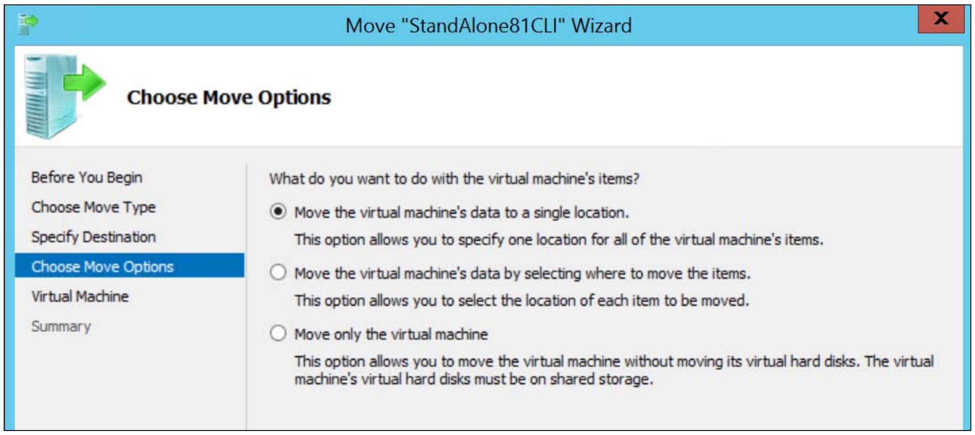
Move "StandAlone81CLI" Wizard

Specify Destination Computer

Specify the name of the destination computer.

Name: moCLAB2

4. In the **Choose Move Options** window, select the **Move the virtual machine's data to a single location** option, which is ok for this example. Otherwise, specify the option according to your situation, and click on **Next**:



What do you want to do with the virtual machine's items?

● Move the virtual machine's data to a single location.

This option allows you to specify one location for all of the virtual machine's items.

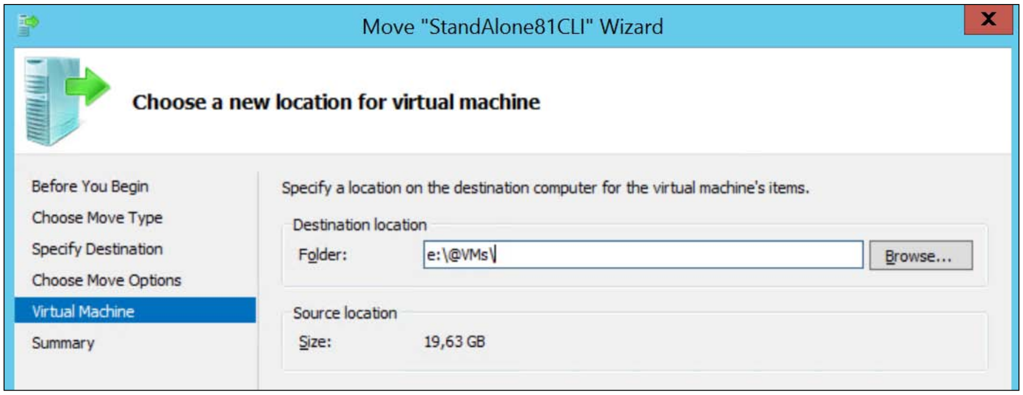
○ Move the virtual machine's data by selecting where to move the items.

This option allows you to select the location of each item to be moved.

○ Move only the virtual machine

This option allows you to move the virtual machine without moving its virtual hard disks. The virtual machine's virtual hard disks must be on shared storage.

5. In the next step, specify the physical location on the destination host where the files should be stored, and click on **Next**:



Specify a location on the destination computer for the virtual machine's items.

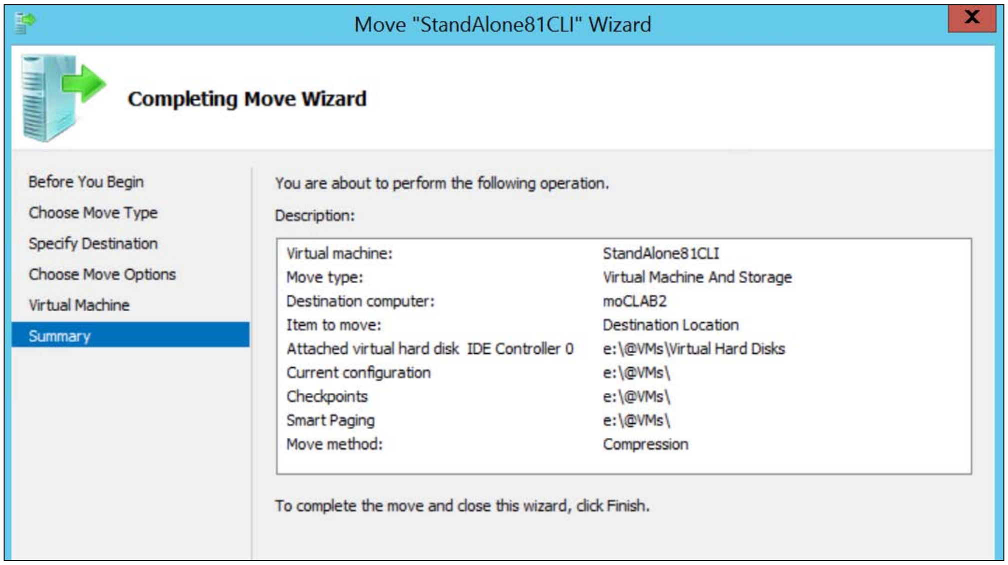
Destination location

Folder: e:\@VMs\

Source location

Size: 19,63 GB

6. Check the summarized overview of the options, and click on **Finish**:



You are about to perform the following operation.

Description:

Virtual machine:	StandAlone81CLI
Move type:	Virtual Machine And Storage
Destination computer:	moCLAB2
Item to move:	Destination Location
Attached virtual hard disk IDE Controller 0	e:\@VMs\Virtual Hard Disks
Current configuration	e:\@VMs\
Checkpoints	e:\@VMs\
Smart Paging	e:\@VMs\
Move method:	Compression

To complete the move and close this wizard, click Finish.

This will start the actual live migration of the virtual machine.

Executing live migration using PowerShell

Moving virtual machines in the described scenario is far from difficult. But the true power lies in automating the process, as, again, live migration is not a true high availability solution (in a sense that it does not move VMs automatically when something goes wrong on the source Hyper-V host). However, by using the great possibilities of PowerShell, one could build a script that does migrate VMs based on a certain scenario (such as for example, when a service goes down on the host, or prior to a scheduled reboot).

By using the following PowerShell cmdlet, a VM named `PDTSRV2` can be moved easily:

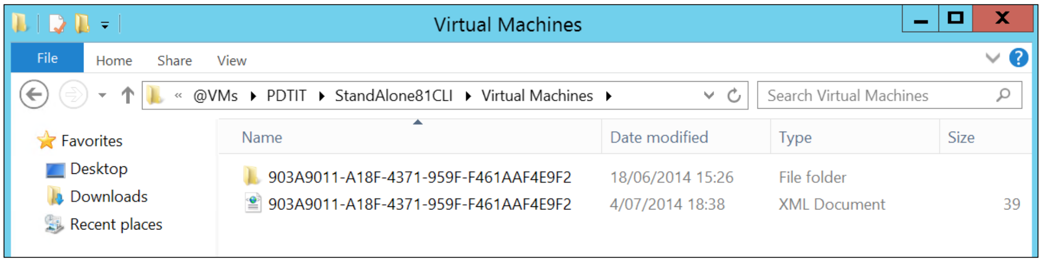
```
Move-VMpdtsrv2 -DestinationHostMOCLAB2 -DestinationStoragePath e:\@vms
```

Verifying the move operation

Let's verify some things that happen during the migration.

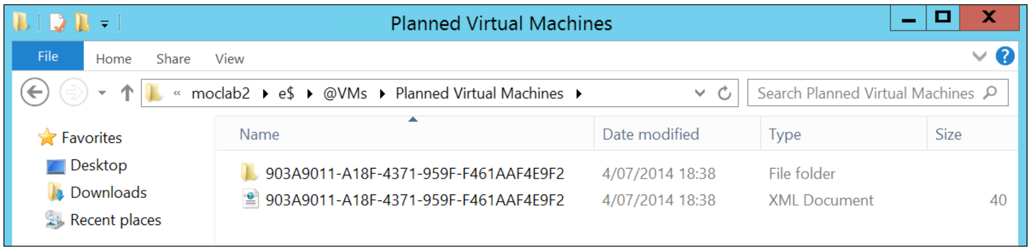
On the target Hyper-V host, browse through the target directory you specified earlier, and notice a temporary directory being created named as `Planned Virtual Machines`. In this directory, a new virtual machine configuration is stored, which has the exact same GUID as on the source Hyper-V host.

The source Hyper-V host is shown in the following screenshot:



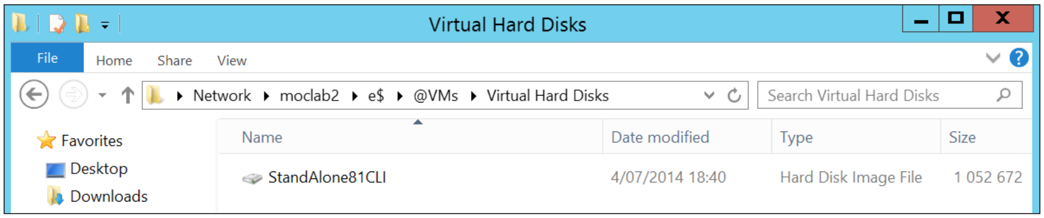
Name	Date modified	Type	Size
903A9011-A18F-4371-959F-F461AAF4E9F2	18/06/2014 15:26	File folder	
903A9011-A18F-4371-959F-F461AAF4E9F2	4/07/2014 18:38	XML Document	39

The target Hyper-V host is shown in the following screenshot:


Planned Virtual Machines
File
Home
Share
View
« moclab2 ▸ e$ ▸ @VMs ▸ Planned Virtual Machines ▸
Search Planned Virtual Machines
Favorites
Desktop
Downloads
Recent places

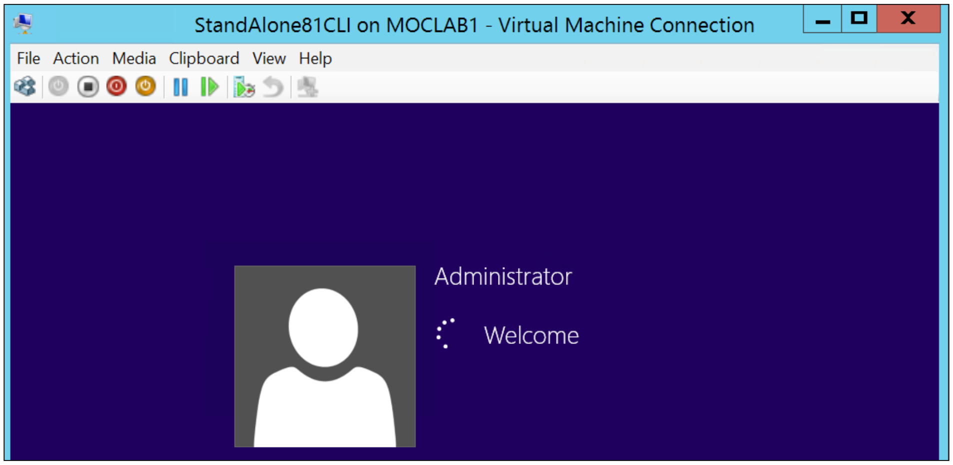
Name	Date modified	Type	Size
903A9011-A18F-4371-959F-F461AAF4E9F2	4/07/2014 18:38	File folder	
903A9011-A18F-4371-959F-F461AAF4E9F2	4/07/2014 18:38	XML Document	40

On the target Hyper-V host, browse through the target directory you specified earlier, and check out the `Virtual Hard Disks` folder. Here, you will notice that the VHD is being copied over from source to target:


Virtual Hard Disks
File
Home
Share
View
▸ Network ▸ moclab2 ▸ e$ ▸ @VMs ▸ Virtual Hard Disks
Search Virtual Hard Disks
Favorites
Desktop
Downloads

Name	Date modified	Type	Size
StandAlone81CLI	4/07/2014 18:40	Hard Disk Image File	1 052 672

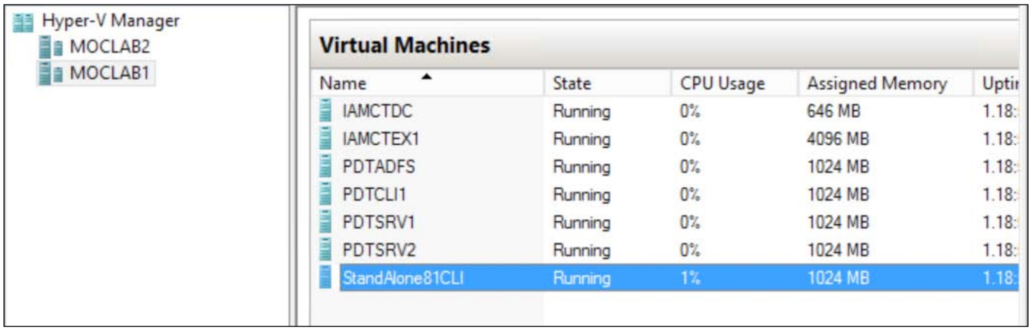
While you can't really verify this from going through the steps in this book, the Windows 8.1 client we are moving during this live migration, is actually still running. Let's try to log on to the client (which works perfectly fine, without having any performance issues), while the move process is active:


StandAlone81CLI on MOCLAB1 - Virtual Machine Connection
File Action Media Clipboard View Help
Administrator
Welcome

After logging on to the virtual machine, you could try a continuous ping to the client's gateway, which shows you there is actual network communication going on, and no downtime of the client is noticeable. Not even a ping gets lost!

Now let's do another verification from within the Hyper-V console itself. As you can see, the VM `StandAlone81CLI` is running on `MOCLAB1`—our Hyper-V source host. Once the move of the virtual machine is completed, it will show up as running on `MOCLAB2`.

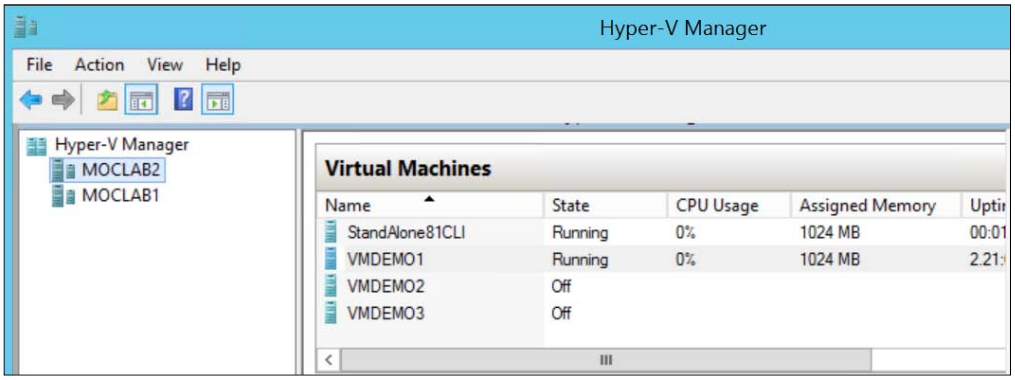
Before the move, the console looks like the following screenshot:



Virtual Machines

Name	State	CPU Usage	Assigned Memory	Uptir
IAMCTDC	Running	0%	646 MB	1.18:
IAMCTEX1	Running	0%	4096 MB	1.18:
PDTADFS	Running	0%	1024 MB	1.18:
PDTCLI1	Running	0%	1024 MB	1.18:
PDTSRV1	Running	0%	1024 MB	1.18:
PDTSRV2	Running	0%	1024 MB	1.18:
StandAlone81CLI	Running	1%	1024 MB	1.18:

After the move, the console shows that the virtual machine is running on `MOCLAB2`:



Virtual Machines

Name	State	CPU Usage	Assigned Memory	Uptir
StandAlone81CLI	Running	0%	1024 MB	00:01
VMDEMO1	Running	0%	1024 MB	2.21:
VMDEMO2	Off			
VMDEMO3	Off			

This concludes the live migration process. Wonderful technology, isn't it?

Live migration authentication protocol

One of the configuration steps in enabling live migration involved specifying the authentication protocol to be used. Here, you can choose between CredSSP or Kerberos. But what is the difference between these two? CredSSP is the default selection when enabling live migration, but in the GUI, it is mentioned that Kerberos is more secure. Hmm, makes you go crazy, right?

CredSSP

Credential Security Support Provider (CredSSP) means that there must be an active logged-on connection between the hosts, on which you want to run the live migration process. CredSSP is the default setting because it is so easy to configure. Actually, there is no configuration needed to make it work. The downside of using CredSSP is that it allows security credentials to be transferred to a single hop, such as for example from the source Hyper-V host to the target Hyper-V host, but no further. It also requires you to log on locally to the Hyper-V source host and start the live migration from there.

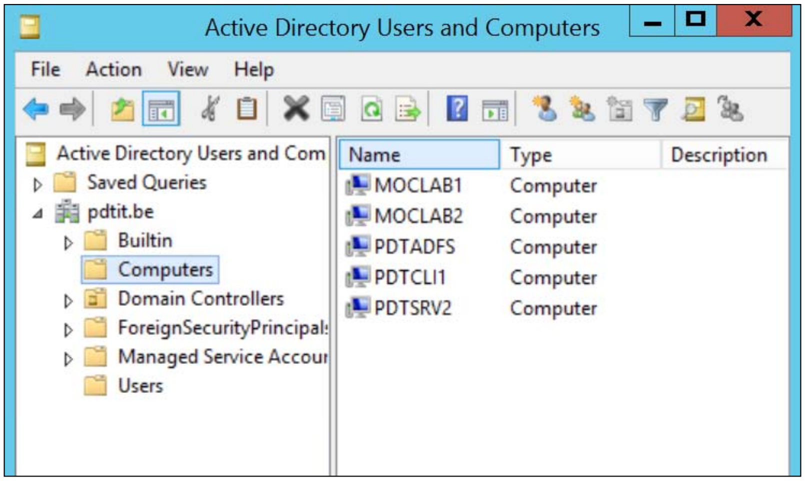
While all this sounds ok, it excludes doing remote management out of Windows 8 clients or for example, through remote PowerShell.

Kerberos

Kerberos is the default Active Directory-based authentication protocol. Its advantage is that it is not limited to the single hop credentials transfer. But this might also be considered as a security risk in your environment. The solution to that is configuring constrained delegation, actually limiting what the account credentials can be used for.

Enabling Kerberos constrained delegation can be done as follows:

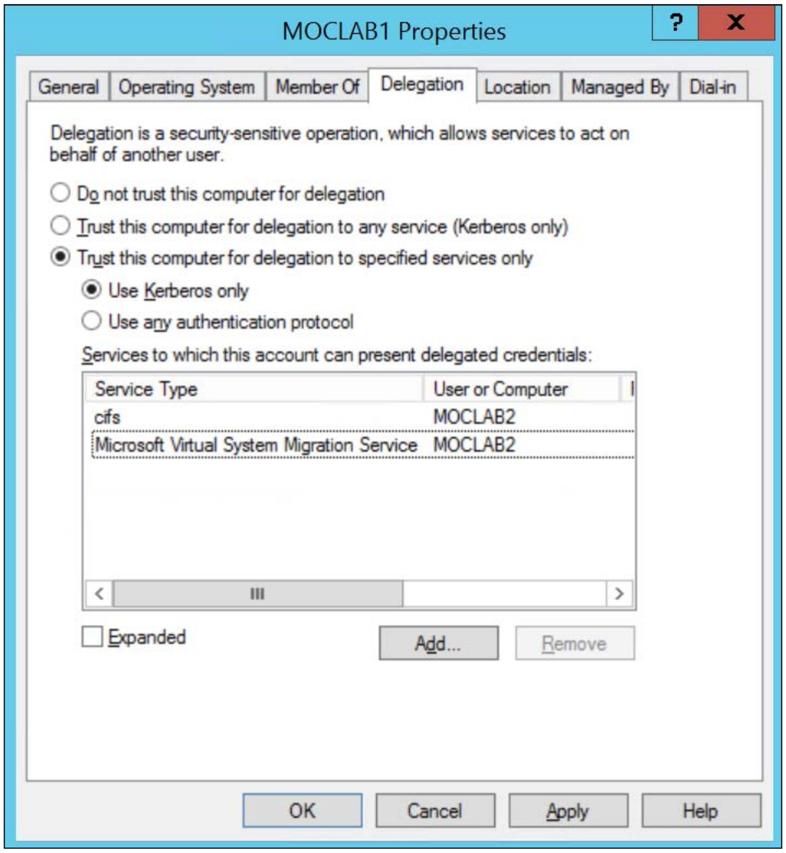
1. From any domain controller in your Active Directory domain, open the **Active Directory Users and Computers** (**ADUC**) console, and browse through the default container named `Computers`:



Name	Type	Description
MOCLAB1	Computer	
MOCLAB2	Computer	
PDTADFS	Computer	
PDTCLI1	Computer	
PDTSRV2	Computer	

2. Select the computer on which you want to configure constrained delegation (your Hyper-V host server), right-click on it and select **Properties**.
3. Go to the **Delegation** tab.
4. Select the **Trust this computer for delegation to specified services only** option and make sure you select the **Use Kerberos Only** option underneath.
5. In the **Services** box, click on **Add** and select the Hyper-V target host computer. From the list of services that show up, select the following:
    - **cifs** (user for file sharing access)
    - **Microsoft Virtual System Migration Service**

These settings are shown in the following screenshot:



Delegation is a security-sensitive operation, which allows services to act on behalf of another user.

○ Do not trust this computer for delegation

○ Trust this computer for delegation to any service (Kerberos only)

● Trust this computer for delegation to specified services only

● Use Kerberos only

○ Use any authentication protocol

Services to which this account can present delegated credentials:

Service Type	User or Computer
cifs	MOCLAB2
Microsoft Virtual System Migration Service	MOCLAB2

□ Expanded

Perform the same configuration steps from the Hyper-V target host to enable constrained delegation to the Hyper-V source host, in order to make live migration possible in the other direction as well. (Think of a situation where you want to move your VMs back.)

From what you have learned in this topic, using CredSSP is a lot easier, but it has some limitations, of which the single hop credential transfer is the main one. Using Kerberos shouldn't be too difficult either, as your Hyper-V hosts are already Active Directory domain members. Just consider configuring constrained delegation to elevate security.

Using SMB 3.0 shared storage for Hyper-V

In the previous section, we mainly assumed the Hyper-V hosts were using local storage (DAS—Direct Attached Storage). Where this is a perfect solution in a Hyper-V environment, a lot of enhancements were brought into Windows Server 2012 and 2012 R2 by using SMB 3.0 shares for storing Hyper-V virtual machines.

By using SMB 3.0 file shares, the following two scenarios are possible:

- **VM storage location for standalone Hyper-V servers**: This doesn't provide high availability.
- **VM storage location for Hyper-V failover cluster**: The storage location is shared between all hosts in the cluster, thus providing full high availability.

In this topic, we continue focusing on the first scenario, which is using Hyper-V hosts without providing high availability. (Hyper-V failover clustering is covered in the next chapter.)

Considerations for using SMB 3.0 in a Hyper-V environment

Before deciding on using SMB 3.0 shares as your virtual machine storage, take the following considerations/requirements into account:

- Hyper-V hosts and SMB 3.0 file share servers need to be part of the Active Directory domain
- The SMB 3.0 file share server(s) must run Windows Server 2012 or 2012 R2, or some other operating system that runs the SMB 3.0 file sharing protocol
- Hyper-V hosts cannot be used themselves as SMB 3.0 file share storage endpoints

> While it is recommended to use SMB 3.0 shared storage only, as that gives you the best performance, Hyper-V itself actually works with previous SMB shared storage locations too. It won't even complain about it. (It is only triggered as an alert in the Hyper-V Best Practices Analyzer.)

Installing and configuring an SMB 3.0 file server

To install and configure Windows Server 2012 or 2012 R2 as an SMB 3.0 file server for Hyper-V storage, go through the following steps (as this shouldn't be that hard, we decided to exclude the screenshots):

1. Log on to your Windows Server with administrative rights, and open **Server Manager**.
2. From the home page of **Server Manager**, chose **Add Roles and features**.
3. On the next page, click on **Role-based or feature-based installation** and then click on **Next**.
4. On the **Select server roles** step, click on **File and Storage Services** and then click on **Next**.
5. Click on **Install** in the **Confirmation** page.

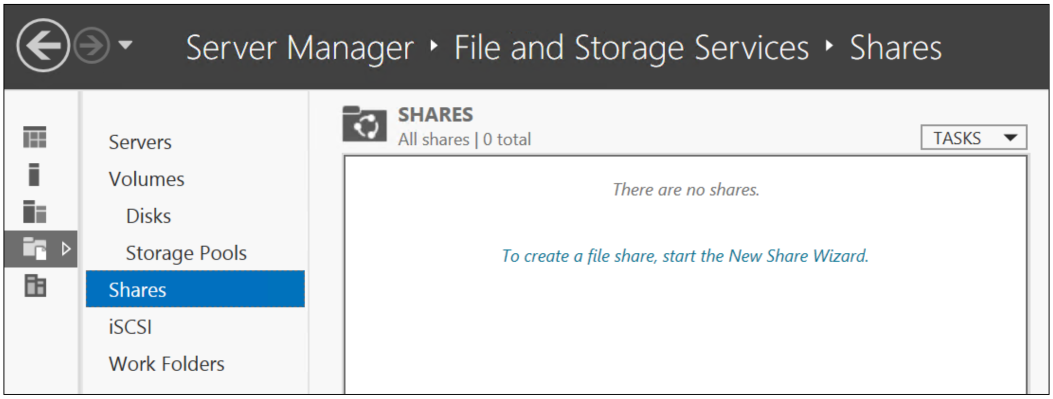
This will install the File and Storage Services role on your Windows Server machine.

If you prefer installing the server role by using PowerShell, run the following cmdlet:

```
Install-WindowsFeature File-Services, FS-FileServer
```

Now the File and Storage Services role is installed, the next step involves configuring an SMB 3.0 file share, which will be used to store the virtual machine data. This is configured as follows:

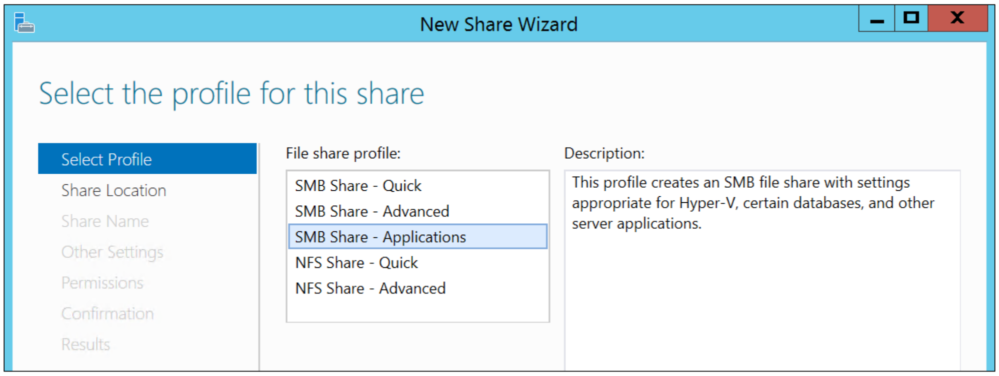
1. From within the **Server Manager** console, select the **File and Storage Services** icon in the left pane, and then click on **Shares**:



There are no shares.

To create a file share, start the New Share Wizard.

2. Click on **Tasks** and then click on **New Share**, which will launch the **New Share Wizard** window:



New Share Wizard

Select the profile for this share

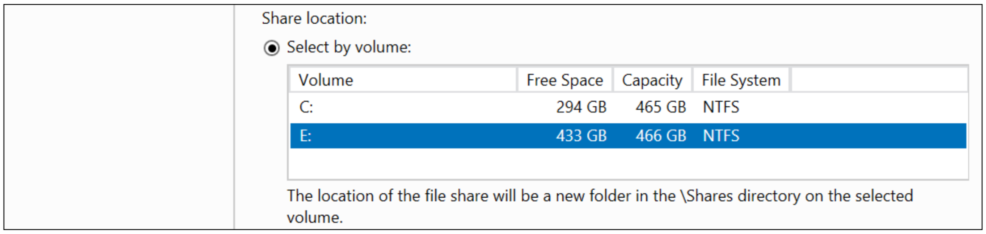
File share profile:

- SMB Share - Quick
- SMB Share - Advanced
- SMB Share - Applications
- NFS Share - Quick
- NFS Share - Advanced

Description:

This profile creates an SMB file share with settings appropriate for Hyper-V, certain databases, and other server applications.

3. Select **SMB Share – Applications** as the **File share profile** option and click on **Next**.
4. Select your **Share location**. In my example, I select the E-drive on my file server, as that one has the most disk space left:



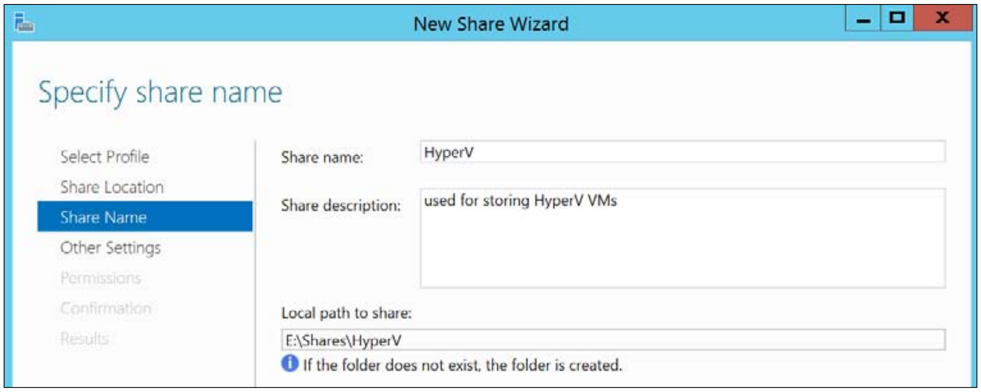
Share location:

● Select by volume:

Volume	Free Space	Capacity	File System
C:	294 GB	465 GB	NTFS
E:	433 GB	466 GB	NTFS

The location of the file share will be a new folder in the \Shares directory on the selected volume.

5. In the next step, provide a **Share name** and click on **Next**:



New Share Wizard

Specify share name

Share name: HyperV

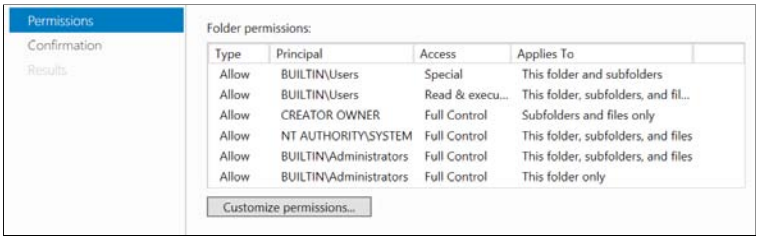
Share description: used for storing HyperV VMs

Local path to share:

E:\Shares\HyperV

If the folder does not exist, the folder is created.

6. Click on **Next** in the **Other Settings** page, without make any selections.
7. In the **Permissions** page, click on the **Customize permissions** button:



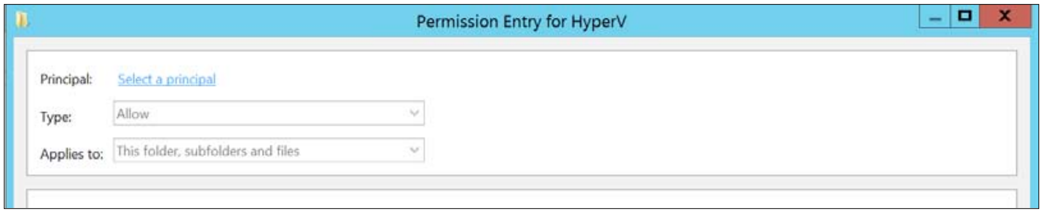
Permissions
Confirmation
Results

Folder permissions:

Type	Principal	Access	Applies To
Allow	BUILTIN\Users	Special	This folder and subfolders
Allow	BUILTIN\Users	Read & execu...	This folder, subfolders, and fil...
Allow	CREATOR OWNER	Full Control	Subfolders and files only
Allow	NT AUTHORITY\SYSTEM	Full Control	This folder, subfolders, and files
Allow	BUILTIN\Administrators	Full Control	This folder, subfolders, and files
Allow	BUILTIN\Administrators	Full Control	This folder only

Customize permissions...

8. In the **Permission Entry** window, click on **Select a principal**:



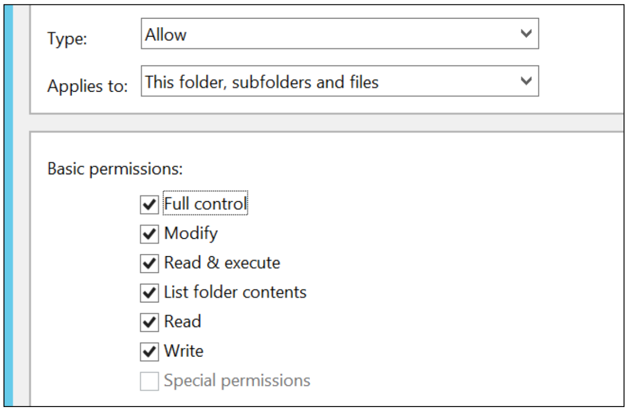
Permission Entry for HyperV

Principal: Select a principal

Type: Allow

Applies to: This folder, subfolders and files

9. In the pop-up window, change the object type to include computers; after that, enter the name of your Hyper-V host that you want to get connected to the file share. Update the permissions to **Full control**:



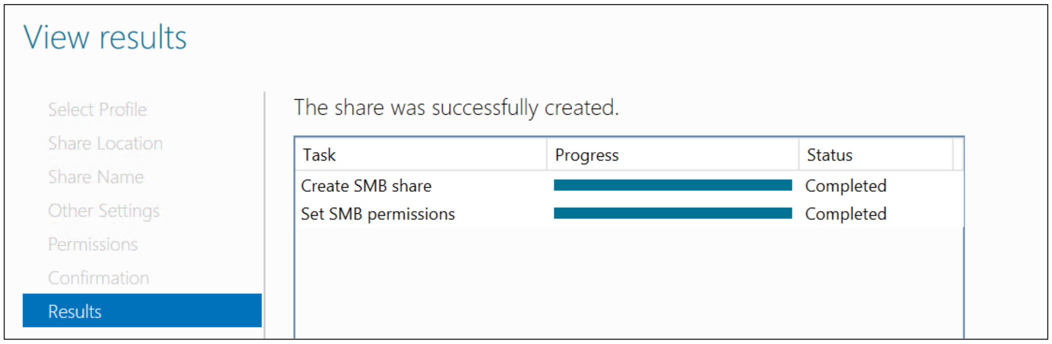
Type: Allow

Applies to: This folder, subfolders and files

Basic permissions:

☑ Full control
☑ Modify
☑ Read & execute
☑ List folder contents
☑ Read
☑ Write
□ Special permissions

10. In the final step, the SMB 3.0 shared folder will be created:



View results

The share was successfully created.

Task	Progress	Status
Create SMB share		Completed
Set SMB permissions		Completed

Repeat the preceding steps (from step 7) to include other Hyper-V hosts that need to be connected to the SMB 3.0 file share as well.

This concludes the configuration of an SMB 3.0 file share for Hyper-V storage.

The same result can be achieved from within PowerShell, by launching the following cmdlet:

```
New-SmbShare -Name HyperV -Path E:\HyperV -FullAccess, PDTIT\MOCLAB1$,
PDTIT\MOCLAB2$
```

Use the following cmdlet to configure the full access rights on NTFS level:

```
Set-SmbPathAcl –Name VMS1
```

If you are looking for an even more in-depth knowledge on SMB 3.0 in a Hyper-V environment, have a look at some of the following blog posts from Aidin Finn, Hyper-V MVP and fellow technical writer for `http://www.petri.com`:

- *A Converged Networks Design For SMB 3.0 Storage & SMB Direct Live Migration*, available at `http://www.aidanfinn.com/?p=14879`.
- *Flow of Storage Traffic In Hyper-V Over SMB 3.0 to WS 2012 R2 SOFS*, available at `http://www.aidanfinn.com/?p=15785`.
- *Testing SMB Live Migration on WS 2012 R2 Hyper-V*, available at `http://www.aidanfinn.com/?p=15291`.

Storage migration

By using storage migration, you can move the virtual hard disks and data that are being used by virtual machines, to a Hyper-V host that is using a different physical storage. It is also possible to move between different storage solutions connected to the same Hyper-V hosts, again, without any downtime to the virtual machine.

In short, storage migration works as follows:

1. Before performing the actual move process, all read/write operations happen on the source virtual hard disk, running on the source storage solution
2. When the storage migration process itself starts, all read/write operations remain on the source virtual hard disk, while the data is already being copied to the destination storage solution
3. Once the initial copy process is completed, write operations for the virtual hard disks are mirrored between the source and target virtual hard disks
4. Once the source and target virtual hard disks are completely in sync, the virtual machine itself switches over as well, and it will start using the virtual hard disk on the destination at that time
5. In the last part of the process, the source virtual hard disk is removed

Storage migration can move the following components:

- Virtual hard disks
- Virtual machine configuration files
- Checkpoints
- Smart paging file of the virtual machine

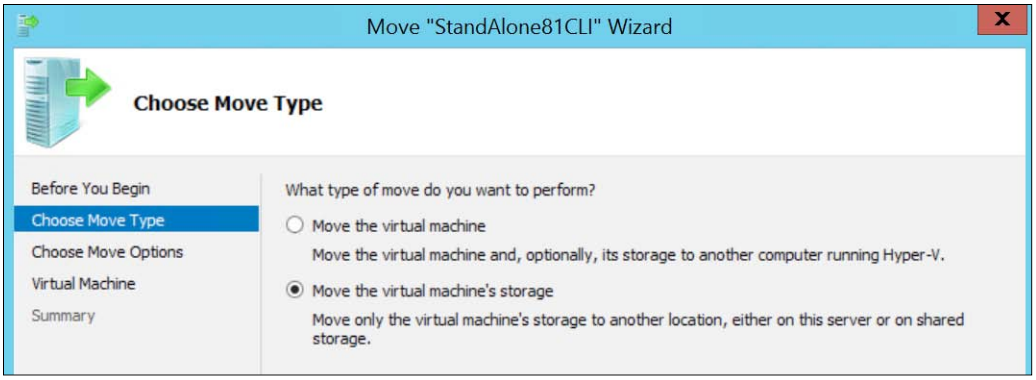
Storage migration does not work in the following scenarios:

- Migrating storage when the virtual machine is using DAS (Direct Attached Storage) directly on the physical disk layer (pass-through disks)
- Migrating data from a Fibre Channel SAN

Moving storage between Hyper-V hosts

By using the following procedure, virtual machine storage data can be moved from one Hyper-V host to another, in a standalone Hyper-V setup:

1. From within the Hyper-V management console, select the virtual machine you want to move the storage location for; right-click on it and select **Move**.
2. This launches the **Move Wizard**; click on **Next** on the first page of the wizard.
3. In the **Choose Move Type** step, select **Move the virtual machine's storage** option and click on **Next**:



Move "StandAlone81CLI" Wizard

Choose Move Type

What type of move do you want to perform?

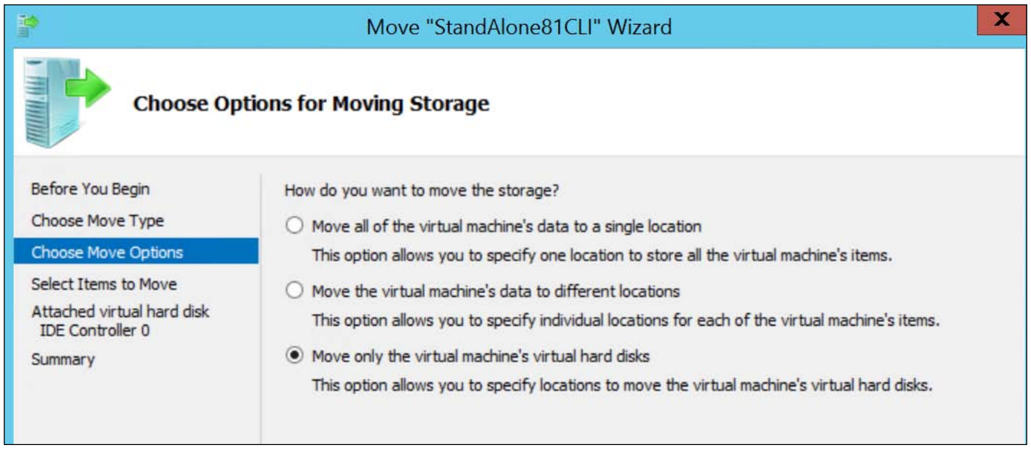
○ Move the virtual machine

Move the virtual machine and, optionally, its storage to another computer running Hyper-V.

● Move the virtual machine's storage

Move only the virtual machine's storage to another location, either on this server or on shared storage.

4. In the **Choose Options for Moving Storage** step, select the **Move only the virtual machine's virtual hard disks** option and click on **Next**:



Move "StandAlone81CLI" Wizard

Choose Options for Moving Storage

How do you want to move the storage?

○ Move all of the virtual machine's data to a single location

This option allows you to specify one location to store all the virtual machine's items.

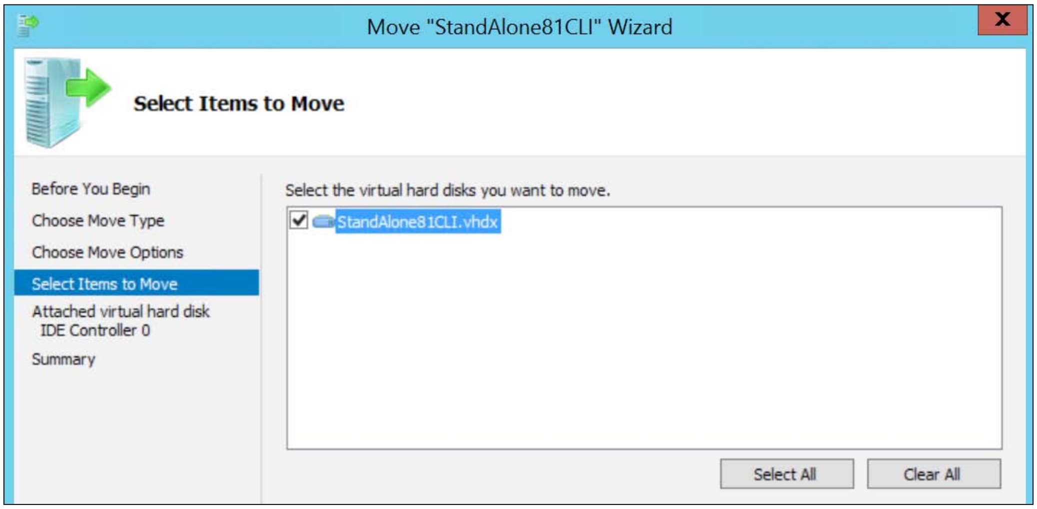
○ Move the virtual machine's data to different locations

This option allows you to specify individual locations for each of the virtual machine's items.

● Move only the virtual machine's virtual hard disks

This option allows you to specify locations to move the virtual machine's virtual hard disks.

5. In the **Select Items to Move** step, select the virtual machine's VHD(s) files that you want to move to the new storage location and click on **Next**:



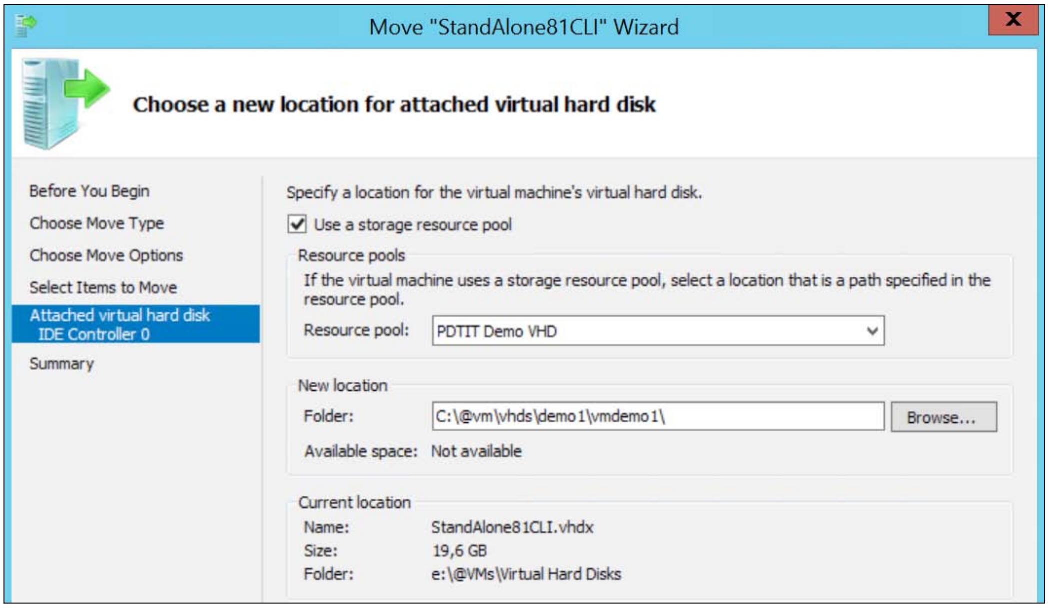
Move "StandAlone81CLI" Wizard

Select Items to Move

Select the virtual hard disks you want to move.

☑ StandAlone81CLI.vhdx

6. In the next step, select the new storage location. In our example, I want to show you two different options; one is to use the resource pool location we specified before (see *Chapter 2*, *Unwrapping Hyper-V 2012 R2 Components*); the second option is the exact location of the shared folder:



Move "StandAlone81CLI" Wizard

Choose a new location for attached virtual hard disk

Specify a location for the virtual machine's virtual hard disk.

☑ Use a storage resource pool

Resource pools

If the virtual machine uses a storage resource pool, select a location that is a path specified in the resource pool.

Resource pool: PDTIT Demo VHD

New location

Folder: C:\@vm\vhds\demo1\vmdemo1\

Available space: Not available

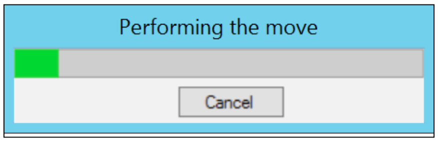
Current location

Name: StandAlone81CLI.vhdx

Size: 19,6 GB

Folder: e:\@VMs\Virtual Hard Disks

7. Clicking on **Next** will give you a summarized view of the selections made; when you click on **Finish**, the actual move process will start:



This concludes the storage move process for a virtual machine between different Hyper-V hosts.

The same can be achieved by using PowerShell, as follows:

```
Move-VMStorage -VMNameStandAlone81CLI -DestinationStoragePath
\\FileServer\@VM\VHDS\demo1\vmdemo1\
```

Hyper-V Replica

In this last section of this chapter, we are going to talk about Hyper-V Replica.

From what you learned in *Chapter 1, Introduction to Microsoft Server 2012 R2 Hyper-V*, Hyper-V Replica is a new feature of Hyper-V in Windows Server 2012 and 2012 R2, which allows you to replicate virtual machines between Hyper-V hosts. And these Hyper-V hosts don't need to use shared storage.

> In Hyper-V on Server 2012 R2, Hyper-V Replica even allows for synchronization from the first Hyper-V Replica server to a third Hyper-V Replica server, providing two copies of a virtual machine.

The main difference between the previous features described in this chapter and Hyper-V Replica is that by using Hyper-V Replica, you are creating a copy of a live virtual machine from one Hyper-V host, which gets copied to a second Hyper-V host. While the copy/replication process runs without any downtime for the live running virtual machine, it will be replicated to an offline (non-running) state on the destination. With live migration or storage migration, the virtual machine is moved.

By leveraging the replication aspect of Hyper-V Replica, it becomes a true disaster recovery scenario solution. First of all, it allows for multiple synchronizations on a timely basis (from 15 minutes up to every 30 seconds!), and second, it allows for failovers. These failovers are a manual process, but under normal conditions, there isn't any loss of data. In the worst case scenario, the data loss is limited to the last successful replication cycle of the virtual machine (5 minutes is the default setting here).

Another interesting aspect of Hyper-V Replica is that it allows for data replication between remote sites. As long as you have networking connectivity between the different data center locations, Hyper-V Replica can synchronize the virtual machine state from the source data center to the target data center, over a WAN link. To make it even more interesting, the remote location can be a partner company's data center or a hosting provider, mainly because there is no Active Directory dependency between the Hyper-V Replica servers.

All these characteristics often make Hyper-V Replica a more interesting scenario than using Hyper-V failover clustering.

Hyper-V Replica considerations and requirements

Before you implement Hyper-V Replica, make note of the following considerations and requirements on your overall infrastructure:

- You need Hyper-V on Windows Server 2012 or R2 to configure the Hyper-V Replica feature. The hosts can be standalone servers or failover cluster members.
- Allow for enough memory, CPU, and storage resources on both Hyper-V hosts.
- Make sure you have a stable and active network connection between the Hyper-V hosts; this can be through a LAN or a WAN connection.
- Make the necessary firewall configurations to allow replication traffic to pass between the Hyper-V servers. This requires HTTP listener and HTTPS listener ports to be opened up.
- Decide on what authentication protocol you want to use between the Hyper-V hosts; this can be certificate-based (internally generated CA or public) or by using Kerberos (requires the Hyper-V hosts to be within the same Active Directory domain).

- Hyper-V hosts can be domain members or be configured in a workgroup.
- Replication can be configured between any authenticated servers, but it is also possible to specify replication between specific servers only. The key thing to consider when performing replication with a hosting company's environment is to configure trust groups. A trust group is a security configuration, limiting access between members of the trust group; this blocks access from one company's virtual machines to another company's virtual machines.
- When using Kerberos / Active Directory authentication, by default the replication passes over HTTP (non-encrypted); when using certificate-based authentication, the replication passes over HTTPS (encrypted) by default.
- After you configure the Hyper-V host machines for Hyper-V Replica, you have to enable the replication mechanism on a per virtual machine level.

Now as you know the key considerations, let's move on with the Hyper-V Replica configuration itself.

Enabling Hyper-V Replica on your Hyper-V host

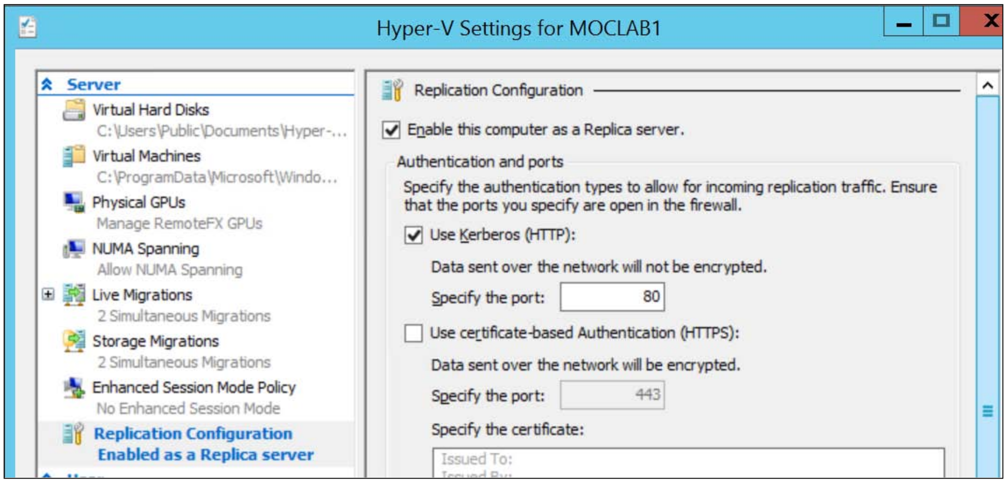
Enabling a Hyper-V host for Hyper-V Replica is actually a pretty easy process, provided that some assumptions such as domain membership, default storage location, and others are already decided on. Let's continue with the same Hyper-V host setup as before, `MOCLAB1` and `MOCLAB2` being members of the same Active Directory domain, and belonging to the same IP subnet.

> Again, if you make sure all points we listed in the considerations/requirements list are covered, enabling Hyper-V Replica should be easy for you.

The steps are as follows:

1. From within the Hyper-V Management console, select your Hyper-V host and then select **Hyper-V Settings** from the **Actions** pane.
2. In the left-hand side column, select the **Replication Configuration** option.
3. Make the following changes in the configuration, as shown in the screenshot:
    1. Check the **Enable this computer as a Replica server** option.
    2. Check the **Use Kerberos (HTTP)** option as authentication.

4. Leave the default port.


Hyper-V Settings for MOCLAB1
Server
Virtual Hard Disks
C:\Users\Public\Documents\Hyper-...
Virtual Machines
C:\ProgramData\Microsoft\Windo...
Physical GPUs
Manage RemoteFX GPUs
NUMA Spanning
Allow NUMA Spanning
Live Migrations
2 Simultaneous Migrations
Storage Migrations
2 Simultaneous Migrations
Enhanced Session Mode Policy
No Enhanced Session Mode
Replication Configuration
Enabled as a Replica server

Replication Configuration

☑ Enable this computer as a Replica server.

Authentication and ports

Specify the authentication types to allow for incoming replication traffic. Ensure that the ports you specify are open in the firewall.

☑ Use Kerberos (HTTP):

Data sent over the network will not be encrypted.

Specify the port: 80

☐ Use certificate-based Authentication (HTTPS):

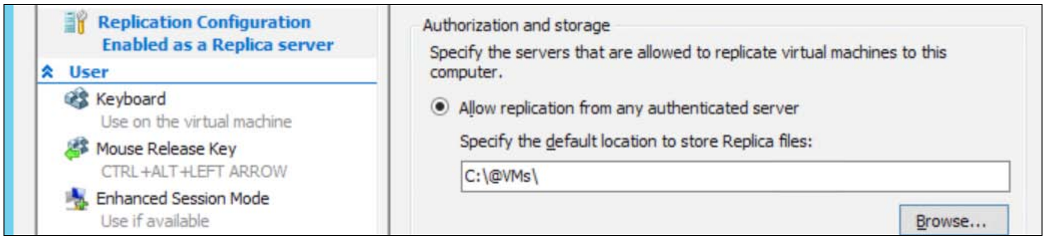
Data sent over the network will be encrypted.

Specify the port: 443

Specify the certificate:

Issued To:

5. In the **Authorization and storage** section, select the **Allow replication from any authenticated server** option, and update the default storage location for Replica files per your situation:



Authorization and storage

Specify the servers that are allowed to replicate virtual machines to this computer.

● Allow replication from any authenticated server

Specify the default location to store Replica files:

C:\@VMs\

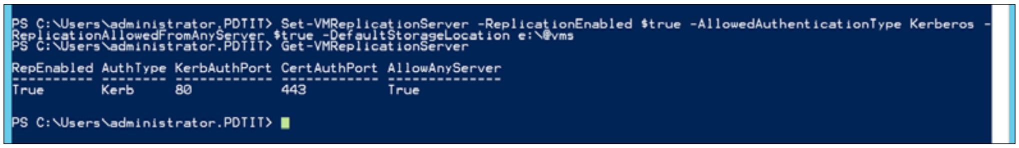
6. Confirm the informational message about the required firewall ports to be configured.

That's all that needs to be done to enable a Hyper-V server for Hyper-V Replica. You should repeat the same steps on the other Hyper-V Replica machine(s) you want to use.

One can achieve the same by using PowerShell as follows:

```
Set-VMReplicationServer -ServernameMOCLAB2 -ReplicationEnabled $True -
AllowedAuthenticationType Kerberos -ReplicationAllowedFromAnyServer $true
-DefaultStorageLocation e:\@VMS\Replica
```

The output will be the following:

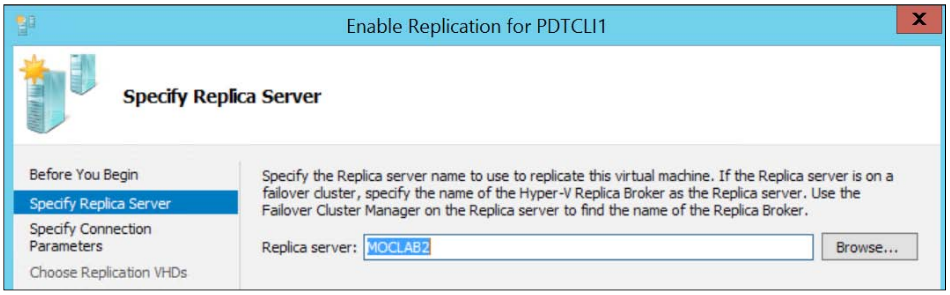


```
PS C:\Users\administrator.PDTIT> Set-VMReplicationServer -ReplicationEnabled $true -AllowedAuthenticationType Kerberos -
ReplicationAllowedFromAnyServer $true -DefaultStorageLocation e:\@vms
PS C:\Users\administrator.PDTIT> Get-VMReplicationServer

RepEnabled AuthType KerbAuthPort CertAuthPort AllowAnyServer
---------- -------- ------------ ------------ --------------
True       Kerb     80           443          True

PS C:\Users\administrator.PDTIT>
```

Replicating a virtual machine

With the Hyper-V Replica feature enabled and configured on both our Hyper-V hosts, it's about time we try to replicate one of our virtual machines. This is achieved as follows:

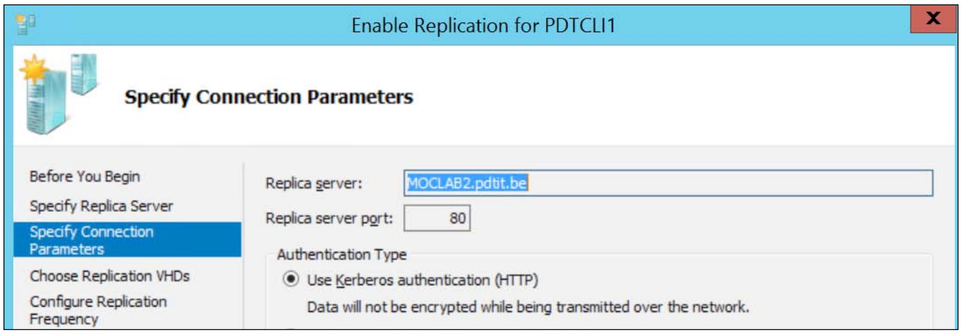
1. Select any virtual machine you want to replicate; right-click on it and click on **Enable Replication…**. This starts the Hyper-V Replica configuration wizard.
2. In the first step, specify the Hyper-V Replica target server (in our example lab, this is MOCLAB2):



Specify the Replica server name to use to replicate this virtual machine. If the Replica server is on a failover cluster, specify the name of the Hyper-V Replica Broker as the Replica server. Use the Failover Cluster Manager on the Replica server to find the name of the Replica Broker.

Replica server: MOCLAB2

3. In the **Specify Connection Parameters** step, verify the settings (which are based on the default settings we specified on the Hyper-V host level before, when enabling the host as Hyper-V Replica server):



Enable Replication for PDTCLI1
Specify Connection Parameters
Before You Begin
Specify Replica Server
Specify Connection Parameters
Choose Replication VHDs
Configure Replication Frequency
Authentication Type

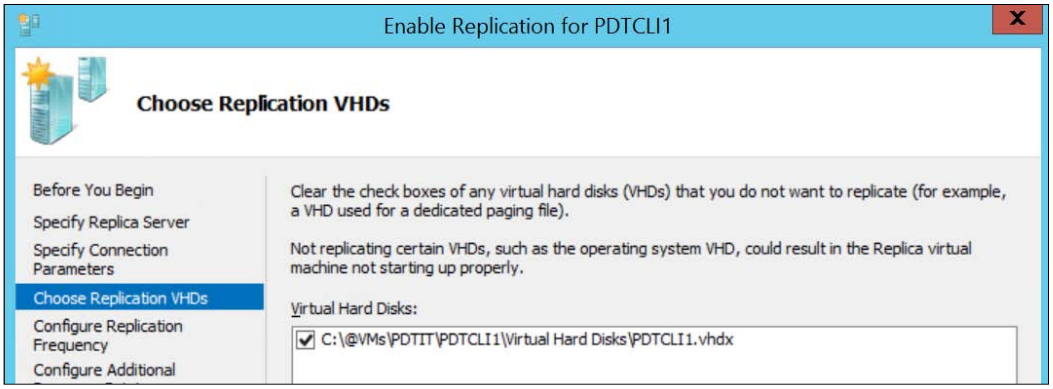
Replica server: MOCLAB2.pdtit.be

Replica server port: 80

● Use Kerberos authentication (HTTP)

Data will not be encrypted while being transmitted over the network.

4. In the next step, specify the virtual machine's VHD files that you want to replicate between the hosts:



Enable Replication for PDTCLI1

Choose Replication VHDs

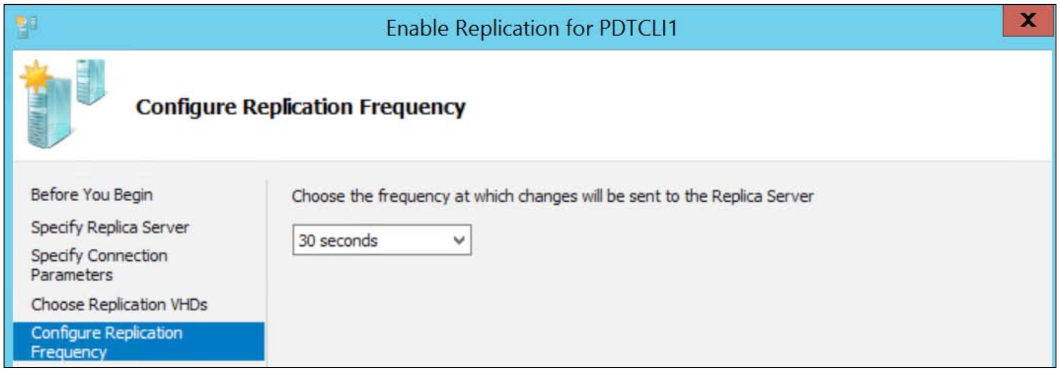
Clear the check boxes of any virtual hard disks (VHDs) that you do not want to replicate (for example, a VHD used for a dedicated paging file).

Not replicating certain VHDs, such as the operating system VHD, could result in the Replica virtual machine not starting up properly.

Virtual Hard Disks:

☑ C:\@VMs\PDTIT\PDTCLI1\Virtual Hard Disks\PDTCLI1.vhdx

5. In the following step, specify the replication frequency (5 minutes, 15 minutes, or as low as 30 seconds):



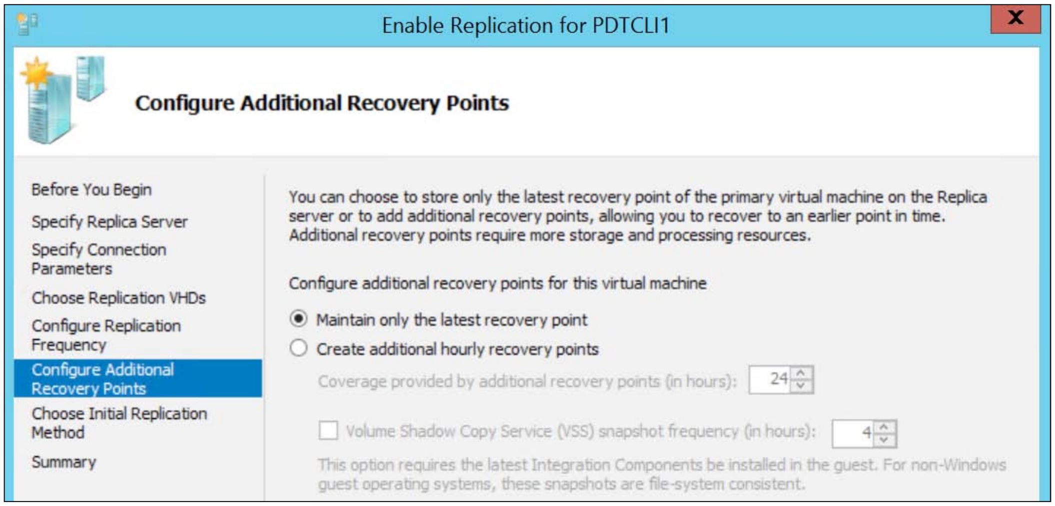
Enable Replication for PDTCLI1

Configure Replication Frequency

Choose the frequency at which changes will be sent to the Replica Server

30 seconds

6. In the **Configure Additional Recovery Points** step, make the appropriate selection based on your environment. The default setting is to check the **Maintain only the latest recovery point** option; however, it is also possible to configure multiple recovery points on the target server. If something goes wrong with the replicated virtual machine, you have the option to restore from a previous recovery point:


Enable Replication for PDTCLI1
Configure Additional Recovery Points
Before You Begin
Specify Replica Server
Specify Connection Parameters
Choose Replication VHDs
Configure Replication Frequency
Configure Additional Recovery Points
Choose Initial Replication Method
Summary

You can choose to store only the latest recovery point of the primary virtual machine on the Replica server or to add additional recovery points, allowing you to recover to an earlier point in time. Additional recovery points require more storage and processing resources.

Configure additional recovery points for this virtual machine

● Maintain only the latest recovery point

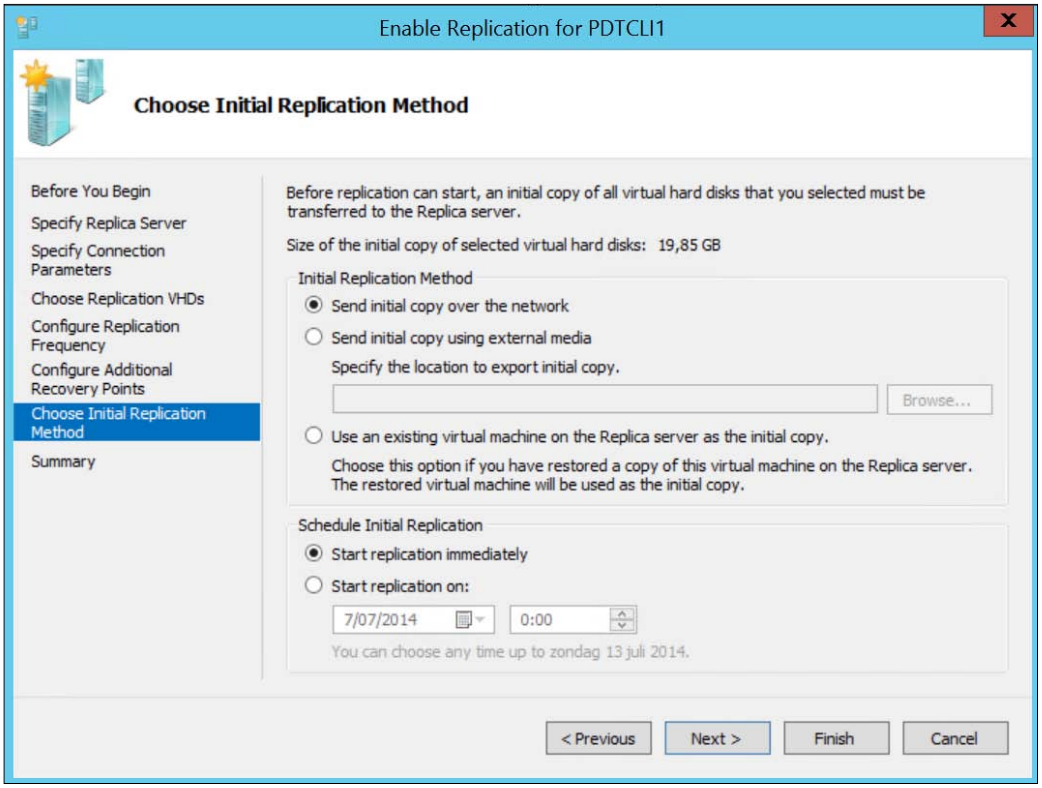
○ Create additional hourly recovery points

Coverage provided by additional recovery points (in hours): 24

□ Volume Shadow Copy Service (VSS) snapshot frequency (in hours): 4

This option requires the latest Integration Components be installed in the guest. For non-Windows guest operating systems, these snapshots are file-system consistent.

7. In the pre-final step, we accept the defaults to send the initial copy over the network and start replication immediately. Note that Hyper-V Replica allows you to start an initial replication from an offline media. (For example, you could take a backup of your VHD files to a USB disk, ship the disk to the remote location, perform a restore of the VHD files, and initiate a final replication to be in sync.)



Enable Replication for PDTCLI1

Choose Initial Replication Method

Before You Begin
Specify Replica Server
Specify Connection Parameters
Choose Replication VHDs
Configure Replication Frequency
Configure Additional Recovery Points
Choose Initial Replication Method
Summary

Before replication can start, an initial copy of all virtual hard disks that you selected must be transferred to the Replica server.

Size of the initial copy of selected virtual hard disks: 19,85 GB

Initial Replication Method

● Send initial copy over the network

○ Send initial copy using external media

Specify the location to export initial copy.

○ Use an existing virtual machine on the Replica server as the initial copy.

Choose this option if you have restored a copy of this virtual machine on the Replica server. The restored virtual machine will be used as the initial copy.

Schedule Initial Replication

● Start replication immediately

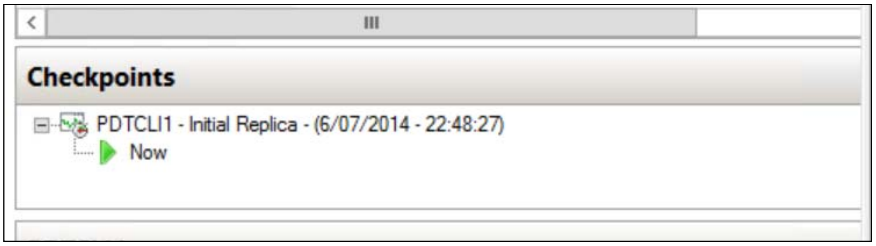
○ Start replication on:

7/07/2014 0:00

You can choose any time up to zondag 13 juli 2014.

This will start the initial replication process, which can be verified as follows:

1. A snapshot will be taken from the virtual machine we are replicating:

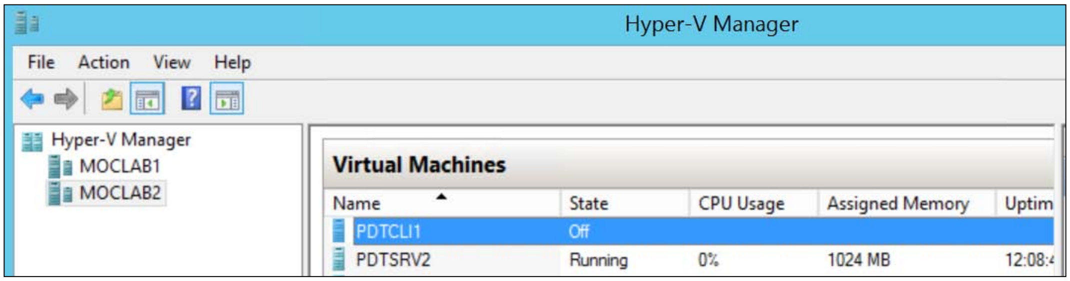


Checkpoints

PDTCLI1 - Initial Replica - (6/07/2014 - 22:48:27)

Now

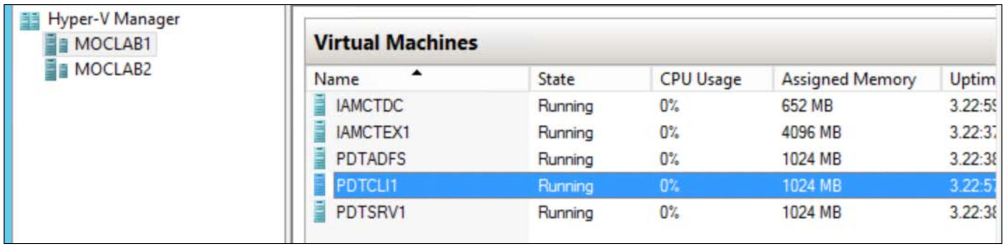
2. The replicated virtual machine will show up (in an offline state) as hosted on the Hyper-V Replica server (in our scenario, on `MOCLAB2`):



Virtual Machines

Name	State	CPU Usage	Assigned Memory	Uptim
PDTCLI1	Off			
PDTSRV2	Running	0%	1024 MB	12:08:4

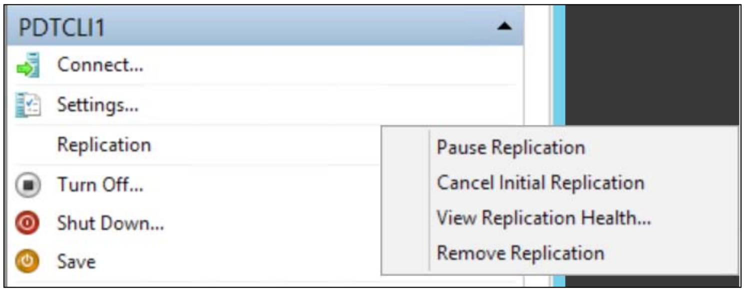
3. However, it will still be shown as a running virtual machine on the Hyper-V Replica source server (in our scenario, on `MOCLAB1`):



Virtual Machines

Name	State	CPU Usage	Assigned Memory	Uptim
IAMCTDC	Running	0%	652 MB	3.22:5
IAMCTEX1	Running	0%	4096 MB	3.22:3
PDTADFS	Running	0%	1024 MB	3.22:3
PDTCLI1	Running	0%	1024 MB	3.22:5
PDTSRV1	Running	0%	1024 MB	3.22:3

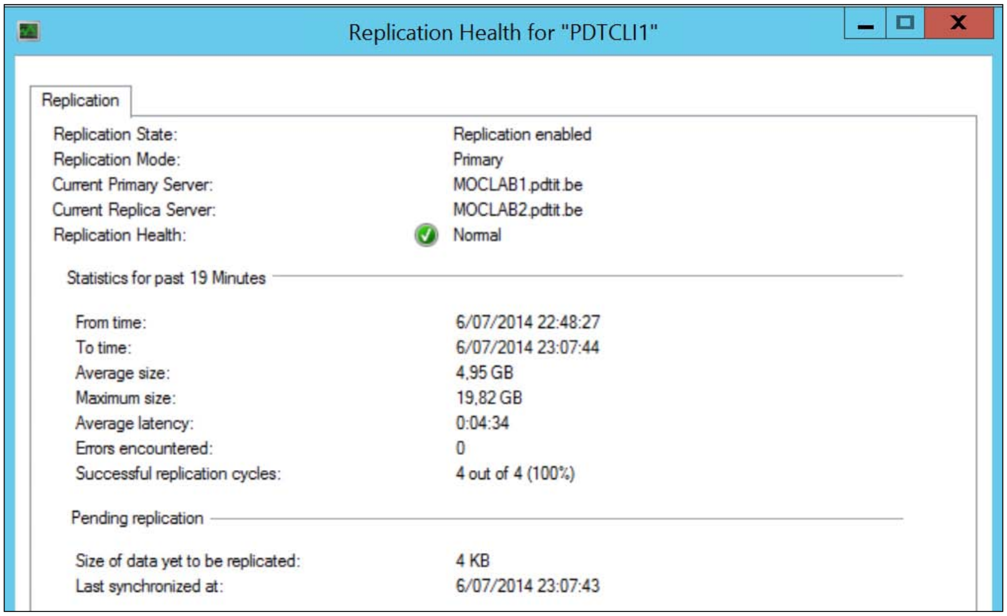
4. When selecting our virtual machine again, the following Hyper-V Replica specific context menu will show up:



5. Select the **View Replication Health...** option, which shows the following information:

    1. **Replication Health**: This can be green, yellow, or red state
    2. **From time** and **To time**: This is the time between initial and last replication
    3. **Average latency**: This is the time between replication attempts
    4. **Successful replication cycles**
    5. **Last synchronized at**: This is the time of the last replication cycle

    These are shown in the following screenshot:



Replication State:	Replication enabled
Replication Mode:	Primary
Current Primary Server:	MOCLAB1.pdtit.be
Current Replica Server:	MOCLAB2.pdtit.be
Replication Health:	✓ Normal

Statistics for past 19 Minutes

From time:	6/07/2014 22:48:27
To time:	6/07/2014 23:07:44
Average size:	4,95 GB
Maximum size:	19,82 GB
Average latency:	0:04:34
Errors encountered:	0
Successful replication cycles:	4 out of 4 (100%)

Pending replication

Size of data yet to be replicated:	4 KB
Last synchronized at:	6/07/2014 23:07:43

Summary

In this chapter, we talked about different mechanisms available in Hyper-V on Windows Server 2012 and 2012 R2, which allow for moving your virtual machines between hosts. Starting with live migration, you learned that it is possible to move virtual machines and virtual machines' data (configuration settings, virtual hard disks, and so on) from one Hyper-V host to another, without any downtime to the virtual machine.

In the second topic of the chapter, we detailed the different components of storage migration, allowing you to migrate a virtual machine's storage files between Hyper-V hosts, or moving from local storage to SMB 3.0 shares. We also explained how Hyper-V can benefit from these kinds of shared folders, and how to configure them.

In the last topic, we explained the concept of Hyper-V Replica, not only allowing you to move virtual machines, but actually create a full replica of a running virtual machine from one Hyper-V host to a second or even third host, allowing for a true disaster recovery solution, without needing failover clustering. Get ready to learn about failover clustering, which is the topic of our next chapter.

4
Building a High Available Hyper-V Cluster

After going through *Chapter 3*, *Live Migration, Storage Migration, and Hyper-V Replica*, and learning about how live migration, storage migration, Shared Nothing Live Migration, and Hyper-V Replica can help you in moving virtual machines between hosts in one way or another, a lot of organizations may see this as an HA and DR solution. While disaster recovery can be realized by using Hyper-V Replica with a configured latency for replicating virtual machines, the same may not be true for the other solutions, which aren't really intended to be full HA or DR solutions.

But, the good news is, Hyper-V has a true built-in solution for that, Hyper-V cluster.

This chapter will describe how to implement such a cluster, what the different components that you need for building such infrastructure are, and lastly, we explain how to execute failover testing.

In this chapter, we will cover the required steps for building a high available Hyper-V cluster. We go into all the details and requirements needed as well as explaining what exactly a high available cluster is.

In the previous chapter, we explained what a Hyper-V Replica is. Let's just quickly recap it. Hyper-V Replica allows you to replicate a copy of one or more virtual machines to another site or server not requiring shared storage where the copy is kept offline. During a failure at the primary site or server, this copy can be brought online, thus providing some sort of disaster recovery and business continuity; bear in mind that there might be some data loss, if not all changes have been replicated.

This brings us to a high available Hyper-V cluster, which is really a failover cluster, providing highly available virtual machines. The failover cluster is made up of multiple Hyper-V servers/nodes. Should one of these nodes fail, another Hyper-V host in the same cluster will take over the workload of that failing node, without any downtime or data loss for the running virtual machine(s) within that cluster. So in contrast with the Hyper-V Replica mechanism, where a copy of the virtual machine is used to create HA/DR, in case of Hyper-V clustering, it is the actual virtual machine that keeps running.

Some see this as the only decent and recommended way in which you should run your Hyper-V server configurations, but I personally think it is up to you to explain to your internal team or to your customer what the true differences and effects between both the solutions are.

For your information, Hyper-V clustering on Windows Server 2012 R2 brings additional functionality and improvements to failover clustering, of which we will be covering most, if not all, in this chapter.

Hyper-V cluster prerequisites and system requirements

Without talking too much in detail about the server specifics, we can go through a few guidelines listed here:

- Try to use Hyper-V hosts that are as identical as possible (same server vendor, server type, CPU type, and alike).
- Don't mix and match CPU architectures. (For example, an Intel processor-based Hyper-V host can't be combined in the same cluster with an AMD processor-based Hyper-V host.)
- Use Microsoft certified hardware on all layers (if in production environment).

> You may also research the Windows Server catalog for Hyper-V as an additional qualification. The Windows Server catalog is available on the Microsoft website at `http://windowsservercatalog.com/default.aspx`.

When the hardware layer as such is covered, you can start thinking about building your cluster. But before it can be used, you need to have an understanding of the following important cluster components:

- Cluster storage
- Cluster shared volumes
- Cluster networking
- Cluster quorum
- Shared virtual disks
- Guest clustering

Cluster storage

Cluster storage is the physical storage location that will be used by the Hyper-V cluster nodes, providing **cluster shared volumes** (**CSV**) for the virtual machines. Windows Server 2012 R2 can utilize different types of device adapters for creating such storage, such as iSCSI, Fibre Channel, and Fibre Channel over Ethernet. Where possible, use MPIO multiple paths for storage, to create the required high available connections between the storage solution and the Hyper-V hosts.

Cluster shared volumes

Cluster shared volumes or CSV allows the different Hyper-V nodes to access individual LUNs (storage volumes) all at the same time. This means, during concurrent connections to the same single LUN, all Hyper-V nodes can be used with Scale out File Server roles. It is interesting to know that CSV supports BitLocker, allowing for encryption of the virtual machines' configuration files and virtual disks, without any performance impact.

Cluster networking

While it is possible to build a Hyper-V cluster in a single NIC environment, I do recommend using multiple network adapters when in production, configured as follows:

- One (or more if teamed) dedicated for connecting to shared storage
- One (or more if teamed) for internal cluster communication
- One (or more if teamed) for communicating from the virtual machines, running on the cluster, to the overall network
- One (or more if teamed) NICs for management and/or Hyper-V administration

Cluster quorum

The failover cluster's quorum configuration determines the collective health of the cluster, as it will verify the running status of the cluster to determine if it is still within a healthy margin with enough nodes to support a running cluster. By default, each node gets a vote. In case of an equal amount of votes, the witness vote is also counted. (This could be a file share or disk witness.)

The different quorum modes define which vote gets counted. The types of quorum are as follows:

- **Node majority**: Every node in the cluster, which is healthy, running, and online and has network connectivity, can use its vote. There is no witness, and quorum is the majority vote.
- **Node and disk or node and file share majority**: Both witness and cluster nodes have a vote. Quorum is majority vote plus witness vote.
- **No majority**: None of the nodes have votes; only disk witness counts as a vote. (This setup is not recommended as disk witness is a single point of failure.)

We will talk more about the different quorum types and how to decide on which type to use later on, so don't worry if it is not completely clear to you yet.

Shared virtual disks

This is a new feature in Hyper-V clustering since Windows Server 2012, where VHDX files can be used as shared virtual disks, which can be used for guest clustering. The VHDX disk format is the only type of disk that supports this for storing both cluster data and quorum disks. In this scenario, the shared virtual disk, which is used by the guest cluster configuration, is in fact a VHDX file, stored on a cluster shared volume and configured as shared storage within a Hyper-V failover cluster.

Another example could be using an SMB 3.0 file share as part of a **Scale Out File Server** (**SOFS**), which is again configured as shared storage within a Hyper-V failover cluster.

Guest clustering

When talking about guest clustering, we refer to a failover cluster, consisting of two or more virtual machines, running inside Hyper-V. Think of a web server cluster or a SQL database cluster or any application cluster.

These virtual machines run inside a Hyper-V cluster, which can even be stretched over Hyper-V nodes in different failover clusters.

> For more details around clustering, you can also have a look at the Microsoft website article at `http://go.microsoft.com/fwlink/?LinkID=386723`.

Configuring shared storage for your cluster

Before we move over to installing and configuring our physical servers as Hyper-V cluster nodes, we need to talk about the cluster shared storage a bit more in detail.

From what you learned in the introduction, we recommend one of the following storage solutions for Hyper-V clustering:

- iSCSI shared storage
- SMB 3.0 Scale Out File Server shared storage

We've already talked about SMB 3.0 and how to configure such shared folders in the previous chapter, so we are not going to talk about that one anymore. However, we'll provide you with some additional information on using iSCSI shared storage, making sure you have a good understanding of the possibilities and differences between both solutions, and how to configure them.

> This doesn't mean these are the only supported ones; for example, Fibre Channel is still a valid and often used solution in larger enterprise environments!

iSCSI shared storage

iSCSI has been around for many years, and a lot of organizations' storage infrastructure is already built on it. If you are not familiar with iSCSI yet, let us give you a brief introduction. By using iSCSI, an iSCSI initiator (for example, a Windows Hyper-V Server) can connect to an iSCSI target (for example, your Storage Area Network (SAN)) storage, mapping the storage volumes as if they were local to the client.

The main benefits of using iSCSI are that it uses standard Ethernet technology, relying on TCP/IP as the transport mechanism, and it doesn't require (but we could recommend it in a production environment however) any dedicated networking connections as is the case with Fibre Channel.

While we referred to a SAN as the iSCSI target, one can also configure a Windows server as the iSCSI target, just by installing the iSCSI target server role. In this scenario, disk volumes that are local to that server can be presented as block-based storage volumes to other Windows servers in the same network.

While the implementation of iSCSI storage should be anything but complex, make sure you take the following points into consideration:

- Use a fast, and if possible, dedicated network for iSCSI communication. Where 1 Gbps is still the standard, 10 Gbps switches and NICs are becoming more and more popular, without the expensive cost of Fibre Channel devices.
- If your iSCSI communication is shared with the client/server network, try using QoS (Quality of Service) to limit bandwidth for specific traffic.
- If you are using a Windows Server as the iSCSI target, make sure you provide the required high availability by using MPIO (multipath) connections. In most SAN storage scenarios, these iSCSI network connections will mostly be high available already.
- While not used in all scenarios and all environments, just remember you can use advanced security on the iSCSI communication. This could include configuring communication only between certain hosts, configuring authentication (CHAP) for the iSCSI traffic, or even configuring encryption.

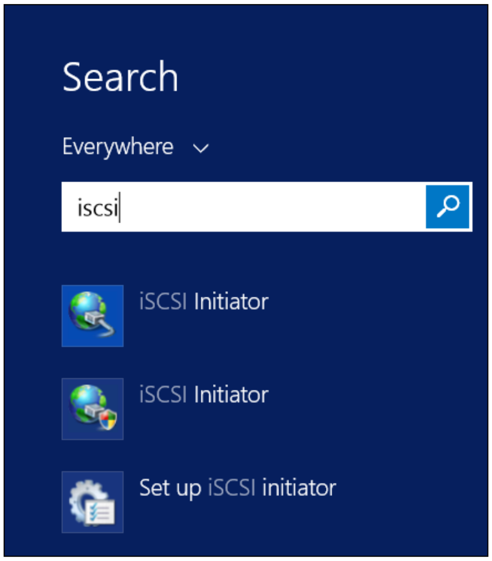
Configuring an iSCSI volume for Hyper-V as a storage solution

Now you know what iSCSI is and how you can use a SAN or Windows Server as an iSCSI target. We are about to show you how to configure an iSCSI volume and use this as a storage location for Hyper-V virtual machines.

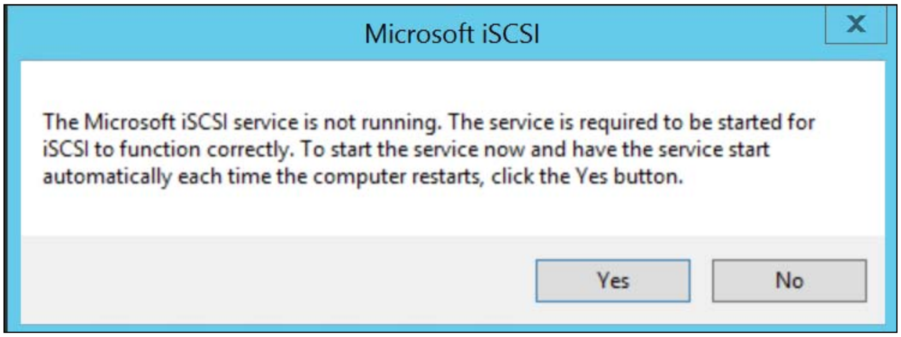
In our example, we use a home office / small office storage solution, which has built-in support for iSCSI. We assume you have your iSCSI volumes already configured on your storage solution.

The steps are as follows:

1. From the Windows Server 2012 R2 Start screen, type `iSCSI` in the search field; this will provide you with a few different iSCSI programs you can launch. From the list that appears (which might be a bit different on your system), select **Set up iSCSI initiator**:



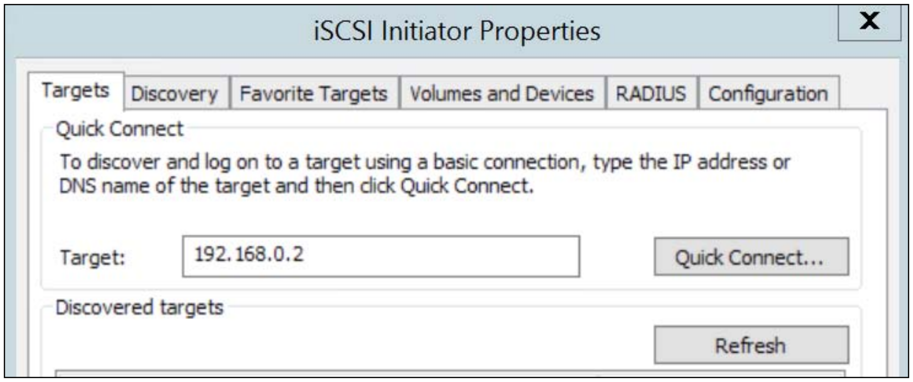
2. Notice the pop up that appears, informing you that the iSCSI service is not yet configured. Confirm this pop up by clicking on **Yes**, as it will configure the iSCSI service for automatic start, making sure the connection with the iSCSI target (your storage device) is set up again in case of a reboot of the server:



The Microsoft iSCSI service is not running. The service is required to be started for iSCSI to function correctly. To start the service now and have the service start automatically each time the computer restarts, click the Yes button.

This will launch the **iSCSI Initiator Properties** window.

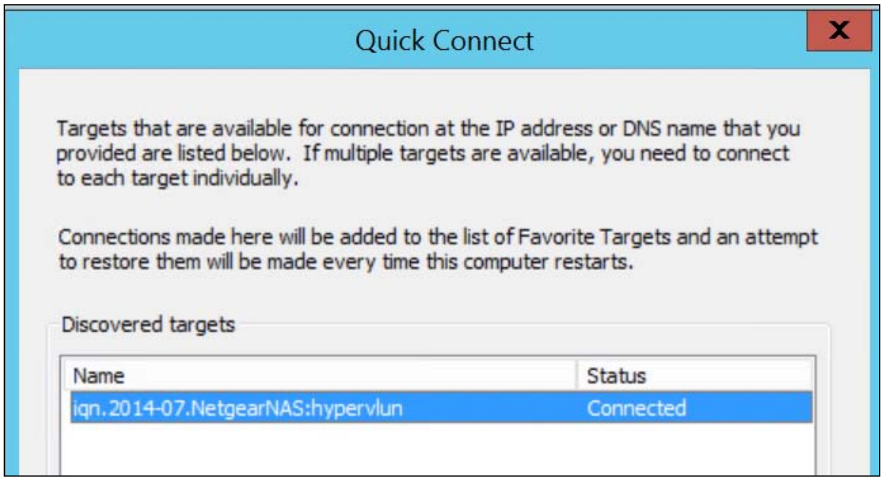
3. In the first tab named **Targets**, enter the IP address of your iSCSI target (your storage device) in the **Target** field, and click on **Quick Connect**. If your storage endpoint is configured correctly for iSCSI, it will get resolved and will be visible as a discovered target:



To discover and log on to a target using a basic connection, type the IP address or DNS name of the target and then click Quick Connect.

Target: 192.168.0.2

From what you can see in the following screenshot, the target has been discovered correctly and has a status of **Connected**:



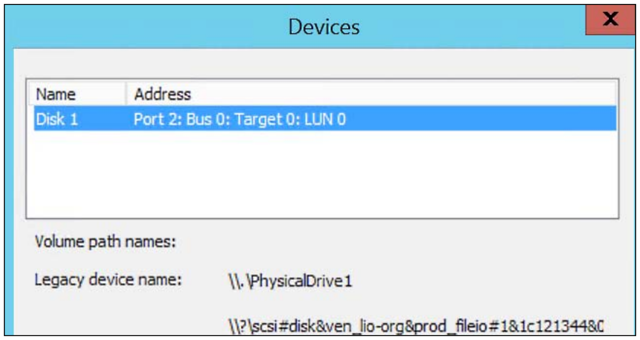
Targets that are available for connection at the IP address or DNS name that you provided are listed below. If multiple targets are available, you need to connect to each target individually.

Connections made here will be added to the list of Favorite Targets and an attempt to restore them will be made every time this computer restarts.

Name	Status
iqn.2014-07.NetgearNAS:hypervlun	Connected

Where this should already be enough to make the iSCSI volumes visible and usable in the Windows Server disk manager, let's go through some additional configuration parameters that are of interest (and sometimes required) in an enterprise environment.

4. Select your iSCSI discovered device, and press the **Devices...** button; this will show you the disk mapping from the local Windows Server's perspective (in my lab setup, **Disk1** is the reference to the iSCSI disk, as seen from the local Disk Manager). While this might not seem to be of importance for the moment, think of a Hyper-V host having multiple iSCSI volumes mounted, and one of the volumes is failing. For example, when opening up your Disk Manager, you could see that specific disk is offline. By using this reference, it should be easier to troubleshoot and find out what disk number is referring to what iSCSI volume on your storage device.



Devices

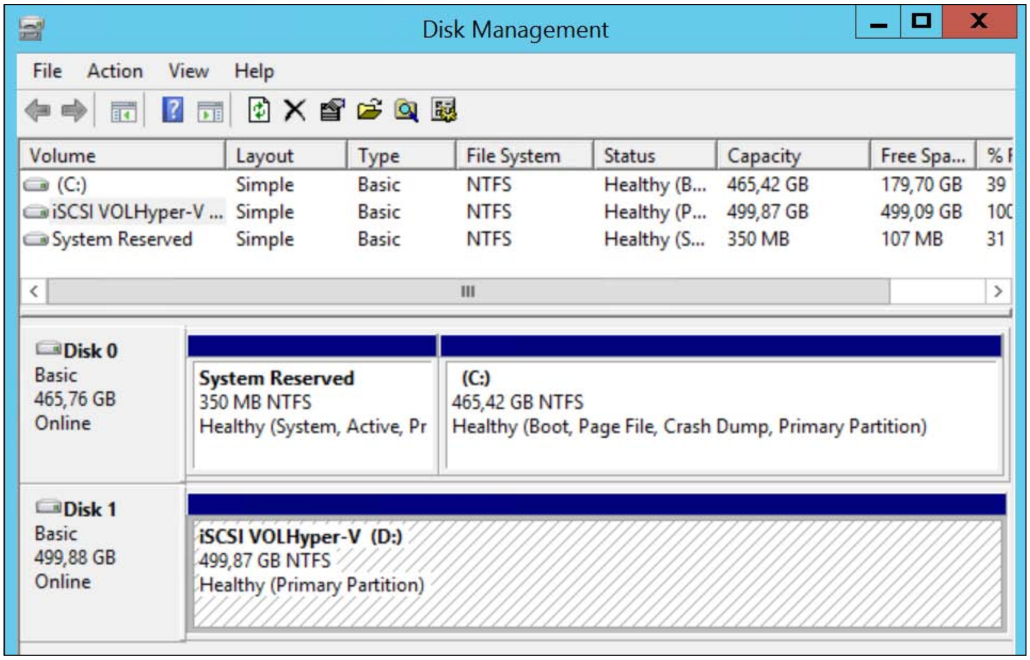
Name	Address
Disk 1	Port 2: Bus 0: Target 0: LUN 0

Volume path names:

Legacy device name: \\.\PhysicalDrive1

\\?\scsi#disk&ven_lio-org&prod_fileio#1&1c121344&C

5. This is what it looks like from within the Disk Manager:



Volume	Layout	Type	File System	Status	Capacity	Free Spa...	% F
(C:)	Simple	Basic	NTFS	Healthy (B...	465,42 GB	179,70 GB	39
iSCSI VOLHyper-V ...	Simple	Basic	NTFS	Healthy (P...	499,87 GB	499,09 GB	10(
System Reserved	Simple	Basic	NTFS	Healthy (S...	350 MB	107 MB	31

Disk 0
Basic
465,76 GB
Online

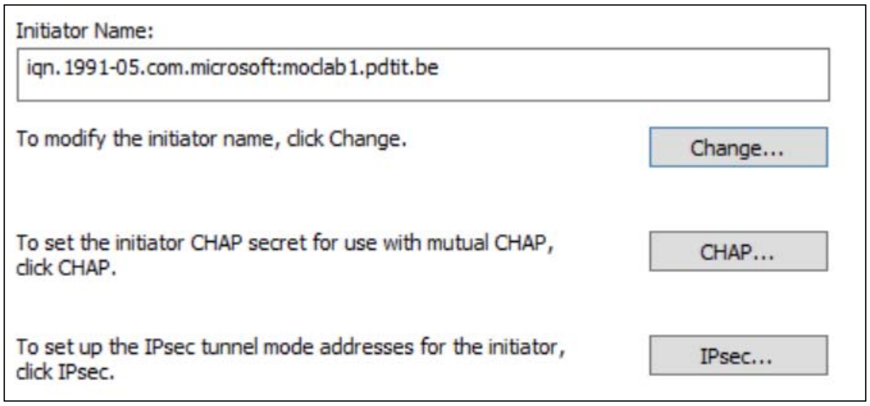
System Reserved
350 MB NTFS
Healthy (System, Active, Pr

(C:)
465,42 GB NTFS
Healthy (Boot, Page File, Crash Dump, Primary Partition)

Disk 1
Basic
499,88 GB
Online

iSCSI VOLHyper-V (D:)
499,87 GB NTFS
Healthy (Primary Partition)

6. If you want to configure specific security or authentication options for the iSCSI communication, you can do so from within the **RADIUS** or **Configuration** tab, as shown in the following screenshot:



Initiator Name:

iqn.1991-05.com.microsoft:moclab1.pdtit.be

To modify the initiator name, click Change.

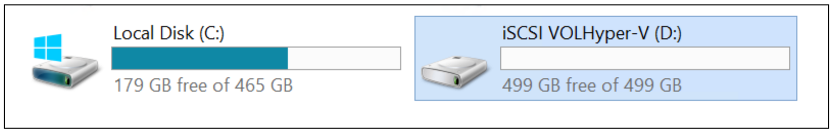
To set the initiator CHAP secret for use with mutual CHAP, click CHAP.

To set up the IPsec tunnel mode addresses for the initiator, click IPsec.

If you are wondering if and why you should use iSCSI security settings in your storage environment, have a look at the Microsoft TechNet knowledge base article at `http://technet.microsoft.com/en-us/library/cc732646.aspx`.

This gives a good explanation of how to manage iSCSI security and what possible reasons could be for implementing advanced security in your storage network.

In the end, your iSCSI mapped disk volume is presented with a drive letter and can be used as Hyper-V storage.



Local Disk (C:)
179 GB free of 465 GB

iSCSI VOLHyper-V (D:)
499 GB free of 499 GB

This concludes the configuration of an iSCSI-based disk volume on a local Hyper-V host.

Implementing a Hyper-V failover cluster

As you learned from the introduction of this chapter, Hyper-V failover cluster is one of the many server roles you can install on Windows Server 2012 and 2012 R2. To a large extent, installing the Hyper-V failover cluster server role is identical to installing any other server role. Of course, there are also some particular steps and parameters to take into consideration when deploying a Hyper-V failover cluster. And that is exactly what we are going to talk about in this section, that is, the implementation and configuration of a Hyper-V failover cluster.

In short, installing the Hyper-V failover cluster requires the following steps:

1. Cluster shared storage should be available to all nodes in the cluster.
2. Windows Server 2012 or 2012 R2 is installed on identically equipped (CPU, memory, and NICs as key resources) and fully operational physical machines and servers are configured as Active Directory domain members.
3. Install the Hyper-V server role on all nodes in the cluster.
4. Validate the cluster configuration and create the failover cluster.
5. Configure the virtual machines within the cluster to be stored on the cluster shared storage volumes.
6. Test the failover of a virtual machine from one host to another node in the cluster.

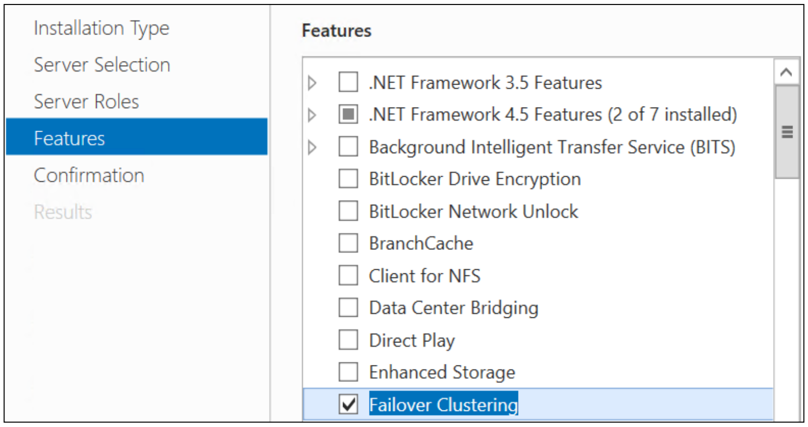
In our lab environment, we continue with the two Hyper-V servers we've configured in the previous chapters, MOCLAB1 and MOCLAB2. Both servers are identical in hardware characteristics, are running Windows Server 2012 R2, both are Active Directory domain members, and have an iSCSI connection to the shared storage volume we configured at the beginning of this chapter.

Installing the Hyper-V failover cluster server role

The steps are as follows:

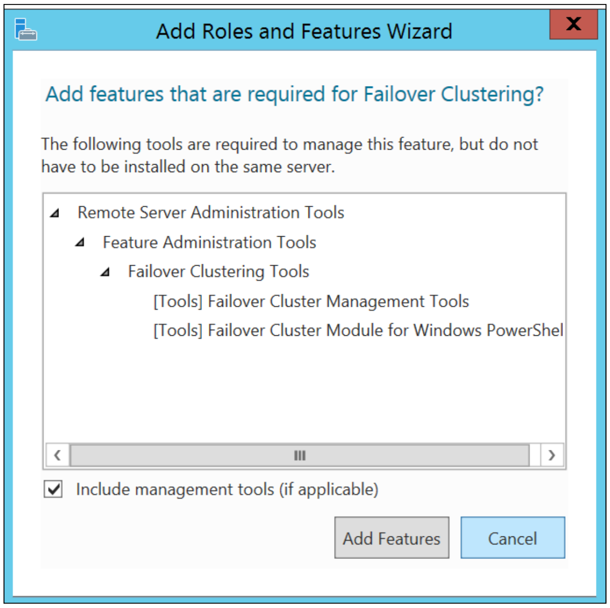
1. Log on to the first Hyper-V server and open up the Server Manager.
2. Click on **Add Roles and Features**.

3. In the **Installation Type** step, select **Role based or feature based installation**; click on **Next** in the **Server Roles** step.
4. In the **Features** installation step, select **Failover Clustering**. Confirm the installation of all subsequent components in the next step as well.



□ .NET Framework 3.5 Features
■ .NET Framework 4.5 Features (2 of 7 installed)
□ Background Intelligent Transfer Service (BITS)
□ BitLocker Drive Encryption
□ BitLocker Network Unlock
□ BranchCache
□ Client for NFS
□ Data Center Bridging
□ Direct Play
□ Enhanced Storage
☑ Failover Clustering

After selecting the **Failover Clustering** feature component, another window will be displayed showing the details of the selected features to be installed. In this window, keep the **Include management tools (if applicable) option selected**; click on **Add Features to continue**.



Add features that are required for Failover Clustering?

The following tools are required to manage this feature, but do not have to be installed on the same server.

- Remote Server Administration Tools
  - Feature Administration Tools
    - Failover Clustering Tools
      - [Tools] Failover Cluster Management Tools
      - [Tools] Failover Cluster Module for Windows PowerShel

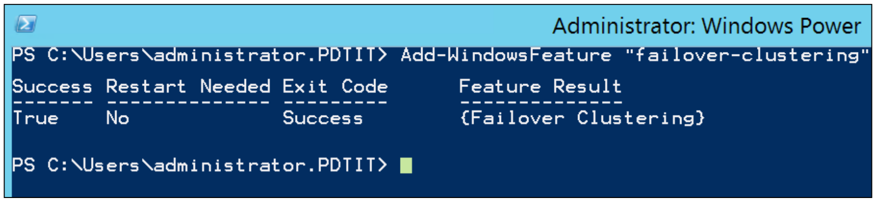
☑ Include management tools (if applicable)

5. Confirm the installation of the failover cluster feature by pressing the **Install** button. Wait for the installation to complete.

   Use the following PowerShell cmdlet to install the Windows feature on the second Hyper-V server:

   ```
   Add-WindowsFeature "failover-clustering"
   ```

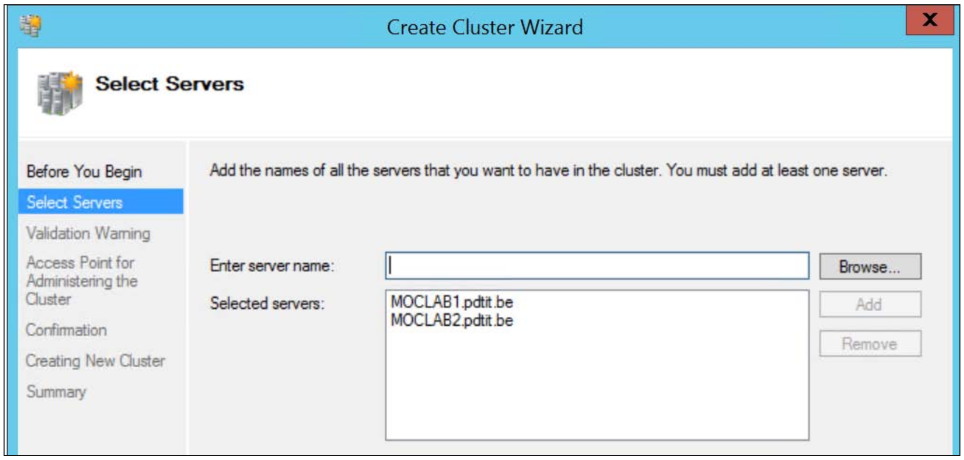
   The output is as follows:



```
PS C:\Users\administrator.PDTIT> Add-WindowsFeature "failover-clustering"
Success Restart Needed Exit Code      Feature Result
------- -------------- ---------      --------------
True    No             Success        {Failover Clustering}

PS C:\Users\administrator.PDTIT>
```

   Now the failover cluster feature is installed on both nodes, we continue with the configuration of the failover cluster for Hyper-V from MOCLAB1.

6. From the Start screen, start **Failover Cluster Manager**.
7. From the left, right-click on **Failover Cluster Manager** and select **Create Cluster**.
8. Click on **Next**. In the **Select Servers** step, add the two servers **MOCLAB1.pdtit.be** and **MOCLAB2.pdtit.be** to the list of selected servers:

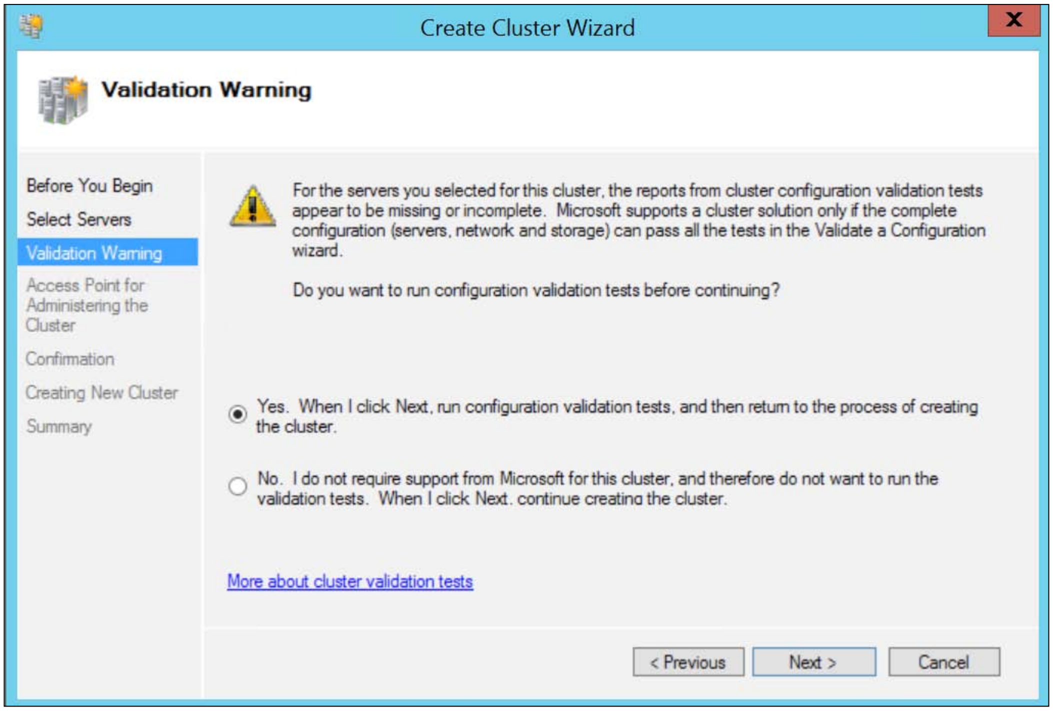


Add the names of all the servers that you want to have in the cluster. You must add at least one server.

Enter server name:

Selected servers: MOCLAB1.pdtit.be MOCLAB2.pdtit.be

9. Click on **Next**.
10. In the **Validation Warning** step, leave the default **Yes. When I click Next, run configuration validation tests, and then return to the process of creating the cluster** option checked. Click on **Next**:



Validation Warning

For the servers you selected for this cluster, the reports from cluster configuration validation tests appear to be missing or incomplete. Microsoft supports a cluster solution only if the complete configuration (servers, network and storage) can pass all the tests in the Validate a Configuration wizard.

Do you want to run configuration validation tests before continuing?

● Yes. When I click Next, run configuration validation tests, and then return to the process of creating the cluster.

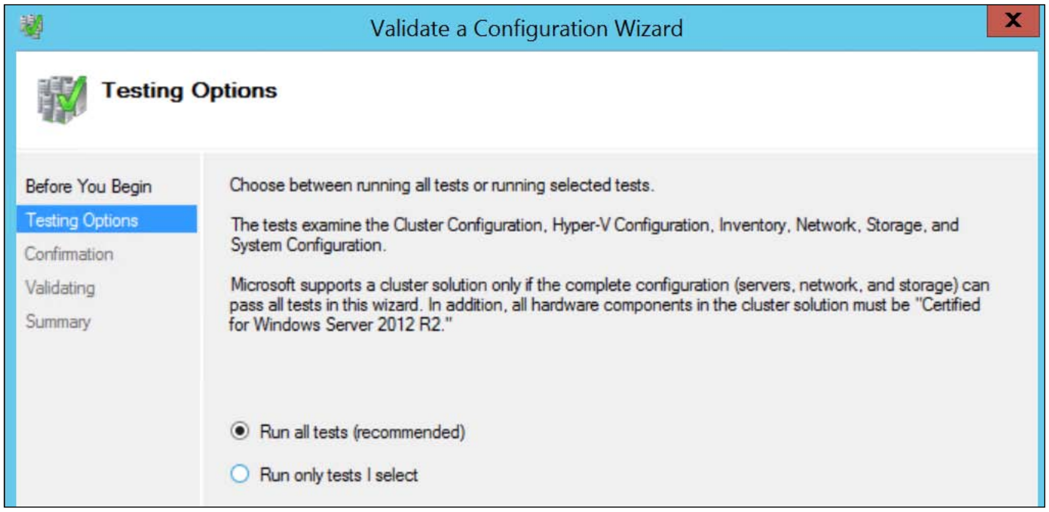
○ No. I do not require support from Microsoft for this cluster, and therefore do not want to run the validation tests. When I click Next, continue creating the cluster.

More about cluster validation tests

This will start the **Validate a Configuration Wizard** window.

11. In the first step of the **Validate a Configuration Wizard**, click on **Next**.

12. In the **Testing Options** step, select the **Run all tests (recommended)** option and then click on **Next**:



Choose between running all tests or running selected tests.

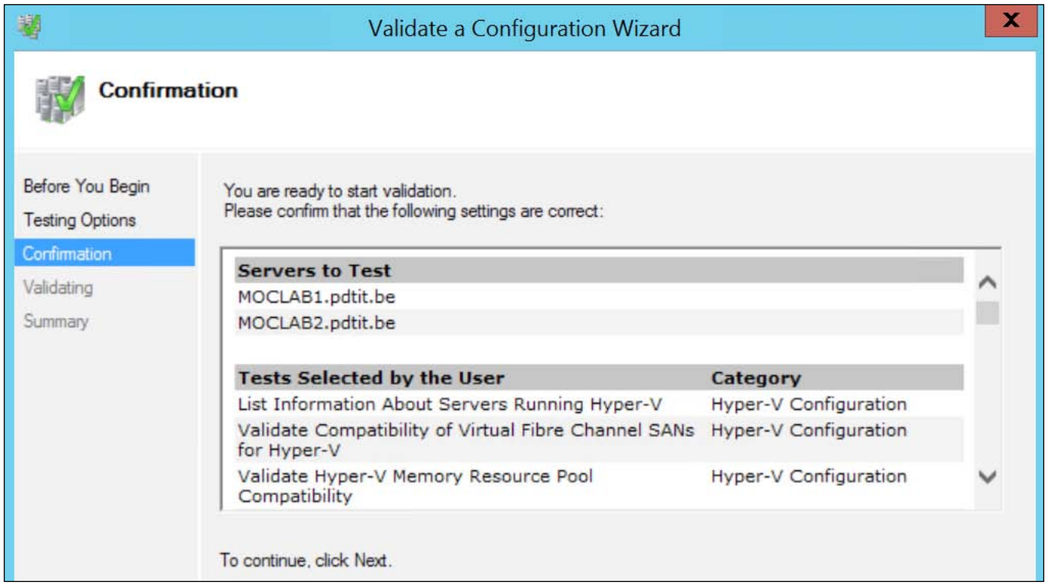
The tests examine the Cluster Configuration, Hyper-V Configuration, Inventory, Network, Storage, and System Configuration.

Microsoft supports a cluster solution only if the complete configuration (servers, network, and storage) can pass all tests in this wizard. In addition, all hardware components in the cluster solution must be "Certified for Windows Server 2012 R2."

● Run all tests (recommended)

○ Run only tests I select

13. In the **Confirmation** step, click on **Next**:



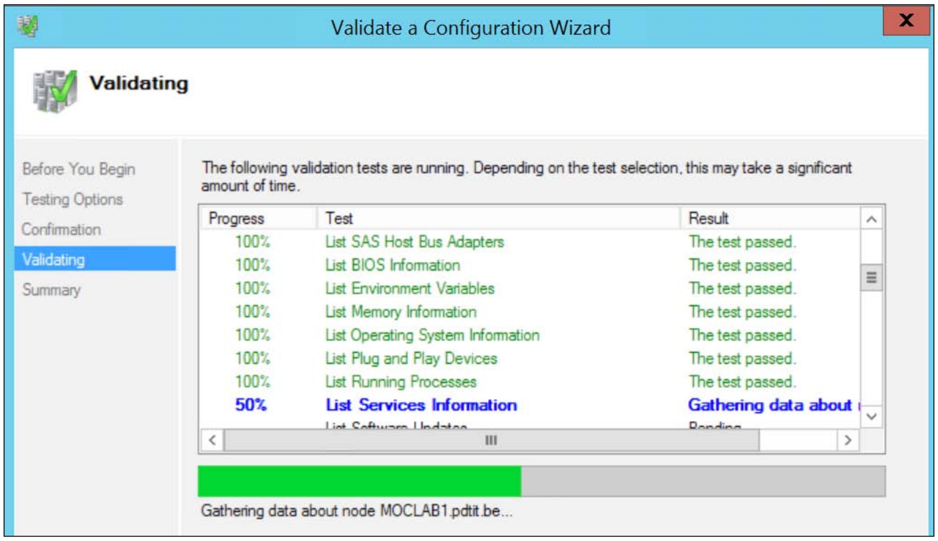
You are ready to start validation.
Please confirm that the following settings are correct:

Servers to Test	
MOCLAB1.pdtit.be	
MOCLAB2.pdtit.be	

Tests Selected by the User	Category
List Information About Servers Running Hyper-V	Hyper-V Configuration
Validate Compatibility of Virtual Fibre Channel SANs for Hyper-V	Hyper-V Configuration
Validate Hyper-V Memory Resource Pool Compatibility	Hyper-V Configuration

To continue, click Next.

14. The cluster node servers will be validated for cluster support; depending on your infrastructure, this might take a few minutes:



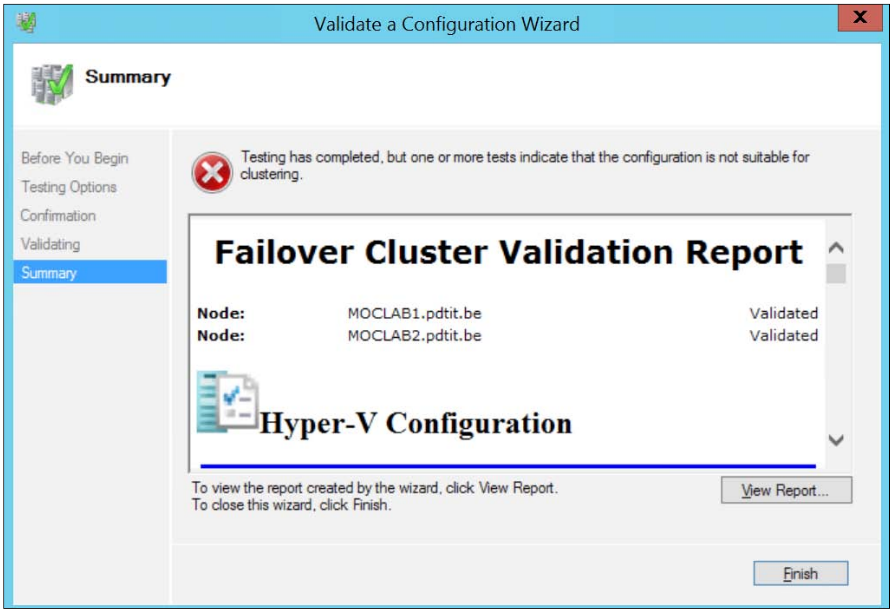
The following validation tests are running. Depending on the test selection, this may take a significant amount of time.

Progress	Test	Result
100%	List SAS Host Bus Adapters	The test passed.
100%	List BIOS Information	The test passed.
100%	List Environment Variables	The test passed.
100%	List Memory Information	The test passed.
100%	List Operating System Information	The test passed.
100%	List Plug and Play Devices	The test passed.
100%	List Running Processes	The test passed.
50%	**List Services Information**	**Gathering data about**
	List Software Updates	Pending

Gathering data about node MOCLAB1.pdtit.be...

> If all hardware is Windows Server 2012 R2-supported, and all previous configurations from this chapter are executed correctly, you shouldn't receive any special error messages. If some errors or warnings do show up, go through them one by one, try to fix them and run the validation again. It will save you a lot of frustration afterwards when your failover cluster is not working correctly in the end.

15. At the end of the validation check, a summarized overview will be shown, including an option to export the results to a report view:



Testing has completed, but one or more tests indicate that the configuration is not suitable for clustering.

Failover Cluster Validation Report

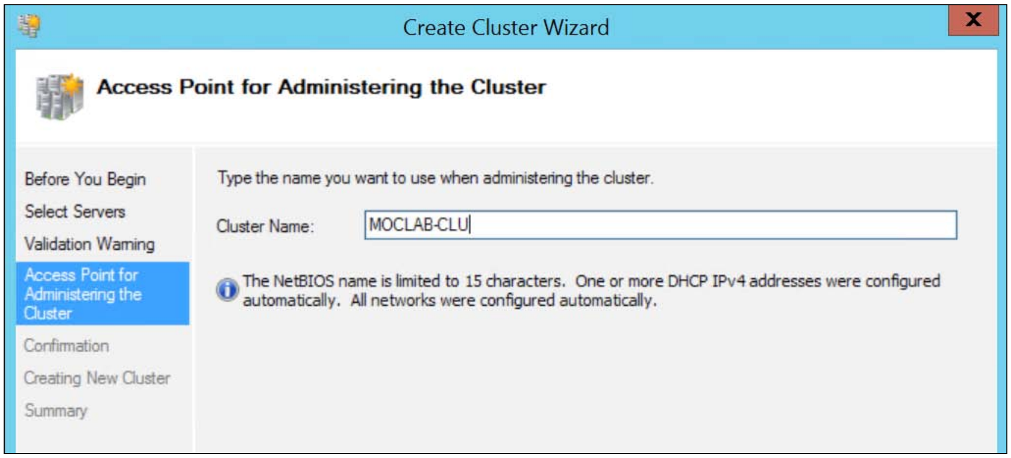
Node:	MOCLAB1.pdtit.be	Validated
Node:	MOCLAB2.pdtit.be	Validated

Hyper-V Configuration

To view the report created by the wizard, click View Report.
To close this wizard, click Finish.

In our lab environment, some errors were thrown because we are using a non-supported Windows Server 2012 R2 iSCSI storage system, but in the end all worked fine.

16. Close the **Validate a Configuration Wizard** by clicking on **Finish**.
17. This brings you back to the cluster configuration wizard; in this step (**Access Point for Administering the Cluster**), you have to enter a name for your Hyper-V failover cluster; in our environment, we chose the name `MOCLAB-CLU`:

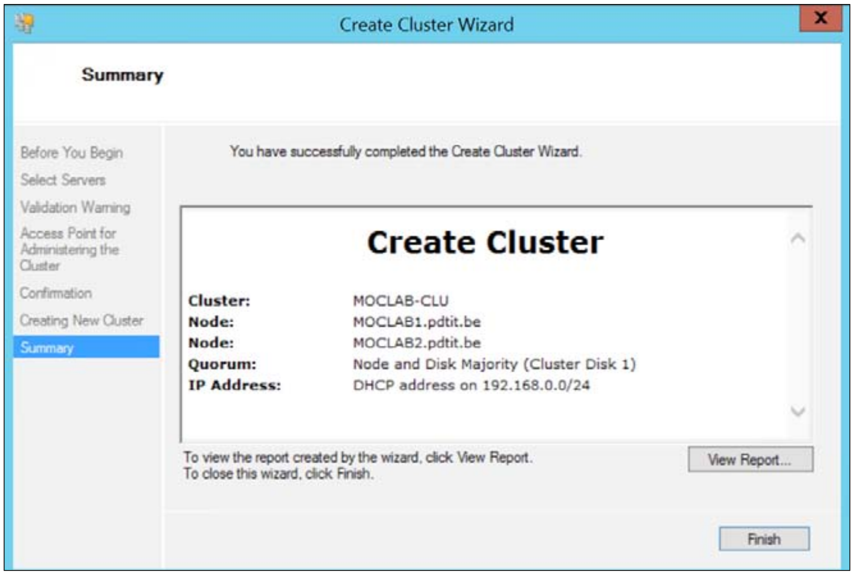


Type the name you want to use when administering the cluster.

Cluster Name: MOCLAB-CLU

The NetBIOS name is limited to 15 characters. One or more DHCP IPv4 addresses were configured automatically. All networks were configured automatically.

18. Click on **Next** to continue. In the **Confirmation** step, you can review all the entered details about the cluster nodes and the cluster name. If all is correct, click on **Next** to actually have the failover cluster get configured.
19. After a few minutes, the cluster should have been configured successfully, as shown in the summarized view:



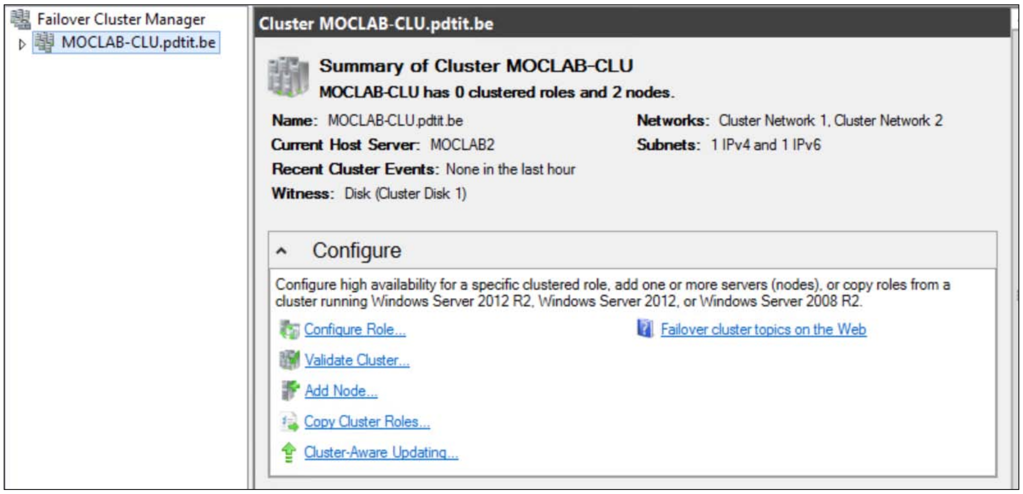
You have successfully completed the Create Cluster Wizard.

Create Cluster

Cluster:	MOCLAB-CLU
Node:	MOCLAB1.pdtit.be
Node:	MOCLAB2.pdtit.be
Quorum:	Node and Disk Majority (Cluster Disk 1)
IP Address:	DHCP address on 192.168.0.0/24

To view the report created by the wizard, click View Report.
To close this wizard, click Finish.

20. From the **Failover Cluster Manager**, it should look like this:



Summary of Cluster MOCLAB-CLU

MOCLAB-CLU has 0 clustered roles and 2 nodes.

Name: MOCLAB-CLU.pdtit.be
Current Host Server: MOCLAB2
Recent Cluster Events: None in the last hour
Witness: Disk (Cluster Disk 1)
Networks: Cluster Network 1, Cluster Network 2
Subnets: 1 IPv4 and 1 IPv6

Configure high availability for a specific clustered role, add one or more servers (nodes), or copy roles from a cluster running Windows Server 2012 R2, Windows Server 2012, or Windows Server 2008 R2.

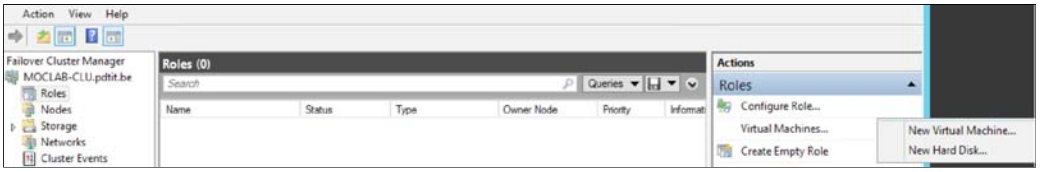
We are almost done here. During the cluster configuration, you should notice a folder path on the `C` drive called `ClusterStorage`. This is nothing but a pointer to the iSCSI volumes on your SAN infrastructure. According to the number of volumes you have, they will all be listed here as subfolders.

This concludes the configuration of the failover cluster itself. In the last part of this chapter, we will show you how to create a high available virtual machine, running on our failover cluster.

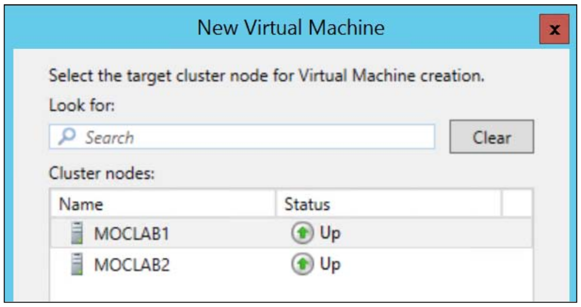
Creating a high available virtual machine

In this last part, the only configuration that is left is actually creating a high available virtual machine. To make things easy, the steps required to create such a VM are almost identical to creating a standalone VM as we did in *Chapter 2*, *Unwrapping Hyper-V 2012 R2 Components*. The steps are as follows:

1. From within the **Failover Cluster Manager**, select your cluster (**MOCLAB-CLU** in our environment), then select **Roles** | **Virtual Machines...** | **New Virtual Machine...**, as shown:



2. Select any of the cluster nodes where you want the VM to be created:



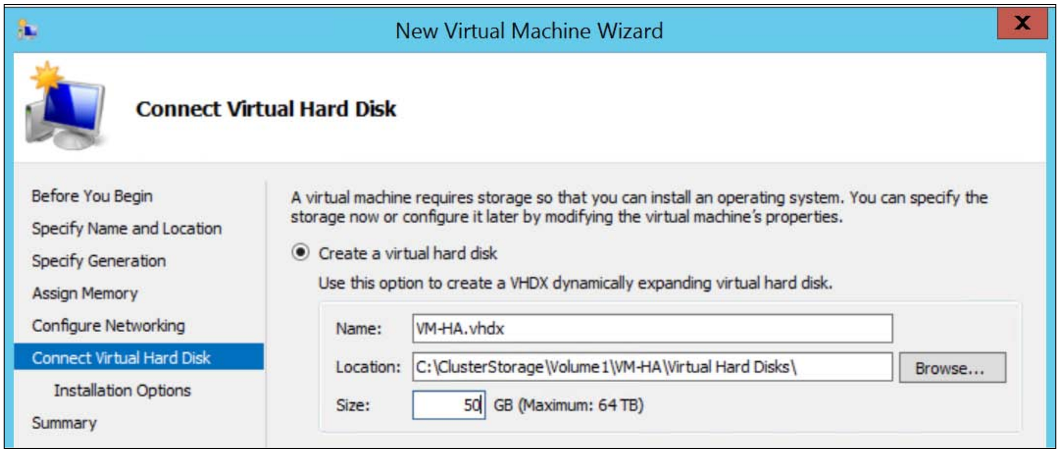
Select the target cluster node for Virtual Machine creation.

Look for:

Cluster nodes:

Name	Status
MOCLAB1	Up
MOCLAB2	Up

3. In the **Specify Name and Location** step, provide a name for your new virtual machine and select the cluster shared volume `c:\clusterstorage\volumeX` as the location. As a result, the VHDX file from the VM will be stored on the cluster shared volume:



A virtual machine requires storage so that you can install an operating system. You can specify the storage now or configure it later by modifying the virtual machine's properties.

● Create a virtual hard disk

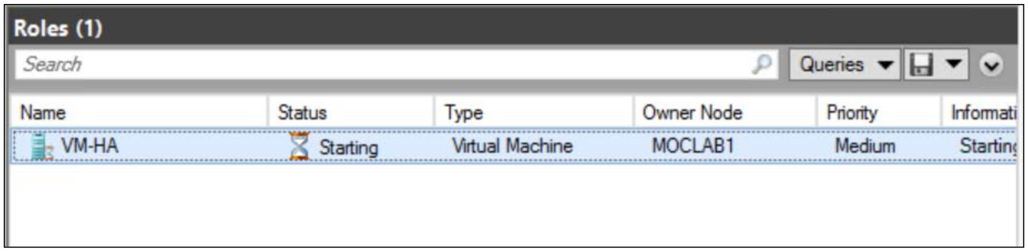
Use this option to create a VHDX dynamically expanding virtual hard disk.

Name: VM-HA.vhdx

Location: C:\ClusterStorage\Volume1\VM-HA\Virtual Hard Disks\

Size: 50 GB (Maximum: 64 TB)

4. Complete all the other steps from the wizard, using their default settings. If you don't remember how to create such a virtual machine, have a look again at *Chapter 2*, *Unwrapping Hyper-V 2012 R2 Components*, where this is described in detail.
5. When the virtual machine is created successfully, it will initially show up as offline in your **Failover Cluster Manager**. Right-click on the VM, and you should be able to start it:



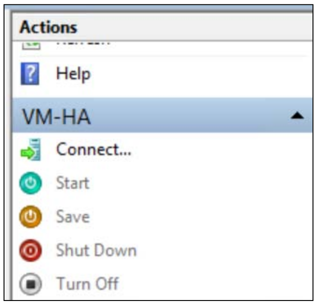
Name	Status	Type	Owner Node	Priority	Informati
VM-HA	Starting	Virtual Machine	MOCLAB1	Medium	Startin

This concludes the creation of a high available virtual machine, running on the CSV storage.

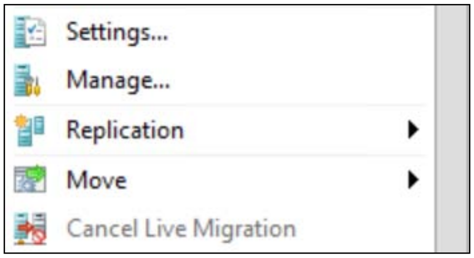
Managing your high available virtual machine

In this last section, we will go over some specific failover cluster configuration options that are only available to high available virtual machines. Most of these settings are configurable from the **Actions** pane in the **Failover Cluster Manager**, after you have selected the virtual machine itself.

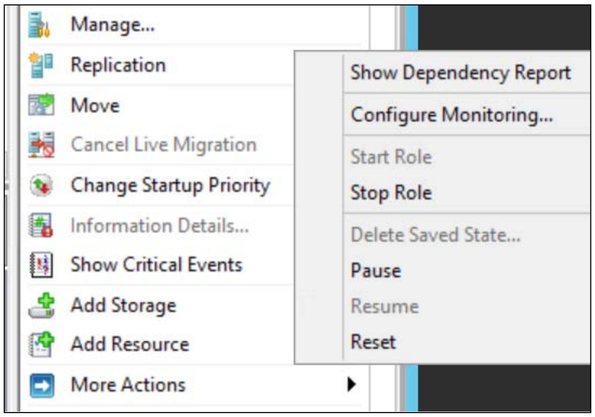
The basic VM commands in the **Actions** pane allow you to connect to the VM, as well as perform start, save, shut down, or turn off operations:



Some other interesting actions that are available here are opening up the **Settings** view, which shows you the parameters of the VM such as memory, CPU, the hard disks it is using, and so on. The **Manage** option will take you back to the Hyper-V Manager, where you can perform most of the same actions as for standalone VMs. The **Replication** and **Move** options allow you to enable Hyper-V replication, moving this VM between Hyper-V hosts, or moving the virtual storage to another location:



The last **Actions** pane topic that can be of interest is the **More Actions** section. Here, you have several actions available such as removing the VM out of the high available cluster (**Stop Role**), **Reset**, **Pause**, or **Resume** for the virtual machine:



Summary

In this chapter, we extensively focused on the aspect of providing high availability to your Hyper-V infrastructure and more importantly to the virtual machines running on the Hyper-V nodes. We started this chapter by explaining why an organization should decide to configure Hyper-V failover clusters; then we explained the interaction between Hyper-V hosts and clustered shared volumes (CSV), as well as showing you how to configure such CSV volumes on an iSCSI storage endpoint.

In the second part of this chapter, you learned how to install the failover cluster component on your Hyper-V hosts.

In the last section, we showed you how to create a new high available virtual machine, and how to manage it from within the Failover Cluster Manager.

In the next chapter, we will dive into a more enterprise-oriented management platform for Hyper-V, which is System Center Virtual Machine Manager. We will start with an overview of the management solution, explaining what is possible from that and how to set it up in your environment. After that, we will discuss the core management tasks you should be able to perform when managing a Hyper-V platform.

5

Hyper-V Management Using System Center Virtual Machine Manager

To date, you have mainly used the Hyper-V management console or its equivalent PowerShell cmdlets to deploy new virtual machines, configure virtual networks, and manage the Hyper-V components. From the last chapter on Hyper-V clustering, you learned that you can also perform a lot of these Hyper-V tasks from within the cluster manager itself.

Where the Hyper-V management console and PowerShell are very powerful and pretty complete, they are missing several features that are a key in an enterprise virtualized infrastructure. In this scenario, you should consider using **Microsoft System Center Virtual Machine Manager** (**SCVMM**), where the 2012 R2 edition is the most recent.

In this chapter, we will cover the main reasons why an IT organization should implement SCVMM as its virtualization platform management solution. We will explain the installation process, followed by giving a detailed walk-through of the main components of the console, clearly showing you how to manage the so-called fabrics (networking, storage, virtual machines, and so on), how to perform virtual machine operations, and more.

Lastly, we will briefly describe the principle of "private clouds", and how this data center topology can be integrated in SCVMM as well.

Reasons for using SCVMM in your virtualized environment

As mentioned in the introduction to this chapter, the standalone Hyper-V management console does a pretty good job in managing and operating your Hyper-V environment, up to a certain scale that is. Imagine you are running a dozen or more physical Hyper-V hosts, each running a few virtual machines on it. But it is not only the number of servers that should be used as a trigger for SCVMM. Think of more complex networking environments, using multiple vLAN configurations, DMZ topologies, complex routing, and alike.

Managing such an environment might become a complex operation if you rely on the Hyper-V manager alone.

This is where SCVMM comes in play. As part of the overall System Center suite of products, VMM is a true virtualized data center management solution.

> If you are new to System Center, have a look again at the *Preface* to find out what the true strengths and reasons of the System Center suite actually are.

The important word here is "virtualized data center". SCVMM enables the management and operation of all aspects of the virtual data center such as your physical hosts, your virtual machines, the virtual networks, the storage interaction, and so on. This is applicable not only for hosts running on Microsoft Hyper-V, but also for hosts that are running on VMware ESX or Citrix XenServer.

> We have dedicated the next chapter to how to integrate VMware ESX hosts in your SCVMM management solution.

By using SCVMM, an organization can leverage the following features (not limited):

- High availability of the SCVMM management tools
- Creating user roles to provide delegated administration of the virtualized data center
- Integration with Citrix XenServer
- Full integration with VMware ESX vCenter

- SCVMM is centralized around a library, which is a catalog of resources to be used in the virtualized data center (templates, virtual hard disks, ISO files, and so on)
- You have the possibility to configure the host and guest server profiles, as well as application and SQL server profiles, streamlining and automating the deployment of such servers
- Creating virtual machine templates and service templates
- Integration with other System Center components and Microsoft Azure
- Monitoring the health and performance of your virtual machines and their hosts

SCVMM version 2012 R2 specifically provides the following features and enhancements to the product:

- Site-to-site network virtualization using NVGRE (Generic Routing Encapsulation) gateways, which enables better networking features to hosting providers such as establishing higher capacity, better performance, better reliability, as well as the creation of tenant scenarios. The SCVMM APIs now also support native networking components better than ever before. A few examples here are the tight integration with Cisco Nexus switches or having the possibility to use load balancers as virtual network gateways.

  For a good explanation on configuring and using NVGRE gateway in a Hyper-V environment, read Kristian Nese's blog post at `http://kristiannese.blogspot.be/2013/06/how-to-add-your-virtualization-gateways.html`.
- Strong integration with Server 2012 R2's IPAM (IP address management). If you are new to IPAM, then have a look at the Microsoft TechNet article at `http://technet.microsoft.com/en-us/library/dn268500.aspx`, which can be used as a good introduction. Some possible usage scenarios here are a better management of IP addresses of physical hosts and virtual machines across the data center. During a virtual machine migration, IP addresses can be configured to match the IP subnet settings of the destination networking subnet by using policies.
- Support of Generation 2 virtual machines is natively built-in (we talked about Generation 1 and 2 virtual machines and the key differences between both in *Chapter 1*, *Introduction to Microsoft Server 2012 R2 Hyper-V*).
- Live cloning of virtual machines; this is actually built on top of a new feature within Hyper-V 2012 R2, which allows for the creation of an exact copy (a clone) of a running VM, without any downtime.

- SCVMM 2012 R2 relies on another new feature in Hyper-V 2012 R2, that is, Hyper-V file transfer APIs. By using these APIs, data can be copied from the HyperV host server to HyperV virtual machines that are running on this host, even when there is no network connection between both servers. The only requirement here is both the physical host and VMs should be running on Server 2012 R2.
- Support for Oracle Linux 5 and 6, as well as Debian GNU/Linux 7 guest operating systems.
- Support for virtual Fibre Channel connections.
- SCVMM leverages on the **Offloaded Data Transfers** (**ODX**) feature of Windows Server 2012 R2, allowing for file fast copy, which enhances the performance of file transfers or virtual machine deployments. A side note here is the physical NICs must also support this feature.

We will discuss several of these listed features and enhancements in the remainder of this chapter.

> One of the key missing features compared to previous versions of SCVMM is the ability to convert physical machines into virtual machines (P2V). Of course, we will explain how you can still make use of this feature within your data center.

Introduction to the Virtual Machine Manager architecture

Before we walk you through the actual installation process of the SCVMM solution, let's start by explaining the different components of a possible SCVMM architecture.

As a minimum, one needs to install an SCVMM management server, as well as an SQL Server database server. You have the option to deploy most components on a single server topology, although it is our recommendation to stretch different components over multiple servers, as it allows for high availability and better performance in larger data center environments.

The following table gives an overview of the different SCVMM components and/or server roles:

Component/server role	**Description**
Management server	This is the main server on which the SCVMM services are running. It is the VMM management server that runs and executes all tasks, processes, and commands within the overall VMM environment. The management server connects to all other components of a possible VMM architecture.
Database component	SCVMM relies on a SQL database to store all information about the VMM topology and its configuration. This can include virtual machines, hosts, libraries, tasks, jobs, clouds, and other virtualized platform information. The SQL Server component can be installed locally on the SCVMM management server, on a separate dedicated SCVMM SQL Server, or as a separate instance on a shared SQL Server infrastructure.
Management console	This is the interface that talks to the SCVMM management server. Through the console, all operations, management tasks, jobs, and configurations are done. You can install this console on the SCVMM server itself, on a separate server (for example, a remote desktop services server to allow for connecting with multiple users), or on a Windows 7/8 client machine.
Library	The SCVMM library is a vital component in the overall topology. It is a shared location where SCVMM stores its resources such as virtual machine templates, virtual hard disks, profiles, and so on. By default, the first SCVMM server that is introduced in the environment is the library server, although it is possible to add additional library servers later on. The library itself is configured on a shared storage as best practice, although it is not required.
PowerShell	While this is not a separate component as such, we want to highlight that most configurations that are visible and usable from within the VMM management console also work from within PowerShell. In fact, the console is built on top of PowerShell.

Now, you know the different components of a possible SCVMM architecture, so let's talk a bit more in detail about the specific topologies and some best practices that I have learned from my own experience.

When we are designing a potential SCVMM topology for a customer, we take note of the following:

- Number of physical hosts, using any SCVMM supported hypervisor
- Number of virtual machines
- Single location data center or multisite geo-stretched locations
- Administrative requirements to use different user roles and groups for management
- Availability of the SCVMM management solution

There is no single answer to design a SCVMM topology. Based on the preceding key components, it should be possible to outline a well-working setup though. My suggestion is to start from your own experience, and do enough monitoring on SCVMM resource performance and availability. If the SCVMM environment needs to be highly available and redundant, it is clear that all components need to be doubled as a minimum.

> There are some additional requirements to the environment when configuring SCVMM in a highly available topology; more details are shared at the end of this chapter.

Don't forget the SQL database component in the high availability design, and neither the shared resources for the library.

There are some numbers reported on Microsoft's TechNet website as well as other Internet resources about how many hosts and virtual machines you can manage with a single SCVMM management server. Without going into too much detail, remember it should be more than enough for about any environment, as it is close to 1,000 hosts and 25,000 virtual machines. (Officially, 400 hosts and 8,000 VMs is the number in SCVMM 2007.)

If you are running such an environment, then let me know as I'm more than interested in configuring the Virtual Machine Manager platform for you.

As the basic topology and architectural components have been touched, let's continue the chapter by guiding you in how to install and configure the SCVMM management server.

Installing and configuring Virtual Machine Manager

The system resources for a SCVMM server are far from extraordinary. Without making these numbers hard, the following specifications should allow you to have a smooth-running SCVMM in a typical Hyper-V environment having 100 hosts / 1,000 VMs on an average as follows:

- Dual core CPU, 2.8 GHz or higher
- 4 to 8 GB of RAM
- 60 GB of disk space for the Windows Server 2008 R2 or Server 2012 or 2012 R2 operating system
- 150 GB of disk space for the SQL database

From a software perspective, you will need the following:

- System Center Virtual Machine Manager 2012 R2
- .Net Framework 4 or higher
- Windows Assessment and Deployment Kit (WADK)
- SQL Server 2008 R2 with SP2 or higher or SQL 2012 with SP1

If you want to take control of the SCVMM server, then nothing stops you from logging on locally to the management server; however, you also have the option to install a separate SCVMM management console on a management station, for example. This could be a remote desktop services server, which allows multiple connections at the same time. If you prefer having a locally installed console, then remember this can also be installed on your Windows 7 or Windows 8/8.1 client machine.

Walk-through of the installation

In this topic, we are going to install a single SCVMM server in our lab environment. This machine will have an SQL Server instance called SCVMM configured locally on the SCVMM server. Remember that in a production environment, it is best practice to separate the SQL database from the SCVMM server. With SQL 2012, you have the option to make use of the new SQL cluster type, known as AlwaysOn Failover Cluster.

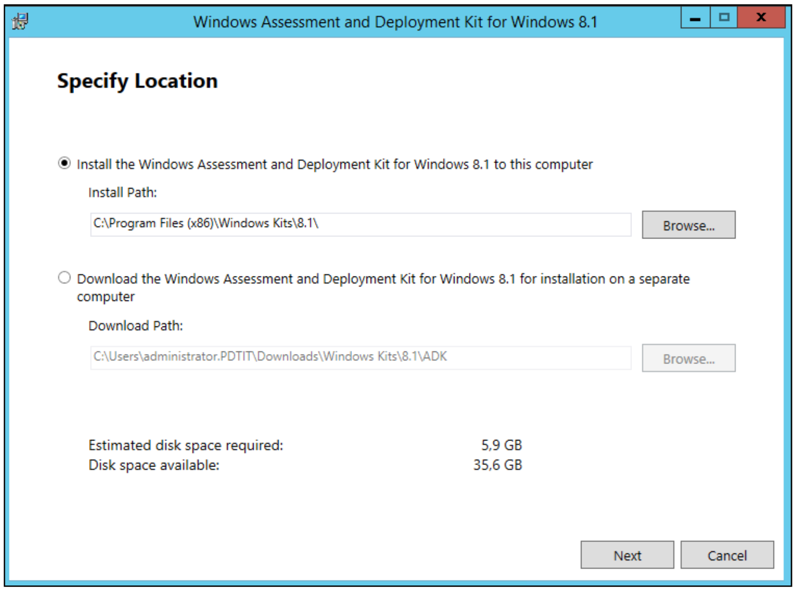
> For more information regarding the SQL AlwaysOn Failover Cluster, have a look at the Microsoft TechNet Knowledgebase article at `http://technet.microsoft.com/en-us/library/ms189134.aspx`.

To save some pages here and skip irrelevant information, we will not describe the SQL Server 2012 instance configuration in detail. Basically, all default settings should be OK. I personally prefer to create a named instance instead of the default instance that is suggested, but all other settings should be straightforward.

Therefore, in this topic, we assume your SQL Server instance is already set up and configured.

Let's start by installing the **WADK for Windows 8.1** update, as this is a required component of SCVMM as follows:

1. Download the WADK installation files from `http://www.microsoft.com/en-us/download/details.aspx?id=39982`.
2. Start the setup; specify the install path for the WADK program files:



Windows Assessment and Deployment Kit for Windows 8.1

Specify Location

● Install the Windows Assessment and Deployment Kit for Windows 8.1 to this computer

Install Path:

C:\Program Files (x86)\Windows Kits\8.1\

○ Download the Windows Assessment and Deployment Kit for Windows 8.1 for installation on a separate computer

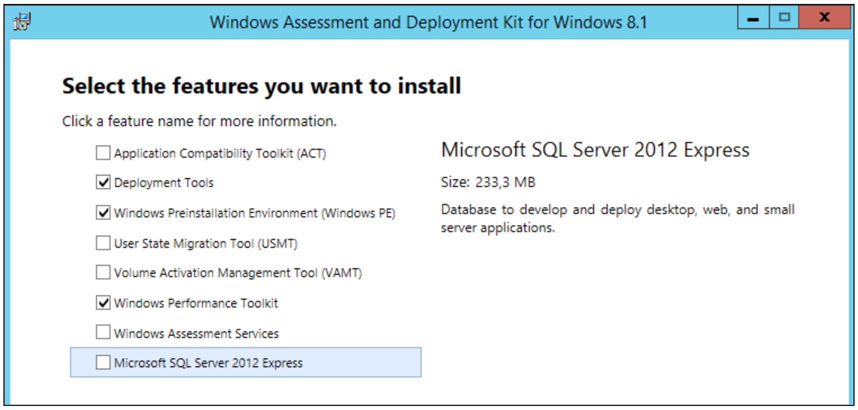
Download Path:

C:\Users\administrator.PDTIT\Downloads\Windows Kits\8.1\ADK

Estimated disk space required: 5,9 GB

Disk space available: 35,6 GB

3. Select the following features to be installed as a minimum for SCVMM:
    - **Deployment Tools**
    - **Windows Preinstallation Environment**
    - **Windows Performance Toolkit**



Windows Assessment and Deployment Kit for Windows 8.1

Select the features you want to install

Click a feature name for more information.

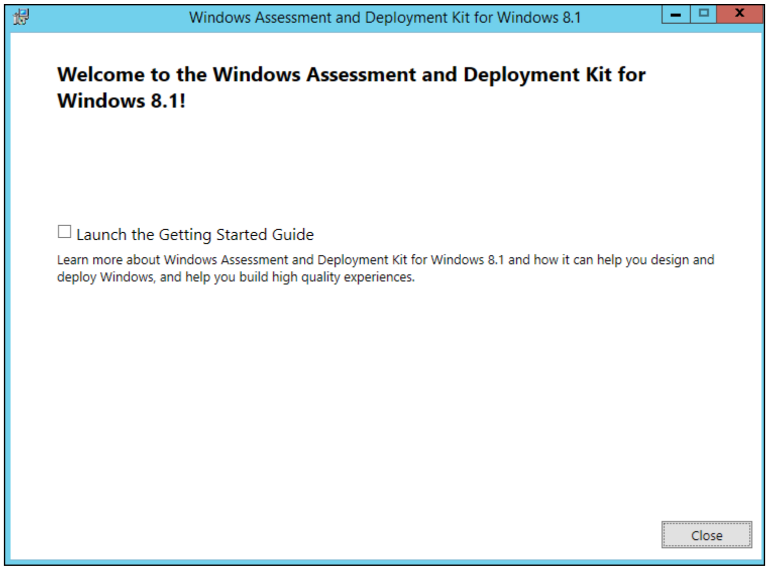
- □ Application Compatibility Toolkit (ACT)
- ☑ Deployment Tools
- ☑ Windows Preinstallation Environment (Windows PE)
- □ User State Migration Tool (USMT)
- □ Volume Activation Management Tool (VAMT)
- ☑ Windows Performance Toolkit
- □ Windows Assessment Services
- □ Microsoft SQL Server 2012 Express

Microsoft SQL Server 2012 Express

Size: 233,3 MB

Database to develop and deploy desktop, web, and small server applications.

4. Clicking on **Next** will install the WADK; at the end of the installation, close the setup wizard without any further configuration required:



Windows Assessment and Deployment Kit for Windows 8.1

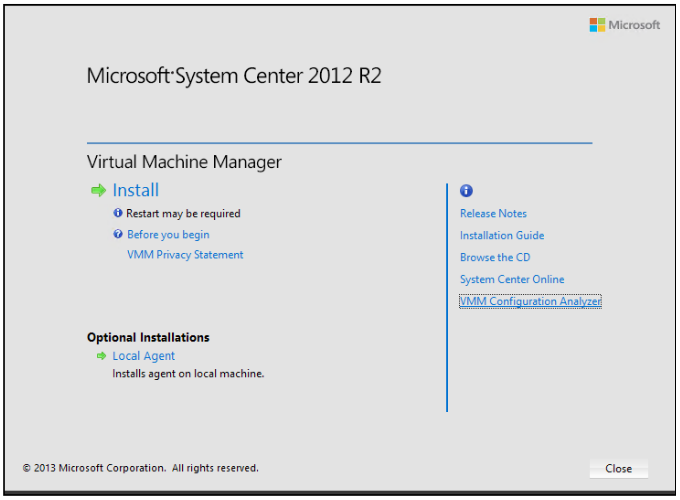
Welcome to the Windows Assessment and Deployment Kit for Windows 8.1!

□ Launch the Getting Started Guide

Learn more about Windows Assessment and Deployment Kit for Windows 8.1 and how it can help you design and deploy Windows, and help you build high quality experiences.

Now, the WADK is installed and we can continue with installing SCVMM itself, including the SCVMM console as follows:

1. Mount the SCVMM ISO file and launch setup. This will launch the SCVMM installation wizard:



Microsoft System Center 2012 R2

Virtual Machine Manager

Install

Restart may be required

Before you begin

VMM Privacy Statement

Release Notes

Installation Guide

Browse the CD

System Center Online

VMM Configuration Analyzer

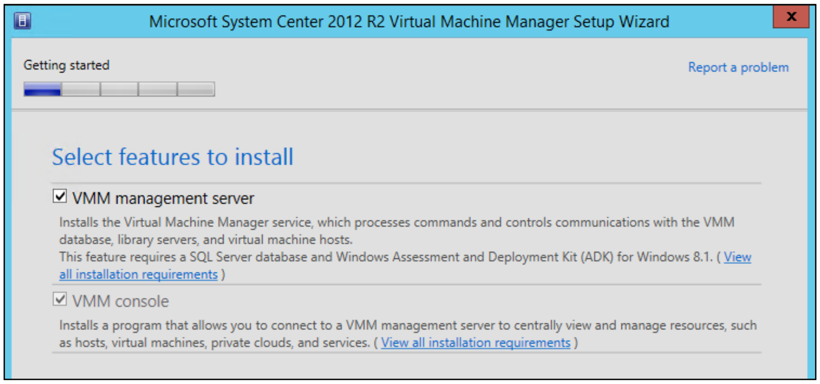
Optional Installations

Local Agent

Installs agent on local machine.

© 2013 Microsoft Corporation. All rights reserved.

2. Click on **Install** to continue:



Microsoft System Center 2012 R2 Virtual Machine Manager Setup Wizard

Getting started

Report a problem

Select features to install

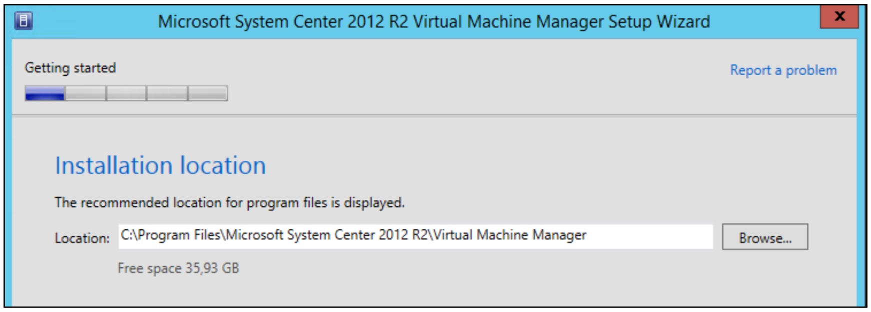
☑ VMM management server

Installs the Virtual Machine Manager service, which processes commands and controls communications with the VMM database, library servers, and virtual machine hosts.
This feature requires a SQL Server database and Windows Assessment and Deployment Kit (ADK) for Windows 8.1. (View all installation requirements)

☑ VMM console

Installs a program that allows you to connect to a VMM management server to centrally view and manage resources, such as hosts, virtual machines, private clouds, and services. (View all installation requirements)

3. Mark both selections for the installation; click on **Next** to go to the license agreement, and make your choices in the **Customer experience improvement program** step, as well as selecting if you want SCVMM to be updated by using Windows update or not:



Microsoft System Center 2012 R2 Virtual Machine Manager Setup Wizard

Getting started

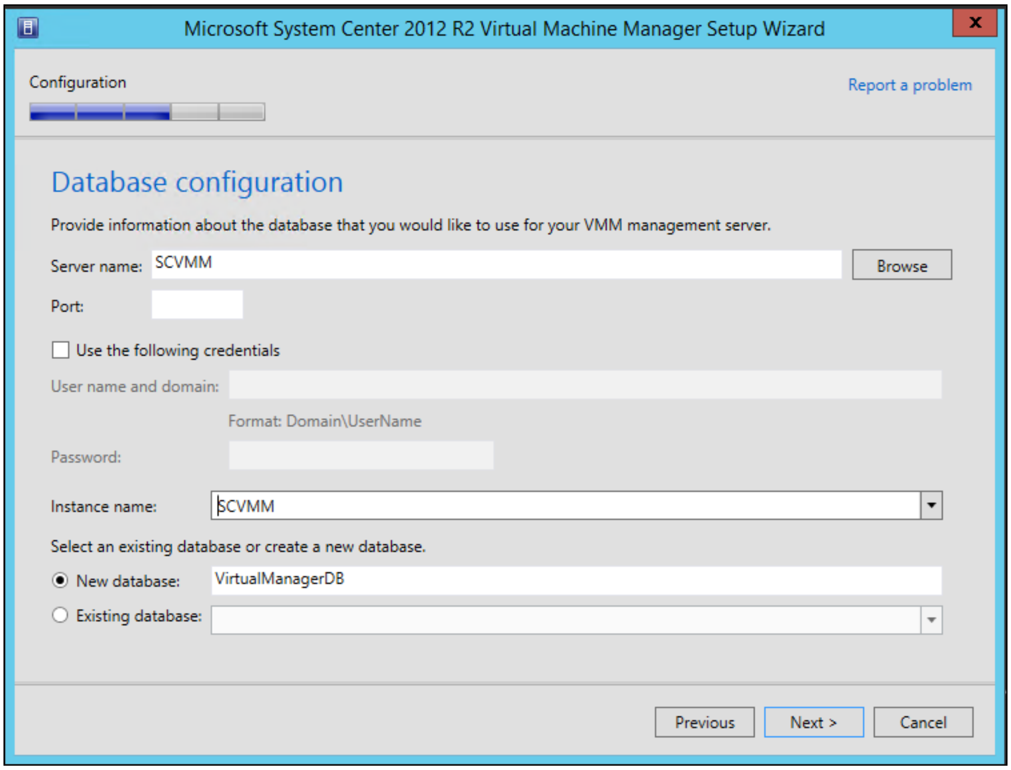
Installation location

The recommended location for program files is displayed.

Location: C:\Program Files\Microsoft System Center 2012 R2\Virtual Machine Manager

Free space 35,93 GB

4. Next, accept or modify the default installation location to your requirements:



Microsoft System Center 2012 R2 Virtual Machine Manager Setup Wizard

Configuration

Database configuration

Provide information about the database that you would like to use for your VMM management server.

Server name: SCVMM

Port:

□ Use the following credentials

User name and domain:

Format: Domain\UserName

Password:

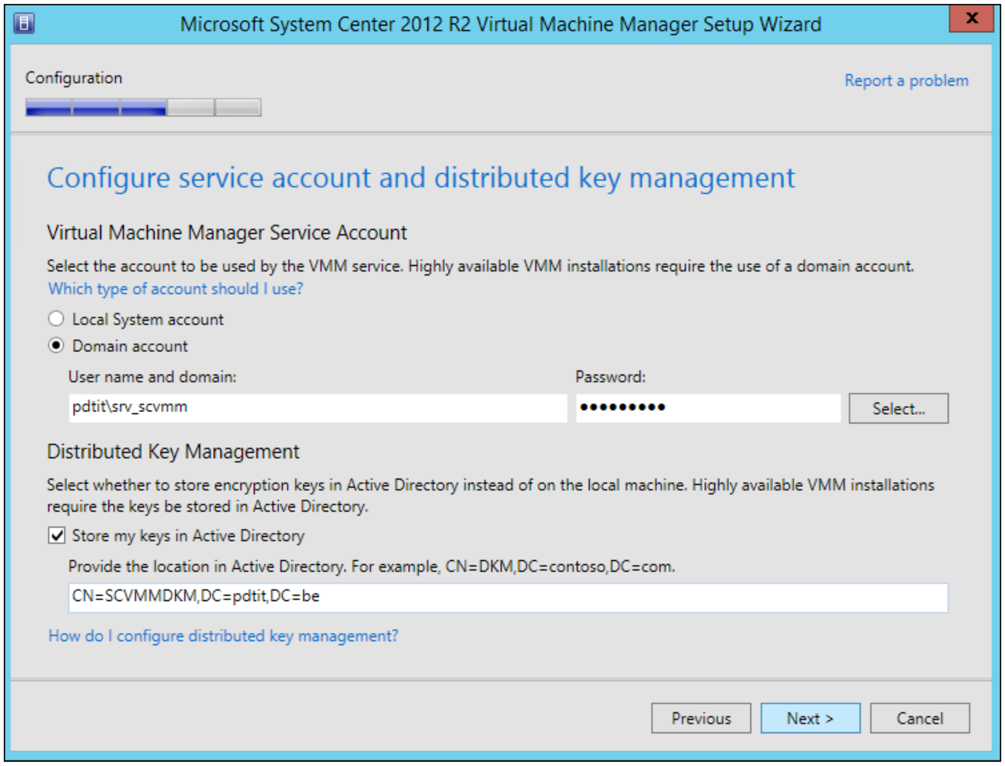
Instance name: SCVMM

Select an existing database or create a new database.

● New database: VirtualManagerDB

○ Existing database:

5. In the **Database configuration** step, enter the server name of the SQL server and the name of the SQL instance you created before (in our example, we installed SQL Server 2012 on the same machine as the SCVMM management server); accept or modify the default database name to your needs:



Microsoft System Center 2012 R2 Virtual Machine Manager Setup Wizard
Configuration
Report a problem

Configure service account and distributed key management

Virtual Machine Manager Service Account

Select the account to be used by the VMM service. Highly available VMM installations require the use of a domain account.

Which type of account should I use?

○ Local System account

● Domain account

User name and domain: pdtit\srv_scvmm

Password: •••••••••

Distributed Key Management

Select whether to store encryption keys in Active Directory instead of on the local machine. Highly available VMM installations require the keys be stored in Active Directory.

☑ Store my keys in Active Directory

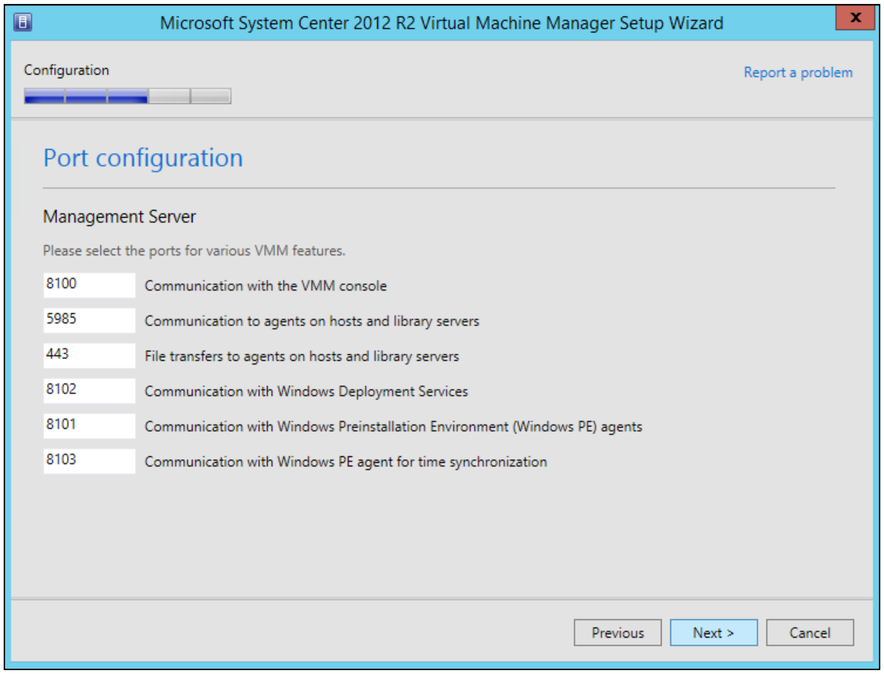
Provide the location in Active Directory. For example, CN=DKM,DC=contoso,DC=com.

CN=SCVMMDKM,DC=pdtit,DC=be

How do I configure distributed key management?

6. In the next step, you can specify to use a specific SCVMM service account, as well as if you want to store the encryption keys in the Active Directory (we will talk about this more in detail in the next few pages, as well as how to configure this; if you are deploying SCVMM in a high availability setup, storing the encryption keys in AD is required. Otherwise, they are stored on the local machine).

   In our environment here, I created an Active Directory domain administrator account called `srv_scvmm`, which I will use here as the SCVMM service account.



Microsoft System Center 2012 R2 Virtual Machine Manager Setup Wizard

Configuration

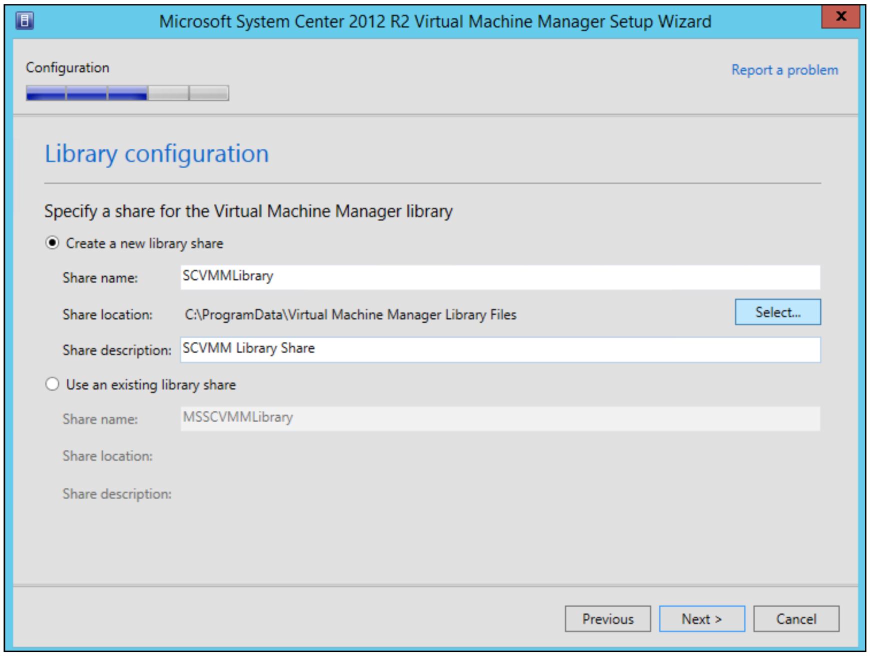
Port configuration

Management Server

Please select the ports for various VMM features.

8100	Communication with the VMM console
5985	Communication to agents on hosts and library servers
443	File transfers to agents on hosts and library servers
8102	Communication with Windows Deployment Services
8101	Communication with Windows Preinstallation Environment (Windows PE) agents
8103	Communication with Windows PE agent for time synchronization

7. Accept or modify the different IP ports for the different SCVMM components:



Microsoft System Center 2012 R2 Virtual Machine Manager Setup Wizard

Configuration

Library configuration

Specify a share for the Virtual Machine Manager library

● Create a new library share

Share name: SCVMMLibrary

Share location: C:\ProgramData\Virtual Machine Manager Library Files

Share description: SCVMM Library Share

○ Use an existing library share

Share name: MSSCVMMLibrary

Share location:

Share description:

8. In the next step of the SCVMM installation, you have to configure the VMM library settings. You have to specify a share name, shared location, and description.
9. In the final steps of the installation wizard, a summary of all your settings is shown, after which the installation itself will start. This should take anywhere from 10 to 20 minutes, depending on your hardware, and hopefully end in a successful state.

Once the installation is complete, you can launch the SCVMM management console.

Now, before we continue with the detailed walk-through of the actual SCVMM console, I promised you a more detailed explanation of **Distributed Key Management** (**DKM**) to store your encryption details in the Active Directory.

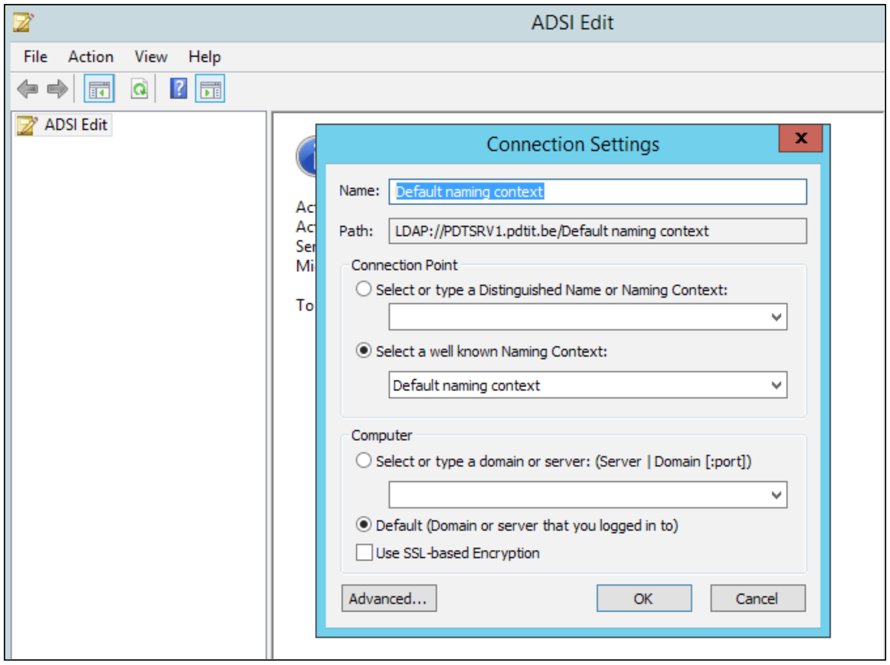
Distributed Key Management

A few pages ago, during setting the installation options of the SCVMM management server, one of the important options that one could activate was using the Active Directory for the DKM. Now, what the heck is that DKM?

Within SCVMM, some information and parameters such as Run As account passwords are pretty sensitive information, which needs to be stored in an encrypted way. Out of the box, SCVMM stores this information on the VMM server itself. While this is OK for single server scenarios, it might cause issues in a multiserver, high availability scenario. That's where the Active Directory comes in play as a storage location for these encryption keys. Imagine you are losing your SCVMM server due to disk failure, having to rebuild your server for whatever reason, and so on, you still have access to the encrypted data, out of the Active Directory.

A minor issue is that there is a little manual configuration required within the Active Directory to create the DKM container, prior to continuing the configuration of SCVMM management server itself as follows:

1. From any domain controller in your Active Directory domain, start **ADSI EDIT**.
2. From within the **ADSI Edit** configuration tool, right-click on **ADSI Edit**, chose **Connect to...**, and select **Default naming context**:



Connection Settings

Name: Default naming context

Path: LDAP://PDTSRV1.pdtit.be/Default naming context

Connection Point

○ Select or type a Distinguished Name or Naming Context:

● Select a well known Naming Context:

Default naming context

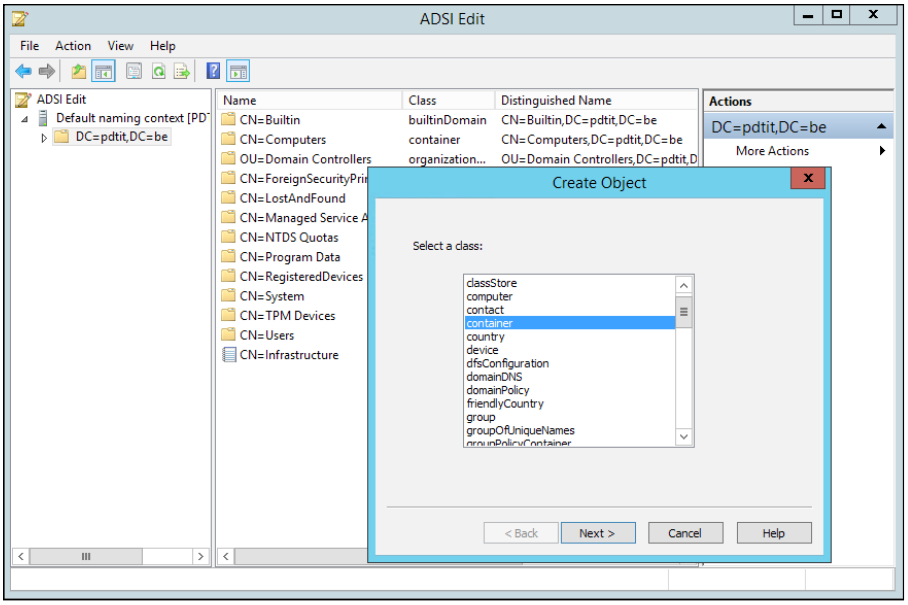
Computer

○ Select or type a domain or server: (Server | Domain [:port])

● Default (Domain or server that you logged in to)

□ Use SSL-based Encryption

3. This will open up your Active Directory container and OU topology. From the top of your domain, right-click, choose **New object...**, click on **Container**, and give it a descriptive name such as `SCVMMDKM` in my example:



Name	Class	Distinguished Name
CN=Builtin	builtinDomain	CN=Builtin,DC=pdtit,DC=be
CN=Computers	container	CN=Computers,DC=pdtit,DC=be
OU=Domain Controllers	organization...	OU=Domain Controllers,DC=pdtit,D
CN=ForeignSecurityPri		
CN=LostAndFound		
CN=Managed Service A		
CN=NTDS Quotas		
CN=Program Data		
CN=RegisteredDevices		
CN=System		
CN=TPM Devices		
CN=Users		
CN=Infrastructure		

Create Object

Select a class:

classStore
computer
contact
container
country
device
dfsConfiguration
domainDNS
domainPolicy
friendlyCountry
group
groupOfUniqueNames
groupPolicyContainer

4. Close ADSI Edit and continue with the installation of the SCVMM manager. Enter the distinguished name of the location of this container in the DKM - Active Directory field.

Once the installation is completed successfully, when you go back into your **Active Directory Users and Computers** (**ADUC**), while having the **Advanced Features** option active in the **View** settings, you will see certain objects being created in the `SCVMMDKM` container.

> Several Internet sources are talking about the fact that SCVMM can create this container automatically upon entering the distinguished name of the location, but I've seen this failing during certain installs. That's why I prefer creating this container manually, allowing for AD replication to complete, and avoiding errors during the SCVMM installation itself.

Managing Hyper-V hosts

Once the SCVMM management server is up and running, it will become your central management solution for the Hyper-V platform (and other hypervisors such as VMware ESX and Citrix XenServer, as you will see in the next chapter). To ease the administration of hosts, virtual machines, network, and storage components, SCVMM can be structured in host groups that contain hosts. Properties you configure on host group level will be applied to all host members in that specific group.

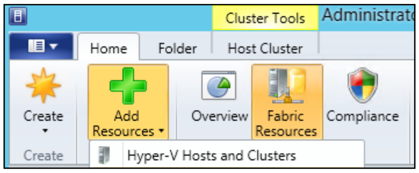
In order to manage Hyper-V hosts from within SCVMM, the SCVMM agent must be installed on the Hyper-V host. This can be done from the VMM console to all Hyper-V hosts that are a member of the same Active Directory domain as the VMM management server. Installing the SCVMM agent on machines in the perimeter network requires a manual installation of the agent.

We will go through both configuration scenarios right away.

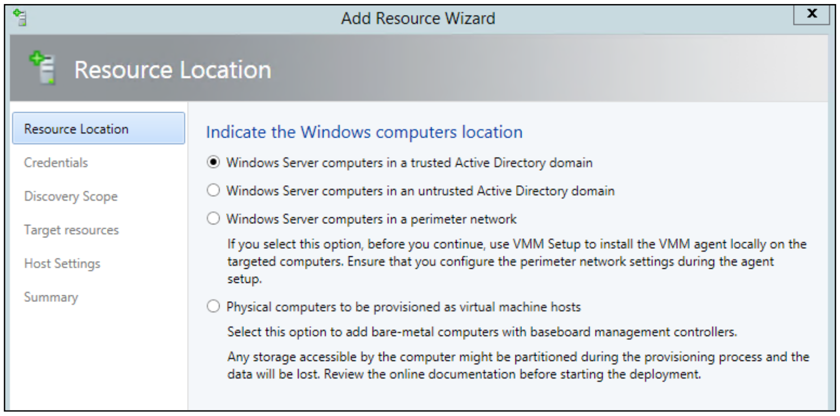
Installing the SCVMM agent on a domain member Hyper-V host

The steps are as follows:

1. From within the SCVMM management console, select **Fabric**. Within the **Fabric** pane, browse to **Servers**, and then click on **All hosts**. From the ribbon above, click on **Add Resources** and then click on **Hyper-V Hosts and Clusters**:



2. Make the proper selection of your Hyper-V hosts from the next step in the **Add Resource Wizard**. In our environment, we select the **Windows Server computers in a trusted Active Directory domain** option:



Indicate the Windows computers location

● Windows Server computers in a trusted Active Directory domain

○ Windows Server computers in an untrusted Active Directory domain

○ Windows Server computers in a perimeter network

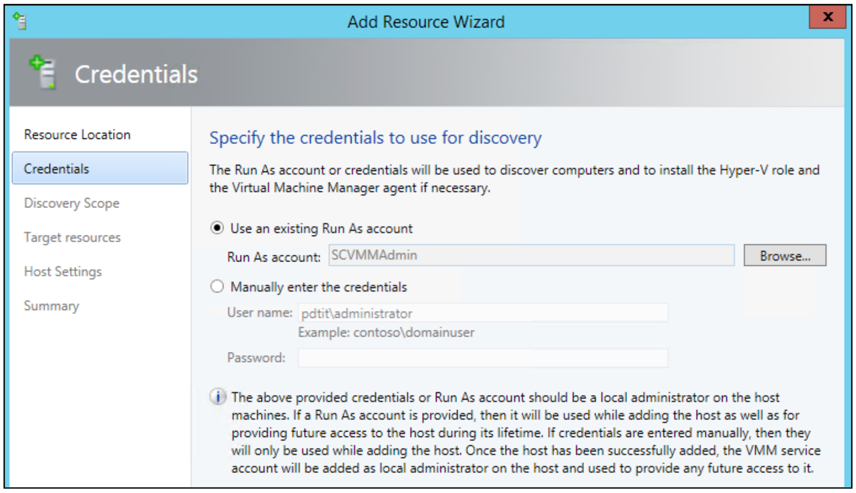
If you select this option, before you continue, use VMM Setup to install the VMM agent locally on the targeted computers. Ensure that you configure the perimeter network settings during the agent setup.

○ Physical computers to be provisioned as virtual machine hosts

Select this option to add bare-metal computers with baseboard management controllers.

Any storage accessible by the computer might be partitioned during the provisioning process and the data will be lost. Review the online documentation before starting the deployment.

3. In the next step, you can specify a Run As account or provide user account credentials directly. In our example, we select the Run As account option, as we will use the SRV_SCVMM account we created earlier, prior to the SCVMM installation:



Specify the credentials to use for discovery

The Run As account or credentials will be used to discover computers and to install the Hyper-V role and the Virtual Machine Manager agent if necessary.

● Use an existing Run As account

Run As account: SCVMMAdmin

○ Manually enter the credentials

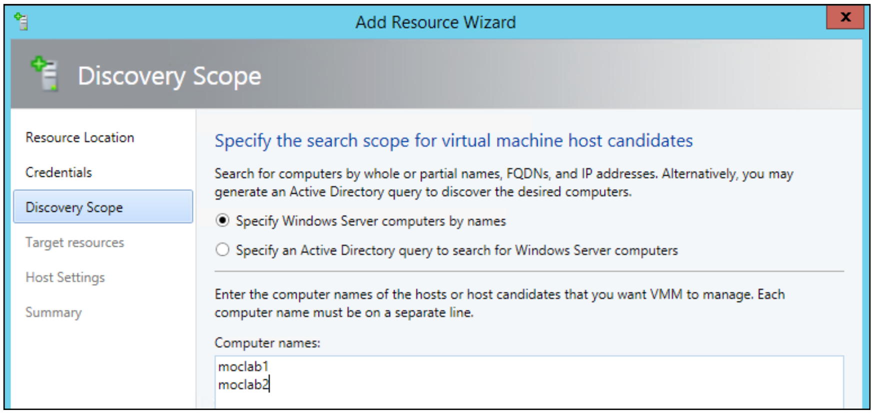
User name: pdtit\administrator

Example: contoso\domainuser

Password:

The above provided credentials or Run As account should be a local administrator on the host machines. If a Run As account is provided, then it will be used while adding the host as well as for providing future access to the host during its lifetime. If credentials are entered manually, then they will only be used while adding the host. Once the host has been successfully added, the VMM service account will be added as local administrator on the host and used to provide any future access to it.

4. In the **Discovery Scope** step, enter the host names of the Hyper-V hosts, or select the option to use an Active Directory search. In our example, the easiest way is to enter the Hyper-V host names directly, as we only have two hosts running:



Specify the search scope for virtual machine host candidates

Search for computers by whole or partial names, FQDNs, and IP addresses. Alternatively, you may generate an Active Directory query to discover the desired computers.

● Specify Windows Server computers by names

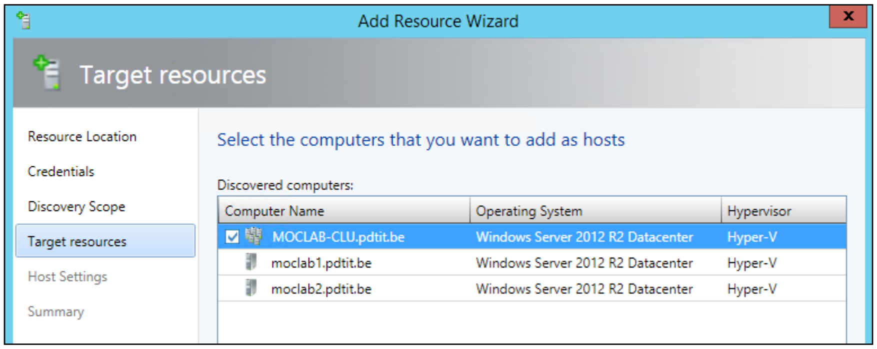
○ Specify an Active Directory query to search for Windows Server computers

Enter the computer names of the hosts or host candidates that you want VMM to manage. Each computer name must be on a separate line.

Computer names:

moclab1
moclab2

5. As you might remember from the previous modules, both machines are members of a Hyper-V cluster, which is visible as well in the window in the following screenshot:

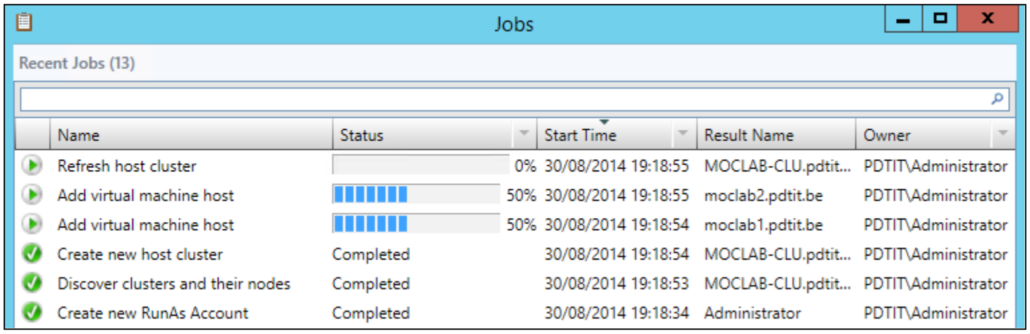


Select the computers that you want to add as hosts

Discovered computers:

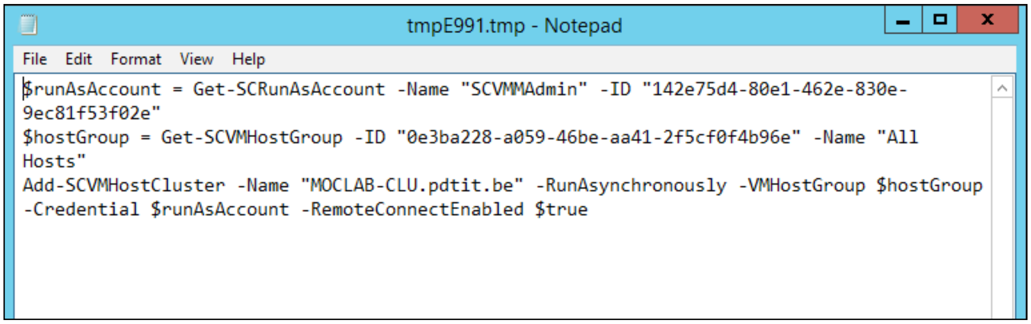
Computer Name	Operating System	Hypervisor
☑ MOCLAB-CLU.pdtit.be	Windows Server 2012 R2 Datacenter	Hyper-V
moclab1.pdtit.be	Windows Server 2012 R2 Datacenter	Hyper-V
moclab2.pdtit.be	Windows Server 2012 R2 Datacenter	Hyper-V

6. After selecting your target resources, allocate the hosts to a specific host group (we will talk about host groups later on in this chapter) and close the **Add Resource Wizard** by clicking on the **Finish** button. This will jump you to the **Jobs** window, where the different tasks that are running out of the selections you made are visible as follows:



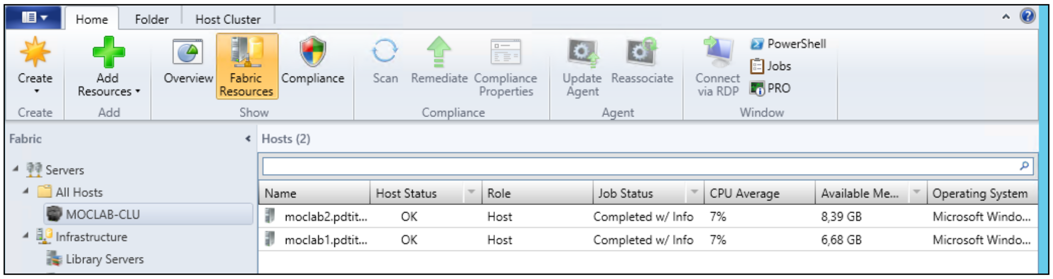
Name	Status	Start Time	Result Name	Owner
Refresh host cluster	0%	30/08/2014 19:18:55	MOCLAB-CLU.pdtit...	PDTIT\Administrator
Add virtual machine host	50%	30/08/2014 19:18:55	moclab2.pdtit.be	PDTIT\Administrator
Add virtual machine host	50%	30/08/2014 19:18:54	moclab1.pdtit.be	PDTIT\Administrator
Create new host cluster	Completed	30/08/2014 19:18:54	MOCLAB-CLU.pdtit...	PDTIT\Administrator
Discover clusters and their nodes	Completed	30/08/2014 19:18:53	MOCLAB-CLU.pdtit...	PDTIT\Administrator
Create new RunAs Account	Completed	30/08/2014 19:18:34	Administrator	PDTIT\Administrator

7. It is interesting to note the **View Script** button at the top right-hand side of the wizard window; this will show us the equivalent PowerShell script that is used to add the Hyper-V hosts to this SCVMM server:



```
$runAsAccount = Get-SCRunAsAccount -Name "SCVMMAdmin" -ID "142e75d4-80e1-462e-830e-
9ec81f53f02e"
$hostGroup = Get-SCVMHostGroup -ID "0e3ba228-a059-46be-aa41-2f5cf0f4b96e" -Name "All
Hosts"
Add-SCVMHostCluster -Name "MOCLAB-CLU.pdtit.be" -RunAsynchronously -VMHostGroup $hostGroup
-Credential $runAsAccount -RemoteConnectEnabled $true
```

8. Once the job is completed successfully, the Hyper-V cluster and hosts we selected are visible under the **Hosts** pane:



Name	Host Status	Role	Job Status	CPU Average	Available Me...	Operating System
moclab2.pdtit...	OK	Host	Completed w/ Info	7%	8,39 GB	Microsoft Windo...
moclab1.pdtit...	OK	Host	Completed w/ Info	7%	6,68 GB	Microsoft Windo...

This concludes the initial addition of a Hyper-V host and the corresponding virtual machines. From here, you can manage and review host properties, as well as manage virtual machine settings. We will talk more about some of these features in the following subsections.

Two of the other important fabric components within SCVMM are networking and storage. Using these components, SCVMM eases the configuration and management of network and storage infrastructure, providing you with a centralized hub for the underlying virtual machines, services, and clouds within our virtualized data center topology. In larger enterprise-oriented data centers, managing these components can already be a real challenge, when only thinking about the physical layer. When adding virtual hosts, virtual machines, virtual networks, and virtual storages on top of this already complex layer, one might lose the administrative control. One of the advantages here is that by using SCVMM, one can take control and manage these resources almost completely from SCVMM. For example, SCVMM allows for a direct integration with several third-party solutions such as Cisco's Nexus virtual switches.

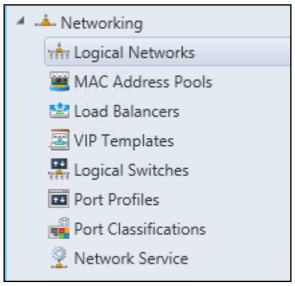
Talking about storage integration, SCVMM leverages, for example, the Storage Spaces feature of Windows Server 2012 R2, allowing for deduplication, storage tiering, and more.

The storage fabric itself can handle Fiber Channel storage, iSCSI storage, and plain file shares (SMB 3.0), thus allowing a true integration between your already existing storage infrastructure and SCVMM, or optimizing the storage infrastructure right out of SCVMM.

Both topics will be discussed more in detail hereafter.

Managing the network fabric

In this section, you will learn about the key features of the networking fabric in SCVMM. When selecting this fabric in the console, it looks like this:



Before we continue with the more technical details, let's start by giving a short description of each of the sections as follows:

Section	Description
Logical Networks	A logical network is a collection of network resources that are related to each other. For example, you can create a production logical network, a test logical network, a DMZ logical network, and so on. Logical networks can be associated with hosts and host groups. Configuring logical networks to separate virtual machines helps the administrators or even nonadministrative users in your environment to deploy new virtual machines, without needing to understand or manage the networking topology behind it.
MAC Address Pools	By using the MAC address pools, SCVMM allows you to manage static MAC addresses per hypervisor. A distinction has been created between Hyper-V, ESX, and VMware MAC address pools. In most cases, the MAC addresses that are generated for the virtual machines are created dynamically. However, you also have an option to manually configure static MAC addresses for your VMs.
Load Balancers	By using load balancing, you can provide high availability to your virtual machines' service connections. When installing SCVMM, it automatically creates the Windows NLB network load balancer for you. SCVMM also allows you to integrate third-party load balancers into the SCVMM management console. This requires you to install a so-called configuration provider from the vendor onto the SCVMM management server. At the time of writing this book, supported vendors and models are F5 BIG IP, Citrix NetScaler, and Brocade. For more information on load balancers within SCVMM, we would like to refer you to the Microsoft TechNet article at `http://technet.microsoft.com/en-us/library/gg610639.aspx`.
VIP Templates	Virtual IP templates are configuration sets of both a load balancer and a network service, for example, HTTPS. These templates can be used by administrators or nonadministrative users to link them to virtual machines and corresponding applications.
Logical Switches	A logical switch allows for centralized management and configuration of host networking settings. You can configure logical to Hyper-V port mappings, port classifications, and virtual switch extensions. For detailed information regarding virtual logical switches and port configuration, have a look at the Microsoft TechNet article at `http://technet.microsoft.com/en-us/library/jj721570.aspx`.
Port Profiles	By creating port profiles, consistent network settings can be applied to hosts and virtual machines, from SCVMM. By default, different port profiles are already preinstalled. The main advantage of using port profiles is it allows you to define specific settings for specific networks or network interfaces or hosts or virtual machines, after which you can specify how and to which resources these settings should be applied.

Section	Description
Port Classifications	This is a logical group of ports with the same settings configured. For example, 10 Gbit port profiles could be grouped as "10 Gbit Ports" under **Port Classifications**.
Network Service	This is the section within SCVMM where you can add third-party component integration such as gateways, virtual switch extensions, Top-Of-Rack switch configurations, and so on.

From the aforementioned table, it should be clear that SCVMM has a lot of options and possible configurations that relate to the network fabric. Discussing all the details about each topic, explaining more in detail what it is used for, how to use it, and configure it from within the SCVMM management console or PowerShell would be way too much out of the scope of this book.

In June 2014, Packt Publishing released *System Center 2012 R2 Virtual Machine Manager Cookbook Second Edition*, *Edvaldo Alessandro Cardoso*, which talks about all these features in much more detail than can be covered here. For more information, please have a look at their website at `https://www.packtpub.com/virtualization-and-cloud/system-center-2012-r2-virtual-machine-manager-cookbook-second-edition`.

An alternative to the preceding book might be this blog post from Aidan Finn, a Microsoft MVP on Hyper-V, talking about SCVMM and networking configurations and features. It can be accessed from `http://www.aidanfinn.com/?p=14694` (part 1) and `http://www.aidanfinn.com/?p=14727` (part 2).

A very interesting aspect of the networking fabric we personally like a lot is the built-in **network isolation**. By using network isolation, one can completely separate the network traffic for different virtual machines, hosts, and even customers if you want. Imagine you are the virtualized data center admin for a hosting company, providing virtual machine environments for multiple customers.

By using network isolation, different customers' virtual machines can be hosted next to one another, having the same IP and other networking settings, without actually causing overlap or conflicts. In this configuration, each virtual machine has two IP addresses, one that is bound to the customer's network and is being used by the services and/or applications the customer is using, and another IP address that is known as the provider IP address and is used for management. This provider IP is not visible to the virtual machine itself.

One last addition to the network fabric within SCVMM is a rather new tool, released by the Microsoft SCVMM product team during summer 2014, which is the **SCVMM Network Builder**.

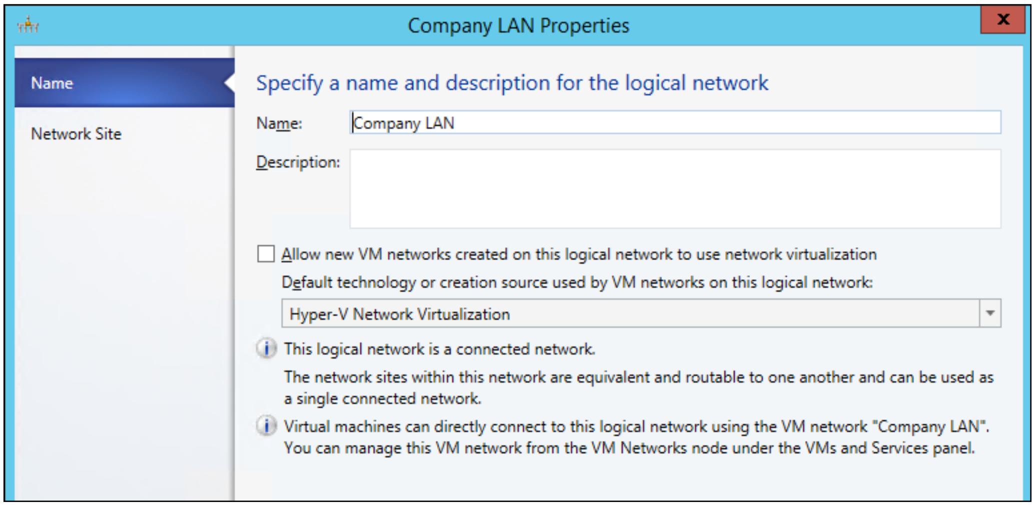
This free add-in to SCVMM allows you to get started quickly in building your VLAN network layout. From the add-in, you can create the overall topology of your virtual networks, VLANs, virtual switches, and such, which afterwards can be applied to hosts and virtual machines.

An important side note about this tool is that it is not intended to create configuration settings for NVGRE-based configurations, but only to be used for VLAN isolation configurations.

The best blog post we could find at the time of writing was the one available at `http://charbelnemnom.com/2014/08/install-and-configure-vmm-network-builder-scvmm-sysctr-cloud/`, which clearly explains what the add-in does, how to integrate it in SCVMM, and how to use it.

As this is still a practical how-to book, it would not be OK to not walk you through some of the mentioned networking features of SCVMM from within our lab environment. Let's look at them step-by-step:

1. From within the SCVMM console, select **Fabric** | **Networking** | **Logical Networks**. In our environment, two logical networks are already visible, **Company LAN** and **Internet**. As you might remember, we have created these in the Hyper-V virtual switch configurations in the previous section.
2. Select your internal network's logical network (**Company LAN** in our setup). Right-click on it and then click on **Properties**. This will open a window showing you the specific details of this logical network:



Specify a name and description for the logical network

Name: Company LAN

Description:

□ Allow new VM networks created on this logical network to use network virtualization

Default technology or creation source used by VM networks on this logical network:

Hyper-V Network Virtualization

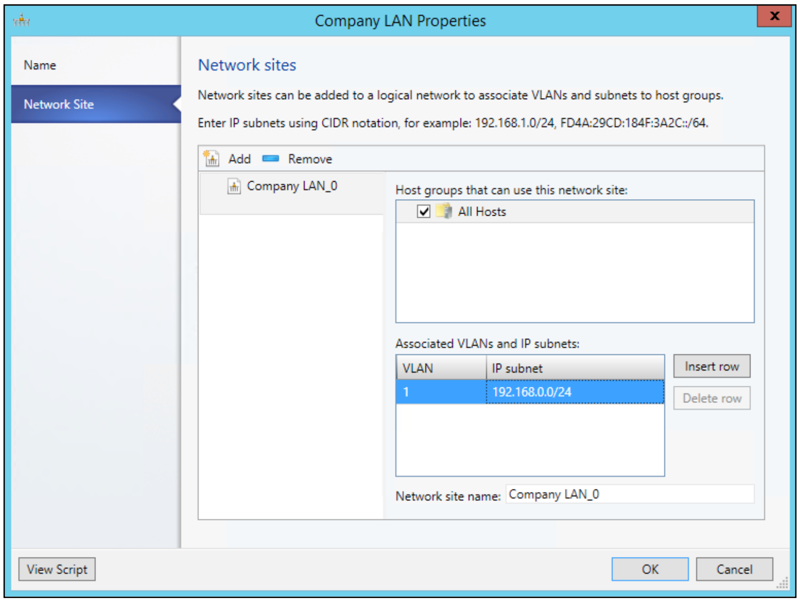
This logical network is a connected network.

The network sites within this network are equivalent and routable to one another and can be used as a single connected network.

Virtual machines can directly connect to this logical network using the VM network "Company LAN". You can manage this VM network from the VM Networks node under the VMs and Services panel.

3. Select the **Network Site** pane; as mentioned earlier, network sites can be used to group logical network settings, associate VLANs, and subnets, and apply them to host groups.

   Let's configure a specific VLAN "VLAN ID 1" for our internal IP subnet `192.168.0.0` in our lab, linking this to the **All Hosts** group. Note that you have to enter the subnet in CIDR notation (subnet/mask) as shown in the following screenshot:



Company LAN Properties

Network sites

Network sites can be added to a logical network to associate VLANs and subnets to host groups.

Enter IP subnets using CIDR notation, for example: 192.168.1.0/24, FD4A:29CD:184F:3A2C::/64.

Host groups that can use this network site:

☑ All Hosts

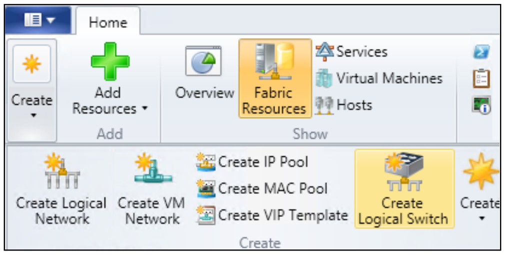
Associated VLANs and IP subnets:

VLAN	IP subnet
1	192.168.0.0/24

Network site name: Company LAN_0

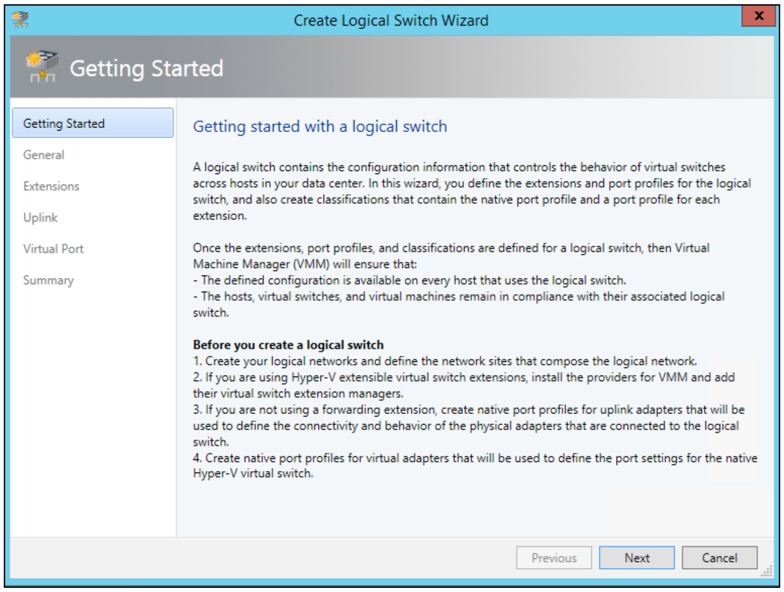
   After this, let's add a new logical switch, taking additional settings such as port profiles into consideration as well during the configuration steps.

4. Select **Fabric** | **Networking** | **Logical Switch**. From within the ribbon, click on the **Create** button, followed by clicking on the **Create Logical Switch** button as shown in the following screenshot:



A shorter way could also be to right-click on the **Logical Switch** topic.

The **Create Logical Switch Wizard** comes up:



Create Logical Switch Wizard

Getting Started

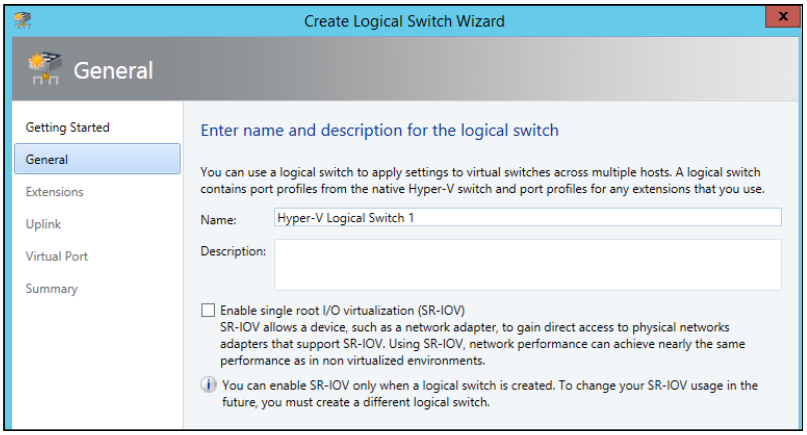
Getting started with a logical switch

A logical switch contains the configuration information that controls the behavior of virtual switches across hosts in your data center. In this wizard, you define the extensions and port profiles for the logical switch, and also create classifications that contain the native port profile and a port profile for each extension.

Once the extensions, port profiles, and classifications are defined for a logical switch, then Virtual Machine Manager (VMM) will ensure that:
- The defined configuration is available on every host that uses the logical switch.
- The hosts, virtual switches, and virtual machines remain in compliance with their associated logical switch.

Before you create a logical switch
1. Create your logical networks and define the network sites that compose the logical network.
2. If you are using Hyper-V extensible virtual switch extensions, install the providers for VMM and add their virtual switch extension managers.
3. If you are not using a forwarding extension, create native port profiles for uplink adapters that will be used to define the connectivity and behavior of the physical adapters that are connected to the logical switch.
4. Create native port profiles for virtual adapters that will be used to define the port settings for the native Hyper-V virtual switch.

5. Click on **Next** to start the wizard; enter a name and optionally a description for the logical switch you are about to create, and then click on **Next**:



Create Logical Switch Wizard

General

Enter name and description for the logical switch

You can use a logical switch to apply settings to virtual switches across multiple hosts. A logical switch contains port profiles from the native Hyper-V switch and port profiles for any extensions that you use.

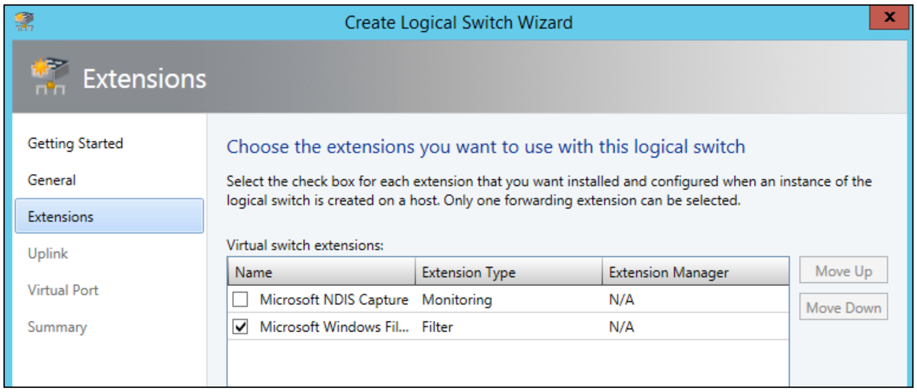
Name: Hyper-V Logical Switch 1

Description:

□ Enable single root I/O virtualization (SR-IOV)
SR-IOV allows a device, such as a network adapter, to gain direct access to physical networks adapters that support SR-IOV. Using SR-IOV, network performance can achieve nearly the same performance as in non virtualized environments.

You can enable SR-IOV only when a logical switch is created. To change your SR-IOV usage in the future, you must create a different logical switch.

6. In the next step, select the logical switch extension you want to use. Note that in our environment, the **Microsoft Windows Filter** option is already selected; click on **Next**:



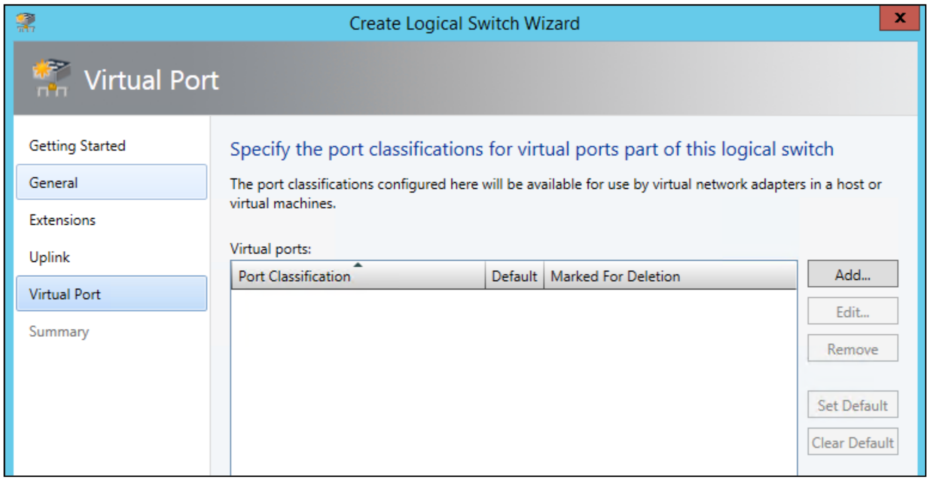
Choose the extensions you want to use with this logical switch

Select the check box for each extension that you want installed and configured when an instance of the logical switch is created on a host. Only one forwarding extension can be selected.

Virtual switch extensions:

Name	Extension Type	Extension Manager
□ Microsoft NDIS Capture	Monitoring	N/A
☑ Microsoft Windows Fil...	Filter	N/A

7. Don't specify any uplink settings in this example; continue by clicking on **Next**.
8. In the **Virtual Port** step of the wizard, we can create mappings to the already available port classifications by clicking on **Add**:



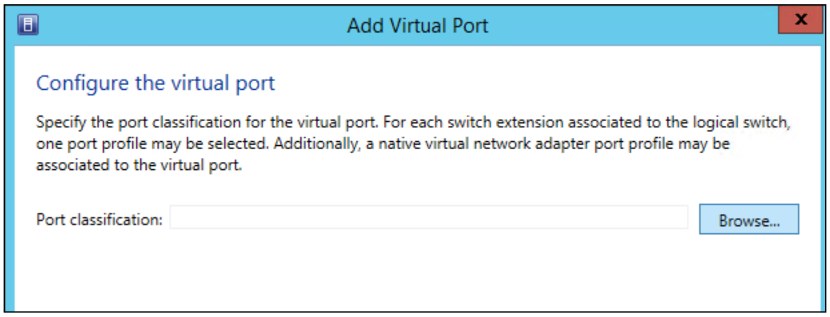
Specify the port classifications for virtual ports part of this logical switch

The port classifications configured here will be available for use by virtual network adapters in a host or virtual machines.

Virtual ports:

Port Classification	Default	Marked For Deletion

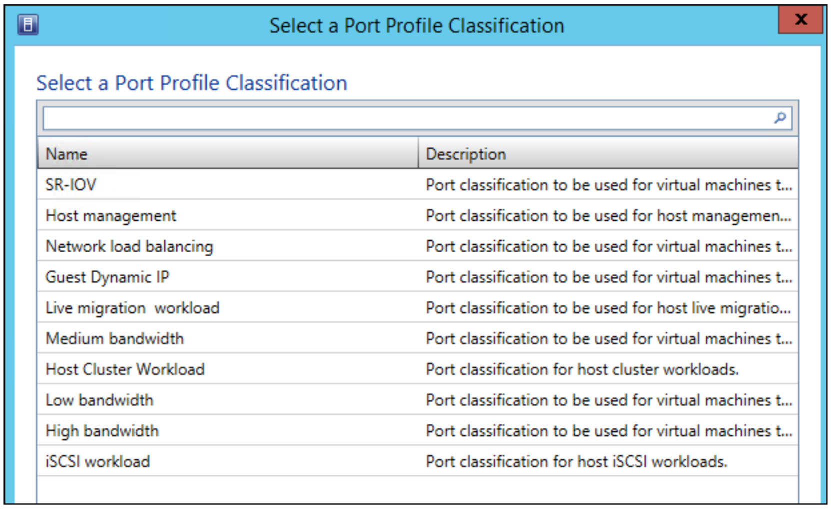
9. This is followed by clicking on **Browse**:



Configure the virtual port

Specify the port classification for the virtual port. For each switch extension associated to the logical switch, one port profile may be selected. Additionally, a native virtual network adapter port profile may be associated to the virtual port.

10. This will show us a list of the already existing port profiles:



Select a Port Profile Classification

Name	Description
SR-IOV	Port classification to be used for virtual machines t...
Host management	Port classification to be used for host managemen...
Network load balancing	Port classification to be used for virtual machines t...
Guest Dynamic IP	Port classification to be used for virtual machines t...
Live migration workload	Port classification to be used for host live migratio...
Medium bandwidth	Port classification to be used for virtual machines t...
Host Cluster Workload	Port classification for host cluster workloads.
Low bandwidth	Port classification to be used for virtual machines t...
High bandwidth	Port classification to be used for virtual machines t...
iSCSI workload	Port classification for host iSCSI workloads.

[Note that you can only select port profile classifications from this list that have been created already in the Networking fabric / Port Classification section before.]

Select any port classification you want to use in your example; click on **Next**.

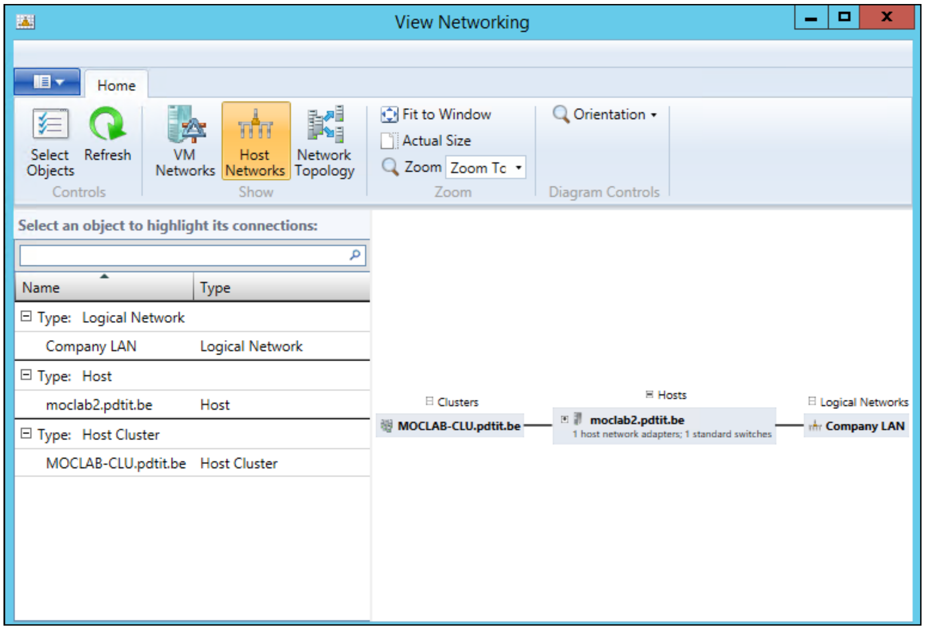
11. In the **Summary** window, click on **Finish**, after which your logical switch will be created; in between, the **Jobs** window pops up, showing you the details of the creation process.

This concludes the configuration of a logical switch within SCVMM.

One last interesting fact we want to show you about SCVMM and its relation to network settings is when viewing the properties of a Hyper-V host itself. Follow these steps:

1. From within SCVMM, select **Fabric** | **Servers** | **All Hosts** and then select your Hyper-V host; right-click on the host and then click on **View Networking**.

   This will bring up a specific pop-up window showing a lot of detailed and different views of the hosts' networking configuration settings and parameters, and how they relate to SCVMM objects:

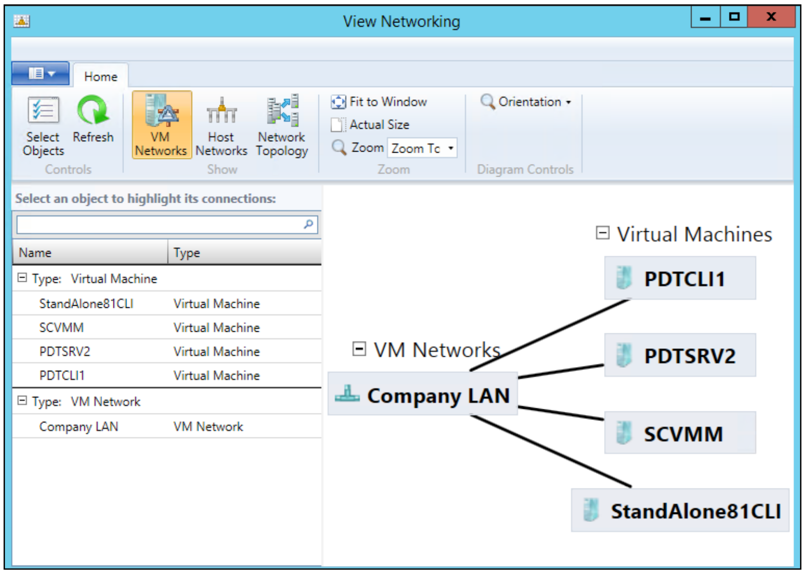


Select an object to highlight its connections:

Name	Type
⊟ Type: Logical Network	
Company LAN	Logical Network
⊟ Type: Host	
moclab2.pdtit.be	Host
⊟ Type: Host Cluster	
MOCLAB-CLU.pdtit.be	Host Cluster

   For example, we see the name of the Hyper-V host (MOCLAB Hyper-V cluster in our example), which logical network is linked to this host (Company LAN in our setup), as well as a simple network diagram, showing the relationship between the different layers.

2. From the ribbon, press the **VM Networks** button, which will result in another diagram, detailing the relation between the VM networks and virtual machines:



Select an object to highlight its connections:

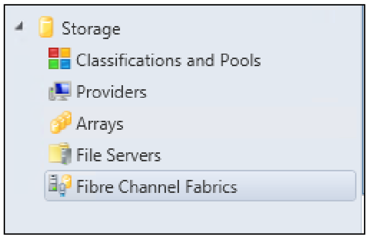
Name	Type
⊟ Type: Virtual Machine	
StandAlone81CLI	Virtual Machine
SCVMM	Virtual Machine
PDTSRV2	Virtual Machine
PDTCLI1	Virtual Machine
⊟ Type: VM Network	
Company LAN	VM Network

Managing the storage fabric

After discussing the network fabric, we would like to continue with providing you with insights into the storage fabric.

Just as with the previous topic, there is a lot of information we can talk about, as it is a very important aspect of SCVMM and your virtualized data center as a whole. However, mentioning all settings, configuration parameters, as well as showing you in detailed steps how the configuration is done is way out of the scope of this book. Therefore, let's refer to the other Packt Publishing book again, released in June 2014, which is, *System Center 2012 R2 Virtual Machine Manager Cookbook Second Edition*, *Edvaldo Alessandro Cardoso*, which talks about all these features way much more in detail than can be covered here. For more information, have a look at the book web page at `https://www.packtpub.com/virtualization-and-cloud/system-center-2012-r2-virtual-machine-manager-cookbook-second-edition`.

When opening up the storage fabric in SCVMM, it looks like this:

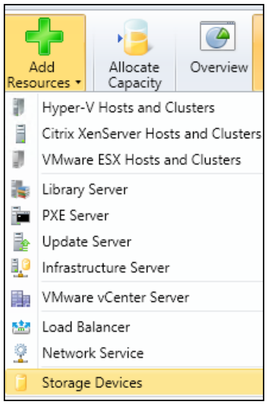


Before we continue with the more technical aspect of storage configuration in SCVMM, let's explain the different concepts here as follows:

Section	Description
Classifications and Pools	Storage classification allows you to link specific storage profiles (for example, based on different storage tiers, speeds, and so on).
Providers	This contains the configuration to your actual storage devices, which can be Fibre Channel or iSCSI-based SANs or NAS devices, or file servers.
Arrays	This contains the configuration to your actual storage devices, which can be Fibre Channel or iSCSI-based SANs or NAS devices, or file servers.
File Servers	If you have Windows-based file servers that are configured as storage solutions for your SCVMM environment, they will be visible here.
Fibre Channel Fabrics	This option will show the details and configuration settings of any Fibre Channel-based SAN, which is used as a storage endpoint for SCVMM.

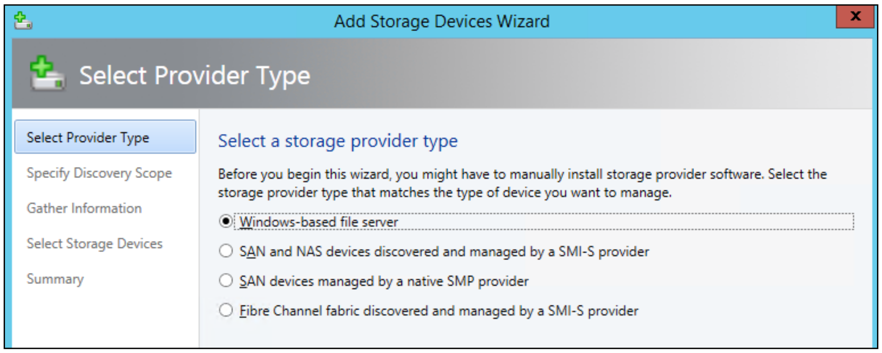
In the following example, let's create a storage integration with a Windows file server, specifying different storage classifications. The steps are as follows:

1. From within the SCVMM console, select **Fabric** | **Select Storage** | **Select File Servers**.
2. From the ribbon, select **Add Resources** | **Storage Devices** as shown here:



This will launch the **Add Storage Devices Wizard**.

3. In the first step of the wizard, select the **Windows-based file server** option as the storage provider type; click on **Next**:



Select a storage provider type

Before you begin this wizard, you might have to manually install storage provider software. Select the storage provider type that matches the type of device you want to manage.

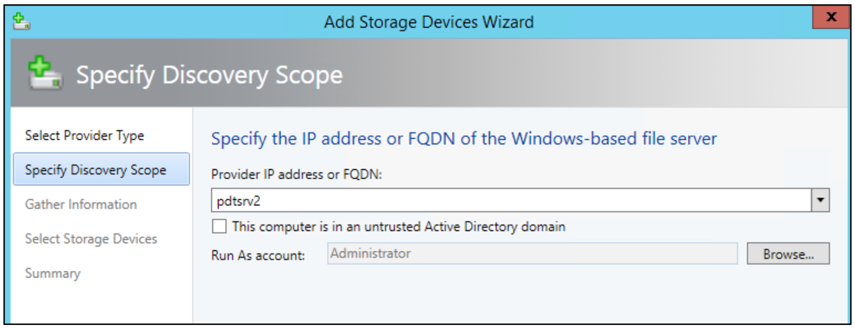
● Windows-based file server

○ SAN and NAS devices discovered and managed by a SMI-S provider

○ SAN devices managed by a native SMP provider

○ Fibre Channel fabric discovered and managed by a SMI-S provider

4. In the next step, enter the hostname or IP address of the Windows file server you want to configure as the storage endpoint for SCVMM. Select your appropriate **Run As account** and click on **Next**:



Specify the IP address or FQDN of the Windows-based file server

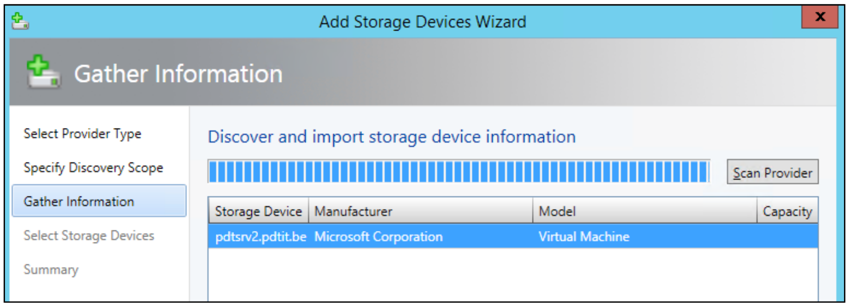
Provider IP address or FQDN:

pdtsrv2

□ This computer is in an untrusted Active Directory domain

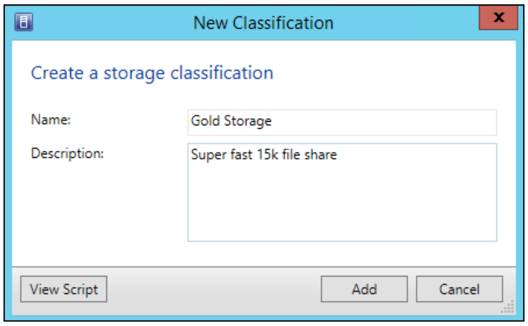
Run As account: Administrator

5. Once the file server is discovered, select it from the list and click on **Next**:



Storage Device	Manufacturer	Model	Capacity
pdtsrv2.pdtit.be	Microsoft Corporation	Virtual Machine	

6. In the next step, you have the option to create specific storage classifications, by pressing the **Create Classification…** button:



Name: Gold Storage

Description: Super fast 15k file share

7. Click on **Next** and then **Finish** to complete the configuration of the Windows file server and storage classifications.

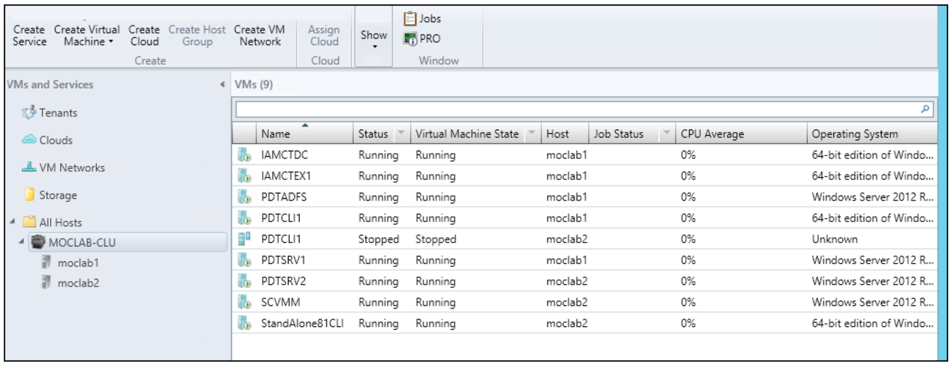
Deploying and managing virtual machines

At this moment, we have talked a lot about SCVMM as the overall management solution for Hyper-V and your centralized virtualized data center. We focused mainly on the network and storage fabric components, as well as showed you how to add Hyper-V hosts.

In this last section, you will learn how to deploy and manage your Hyper-V virtual machines from within SCVMM. Because, after all, this is one of the main tasks an administrator will probably perform from the SCVMM console or by using the corresponding SCVMM PowerShell cmdlets.

By using SCVMM, administrators can deploy new virtual machines both in a manual and in an automatic way—for example, by using a prebuilt template from the SCVMM library and allocating this VM to a specific Hyper-V host. Configuration features such as Hyper-V Replica, moving Hyper-V VMs, or Hyper-V VMs storage are also possible. Let's look at them step-by-step:

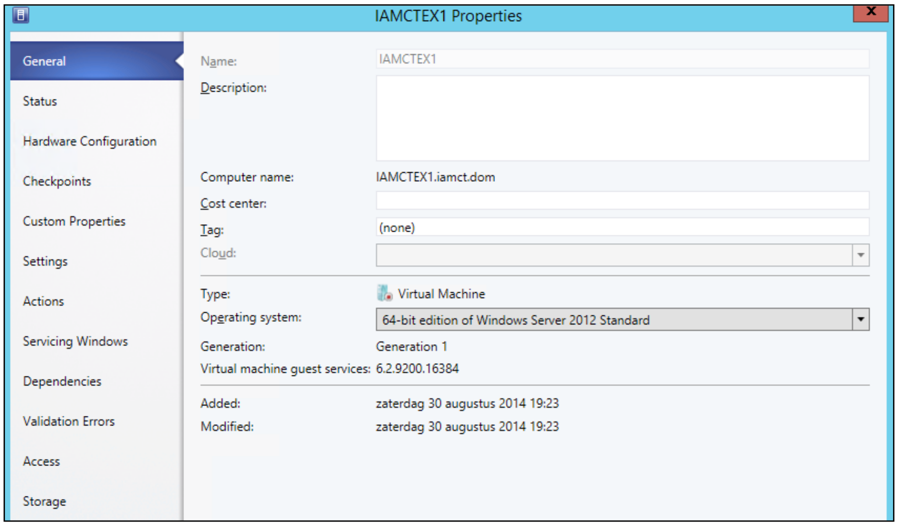
1. In the SCVMM console, select **VMs and Services** and then select your Hyper-V host. In the middle pane, you will see all Hyper-V VMs that are configured on that specific host:



Name	Status	Virtual Machine State	Host	Job Status	CPU Average	Operating System
IAMCTDC	Running	Running	moclab1		0%	64-bit edition of Windo...
IAMCTEX1	Running	Running	moclab1		0%	64-bit edition of Windo...
PDTADFS	Running	Running	moclab1		0%	Windows Server 2012 R...
PDTCLI1	Running	Running	moclab1		0%	64-bit edition of Windo...
PDTCLI1	Stopped	Stopped	moclab2		0%	Unknown
PDTSRV1	Running	Running	moclab1		0%	Windows Server 2012 R...
PDTSRV2	Running	Running	moclab2		0%	Windows Server 2012 R...
SCVMM	Running	Running	moclab2		0%	Windows Server 2012 R...
StandAlone81CLI	Running	Running	moclab2		0%	64-bit edition of Windo...

In our example, all VMs that are configured on our Hyper-V cluster are shown.

2. Select any of the configured VMs, right-click on it and click on **Properties**:



Name: IAMCTEX1
Description:
Computer name: IAMCTEX1.iamct.dom
Cost center:
Tag: (none)
Cloud:
Type: Virtual Machine
Operating system: 64-bit edition of Windows Server 2012 Standard
Generation: Generation 1
Virtual machine guest services: 6.2.9200.16384
Added: zaterdag 30 augustus 2014 19:23
Modified: zaterdag 30 augustus 2014 19:23

Notice there are a lot of similarities with the **Properties** view from the standalone Hyper-V console we talked about in the first four chapters.

Close this **Properties** window; make sure your VM is still selected, and right-click on it again. Notice several features from within this context menu, which should be clear by now, such as migrating the storage, migrating the virtual machine, snapshot, stop, restart, and so on.

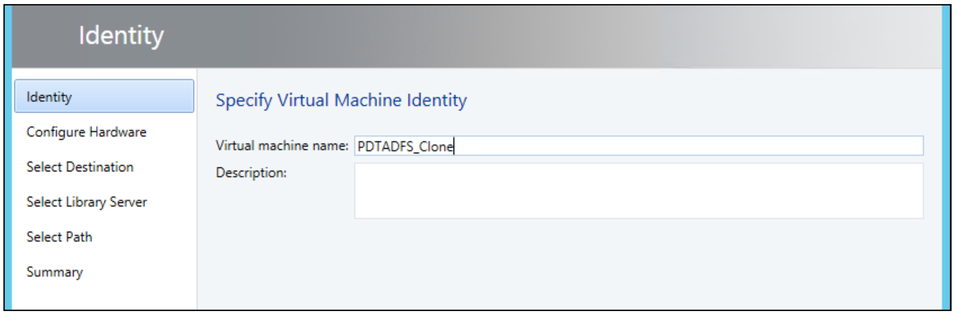
Just as before, a lot of similarities are recognizable from the standalone Hyper-V console. We only hope you are just as convinced as we are about the fact that SCVMM offers a lot more features and functionalities, compared to the Hyper-V manager.

In the next section, you will learn how to deploy a new VM from scratch, as well as out of the library.

Cloning a VM to the library and reusing the clone

The steps are as follows:

1. Select any of the VMs, click on **Create** in the ribbon, and then click on **Clone**. This will start the clone wizard. Provide a VM name and optionally a description for this clone. The VM name doesn't need to match the actual machine's host name, but it makes your life a lot easier if you do that. Click on **Next**:



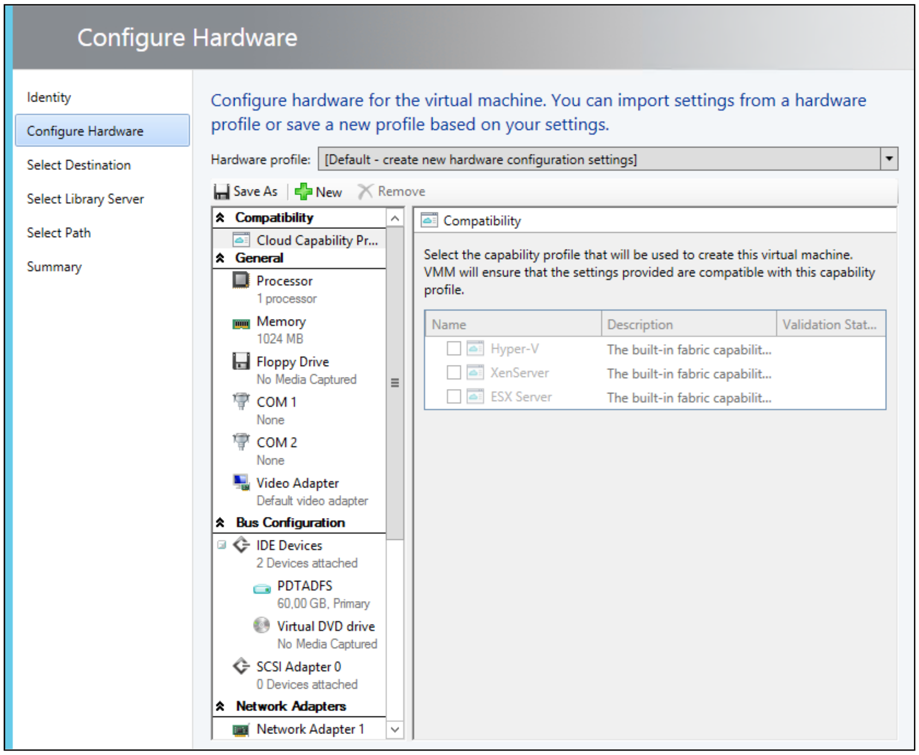
Identity

Specify Virtual Machine Identity

Virtual machine name: PDTADFS_Clone

Description:

2. In the **Configure Hardware** step, accept the default hardware settings:



Configure Hardware

Identity
Configure Hardware
Select Destination
Select Library Server
Select Path
Summary

Configure hardware for the virtual machine. You can import settings from a hardware profile or save a new profile based on your settings.

Hardware profile: [Default - create new hardware configuration settings]

Save As | New | Remove

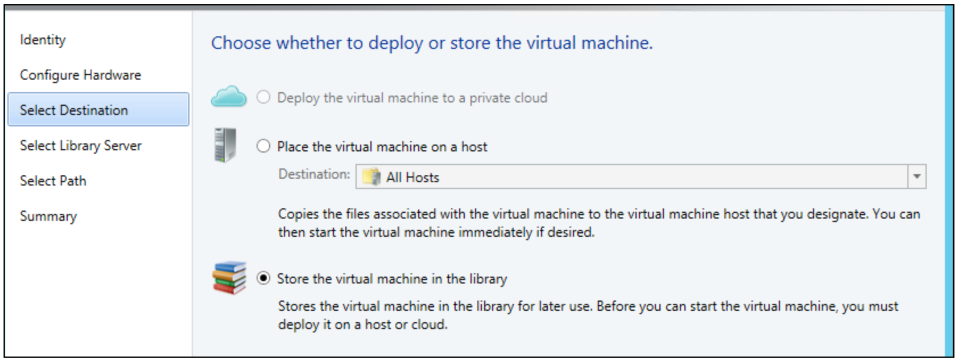
- **Compatibility**
  - Cloud Capability Pr...
- **General**
  - Processor
1 processor
  - Memory
1024 MB
  - Floppy Drive
No Media Captured
  - COM 1
None
  - COM 2
None
  - Video Adapter
Default video adapter
- **Bus Configuration**
  - IDE Devices
2 Devices attached
    - PDTADFS
60,00 GB, Primary
    - Virtual DVD drive
No Media Captured
  - SCSI Adapter 0
0 Devices attached
- **Network Adapters**
  - Network Adapter 1

Compatibility

Select the capability profile that will be used to create this virtual machine. VMM will ensure that the settings provided are compatible with this capability profile.

Name	Description	Validation Stat...
□ Hyper-V	The built-in fabric capabilit...	
□ XenServer	The built-in fabric capabilit...	
□ ESX Server	The built-in fabric capabilit...	

3. In the next step, specify the target of the cloned VM. This can be a running Hyper-V host or the SCVMM library; following our example, select the **Store the virtual machine in the library** option and select the discovered library server in the next step; click on **Next** to continue:



Identity
Configure Hardware
Select Destination
Select Library Server
Select Path
Summary

Choose whether to deploy or store the virtual machine.

○ Deploy the virtual machine to a private cloud

○ Place the virtual machine on a host

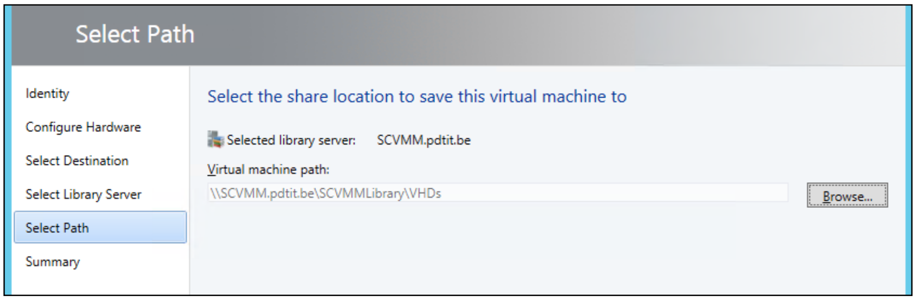
Destination: All Hosts

Copies the files associated with the virtual machine to the virtual machine host that you designate. You can then start the virtual machine immediately if desired.

● Store the virtual machine in the library

Stores the virtual machine in the library for later use. Before you can start the virtual machine, you must deploy it on a host or cloud.

4. Specify the target folder in the SCVMM library configuration, where the cloned VM should be stored:

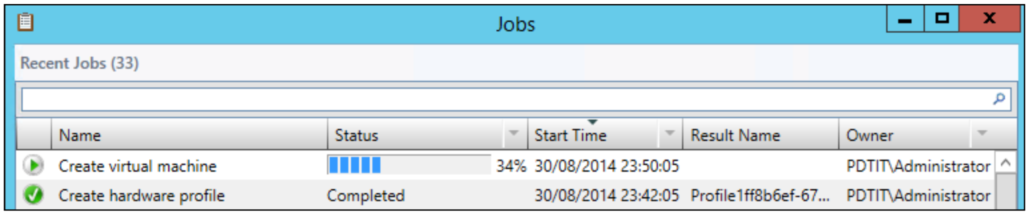


Select the share location to save this virtual machine to

Selected library server: SCVMM.pdtit.be

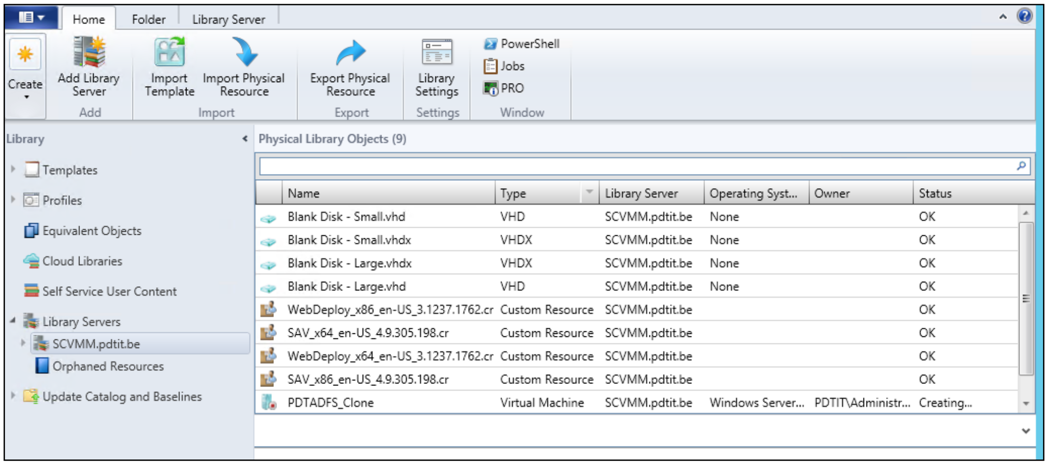
Virtual machine path:
\\SCVMM.pdtit.be\SCVMMLibrary\VHDs

5. Click on **Next**, followed by **Create** to complete the cloning wizard:



Name	Status	Start Time	Result Name	Owner
Create virtual machine	34%	30/08/2014 23:50:05		PDTIT\Administrator
Create hardware profile	Completed	30/08/2014 23:42:05	Profile1ff8b6ef-67...	PDTIT\Administrator

6. Lastly, Select **Library** from within the SCVMM console, followed by selecting your library server. Notice the different resources that are already provided out of the SCVMM library, of which our cloned server is one of them:



Name	Type	Library Server	Operating Syst...	Owner	Status
Blank Disk - Small.vhd	VHD	SCVMM.pdtit.be	None		OK
Blank Disk - Small.vhdx	VHDX	SCVMM.pdtit.be	None		OK
Blank Disk - Large.vhdx	VHDX	SCVMM.pdtit.be	None		OK
Blank Disk - Large.vhd	VHD	SCVMM.pdtit.be	None		OK
WebDeploy_x86_en-US_3.1237.1762.cr	Custom Resource	SCVMM.pdtit.be			OK
SAV_x64_en-US_4.9.305.198.cr	Custom Resource	SCVMM.pdtit.be			OK
WebDeploy_x64_en-US_3.1237.1762.cr	Custom Resource	SCVMM.pdtit.be			OK
SAV_x86_en-US_4.9.305.198.cr	Custom Resource	SCVMM.pdtit.be			OK
PDTADFS_Clone	Virtual Machine	SCVMM.pdtit.be	Windows Server...	PDTIT\Administr...	Creating...

Summary

In this chapter, you learned the basics of SCVMM (System Center Virtual Machine Manager), how to install the SCVMM management server, how to configure the different fabric components such as networking and storage, as well as how to integrate the already existing Hyper-V hosts into the SCVMM console. In the last section, you learned how to clone an existing VM and store it into the library. One of the topics we didn't touch is how to create a new VM. As this is 95 percent identical to creating a VM from within the Hyper-V manager, we decided not to repeat those steps here. The key thing is if you spend enough time configuring your Hyper-V hosts, followed by clearly, correctly, and completely configuring the required network and storage fabrics, deploying new VMs will become a no-brainer.

In the next chapter, we will discuss some of the additional features of SCVMM we slightly touched on in the beginning of this chapter, such as managing your VMware ESX hosts and VMs.

6
Integrating System Center VMM 2012 R2 with Your VMware Environment

As a lot of companies are running VMware nowadays, it is important for them to learn how to achieve centralized management out of System Center Virtual Machine Manager 2012 R2, allowing for a true virtualization platform management.

This chapter will detail how the integration works, how to configure the interaction between SCVMM and VMware's vCenter, and explain the possibilities that can arise out of the integration between both the platforms.

We will discuss the following topics:

- The purpose in using SCVMM as your overall virtualization platform management solution
- How to establish the integration between VMware vCenter and SCVMM
- Managing and operating your VMware hosts out of SCVMM
- How to migrate virtual machines from VMware to Virtual Machine Manager
- Best practices in migrating from VMware to Virtual Machine Manager

As of SCVMM 2012 R2, a tight integration with VMware vCenter Server is possible. From the VMM console, the most typical operations can be performed on the VMware hosts or VMware virtual machines, such as creating, managing, and deploying VMware virtual machines and more.

> A very important factor is the vCenter relationship. That's the "gateway" you need to establish the integration. It's not possible to connect to a VMware ESX server directly, and perform integration or management from there.

The following table gives an overview of what features are available out of SCVMM for managing an ESX platform:

Virtual machine management	ESX host servers can be integrated in the VMM private cloud concept, providing most management operations for virtual machines.
Dynamic optimization	The new dynamic optimization and power optimization features in SCVMM can also be used for ESX hosts.
Migration	Live migration of VMs is supported, leveraging VMware vMotion. Storage migration of ESX hosts is also supported, leveraging VMware Storage vMotion.
VMM Library	VMware ESX components are integrated in the VMM library, allowing you to store VMware VM templates, VMware virtual machines, and plain VMDK files in there.
Networking management	While VMM supports vSwitches and port groups, these components need to be configured out of VMware's vCenter.
Storage management	VMM supports the VMware's paravirtual SCSI adapters. Other storage options that are available and supported are dynamic disks, thin and thick provisioned disks, and hot replacement of virtual disks.
Virtual machine conversions	By using the V2V mechanism, it is possible to convert a VMware virtual machine to a Hyper-V virtual machine.

VM machine specs	The following virtual machine specs are supported out of SCVMM: • 255 GB RAM per VM • 8 vCPUs per VM However, the following features are not supported: • IDE storage bus • VM update management • Bare-metal conversion to VM • Dynamic memory

Why should you use SCVMM as your overall virtualization platform management solution?

It should be clear by now that most of the common day-to-day operations that are required on a VMware ESX platform are possible out of System Center Virtual Machine Manager, but not all. The more advanced features such as, for example, configuring virtual networks and vSwitches or managing datastores still need to be performed out of VMware's vCenter management console or by using VMware's command-line interface.

That said, there are still many advantages to configuring the integration, if you are to administer both VMware and Hyper-V platforms.

Think back to the System Center Virtual Machine Manager concept of a **fabric**. A VMM fabric is made up of hosts, host groups, library servers, storage components, and networking components. By using this fabric concept, those rather technical components are presented to users in a more familiar, controlled way, without the need of administrative rights or even deep, technical IT or virtualized platform knowledge. Using fabric, users can deploy new VMs, applications, services, and service templates, irrespective of which platform they are deploying these resources.

In a mixed Hyper-V and VMware environment, by default, one should administer the Hyper-V platform by using the Hyper-V management tools such as SCVMM, and VMware management should be done through VMware's vCenter. But that might confuse people, and may put a lot of administrative overhead on the platform administrators. They need to know both the management environments and should be trained on both the platforms.

Thinking about migration, where a company could, for example, decide to move specific workloads from Hyper-V to VMware or the other way around, shouldn't even be possible without using the tight integration between both platforms. Or at least not in such an easy and online way as is possible today out of SCVMM.

As long as the specific details of each platform such as networking, vSwitches, storage components, and so on are configured, a hybrid configuration and management solution will make your administrator's job a lot easier, with the nice addition of having both platforms running independent or completely integrated if you want. Another reason I could personally think of for why Microsoft spent so much time in making this tight integration a given, is to attract and persuade customers to move to Hyper-V, instead of renewing their VMware maintenance and license contracts. But that's just my personal opinion… Yet, think back to our introduction chapter, where we talked about Microsoft's CloudOS vision, which creates a cohesive platform for your on-premise private cloud, the Microsoft public cloud with Azure, or a hybrid setup. CloudOS is triggered out of Windows Server 2012 R2 and Hyper-V. That story wouldn't be as powerful and complete as it is today, if there was no possibility to manage and integrate VMware ESX in the story. And that's exactly where SCVMM comes—again—in play.

Although not all features are available, we are pretty sure people love it. And yes, it is not a 100 percent single pane of glass, but thinking back on what was only possible with the previous versions of VMM in relation to ESX integration, I'm confident to say Microsoft did a great job in VMM 2012 R2 regarding this subject.

> Organizations will welcome the ease of management of SCVMM, allowing them to manage both Hyper-V and VMware ESX platforms from one solution.

How to establish the integration between VMware vCenter and SCVMM

SCVMM supports the following VMware ESX platforms:

- VMware vCenter 4.1
- VMware vCenter 5.0
- VMware vCenter 5.1
- VMware ESX / ESXi 4.1
- VMware ESX / ESXi 5.0
- VMware ESX / ESXi 5.1

vCenter 5.5 and ESXi 5.5 are still not officially supported in System Center 2012 R2, but in practice, it appears that simple management tasks can be carried out as with version 5.1, the support for which was added in SCVMM SP1.

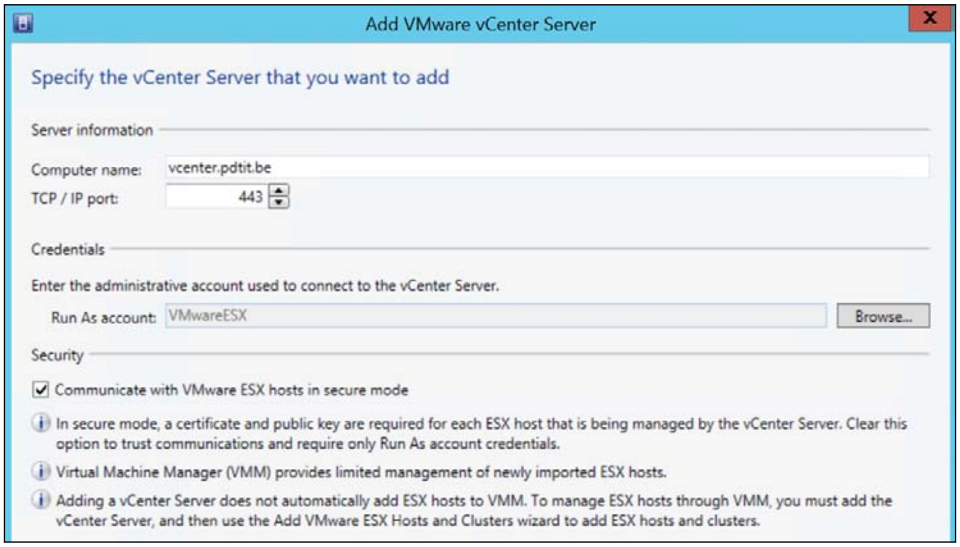
Adding VMware vCenter to VMM

When your environment has all the prerequisites covered, we can start with the actual integration configuration from within SCVMM, using the following procedure:

1. Log on to the SCVMM server with administrative rights and open the SCVMM console.
2. From within the SCVMM console, select **Fabric**.
3. Select **Servers** and then right-click on it.

> Before adding an ESX host itself, you must first add at least one vCenter server to SCVMM.

4. From the context menu, select **Add VMware vCenter Server**:



Add VMware vCenter Server

Specify the vCenter Server that you want to add

Server information

Computer name: vcenter.pdtit.be

TCP / IP port: 443

Credentials

Enter the administrative account used to connect to the vCenter Server.

Run As account: VMwareESX

Security

☑ Communicate with VMware ESX hosts in secure mode

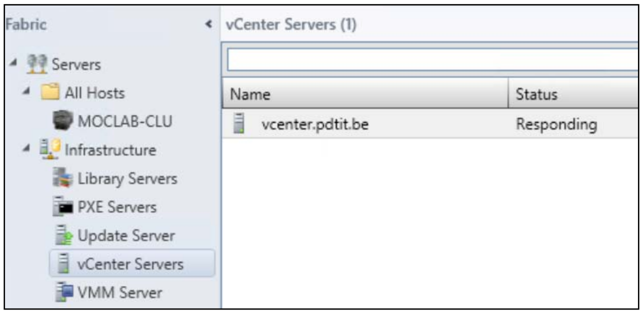
In secure mode, a certificate and public key are required for each ESX host that is being managed by the vCenter Server. Clear this option to trust communications and require only Run As account credentials.

Virtual Machine Manager (VMM) provides limited management of newly imported ESX hosts.

Adding a vCenter Server does not automatically add ESX hosts to VMM. To manage ESX hosts through VMM, you must add the vCenter Server, and then use the Add VMware ESX Hosts and Clusters wizard to add ESX hosts and clusters.

> I created a Run As account from my VMware vCenter administrative account, which is the default `administrator@vsphere.local` account.

5. Enter the IP address or hostname from your VMware vCenter server, as well as the communication port, which by default is 443, but could be anything else in your specific configuration.
6. Select the **Run As account** from the list.
7. Flag the **Communicate with VMware ESX hosts in secure mode** option. Click on **OK** to continue.



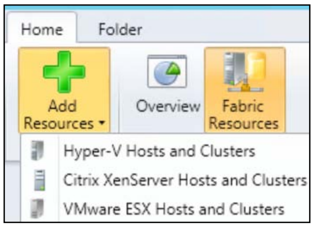
Name	Status
vcenter.pdtit.be	Responding

8. The VMware vCenter server is successfully added with the status **Responding**, as shown in the preceding screenshot.

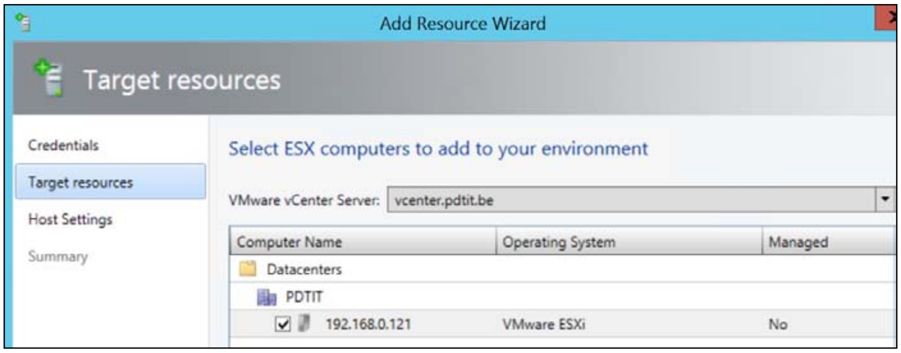
Adding VMware ESX/ESXi hosts to VMM

In order to be able to manage our VMware ESX/ESXi host servers itself, we need to add them explicitly to our VMM environment, using a similar process:

1. Through **Fabric** | **Home** | **Add Resources**, select **VMware ESX Hosts and Clusters**:


Home
Folder
Add Resources
Overview
Fabric Resources
Hyper-V Hosts and Clusters
Citrix XenServer Hosts and Clusters
VMware ESX Hosts and Clusters

2. This will launch the **Add Resource Wizard** window:

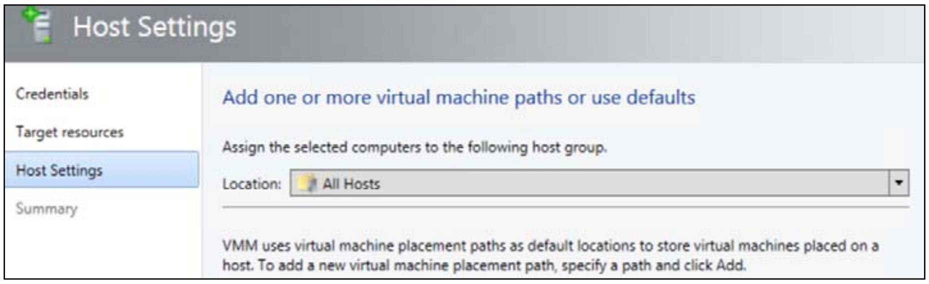


Select ESX computers to add to your environment

VMware vCenter Server: vcenter.pdtit.be

Computer Name	Operating System	Managed
Datacenters		
PDTIT		
☑ 192.168.0.121	VMware ESXi	No

3. Here, select the IP address or hostname of the ESX server host you want to add and then click on **Next**:



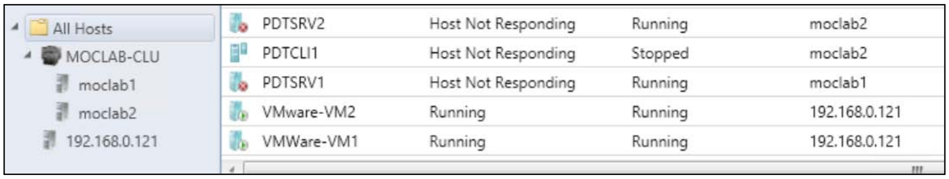
Add one or more virtual machine paths or use defaults

Assign the selected computers to the following host group.

Location: All Hosts

VMM uses virtual machine placement paths as default locations to store virtual machines placed on a host. To add a new virtual machine placement path, specify a path and click Add.

4. In the **Host Settings** step of the wizard, select a specific location where you want to have the VMware ESX hosts to be added. In my demo VMM environment, I only have the top section called `All Hosts` in my VMM setup. This is ok for now.
5. Go through the summary at the end of the wizard and then click on **Next**:



All Hosts
MOCLAB-CLU
moclab1
moclab2
192.168.0.121

PDTSRV2	Host Not Responding	Running	moclab2
PDTCLI1	Host Not Responding	Stopped	moclab2
PDTSRV1	Host Not Responding	Running	moclab1
VMware-VM2	Running	Running	192.168.0.121
VMWare-VM1	Running	Running	192.168.0.121

6. From within the VMM console, select the **VMs and Services** pane, browse to the `All Hosts` folder, and notice that the VMware ESXi server is added here. In the list of VMs, notice that two of my VMware virtual machine clients, **VMware-VM1** and **VMware-VM2**, are running on host **192.168.0.121**. This is the management IP address from my physical ESXi machine.

To achieve the same using PowerShell, use the following cmdlet sample:

```
$RunAsAccount = Get-SCRunAsAccount -Name "VMwareESXi" -ID "a69fe1d9-3930-
4a97-9911-dcbd937dc2d5"

$HostGroup = Get-SCVMHostGroup -ID "0e3ba228-a059-46be-aa41-2f5cf0f4b96e"
-Name "All Hosts"

$VirtualizationManager = Get-SCVirtualizationManager -ComputerName
"vcenter.pdtit.be"

Add-SCVMHost -ComputerName "192.168.0.122" -VirtualizationManager
$VirtualizationManager -VMHostGroup $HostGroup -VMPaths "" -Credential
$RunAsAccount -RunAsynchronously
```

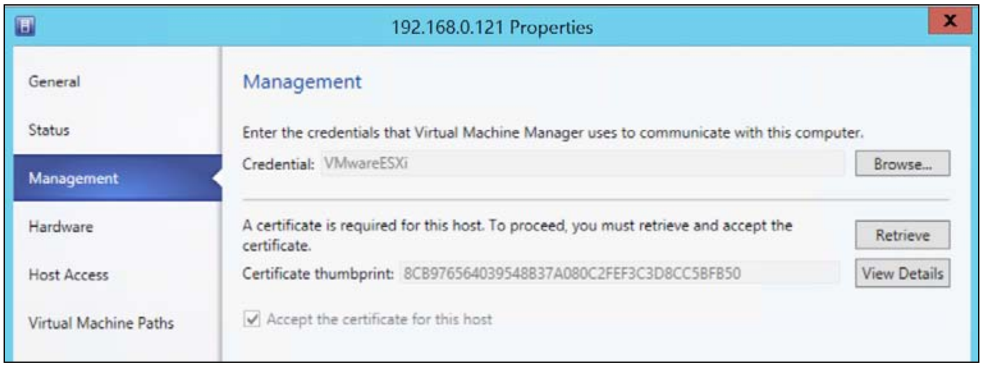
Without going into much detail here, VMM shows you already the integration with VMware vCenter from these steps. When selecting one of the VMware virtual machine clients, you can not only view the VM machine settings, but also edit and change them. Changes will be propagated from VMM through VMware vCenter and be applied on the virtual machine(s) immediately.

Managing and operating your VMware hosts out of SCVMM

After updating or viewing a virtual machine's settings, it's also possible to shut down or power off a virtual machine, or create a snapshot of it.

Finally, it is also possible to create new virtual machines on the VMware ESX/ESXi environment, directly out of VMM. However, in order to make this work, we have to configure the correct security and management settings on the ESX host as follows:

1. Go to **VMM Console** | **VMs and Services** | **All Hosts**, select your VMware ESX host, right-click on it, and click on **Properties**.
2. Select the **Management** option:



Management

Enter the credentials that Virtual Machine Manager uses to communicate with this computer.

Credential: VMwareESXi

A certificate is required for this host. To proceed, you must retrieve and accept the certificate.

Certificate thumbprint: 8CB97656403954B837A080C2FEF3C3D8CC5BFB50

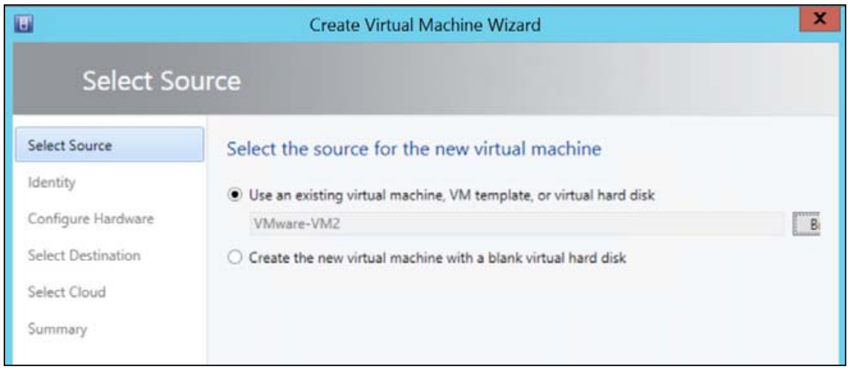
☑ Accept the certificate for this host

3. Press the **Retrieve** button. Mark the **Accept the certificate for this host** option.

This will configure additional and elevated management permissions between the VMware ESX host and our VMM server. This is, for example, required when converting VMware VMs to Hyper-V VMs (see later on in this chapter to achieve this).

Once this step is complete, we have "unlocked" additional management features, allowing us, for example, to create a new virtual machine on the ESX server. This can be done as follows:

1. Select your ESX host, right-click on it, and then click on **Create Virtual Machine**. This will launch the **Create Virtual Machine Wizard** window:



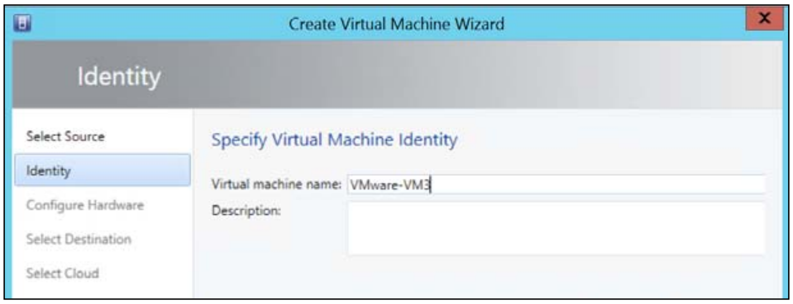
Select the source for the new virtual machine

● Use an existing virtual machine, VM template, or virtual hard disk

VMware-VM2

○ Create the new virtual machine with a blank virtual hard disk

2. In the **Select Source** step, we browse to our existing **VMware-VM2** machine, to be used. (Make sure this VM is powered-off, as it is not possible to deploy a new VM in this way when the source VM is running.)



Create Virtual Machine Wizard

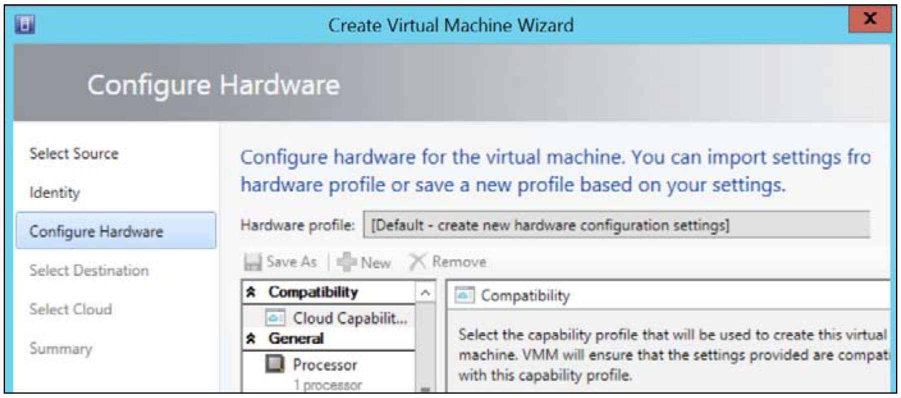
Identity

Specify Virtual Machine Identity

Virtual machine name: VMware-VM3

Description:

3. Specify a VM name and description.



Create Virtual Machine Wizard

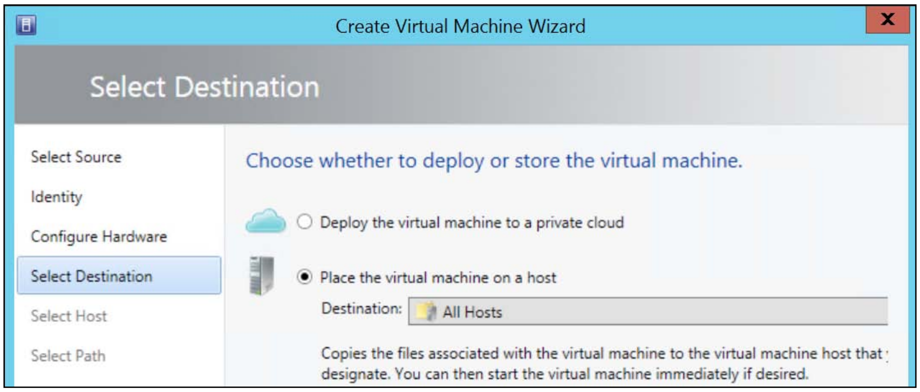
Configure Hardware

Configure hardware for the virtual machine. You can import settings fro hardware profile or save a new profile based on your settings.

Hardware profile: [Default - create new hardware configuration settings]

Select the capability profile that will be used to create this virtual machine. VMM will ensure that the settings provided are compati with this capability profile.

4. In the **Configure Hardware** step, we select the default option to create a new hardware profile.



Create Virtual Machine Wizard

Select Destination

Choose whether to deploy or store the virtual machine.

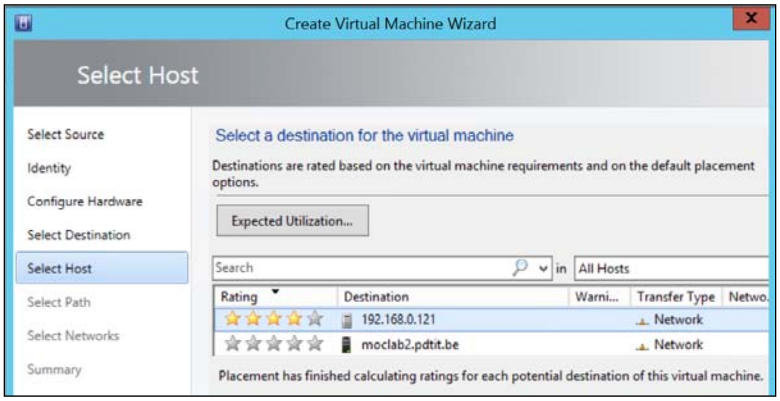
○ Deploy the virtual machine to a private cloud

● Place the virtual machine on a host

Destination: All Hosts

Copies the files associated with the virtual machine to the virtual machine host that designate. You can then start the virtual machine immediately if desired.

5. In the **Select Destination** step, check the **Place the virtual machine on a host** option, and select **All Hosts** as destination.



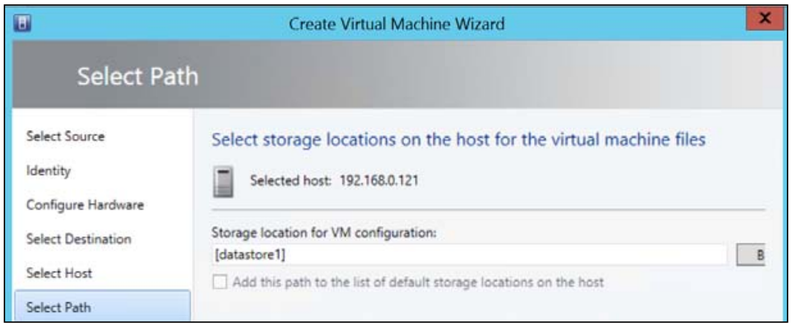
Select a destination for the virtual machine

Destinations are rated based on the virtual machine requirements and on the default placement options.

Rating	Destination	Warni...	Transfer Type	Netwo...
★★★★☆	192.168.0.121		Network	
☆☆☆☆☆	moclab2.pdtit.be		Network	

Placement has finished calculating ratings for each potential destination of this virtual machine.

6. In the **Select Host** option, the different VMware ESX hosts and Hyper-V hosts will be shown here. Select the VMware ESX host here.



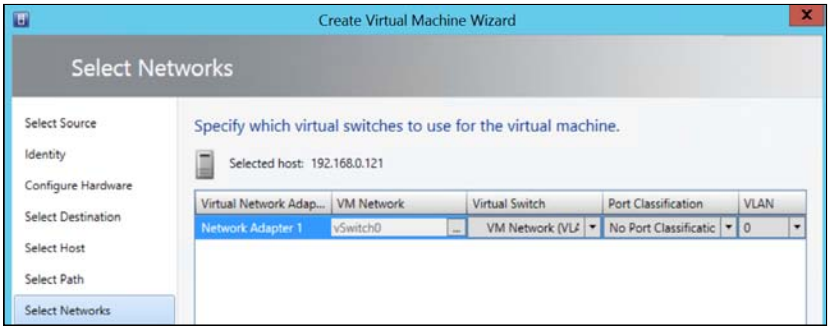
Select storage locations on the host for the virtual machine files

Selected host: 192.168.0.121

Storage location for VM configuration:
[datastore1]

□ Add this path to the list of default storage locations on the host

7. In the **Select Path** step, select the storage location for the VM to be created; in our example, we place them in the `[datastore1]` location, which is the local hard disk in the physical VMware ESX server.

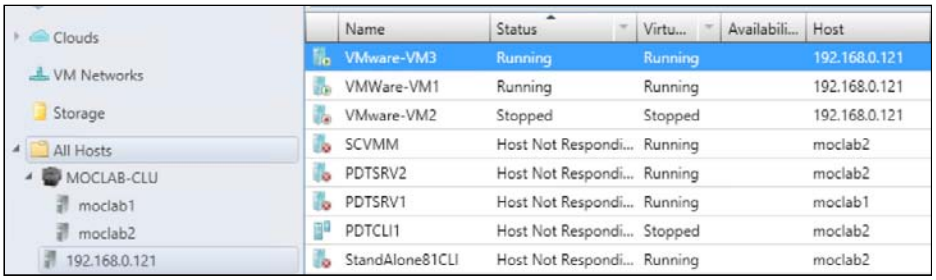


Specify which virtual switches to use for the virtual machine.

Selected host: 192.168.0.121

Virtual Network Adap...	VM Network	Virtual Switch	Port Classification	VLAN
Network Adapter 1	vSwitch0	VM Network (VL/	No Port Classificatic	0

8. After selecting the path where the VM should be stored, the VMware virtual networks that are available are shown. Select the appropriate network(s) you want to configure on this new VM.



Name	Status	Virtu...	Availabili...	Host
VMware-VM3	Running	Running		192.168.0.121
VMWare-VM1	Running	Running		192.168.0.121
VMware-VM2	Stopped	Stopped		192.168.0.121
SCVMM	Host Not Respondi...	Running		moclab2
PDTSRV2	Host Not Respondi...	Running		moclab2
PDTSRV1	Host Not Respondi...	Running		moclab1
PDTCLI1	Host Not Respondi...	Stopped		moclab2
StandAlone81CLI	Host Not Respondi...	Running		moclab2

9. Once the wizard is completed, wait a few minutes before the job is completed, and then verify the creation of the new VMware VM has been successful and the possibility to start the VM as well.

This completes the topic on how to create a new VMware VM on an ESX host from SCVMM.

How to migrate virtual machines from VMware to Virtual Machine Manager using VMM 2012 R2

It is of course already powerful and very interesting for organizations to have the possibility of integrating an existing VMware ESX platform into System Center Virtual Machine Manager. While there are still some really specific features and configuration options only available from within the ESX side, in a mixed environment, it's almost "logic" to start using SCVMM to manage both your Hyper-V and VMware platforms.

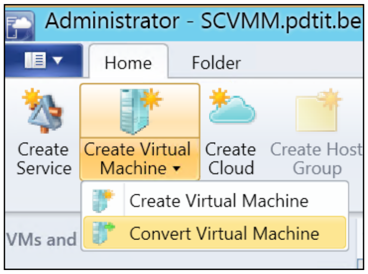
Another really interesting feature that is part of the integration between both hypervisor management tools, is migrating/transforming from a VMware VM to a Hyper-V VM.

Before I walk you through the different steps on how to achieve this, I want you to understand the following requirements and limitations in doing this:

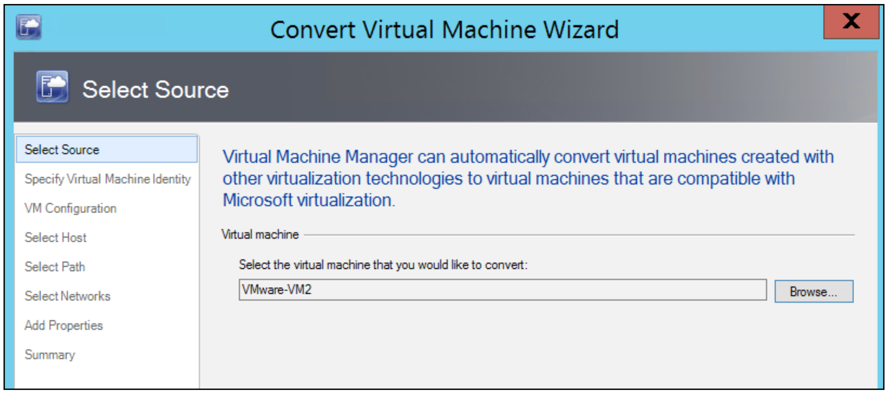
- Conversion only works from VMware ESX and/or ESXi platforms
- Integration between VMM 2012 R2 and VMware vCenter must be established
- The VMware VM(s) you want to convert need to be turned off
- VMware tools that are installed on the VM must be uninstalled prior to performing the conversion
- VMware virtual machines with IDE-connected virtual hard disks cannot be converted

Use the following steps to convert VMware virtual machines to Hyper-V virtual machines, using **Convert Virtual Machine Wizard**:

1. From within the **VMs and Services** pane, select **Home** in the menu ribbon and then go to **Create Virtual Machine** | **Convert Virtual Machine**:



This will launch the **Convert Virtual Machine Wizard** window:

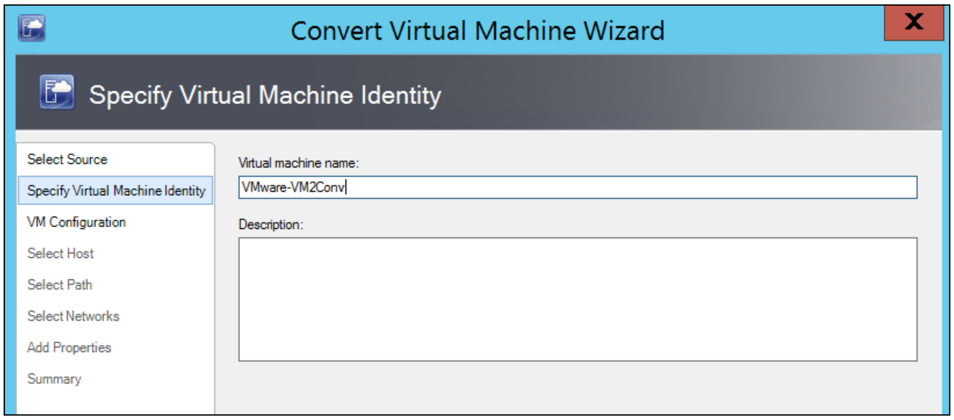


Virtual Machine Manager can automatically convert virtual machines created with other virtualization technologies to virtual machines that are compatible with Microsoft virtualization.

Virtual machine

Select the virtual machine that you would like to convert: VMware-VM2

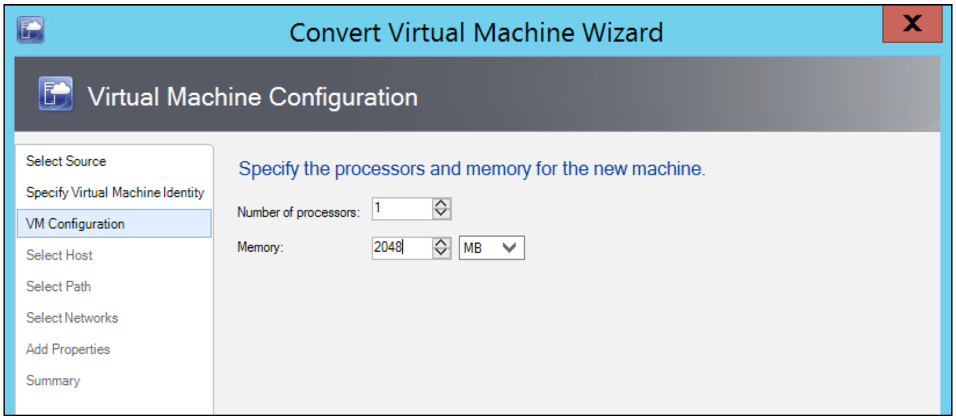
2. Click on the **Browse** button to select the VMware virtual machine you want to convert. Again, make sure the VM is powered off, otherwise it will not be visible in the list of possible VMs to convert. Click on **Next**.


Convert Virtual Machine Wizard
Specify Virtual Machine Identity
Select Source
Specify Virtual Machine Identity
VM Configuration
Select Host
Select Path
Select Networks
Add Properties
Summary

Virtual machine name: VMware-VM2Conv

Description:

3. Enter an appropriate name for the new VM you want to create. This name can be the same as the source VMware VM, but I recommend using a different name, to keep it consistent in the console. Also, this name can be anything, but should not be the same as the Windows machine name itself. Click on **Next**.

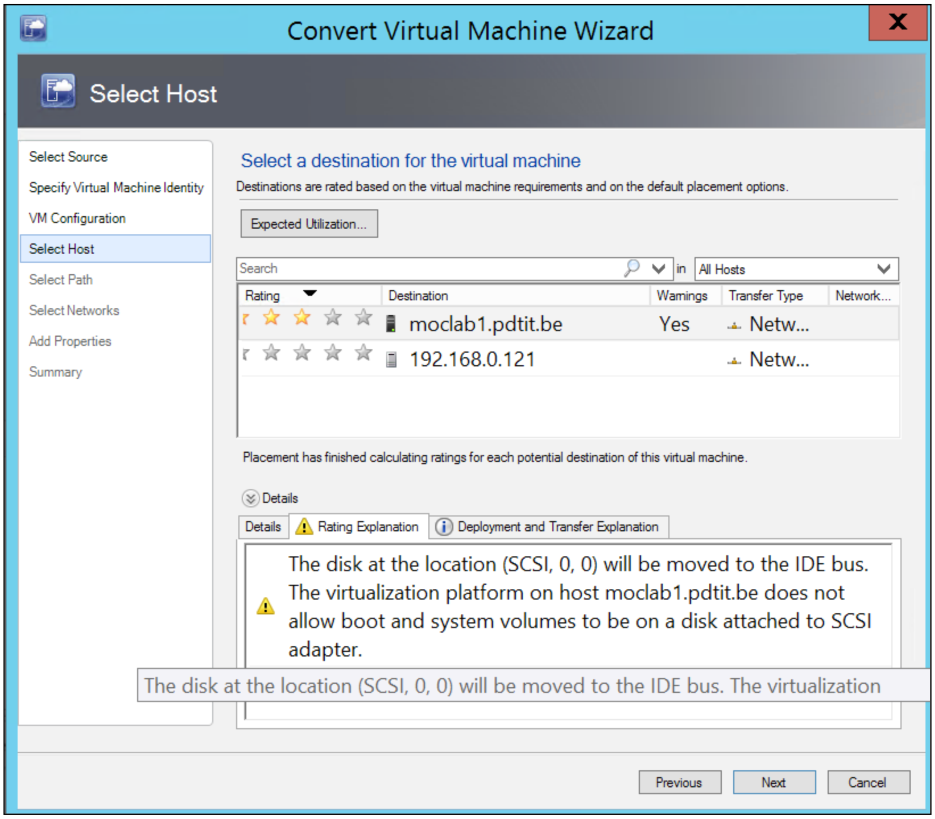

Convert Virtual Machine Wizard
Virtual Machine Configuration
Select Source
Specify Virtual Machine Identity
VM Configuration
Select Host
Select Path
Select Networks
Add Properties
Summary

Specify the processors and memory for the new machine.

Number of processors: 1

Memory: 2048 MB

4. Specify the processors and memory settings for the new VM and then click on **Next**.



Select a destination for the virtual machine

Destinations are rated based on the virtual machine requirements and on the default placement options.

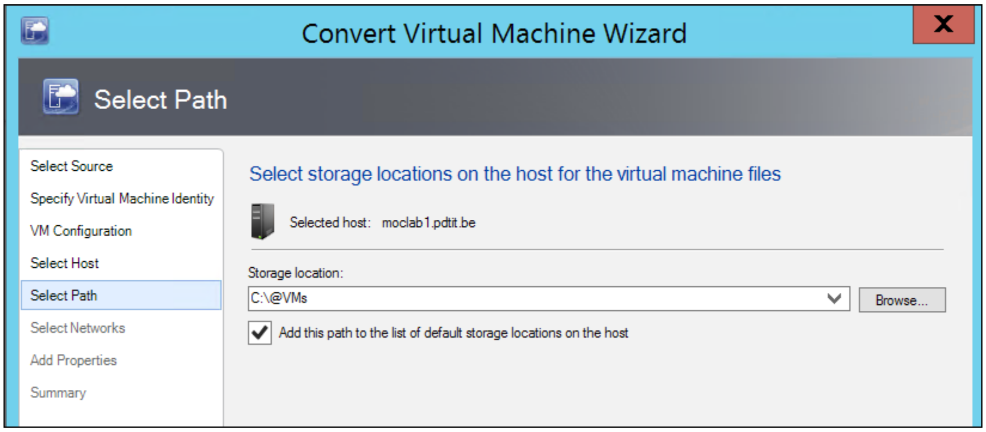
Rating	Destination	Warnings	Transfer Type	Network...
★★☆☆	moclab1.pdtit.be	Yes	Netw...	
☆☆☆☆	192.168.0.121		Netw...	

Placement has finished calculating ratings for each potential destination of this virtual machine.

The disk at the location (SCSI, 0, 0) will be moved to the IDE bus. The virtualization platform on host moclab1.pdtit.be does not allow boot and system volumes to be on a disk attached to SCSI adapter.

The disk at the location (SCSI, 0, 0) will be moved to the IDE bus. The virtualization

5. Select the target Hyper-V host (**moclab1.pdtit.be** in my setup) and click on **Next**. Note the exclamation mark informing us about the change in disk type that will occur during the conversion.



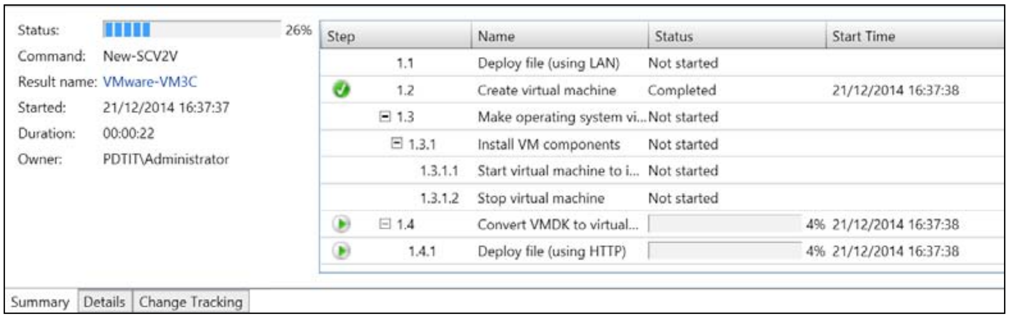
Select storage locations on the host for the virtual machine files

Selected host: moclab1.pdtit.be

Storage location: C:\@VMs

☑ Add this path to the list of default storage locations on the host

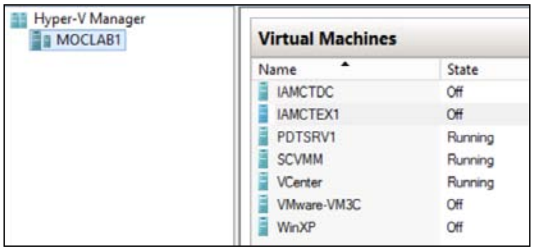
6. Browse to the destination location where the converted VM should be stored. Optionally, flag the **Add this path to the list of default storage locations on the host** option. Click on **Next**.



Status: 26%
Command: New-SCV2V
Result name: VMware-VM3C
Started: 21/12/2014 16:37:37
Duration: 00:00:22
Owner: PDTIT\Administrator

Step	Name	Status	Start Time
1.1	Deploy file (using LAN)	Not started	
1.2	Create virtual machine	Completed	21/12/2014 16:37:38
1.3	Make operating system vi...	Not started	
1.3.1	Install VM components	Not started	
1.3.1.1	Start virtual machine to i...	Not started	
1.3.1.2	Stop virtual machine	Not started	
1.4	Convert VMDK to virtual...	4%	21/12/2014 16:37:38
1.4.1	Deploy file (using HTTP)	4%	21/12/2014 16:37:38

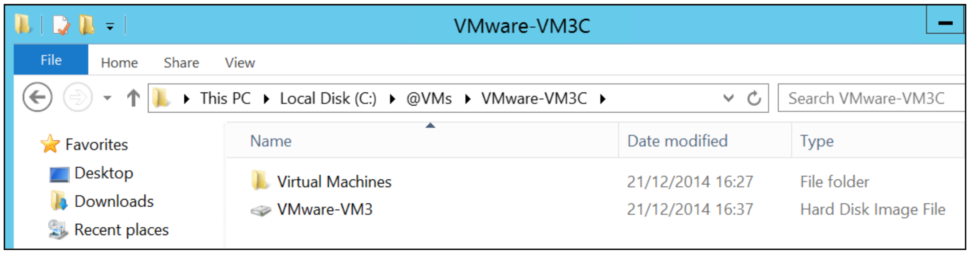
7. After clicking on **Next** on the **Summary** screen, the **Jobs** window will be shown, where you can view and follow the progress of the conversion process. Depending on the size of the VM, as well as the system specifications of both the source and target hypervisor hosts, this process might take some time.



Virtual Machines

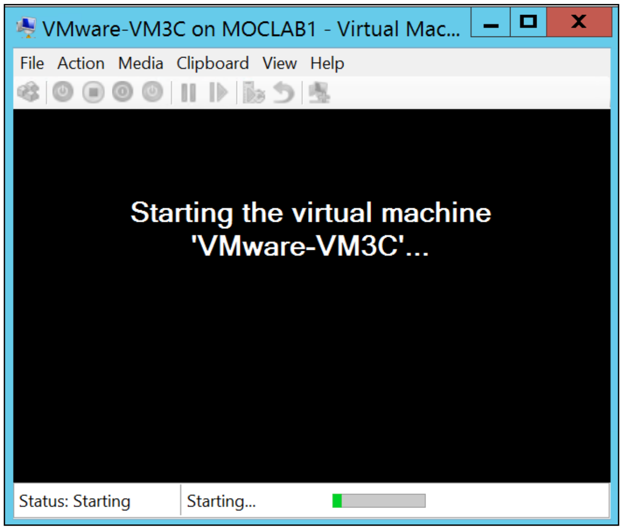
Name	State
IAMCTDC	Off
IAMCTEX1	Off
PDTSRV1	Running
SCVMM	Running
VCenter	Running
VMware-VM3C	Off
WinXP	Off

8. During the conversion process, we can see that the Hyper-V VM is created successfully on our Hyper-V host already (**VMware-VM3C**).



Name	Date modified	Type
Virtual Machines	21/12/2014 16:27	File folder
VMware-VM3	21/12/2014 16:37	Hard Disk Image File

9. Similarly, when browsing to the destination folder we selected during the conversion wizard, we can see that the Hyper-V VM is being created successfully, and also the virtual hard disk is being created.


VMware-VM3C on MOCLAB1 - Virtual Mac...
File Action Media Clipboard View Help

Starting the virtual machine 'VMware-VM3C'...

Status: Starting Starting...

10. Once the conversion process has been completed on the VMM 2012 R2 server, we can start the VM itself on the Hyper-V host.

This completes the conversion of a VMware ESX virtual machine to a Hyper-V virtual machine, by using VMM 2012 R2.

While this conversion mechanism is interesting and very useful in many cases, there are also some drawbacks, sort of. For me personally, the biggest "concern" is the fact you need to have both VMM 2012 R2 and VMware vCenter running to make this work. But what if you have an SMB without the full VMM or vCenter tools? Or you want to use this as a recovery mechanism in case of a disaster?

Or, maybe even more important, when you want to perform a physical-to-virtual conversion? That feature was possible in VMM in the past, but has been taken out since VMM 2012 R2.

That's where the **Microsoft Virtual Machine Converter 3.0** comes in play.

The MVMC 3.0, as Microsoft calls it, is a standalone and free Microsoft tool, which allows you to convert VMware VMDK files into Hyper-V VHD/VHDX files, as well as performing a conversion from a physical machine and disks into a virtual Hyper-V machine.

> It is interesting to note that the MVMC can convert virtual machines which are both powered off and online. When they are online, however, the VMware tools must be uninstalled and must be running a Windows operating system to make the conversion work.

Before explaining how the MVMC works, I would like to draw your attention to the following requirements and considerations:

- The conversion is supported for VMs running Windows Server 2008 R2 with SP1 or above, Server 2012, and Server 2012 R2
- Full VMware configurations are converted (disk, memory, CPU, and so on)
- Virtual network connections are added during the migration, to allow the VM to boot up and connect to the network, once converted
- Conversion is supported from VMware vSphere 4.1 and above
- VMware tools are uninstalled if not done before the conversion starts
- Full PowerShell is enabled

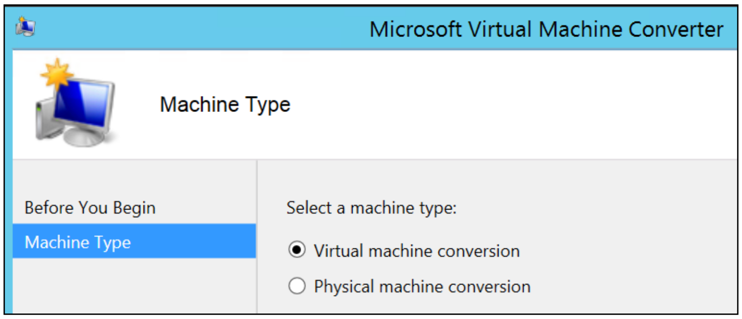
In the following section, I will show you how the conversion process is achieved, both by using the MVMC 3.0 GUI and by using PowerShell.

How to convert virtual machines from VMware to Hyper-V using Microsoft Virtual Machine Convertor 3.0

Now that you understand where the MCVM can be used, we will walk you through the actual process.

1. Download Microsoft Virtual Machine Convertor 3.0 from the Microsoft website at `http://www.microsoft.com/en-gb/download/details.aspx?id=42497`.

Run the installation (I didn't include this part here, as it is basically a next/next/finish install). Once the installation is done, start the tool from the Start menu:



Microsoft Virtual Machine Converter

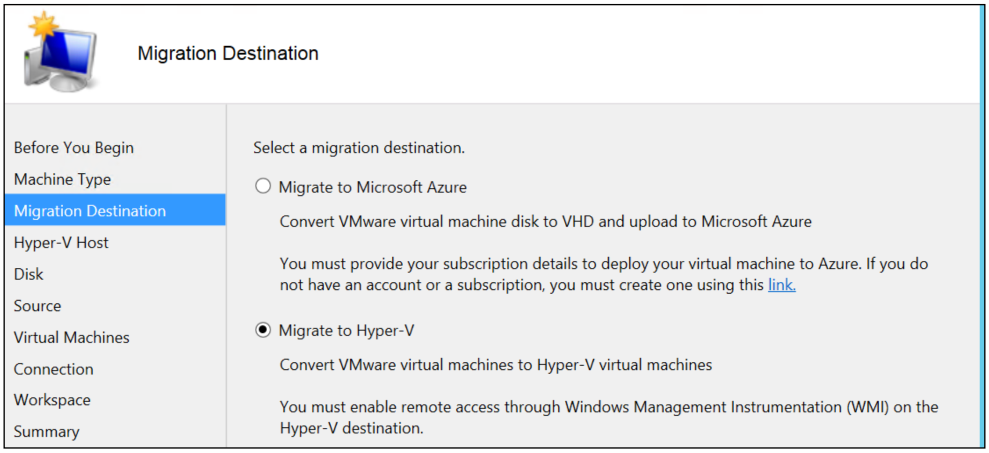
Machine Type

Select a machine type:

● Virtual machine conversion

○ Physical machine conversion

2. In the first step of the wizard, select the machine type you want to convert from. Click on **Next**.



Migration Destination

Select a migration destination.

○ Migrate to Microsoft Azure

Convert VMware virtual machine disk to VHD and upload to Microsoft Azure

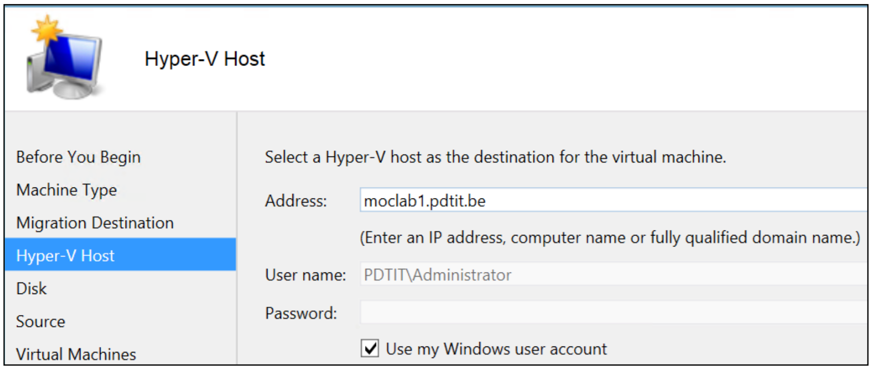
You must provide your subscription details to deploy your virtual machine to Azure. If you do not have an account or a subscription, you must create one using this link.

● Migrate to Hyper-V

Convert VMware virtual machines to Hyper-V virtual machines

You must enable remote access through Windows Management Instrumentation (WMI) on the Hyper-V destination.

3. In the **Migration Destination** step, select if you want to convert the source VM directly to Microsoft Azure or to an on-premise Hyper-V host. Click on **Next**.



Hyper-V Host

Select a Hyper-V host as the destination for the virtual machine.

Address: moclab1.pdtit.be

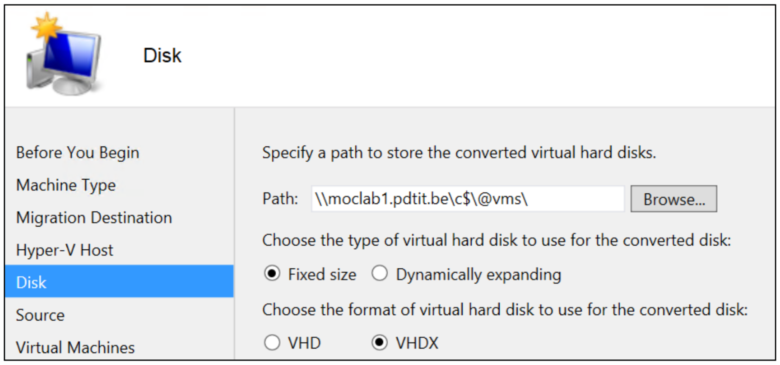
(Enter an IP address, computer name or fully qualified domain name.)

User name: PDTIT\Administrator

Password:

☑ Use my Windows user account

4. In the **Hyper-V Host** step, provide the details from the Hyper-V host as destination and then click on **Next**.



Disk

Specify a path to store the converted virtual hard disks.

Path: \\moclab1.pdtit.be\c$\@vms\

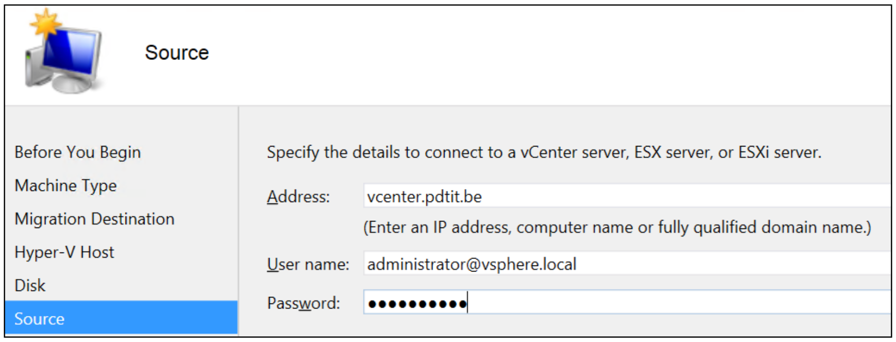
Choose the type of virtual hard disk to use for the converted disk:

● Fixed size ○ Dynamically expanding

Choose the format of virtual hard disk to use for the converted disk:

○ VHD ● VHDX

5. In the **Disk** step, provide the details and settings regarding the converted virtual hard disks. Click on **Next**.



Source

Specify the details to connect to a vCenter server, ESX server, or ESXi server.

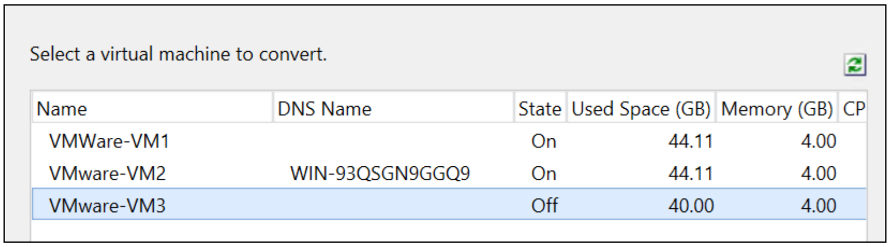
Address: vcenter.pdtit.be

(Enter an IP address, computer name or fully qualified domain name.)

User name: administrator@vsphere.local

Password: ••••••••••

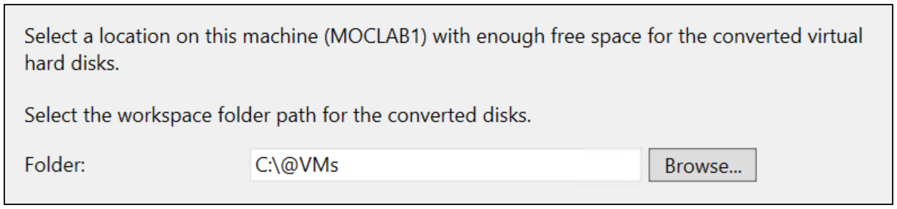
6. In the **Source** step of the wizard, enter the details of the source VMware vCenter machine, or a ESX/ESXi server directly. Click on **Next**.



Select a virtual machine to convert.

Name	DNS Name	State	Used Space (GB)	Memory (GB)	CP
VMWare-VM1		On	44.11	4.00	
VMware-VM2	WIN-93QSGN9GGQ9	On	44.11	4.00	
VMware-VM3		Off	40.00	4.00	

7. In the next step, select the VMware VM you want to convert. In my example, I will perform another conversion from the VM3 virtual machine.

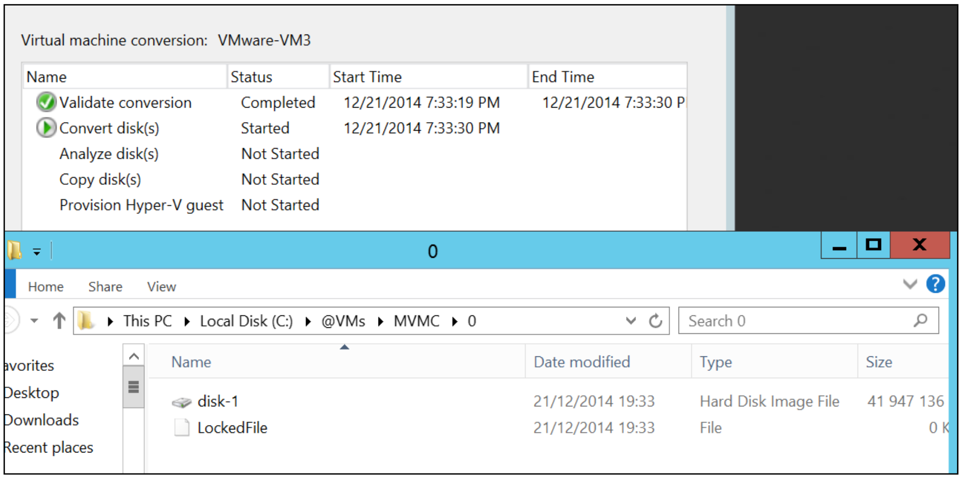


Select a location on this machine (MOCLAB1) with enough free space for the converted virtual hard disks.

Select the workspace folder path for the converted disks.

Folder: C:\@VMs

8. Next, enter a folder path on the Hyper-V host where the MVMC tool can store the converted disk(s). Click on **Next**.



Virtual machine conversion: VMware-VM3

Name	Status	Start Time	End Time
✓ Validate conversion	Completed	12/21/2014 7:33:19 PM	12/21/2014 7:33:30 P
▶ Convert disk(s)	Started	12/21/2014 7:33:30 PM	
Analyze disk(s)	Not Started		
Copy disk(s)	Not Started		
Provision Hyper-V guest	Not Started		

Name	Date modified	Type	Size
disk-1	21/12/2014 19:33	Hard Disk Image File	41 947 136
LockedFile	21/12/2014 19:33	File	0 K

9. In the final step, the virtual machine conversion process is started. To verify that the process is working ok, browse to your Hyper-V host's VM folder that you specified during the conversion, notice the `MVMC` subfolder, in which you can see the virtual disk being created.

This completes the conversion of a VMware virtual machine to a Hyper-V virtual machine by using the latest Microsoft Virtual Machine Converter 3.0.

Best practices in migrating from VMware to Virtual Machine Manager

Although I didn't plan to make this an official section in this chapter, I wanted to give you some additional "best practices" from my own experiences with customers, when performing migrations from VMware ESX platforms to Hyper-V and Virtual Machine Manager.

Don't consider this as an official list of any kind, nor as the "one word of truth", as some information might not be relevant to your specific situation:

- Always take enough time to do a thorough analysis of the current VMware ESX/ESXi and vCenter infrastructure of the customer. Know that the **Microsoft Assessment and Planning Toolkit** (**MAPT**), which has been discussed in *Chapter 1*, *Introduction to Microsoft Server 2012 R2 Hyper-V*, is your best friend here.
- Know the details around licensing, both the as-is situation on the VMware side, as well as the to-be situation on Hyper-V and System Center Virtual Machine Manager. (Remember that we have added a specific chapter to talk only about licensing Hyper-V and System Center, which is *Chapter 9*, *Hyper-V and System Center Licensing*).
- Create an inventory of the VMware specific features such as clustering, high availability, dynamic memory, and so on, and make sure you are able to recreate them by using the Hyper-V alternatives.
- Specifically about the conversion process, do talk with your customer about the requirements, business impact on virtual machines (for example powering off the VM or removing VMware tools, which often requires a reboot, besides possible other impacts), maintenance window where the conversion process can take place, what is the fall-back scenario in case of failure, and so on, to name just a few.

- Allow for a well-performing host on both sides if possible, to speed up the process of the conversion. The conversion itself is mainly network traffic, so make sure you have enough bandwidth that is preferably performing the conversion on a LAN.

This should hopefully help you in thoroughly preparing the conversion process.

Other solutions to perform virtual machine conversions

In this last section of this chapter, I wanted to highlight some of the other solutions available to perform virtual machine conversions, besides the already detailed VMM 2012 R2 and MVMC 3.0 possibilities.

I only want to avoid having this sound like a commercial/marketing trick, as I have no intention at all in that direction. The tools mentioned hereafter are solutions I have been using with customers myself, or the customers were already using them. I suggest you have a look at the following websites to get a better understanding of the products, features, the licensing cost involved, and alike, to decide for yourself if they are valid solutions in your specific situation:

- Double Take by Vision Solutions (`http://www.visionsolutions.com/products/dt-move.aspx`)
- Acronis Backup and Recovery (`http://www.acronis.com/en-eu/business/backup/virtual-machine/`)
- 5Nine Easy V2V Converter (`http://www.5nine.com/vmware-hyper-v-v2v-conversion.aspx`)
- StarWind V2V Converter (`http://www.starwindsoftware.com/converter`)

And most probably, many more other solutions are out there…

Summary

In this chapter, we mainly focused on how existing VMware platforms can be integrated with Virtual Machine Manager 2012 R2. We started by showing you how management of VMware environments can be achieved from within VMM, by using the intermediate vCenter as a prerequisite. Without being complete, we walked you through possibilities such as overall VM management, true fabric integration, how to create new VMware VMs, and more.

In the second part of this chapter, we showed you how to migrate/convert existing VMware virtual machines to Hyper-V virtual machines, both from within VMM 2012 R2 or by using the Microsoft Virtual Machine Converter.

In the last section, we shared some of our own best practices in such conversion projects, as well as providing an overview list of third-party solutions that offer similar and comparable features.

In the next chapter, we step away a bit from VMM as such, talking about establishing a more "private cloud" enabled platform by using System Center App Controller as a self-service portal for your overall virtualized resources platform.

7

Operating Your Private Cloud Using SCVMM and App Controller

So far, we have been positioning SCVMM as a virtualized platform management solution, but mainly from a features and technical view. In *Chapter 5*, *Hyper-V Management Using System Center Virtual Machine Manager*, you learned how to use SCVMM to manage your Hyper-V environment, whereas *Chapter 6*, *Integrating System Center VMM 2012 R2 with Your VMware Environment*, showed you how to integrate SCVMM into your existing VMware platform. In both scenarios, we approached the possibilities from a technical standpoint. What can be achieved from within the SCVMM management console? How can you administer the network and storage fabrics?

In this chapter, we will focus more on the "services side" of virtualization, better known as clouds, whether that be a private cloud and/or a public cloud. Where some organizations see cloud computing as another technology bubble in IT, it has become very important for others. Cloud computing allows companies to save on IT costs, working more efficiently and dynamically, as the IT assets you need (servers, storage, applications, and alike) are also offered in a very efficient and dynamic way.

By using SCVMM in combination with some other tools, one can apply such cloud features in their private network, thus talking about private cloud infrastructures. On the other hand, SCVMM can also be integrated with public cloud infrastructures such as Windows Azure, allowing a true hybrid topology of an organization's IT infrastructure.

However, before we dive again right into the technology, we want you to have a good understanding of the concepts of cloud computing.

The concepts of cloud computing

In the past, companies have been using data centers all along. Massive computer rooms with large units of computing power were in place. From a technical perspective, applications and resources were allocated to a specific machine. With only a few examples such as database servers or terminal servers, most servers were underused, consuming on a average 15 percent CPU power, 40 percent memory, and so on.

About 10 years ago, the implementation of server virtualization became popular in data centers. By implementing virtualization, organizations still need physical servers, but on top of the physical layer, a hypervisor is installed, which allows for running virtual machines. Each virtual machine runs isolated inside this virtual environment. Typically, a single physical server runs 20 or more virtual machines, thus using about the maximum of the physical machine's resources, but still running inside the company's own data center or using hosting centers. This architecture was quickly followed by a new concept, called cloud computing.

In short, there are three possible cloud scenarios:

Cloud scenario	**Description**
Private cloud	A private cloud is managed and owned by the organization itself, providing IT resources using in-house infrastructure. The resources are used within that organization only.
Public cloud	A public cloud provides on-demand resources through the Internet, for about anybody. Typical resources in a public cloud services are applications, servers, and storage. In most cases, organizations pay on a pay-per-use basis, using monthly subscriptions. Typical examples of public cloud are Microsoft Azure, Office 365, Google Mail, Amazon AWS, and many others.
Hybrid cloud	A hybrid cloud is the combination of private and public cloud. Parts of the infrastructure remain inside the organization's data center, but other components are running inside the cloud provider's data center. In most cases, applications allow extensive integration between both environments. Exchange Online, Lync Online, and SharePoint Online are great examples of applications that can be configured in a hybrid topology.

What are services and service templates within SCVMM?

When talking about cloud computing, one of the key differentiators in comparison to "the classic data center as we always knew it" is servicing and automation and to an even greater extent, self-service. Self-service refers to the situation where—in most cases—a non-IT-savvy person is capable of deploying a server or an application. The true magic lies in automation activities (so-called runbooks) and scripts that are running in the background.

The following cloud service models are available nowadays:

- **Infrastructure as a Service** (**IaaS**): IaaS is the model where you manage a virtual infrastructure within your organization. For example, a lot of organizations are building such cloud models by using Windows Server 2012 R2 Hyper-V and System Center 2012 R2 as a management and administration solution.
- **Software as a Service** (**SaaS**): In this model, the focus lies on using an application, without needing the infrastructure as such. A good example is SalesForce.com or Office 365. You manage the application, but not the infrastructure behind it.
- **Platform as a Service** (**PaaS**): Much like the hybrid topology, PaaS is in between IaaS and SaaS; you are using infrastructure that is provided out of the cloud, and on top of that, you are also managing the applications that are running inside the cloud infrastructure. Microsoft Azure is a good example of PaaS.

As already mentioned, the true success of cloud services lies in the usability and ease of configuration of the given cloud model. Within Microsoft Azure for example, by using an easy 7-step wizard, one can deploy new virtual machines, new websites, allocate cloud storage, and more, without actually needing to be an expert in the underlying infrastructure layer (for example storage, networking, firewalling, server administrator, and so on).

It should be clear by now, and we actually already used it as an example, Windows Server 2012 R2 Hyper-V in combination with System Center 2012 R2 is the perfect combination for building and deploying your own cloud infrastructure.

Configuring services and clouds in SCVMM

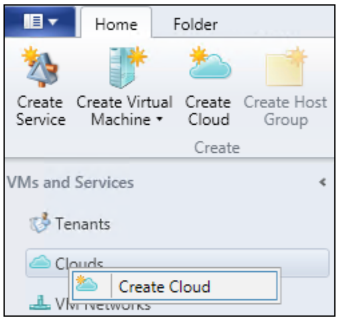
While working with SCVMM in *Chapter 5*, *Hyper-V Management Using System Center Virtual Machine Manager*, you have probably noticed the **Cloud** topic inside the SCVMM management console. The idea behind that "cloud" concept is creating a combination of hosts, host groups, virtual networks, storage classifications, and other resource objects you learned about in *Chapter 5*, *Hyper-V Management Using System Center Virtual Machine Manager*.

On top of the pure infrastructure components, user roles define what permissions a user can get to manage and deploy cloud resources, without actually touching or managing the infrastructure itself. A cloud can be configured in such a way that the provided resources are limited in capacity. Think back to the pay-per-use concept. If your sales department within the company needs more powerful virtual machines, they will pay more for these resources. As such, you could build a cloud "sales app", containing virtual machines, application servers, web servers, storage, backup services, and alike, which in turn are configurable by and visible only to the members of the sales team.

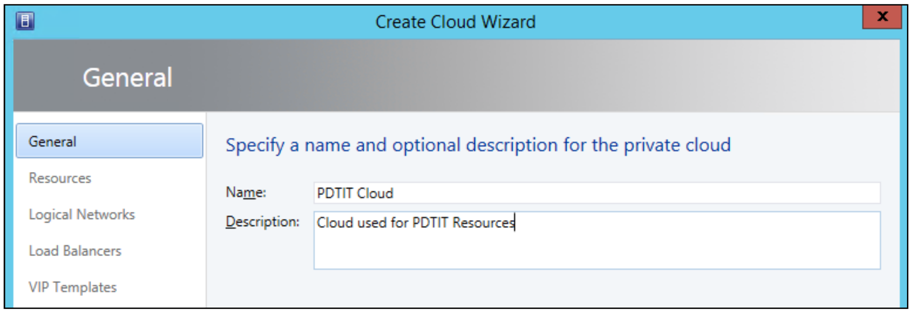
Configuring a cloud within SCVMM

After the more theoretical approach of cloud computing, cloud computing models, and services, let's get back to the SCVMM management console and configure a cloud in our lab environment:

1. From within the SCVMM console, click on **VMs and Services** and select **Cloud** | **Create Cloud**:



2. This launches the **Create Cloud Wizard** window:

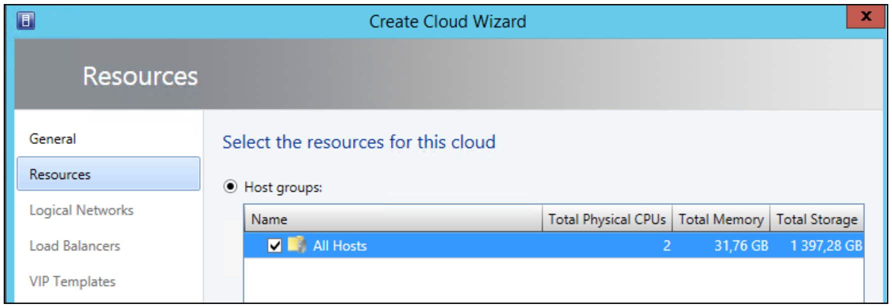


Specify a name and optional description for the private cloud

Name: PDTIT Cloud

Description: Cloud used for PDTIT Resources

3. Provide a name and description for your cloud, for example **PDTIT Cloud** as shown in the preceding screenshot. Click on Next.

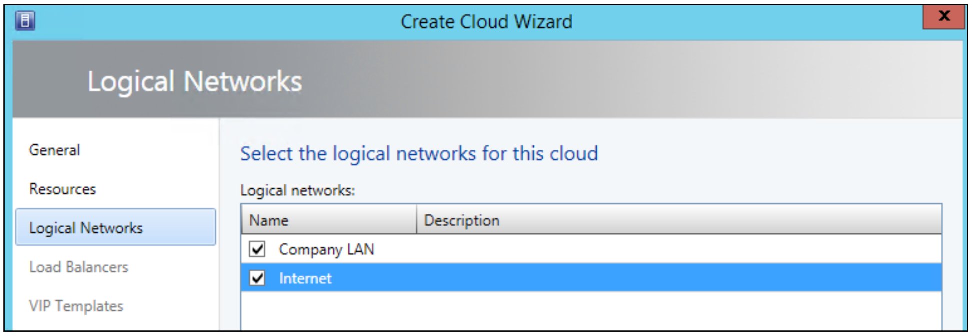


Select the resources for this cloud

● Host groups:

Name	Total Physical CPUs	Total Memory	Total Storage
☑ All Hosts	2	31,76 GB	1 397,28 GB

4. Select the resources you want to allocate to this cloud. In this example, we select **All Hosts** as demonstrated in the previous screenshot. Click on **Next**.

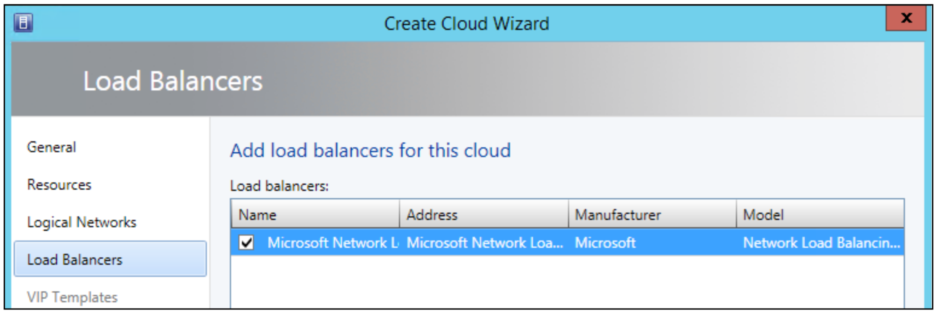


Select the logical networks for this cloud

Logical networks:

Name	Description
☑ Company LAN	
☑ Internet	

5. Select the logical networks that can be used for this cloud and click on **Next**.

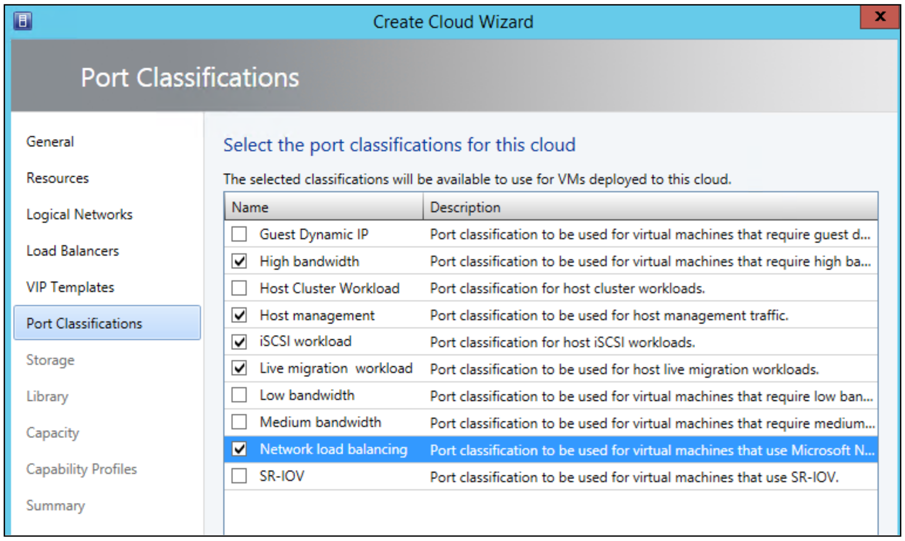


Add load balancers for this cloud

Load balancers:

Name	Address	Manufacturer	Model
☑ Microsoft Network L	Microsoft Network Loa...	Microsoft	Network Load Balancin...

6. Select the default **Microsoft Network Load Balancer** and click on **Next**.

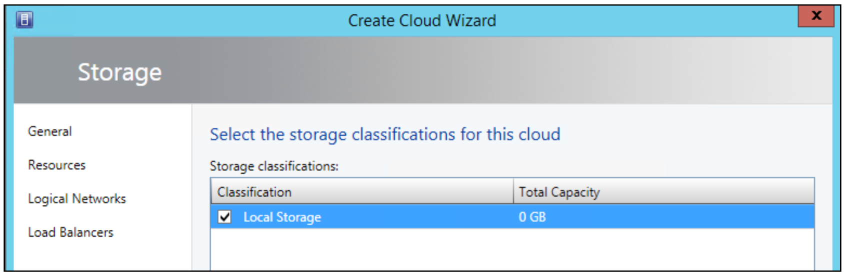


Select the port classifications for this cloud

The selected classifications will be available to use for VMs deployed to this cloud.

Name	Description
☐ Guest Dynamic IP	Port classification to be used for virtual machines that require guest d...
☑ High bandwidth	Port classification to be used for virtual machines that require high ba...
☐ Host Cluster Workload	Port classification for host cluster workloads.
☑ Host management	Port classification to be used for host management traffic.
☑ iSCSI workload	Port classification for host iSCSI workloads.
☑ Live migration workload	Port classification to be used for host live migration workloads.
☐ Low bandwidth	Port classification to be used for virtual machines that require low ban...
☐ Medium bandwidth	Port classification to be used for virtual machines that require medium...
☑ Network load balancing	Port classification to be used for virtual machines that use Microsoft N...
☐ SR-IOV	Port classification to be used for virtual machines that use SR-IOV.

7. Select the port classifications you want to use for this cloud; you can use my selections as an example as shown in the previous screenshot. Click on **Next**.

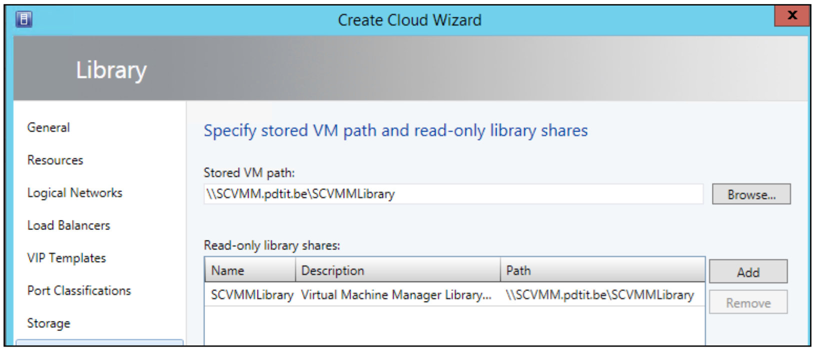


Select the storage classifications for this cloud

Storage classifications:

Classification	Total Capacity
☑ Local Storage	0 GB

8. Select the storage classification(s) you want to use; click on **Next**.



Specify stored VM path and read-only library shares

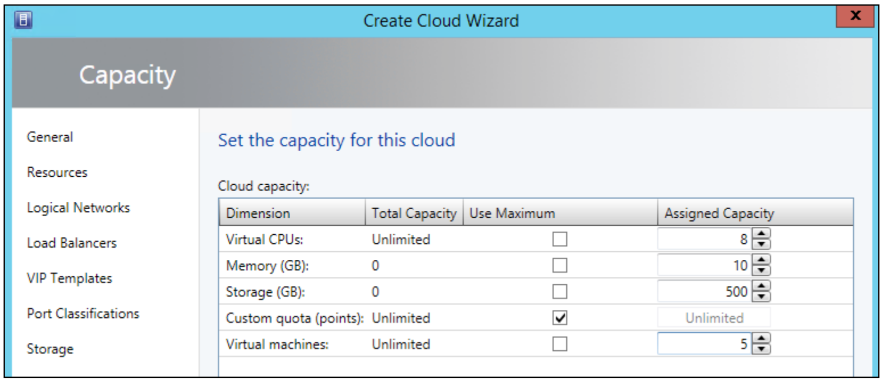
Stored VM path:

\\SCVMM.pdtit.be\SCVMMLibrary

Read-only library shares:

Name	Description	Path
SCVMMLibrary	Virtual Machine Manager Library...	\\SCVMM.pdtit.be\SCVMMLibrary

9. Browse to `SCVMMLibrary` and optionally define the read-only library shares you want to allocate to this cloud; click on **Next**.

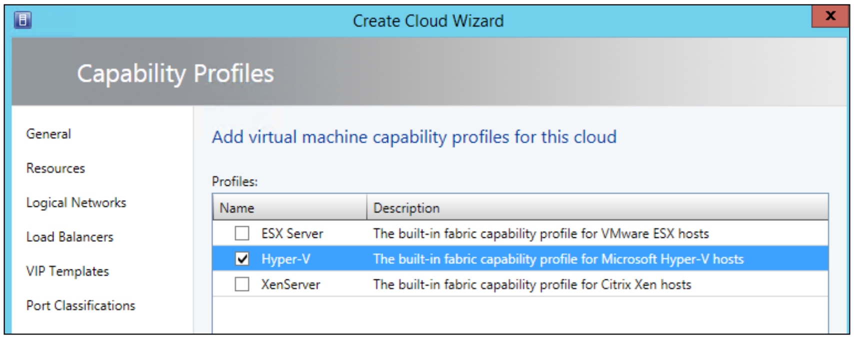


Set the capacity for this cloud

Cloud capacity:

Dimension	Total Capacity	Use Maximum	Assigned Capacity
Virtual CPUs:	Unlimited	☐	8
Memory (GB):	0	☐	10
Storage (GB):	0	☐	500
Custom quota (points):	Unlimited	☑	Unlimited
Virtual machines:	Unlimited	☐	5

10. In the next step, you will define the different capacity parameters for the resources inside the cloud. In our example, we want to show you how you can allocate assigned capacity settings or use all available resource capacity.



Add virtual machine capability profiles for this cloud

Profiles:

Name	Description
☐ ESX Server	The built-in fabric capability profile for VMware ESX hosts
☑ Hyper-V	The built-in fabric capability profile for Microsoft Hyper-V hosts
☐ XenServer	The built-in fabric capability profile for Citrix Xen hosts

11. In the last step, you can define what VM capability profiles can be used inside the given cloud and then click on **Next**. Click on **Finish** to have the cloud configured using the defined settings.

This completes the configuration of the cloud within SCVMM. Once the cloud is configured, we can assign this cloud to specific users or user groups, as well as deploy new virtual machines within this cloud.

As deploying a virtual machine within a cloud is no different from deploying a virtual machine inside SCVMM, you just rely on the configured resources of the cloud. We refer you to *Chapter 5*, *Hyper-V Management Using System Center Virtual Machine Manager*, again if you are not familiar with that.

How to configure service templates in SCVMM

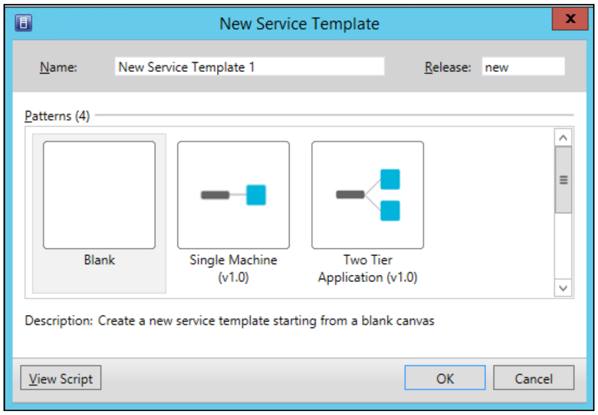
In this next topic, we will dive a bit deeper in the concept of services, by explaining and configuring the principal of service templates.

A service template is a preconfigured state of a service, such as the one we created in the previous topic. A service template contains information settings for a virtual machine, parameters about what application(s) needs to be installed on the virtual machine, the networking configuration settings of the virtual machine, and so on. A good example of "a service" could be "Web Shop", where a "service template" would be created for the "Web Shop Web Server" and the "Web Shop Database Server". All these separate blocks could also be bundled in a single service template. The service template "Web Shop Web Server" would define the characteristics of the Windows operating system, as well as the configuration settings for IIS running inside that VM. For the "Web Shop Database Server" service template, the virtual machine specifications would be different than those of the web server. Moreover, it will also contain configuration settings regarding the SQL Server database installation.

Creating a service template

Let's start by creating a service template in SCVMM 2012 R2:

1. From within the SCVMM console, go to the workspace in the left bottom corner, and select **Library**. In the top left bar that appears, notice the **Templates** section. Open up this section and notice the **Service Templates** topic. Right-click and choose **Create New Service Template**, or click on the button for this in the ribbon. This will launch the **New Service Template** wizard:



New Service Template

Name: New Service Template 1

Release: new

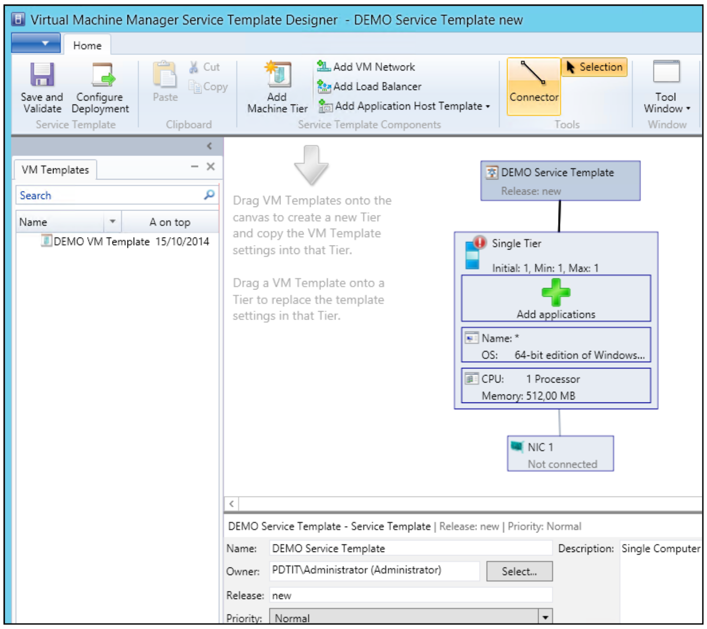
Patterns (4)

Description: Create a new service template starting from a blank canvas

2. Give a descriptive name and optionally a release reference number for this service template. In our example, we will build a service template for a new VM; therefore, select **Single Machine (v1.0)** from the available patterns; click on **OK** to continue.

   This will bring up the template designer detailed view, which will show the different "building blocks" that form this service template.

   Note the red alert icon saying that a virtual disk should be added to this template, before we can start using it for future deployment. See the following screenshot for reference:



Virtual Machine Manager Service Template Designer - DEMO Service Template new

DEMO VM Template 15/10/2014

Drag VM Templates onto the canvas to create a new Tier and copy the VM Template settings into that Tier.

Drag a VM Template onto a Tier to replace the template settings in that Tier.

DEMO Service Template
Release: new

Single Tier
Initial: 1, Min: 1, Max: 1

Name: *
OS: 64-bit edition of Windows...

CPU: 1 Processor
Memory: 512,00 MB

NIC 1
Not connected

DEMO Service Template - Service Template | Release: new | Priority: Normal

Name: DEMO Service Template

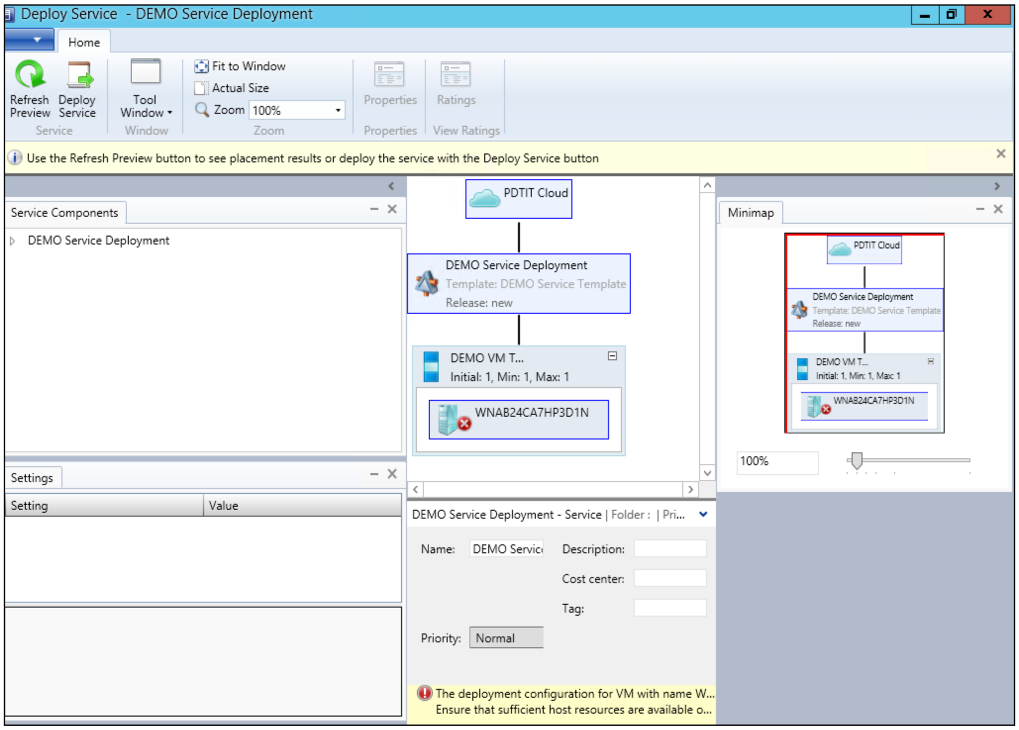
Description: Single Computer

Owner: PDTIT\Administrator (Administrator)

Release: new

Priority: Normal

3. To fix the error message regarding the virtual disk, you can drag and drop the available **DEMO VM Template** we created in one of the previous topics to the **Single Tier** dialog in the middle pane.



Use the Refresh Preview button to see placement results or deploy the service with the Deploy Service button

Name: DEMO Servic

Description:

Cost center:

Tag:

Priority: Normal

The deployment configuration for VM with name W...
Ensure that sufficient host resources are available o...

4. Once this is done, the service template needs to be saved (just click on the **Save** button in the menu ribbon for this).

> Note the error message **The deployment configuration for VM with name… cannot be completed. Ensure that sufficient host resources are available…**.
>
> This error message shouldn't be there at all, but is limited to my demo environment. As I'm using all resources I can get for my environment already, there is no resource space left for deploying this service template.

This completes the configuration of a service template. Our goal is to use this service template out of App Controller, which is what we will do later in this chapter.

If you are really interested in how to build and use service templates in the real world, have a look at the following blog posts:

- *Service Templates in System Center 2012 SP1 VMM: Build your Private Cloud* from Blain Barton's blog available at `http://blogs.technet.com/b/blainbar/archive/2013/04/26/service-templates-in-system-center-2012-sp1-vmm-build-your-private-cloud-series-part-5-of-5.aspx`.
- *Service Templates and SCVMM 2012 Demo* from Channel 9's Edge Show Series available at `http://channel9.msdn.com/posts/Service-Templates-and-SCVMM-2012-Demo`.

What are user roles and how to configure them in SCVMM

A user role in SCVMM is quite similar to other applications, where a user gets access to specific parts in the application, based on his user profile or group membership. If you are familiar with Exchange Server or Lync Server, by using **Role-based Access Control** (**RBAC**), one might give more or less administrative control to a non-administrative user, allowing that user to perform certain actions within the application (for example, create users, define new databases, and so on).

Related to SCVMM, a user profile (actually called user role within SCVMM) defines what resources a user has access to in the SCVMM cloud, from an administrative perspective. Let's say we created two different clouds within SCVMM. One user role could allow administration and deployment of new virtual machines on one cloud, without having access to the other cloud. Or, another example, in which a user can have the user role defined to deploy new virtual machines, but only to a gold tier storage layer that was configured before.

Within SCVMM, there are four possible user profiles available:

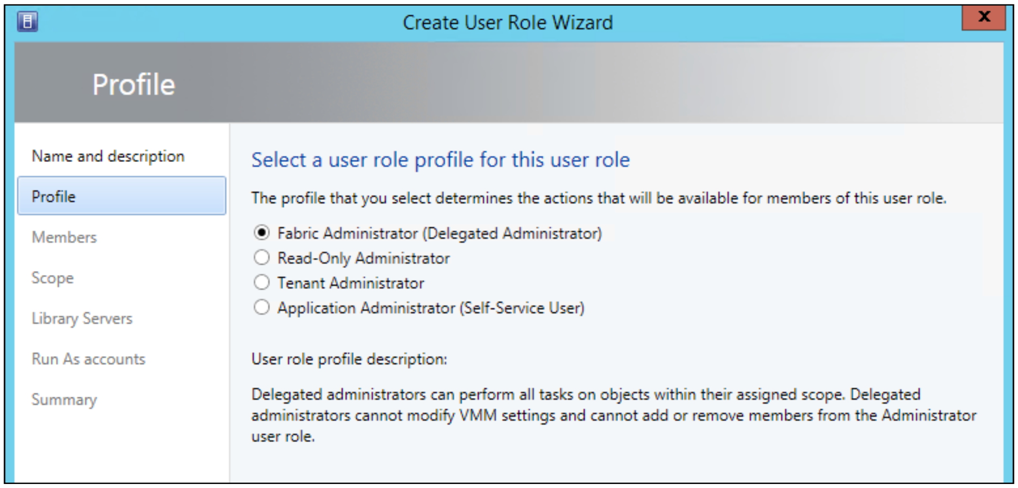
Fabric admin	The Fabric admin is similar to the SCVMM Administrator role, except missing the rights to add Citrix XenServer and WSUS servers (don't ask me why). The Fabric admin is also missing rights to change global SCVMM settings or add other administrator users.
Read-Only admin	A Read-Only admin has viewing rights to all settings, configurations, VMs, storage, and objects within SCVMM to which he is assigned.
Tenant admin	A Tenant admin has administrative rights for the tenant he/she is assigned to. Within the tenant, options for adding, changing, deleting, creating, SCVMM tenant objects are all available.
Application admin	An Application admin is also known as a self-service user. This user profile is able to deploy, create, manage, and administer their own virtual machines and services, without needing specific full administrative rights for the platform. Management can be done from within the SCVMM console, from the SCVMM web console, or by integrating with System Center App Controller or Windows Azure Pack for example.

User role configuration can be managed from within the SCVMM console or PowerShell.

Configuring user roles and mapping them to cloud resources

In this section, you will learn how to configure a user role, and have that user role linked to a cloud resource, limiting access to specific resource objects within that cloud. The steps are as follows:

1. From within the SCVMM console, go to the workspace, and select **Settings**.
2. From the menu ribbon, select **Create User Role**; this will start the **Create User Role Wizard** window.
3. Enter a name and description for the Admin user role you are creating; click on **Next**.



Select a user role profile for this user role

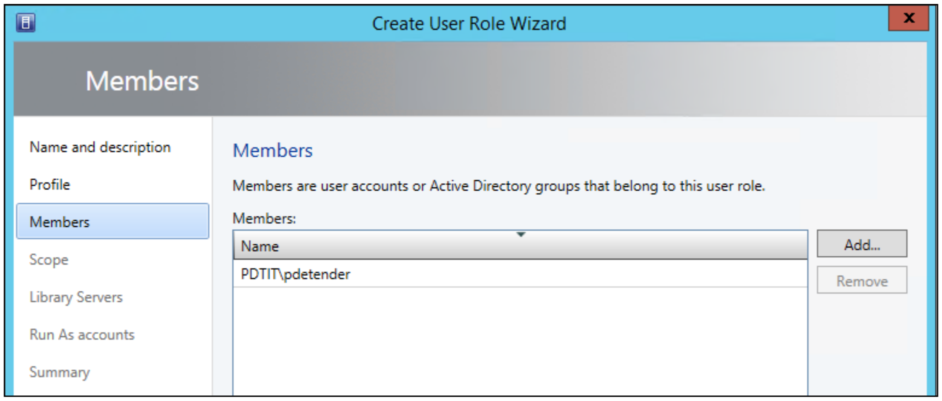
The profile that you select determines the actions that will be available for members of this user role.

● Fabric Administrator (Delegated Administrator)
○ Read-Only Administrator
○ Tenant Administrator
○ Application Administrator (Self-Service User)

User role profile description:

Delegated administrators can perform all tasks on objects within their assigned scope. Delegated administrators cannot modify VMM settings and cannot add or remove members from the Administrator user role.

4. Select the **Fabric Administrator (Delegated Administrator)** option as shown in the preceding screenshot. Click on **Next**.



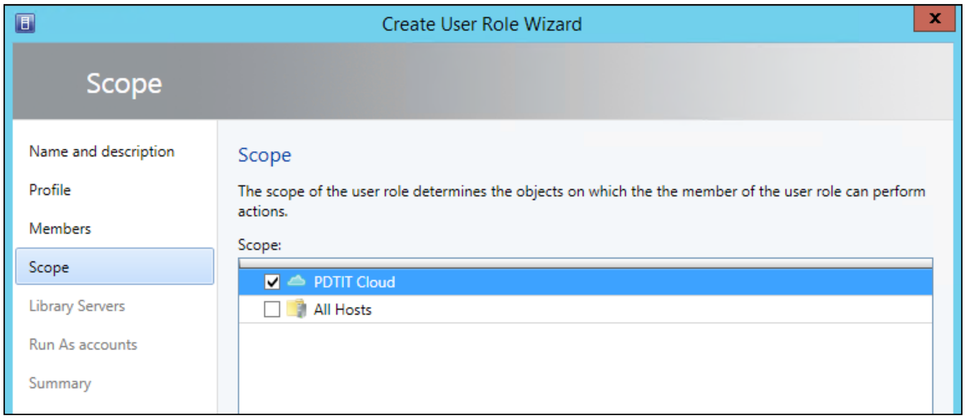
Members

Members are user accounts or Active Directory groups that belong to this user role.

Members:

Name
PDTIT\pdetender

5. Select the Active Directory domain members you want to nominate as Fabric administrators; in my example, I selected "Peter De Tender", which is a non-administrative user within the Active Directory domain. Click on **Next**.



Scope

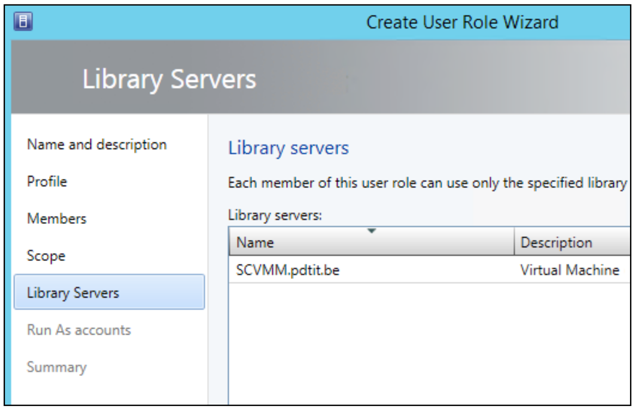
The scope of the user role determines the objects on which the the member of the user role can perform actions.

Scope:

☑ PDTIT Cloud
□ All Hosts

6. In the next step, select the scope to which this Administrator role should have access to; in our example, we select the **PDTIT Cloud** option we created before. Click on **Next**.

7. In the next step, we point the user role to our library servers in our environment, and click on **Next** as shown in the following screenshot:



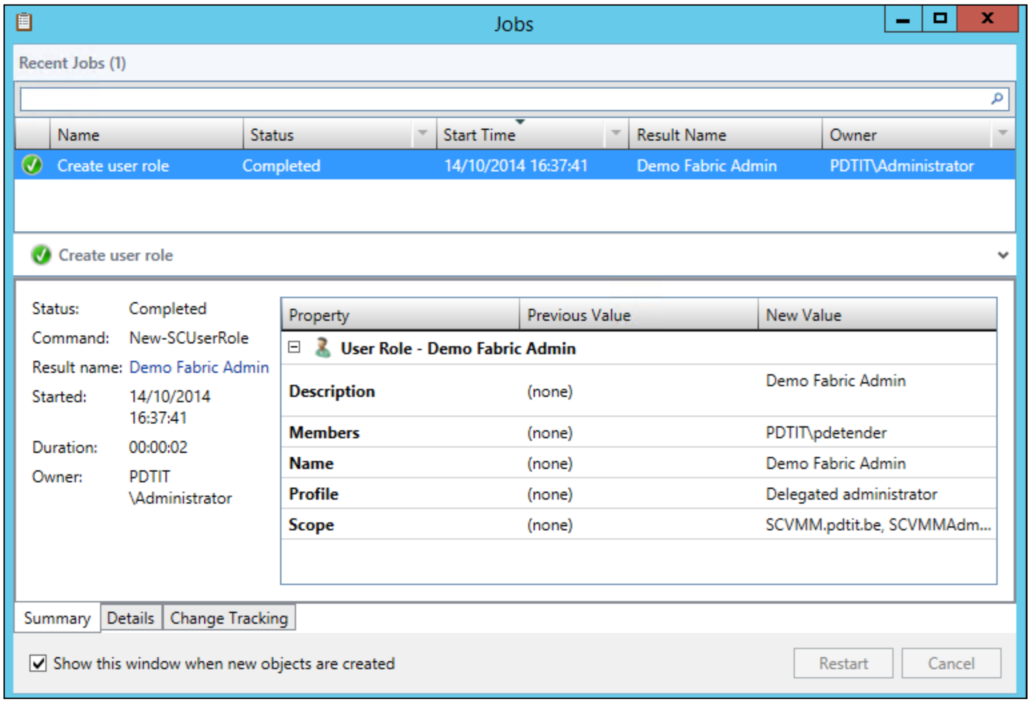
Library servers

Each member of this user role can use only the specified library

Library servers:

Name	Description
SCVMM.pdtit.be	Virtual Machine

8. In the last step, we select the Run As accounts this Fabric admin will have access to; after clicking on **Next**, a summary is shown, which can be completed by pressing the **Finish** button. After this, our new user role will be created as demonstrated in the following screenshot:



Name	Status	Start Time	Result Name	Owner
✓ Create user role	Completed	14/10/2014 16:37:41	Demo Fabric Admin	PDTIT\Administrator

✓ Create user role

Status: Completed
Command: New-SCUserRole
Result name: Demo Fabric Admin
Started: 14/10/2014 16:37:41
Duration: 00:00:02
Owner: PDTIT \Administrator

Property	Previous Value	New Value
⊟ **User Role - Demo Fabric Admin**		
Description	(none)	Demo Fabric Admin
Members	(none)	PDTIT\pdetender
Name	(none)	Demo Fabric Admin
Profile	(none)	Delegated administrator
Scope	(none)	SCVMM.pdtit.be, SCVMMAdm...

☑ Show this window when new objects are created

This completes the creation of a custom user role, linking it to an existing Active Directory user, and giving that user specific admin rights within the SCVMM console.

While we will not explain the detailed steps to test this, you can install the SCVMM management console on a Windows client machine, log on to that client using the Active Directory user account we linked earlier, and you should be able to "manage the fabric". It should be possible to create a new virtual machine, manage existing VMs in the PDTIT cloud, and so on.

In the next part of this module, we will re-use this user role in a self-service context, by leveraging the capabilities of SCVMM, using System Center App Controller.

Lastly, an interesting read on how to configure these user roles using PowerShell can be found at `http://www.yusufozturk.info/virtual-machine-manager/how-to-grant-user-role-on-vms-on-scvmm-2012-r2-via-powershell.html`.

Installing and configuring App Controller

As mentioned in the introduction of this chapter, App Controller is a web-based self-service solution, providing access to SCVMM applications and services to—in many cases—non-administrative users.

> In previous versions of SCVMM—before 2012—this feature was known as the VMM Self-Service Portal.

The biggest change to the older self-service portal and the new App Controller in my opinion is the possibility of not only managing your on-premises SCVMM environment, but also the management of Azure environments out of the same console. As such, it gives flexibility to manage resources in both environments. Fundamentally, App Controller connects to "a VMM cloud", which can either be a private cloud or the public Azure cloud.

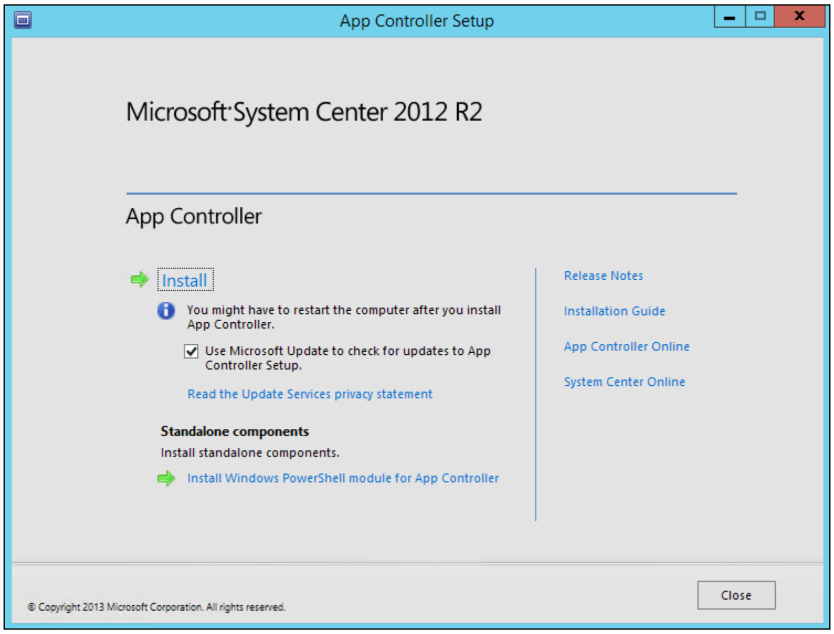
Before installing the App Controller from the website, take note of the following prerequisites:

- App Controller requires Silverlight, which means, your client should use Internet Explorer.
- App Controller can only be installed on a domain-joined machine (SCVMM also has this dependency).
- While not required for all features, we recommend using an SSL certificate to secure the application.

- Your App Controller server operating system needs to be Windows Server 2008 R2 SP1 or higher.
- As it is a web application, your App Controller server relies on IIS. It suffices to have the default IIS component installed; any missing features will be installed when going through the App Controller configuration wizard later on.
- Just like SCVMM itself, App Controller makes use of a SQL Server database. This can be SQL Server version 2008 SP2 and higher; there is no requirement to have a dedicated SQL Server though.
- App Controller requires the SCVMM console to be installed on the App Controller server. (In most situations, it's actually the other way around, people are installing the App Controller component on their SCVMM server.)

When all the previous requirements are ok in your environment, we can continue with the installation itself, as follows:

1. Launch the `setup.exe` file from the System Center App Controller 2012 R2 install media window:



App Controller Setup
Microsoft System Center 2012 R2
App Controller
Install
Release Notes
Installation Guide
App Controller Online
System Center Online
Read the Update Services privacy statement
Install Windows PowerShell module for App Controller
Close

You might have to restart the computer after you install App Controller.

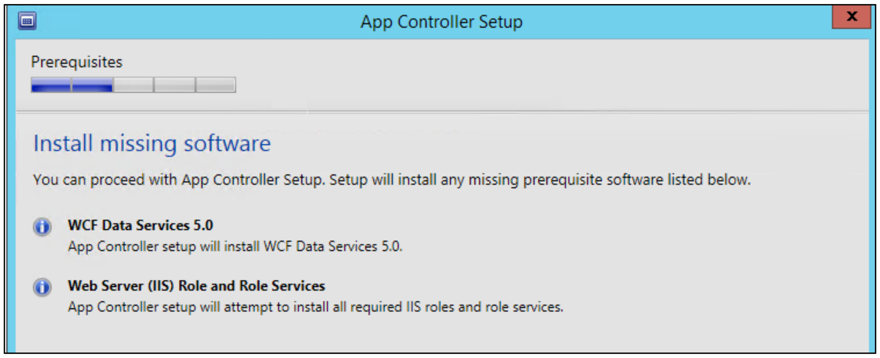
☑ Use Microsoft Update to check for updates to App Controller Setup.

Standalone components

Install standalone components.

© Copyright 2013 Microsoft Corporation. All rights reserved.

2. From the **App Controller Setup** window, select **Install**; we recommend leaving the option to use Microsoft Update active, so you get the latest updates installed.
3. In the next step, a check will run to find out if the server has all the required components installed to continue with the setup:



App Controller Setup

Prerequisites

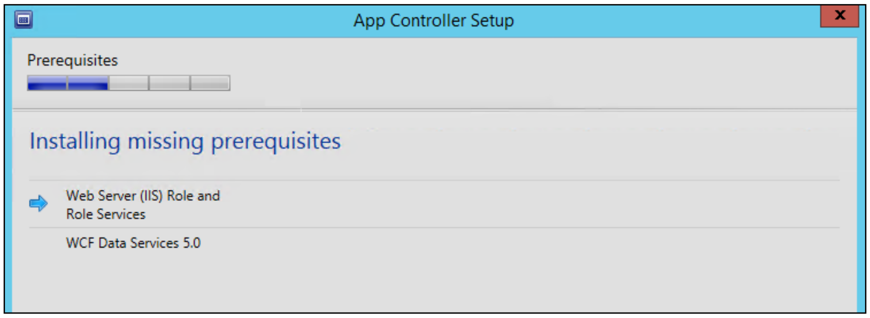
Install missing software

You can proceed with App Controller Setup. Setup will install any missing prerequisite software listed below.

WCF Data Services 5.0
App Controller setup will install WCF Data Services 5.0.

Web Server (IIS) Role and Role Services
App Controller setup will attempt to install all required IIS roles and role services.

4. I have added the preceding step on purpose, to prove that the missing components (basically, these are Windows 2012 R2 features) will be installed automatically during the App Controller setup. Click on **Install** to continue.



App Controller Setup

Prerequisites

Installing missing prerequisites

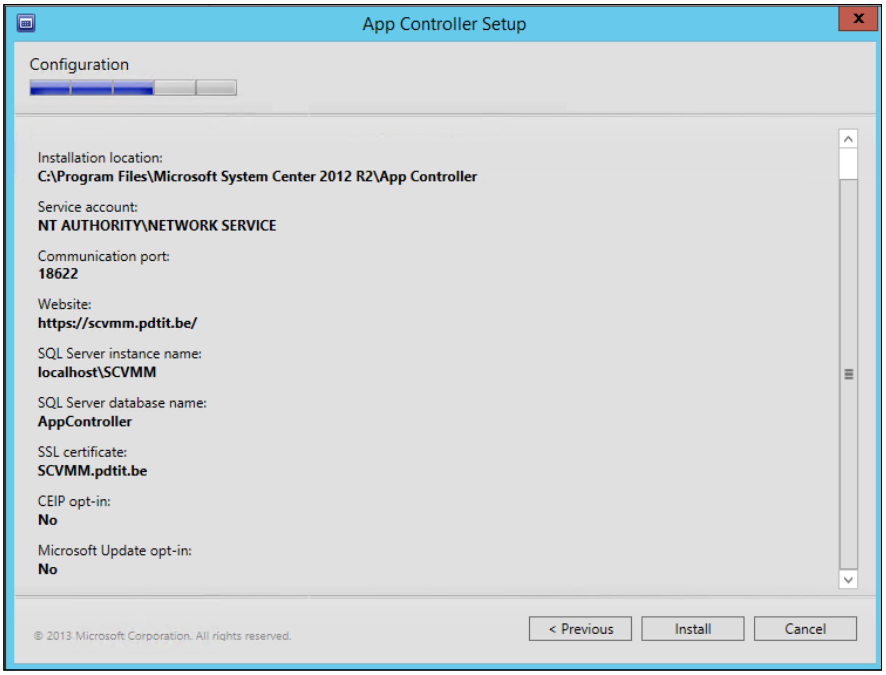
Web Server (IIS) Role and Role Services

WCF Data Services 5.0

5. The remaining steps of the installation wizard are rather easy, so I won't publish the screenshots here. It continues as follows:
   1. Selecting the installation path (which is the `C` drive by default).
   2. Specify the network service account or a specific App Controller service account. (I created the SCAPPCont_SRV user account for this.)

3. Configure the website, where I have accepted the defaults; I also selected the already existing SCVMM-machine's SSL-certificate (hence I'm installing the App Controller components on the SCVMM server itself)
4. Configure the SQL Server database settings, where I also accepted the defaults. (As SQL Server is already running on this machine, it shouldn't cause many problems. Just remember you can also configure a dedicated SQL Server for this.)

6. For your information, the following screenshot explains the summary of my settings during the installation process:



Installation location:
C:\Program Files\Microsoft System Center 2012 R2\App Controller

Service account:
NT AUTHORITY\NETWORK SERVICE

Communication port:
18622

Website:
https://scvmm.pdtit.be/

SQL Server instance name:
localhost\SCVMM

SQL Server database name:
AppController

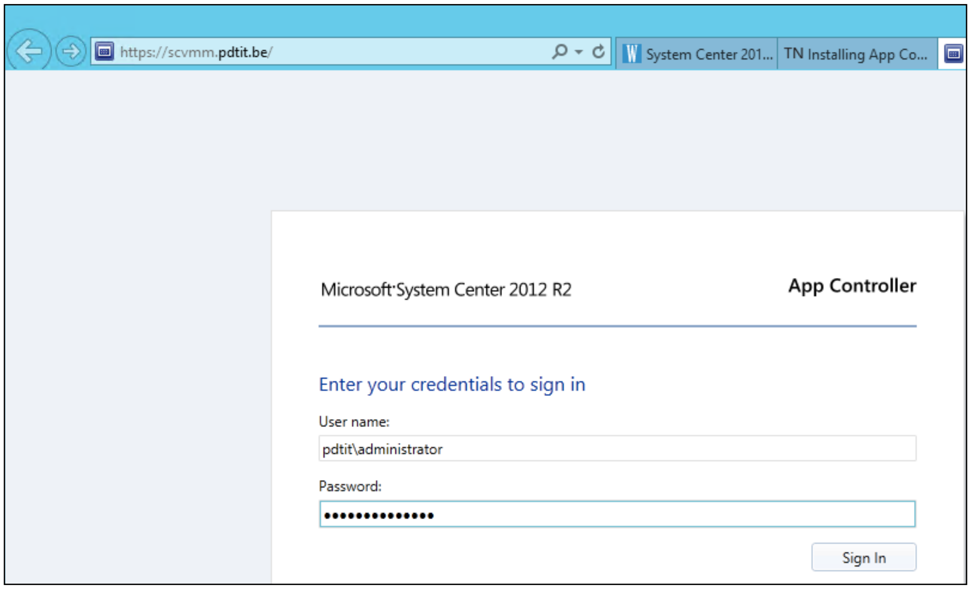
SSL certificate:
SCVMM.pdtit.be

CEIP opt-in:
No

Microsoft Update opt-in:
No

7. If all goes well, the setup should finish successfully, showing all green buttons.

8. Let's see if App Controller is indeed working, by connecting to our App Controller URL (`https://<AppControllerServerhttps://<AppControllerServerhttps://<AppControllerServer>;`) to make sure everything works, I generally test with the SCVMM admin or domain administrator account first:



Microsoft System Center 2012 R2

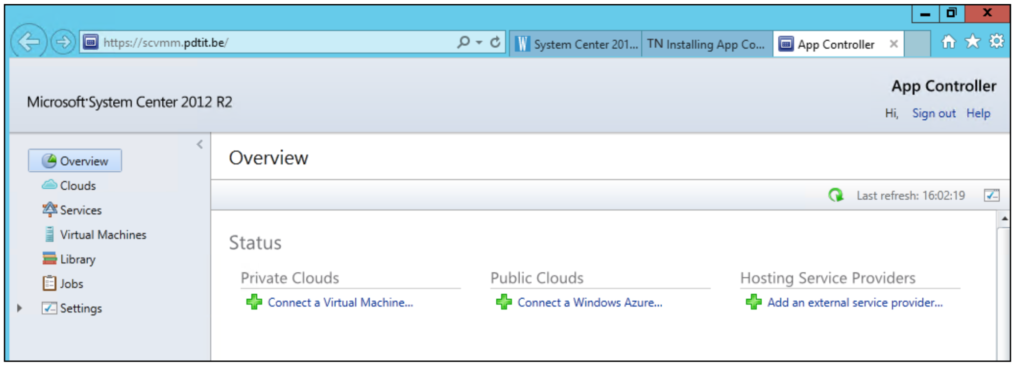
App Controller

Enter your credentials to sign in

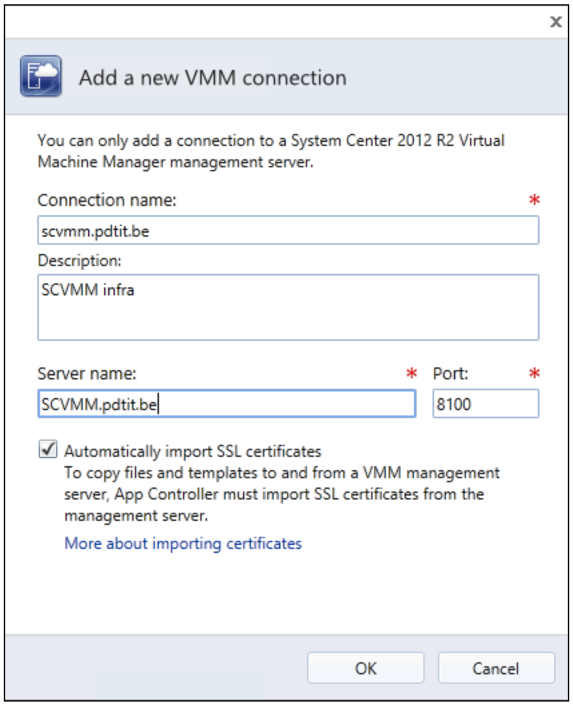
User name: pdtit\administrator

Password: ••••••••••••••

9. After signing in, we see the App Controller home page:



10. On the home page, select **Private Clouds**, followed by **Connect a Virtual Machine…** as shown in the previous screenshot.



Add a new VMM connection

You can only add a connection to a System Center 2012 R2 Virtual Machine Manager management server.

Connection name: * scvmm.pdtit.be

Description: SCVMM infra

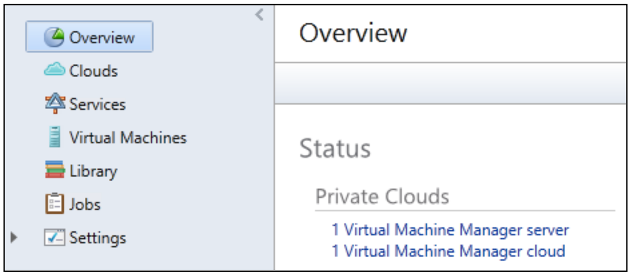
Server name: * SCVMM.pdtit.be

Port: * 8100

☑ Automatically import SSL certificates
To copy files and templates to and from a VMM management server, App Controller must import SSL certificates from the management server.

More about importing certificates

11. Enter the information details from your SCVMM environment. This should go well and provide you immediate information about the Hyper-V host, the virtual cloud it is part of, as well as details about the available VMs on that host.



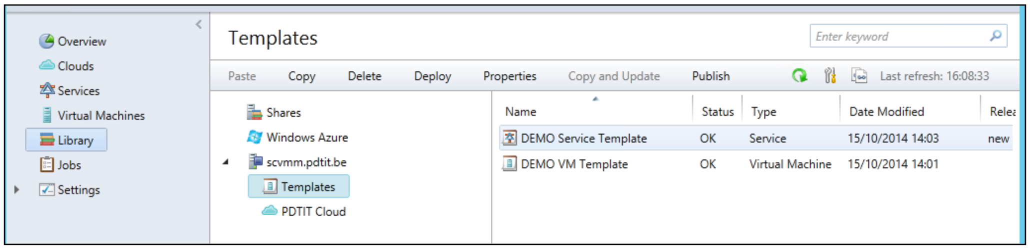
Overview

Status

Private Clouds

1 Virtual Machine Manager server
1 Virtual Machine Manager cloud

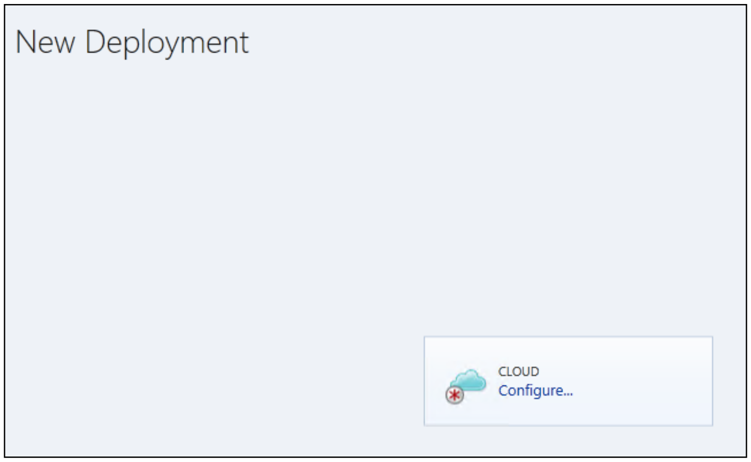
12. By going through the console, further on, for example, I select the VMM library, which will show me the service templates and VM templates we created earlier in this chapter:



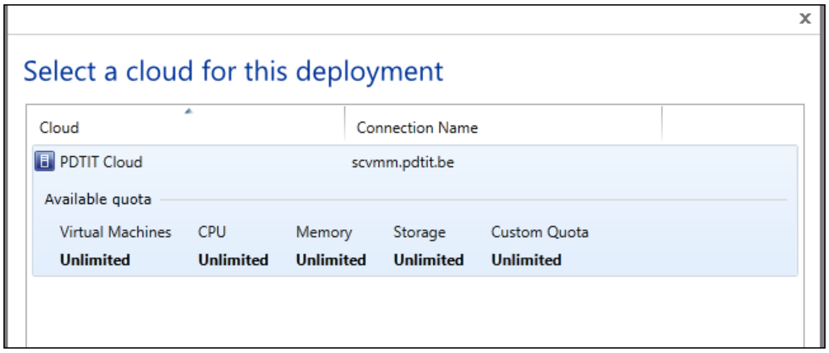
Name	Status	Type	Date Modified	Relea
DEMO Service Template	OK	Service	15/10/2014 14:03	new
DEMO VM Template	OK	Virtual Machine	15/10/2014 14:01	

Let's log out of this console now, and log back in with our non-admin user we created earlier in the user role section, and try to deploy a new virtual machine from here.

13. From within App Controller, select the **Virtual Machines** section. Next, click on the **Deploy** button.



14. The **New Deployment** window will show up, having an option to configure a cloud. Click on **Configure…** here.



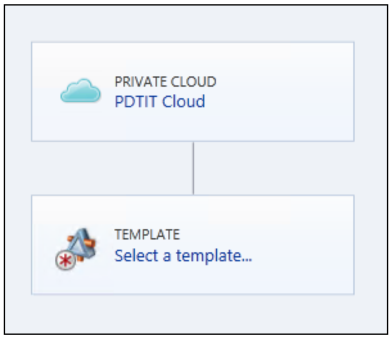
Select a cloud for this deployment

Cloud	Connection Name
PDTIT Cloud	scvmm.pdtit.be

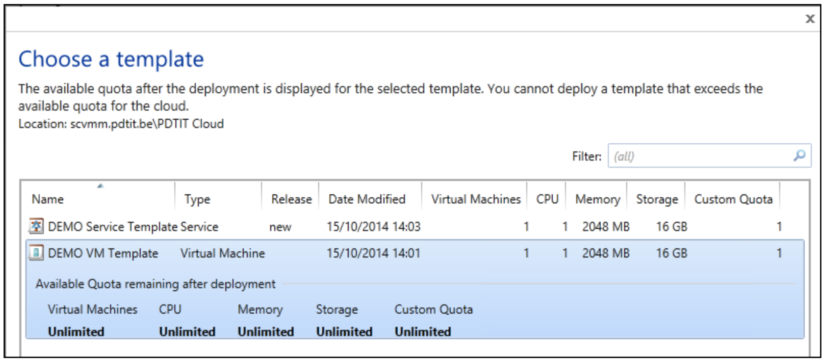
Available quota

Virtual Machines	CPU	Memory	Storage	Custom Quota
Unlimited	**Unlimited**	**Unlimited**	**Unlimited**	**Unlimited**

15. Select the **PDTIT Cloud** we created earlier in this chapter from SCVMM itself as shown in the preceding screenshot.



16. In the next step, the Private cloud **PDTIT Cloud** is recognized, allowing us to select templates out of this cloud, based on our user role access. Click on the **Select a template…** button.



Choose a template

The available quota after the deployment is displayed for the selected template. You cannot deploy a template that exceeds the available quota for the cloud.
Location: scvmm.pdtit.be\PDTIT Cloud

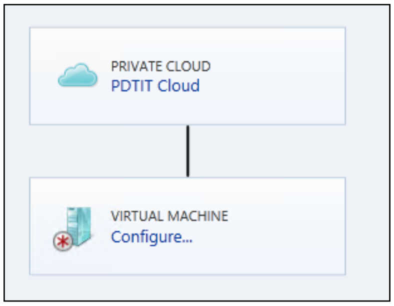
Filter: (all)

Name	Type	Release	Date Modified	Virtual Machines	CPU	Memory	Storage	Custom Quota
DEMO Service Template	Service	new	15/10/2014 14:03	1	1	2048 MB	16 GB	1
DEMO VM Template	Virtual Machine		15/10/2014 14:01	1	1	2048 MB	16 GB	1

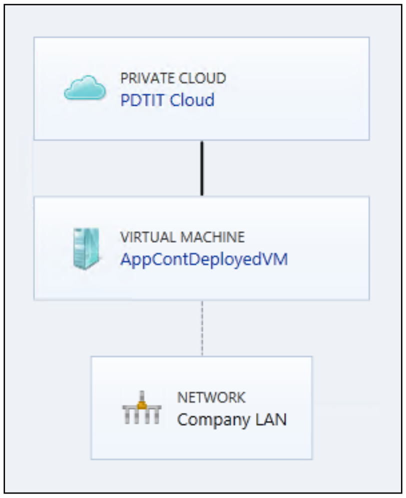
Available Quota remaining after deployment

Virtual Machines	CPU	Memory	Storage	Custom Quota
Unlimited	**Unlimited**	**Unlimited**	**Unlimited**	**Unlimited**

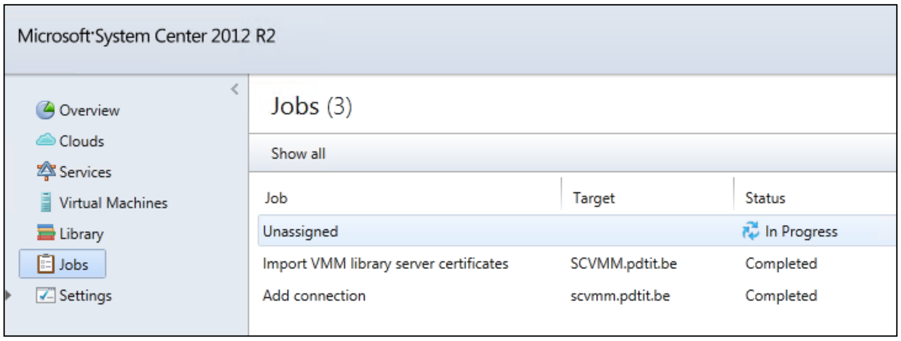
17. This will show us the different templates we can choose from. Notice both service templates and virtual machine templates that are available. In this example, select the **DEMO VM Template** option as demonstrated in the previous screenshot.



18. Click on **Configure…** to define specific configuration settings for the new VM we want to deploy in the next step. Core settings are server name and description, as well as providing the network settings.



19. As the last step, click on the **Deploy** button on the right.



Job	Target	Status
Unassigned		In Progress
Import VMM library server certificates	SCVMM.pdtit.be	Completed
Add connection	scvmm.pdtit.be	Completed

20. Finally, you are brought to the **Jobs** window, which will show you the VM deployment is **In Progress**. After some time, this job will be shown as completed, and the virtual machine is deployed successfully.

Summary

In this chapter, we started with explaining the concepts of cloud computing and detailing the differences between IaaS, PaaS, and SaaS. After the introduction, we focused on the configuration of a private cloud within SCVMM, what service templates are, and how you can use them within your private cloud, as well as what VM templates are and how they integrate in the overall SCVMM cloud story.

After that, we walked you through the purpose of using and configuring user role profiles to streamline self-service possibilities inside your data center cloud platform.

In the last section, we leveraged the self-service features by showing you how to install and configure System Center App Controller, a web portal application allowing true self-service to your end users, giving them the means to manage parts of the private cloud, based on configured access rights and service templates we created earlier.

In the next chapter, we will take another interesting and important step, explaining to you the possibilities of Microsoft Azure, the public cloud data center platform of Microsoft.

8

The Road to a Public Cloud Data Center Infrastructure Using Microsoft Azure

As IT administrators, architects, and technology leaders, very often, you're in the spotlight on deciding what technological road a company can take. Whatever technology is chosen, it's your job to make things happen. Whether business is booming and new branch offices have to be set up, or you're entering new markets with new products, or there's a vision to deliver more efficient IT operations, your team is always expected to deliver. And all this is happening at a much faster pace than before. Provisioning capacity right away, upgrading apps with new databases as quick as yesterday, taking non-compliant and old hardware out of the data center immediately, building test labs for your developers in a snap, prototyping that new partner collaboration site, and so on; as you know, the list goes on.

That's one of the reasons why virtualization was adopted at a high speed in a lot of organizations, as it helped IT to deliver fast(er). The process for requesting a new departmental server or application was about the same, but the delivery was a lot quicker. No waiting for physical server and storage ordering, no waiting on IT to actually install the server and the application. While the installation and configuration is still needed, a lot of the—mostly manual—steps from that time are now automated by using scripts or starting deployment from (service) templates, as mentioned in the previous chapter.

But with that fast pace comes also a downside. With virtualization being a standard nowadays, heads of departments and even non-IT people take it for granted that a new application is available instantaneously. Even referring to the store-based installation of software, such as with Apple Store, Google Market, and Windows 8 Store, a user is so "spoiled" for having an application installed and available immediately that they are setting the same expectations in the office.

Just because it "looks" so easy to deploy new applications, IT has to adapt again. By going through the previous chapter, it should be clear that a lot of operational performance can be gained by working towards "a cloud". *Chapter 7*, *Operating Your Private Cloud Using SCVMM and App Controller*, primarily focused on the private cloud, where your servers and applications are running inside your own managed data center.

But there's more. We are not only talking about private cloud anymore. For about 3 years, technology companies such as Microsoft, Google, Amazon, RackSpace, and many others have also offered "virtual machines in the cloud", and "applications in the cloud", actually providing a mix of IaaS, PaaS, and SaaS we talked about in the previous chapter as well.

In this chapter, we will focus on Microsoft Azure, the public cloud data center infrastructure (IaaS and PaaS), which offers a full set of features such as deploying virtual machines, virtual networks, SQL Server databases, storage blobs, web server platforms, and much, much more. In fact, features are being integrated, deployed, and optimized in Microsoft Azure faster than I can write about them.

This chapter is structured as follows:

- Overview of Microsoft Azure
- Creating, configuring, and managing virtual machines (VMs) in Microsoft Azure
- Integrating your on-premises VMs into Microsoft Azure

Let's have a closer look…

Overview of Microsoft's public cloud data center infrastructure – Microsoft Azure

When I was attending a train-the-trainer session on Windows Azure about 4 years ago, the presenter of the session asked the question "What is Windows Azure?". Given the fact it was rather new at that time, and far from the full and almost unlimited feature-set it has today, his answer was:

> "*Windows Azure can be about anything you want it to be*".

> Note that Windows Azure has been officially renamed Microsoft Azure as of August 2014.

In short, that "anything" is about true nowadays. Azure provides an almost complete feature-set, starting from deploying virtual machines, virtual networks, high availability, and disaster recovery, both within Azure (in the same data center or stretched over different Azure locations) or even in a hybrid topology between Azure and an organization's own private cloud data center. Next to that, it allows for deployment and management of applications and application servers. SQL database servers and storage, web server applications such as WordPress and Joomla, to name just a few, are part of the default capabilities. Azure provides a full set of storage blobs, actually allowing an organization to store data in the cloud accessible from web applications or connected directly to this virtual online storage or from integration with the native Windows backup tool in Server 2012 or System Center Data Protection Manager 2012.

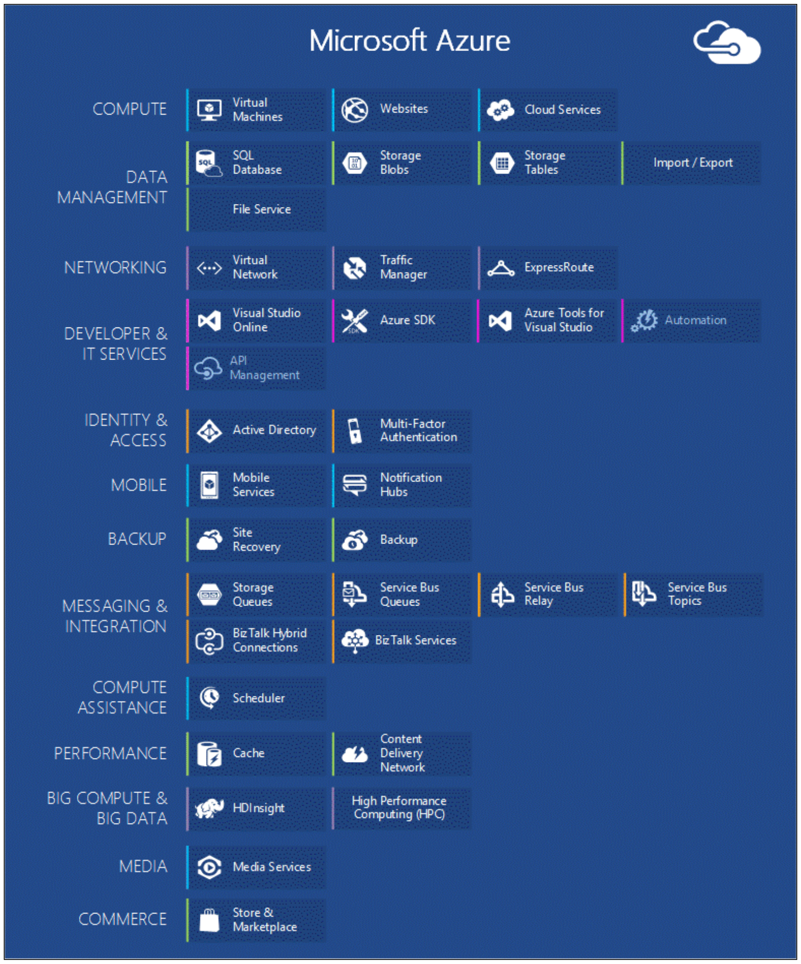
From a developer's perspective, Microsoft Azure provides complete availability and scalability of test and dev platforms, just as is custom in an on-premises environment. In addition to that, one of the newer additions to the portfolio is Visual Studio online, an almost identical copy of the well-known Visual Studio developer suite, but running in the cloud.

When talking about identity and security, Microsoft Azure provides Azure Active Directory as well as Multi-Factor Authentication.

More on the application side, components such as storage queues, service bus queues, BizTalk hybrid connections, media services, Store, and Marketplace, along with many more, are welcome features for developers as well.

All of this is available from an intuitive and crisp Microsoft Azure management portal and by using Azure PowerShell plugins.

The following diagram gives a clear overview of the different Microsoft Azure components at the time of writing this book the (source is `http://www.microsoft.com/azure`):



Let's have a look at it from a more official side. Here is Microsoft's own official description of Windows Azure:

> *"Windows Azure is an open and flexible cloud platform that enables you to quickly build, deploy, and manage applications across a global network of Microsoft-managed data centers. You can build applications using any language, tool, or framework. And you can integrate your public cloud applications with your existing IT environment."*

From a more functional approach, Microsoft Azure provides the following services:

- Computing resources
- Web and mobile
- Data and storage
- Analytics
- Networking resources
- Storage and backup services
- Media and CDN
- Identity and access management
- Applications and developer services

Besides these standard classifications, more and more components are added on a regular basis. One of the latest ones is Azure Operational Insights, the new version of the former System Center Advisor. But this could also fit in the computing resources classification for now.

Let's talk a bit more in detail about each classification, and what resources are available in there.

Computing resources

The computing resources category contains the options to deploy and manage new virtual machines and cloud services that allow for building scalable and highly available cloud applications and APIs. From within the **Batch Jobs** section, one can create and manage large-scale parallel and batch computing jobs. The **Scheduled Jobs** section provides tools to run recurring and scheduled jobs. Finally, **Azure RemoteApps** is the section from where one can deploy Windows client applications in the cloud, which can be run on any device, including iOS and Android devices.

Web and mobile

The web and mobile category contains the options to deploy new websites as well as build and manage mobile services backend for any mobile app you want; there is also a subsection allowing you to manage and build APIs and provide them to developers, partners, and employees. The newest addition to this section is the **Notifications Hub**, which allows for scalable, cross-platform push-notification infrastructure services.

Data and storage

The data and storage category provides database services such as **SQL Azure**, which is very similar to the on-premises SQL Server database services you might already be familiar with. **DocumentDB** is a managed NoSQL document database platform service, which allows you to build a document management web application backend. **Storage blobs** can be accessed by your Azure VMs or website applications. **StorSimple** is a hybrid cloud storage for enterprise organizations that uses an on-premise appliance to synchronize data to the Azure storage environment in a secure and cost-effective way.

Analytics

The analytics section is mainly used for data-mining services such as **HDInsight**, providing Hadoop large data clusters. **Machine learning** is a rather new feature, allowing for cloud-based predictive statistics. **Data factory** is the component that can be used to orchestrate and manage data transformation and movement. The **event hub** component is the one that allows for the ingestion, persistence, and processing of millions of events per second.

Networking resources

In the networking resources section, three components are present at the time of writing. **Virtual network** allows you to provision private networks in a pure-cloud configuration or hybrid topology. **ExpressRoute** is a dedicated private network fiber connection that is only available from specific telecom providers. **Traffic Manager** is the load balancer engine to traffic for high performance and availability.

Storage and backup services

The storage and backup services section provides you with **storage** options, **backup** for the cloud configuration options, and **Azure Site Recovery** (which provides disaster recovery services in a hybrid topology with your private cloud data centers).

Media and CDN

Media services allow you to encode, store, and stream video and audio at a large scale. **Content delivery network** (**CDN**) can be used to deliver content to end users by using the robust network of global data centers.

Identity and access management

The two main components that can be found in this classification are **Active Directory**, which allows for a synchronization between your on-premises directory and enables single sign-on features, and **multi-factor authentication**, which secures your data and apps by leveraging an extra level of authentication (for example, a text message on your cell phone).

Applications and developer services

At present, this refers to Visual Studio Online, the well-known Visual Studio software development environment, but running in the cloud.

Think back to our introduction, Azure can be about anything you want it to be. I think the preceding listing and description of the core features of Microsoft Azure make this statement true.

While going back to the key topic of this book, Microsoft virtualization, we won't discuss all the mentioned features in more detail (might be a good idea for a future book though, who knows…), but will focus on the ones that are related to virtualization. They are as follows:

- Computing resources and managing and administering virtual machines
- Using Azure as a backup-to-the-cloud solution
- Managing your hybrid data center by using App Controller

Keeping in mind that this list is (again) already pretty extensive, we will only be able to show you the surface of the specific feature, with maybe a little bit more under-the-hood guidelines on how to configure, use, and administer them.

Creating your Microsoft Azure account

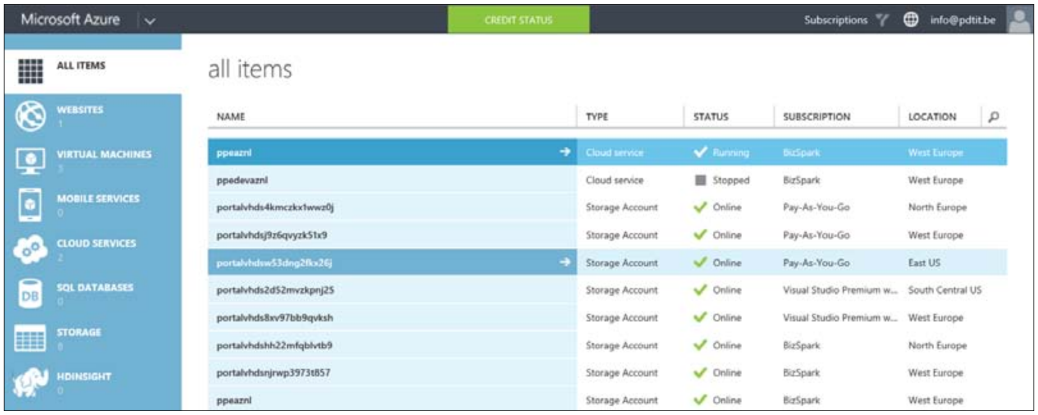
While we won't focus too much on how to actually get a Microsoft Azure account (check with your Microsoft contact inside the company or within Microsoft directly, if you have one, about what the options are for your organization), we thought it might be of interest—if you are totally new to Azure—to show you how to start from scratch. Given the fact you get access to most of the Azure features for free for 30 days, it makes it an ideal solution for trying things out without needing to break the bank.

> Please note that you have to enter valid credit card information for identity verification purposes only. Your credit card will not be charged for this offer unless you explicitly remove the spending limit that is built-in in the trial mode.

The steps are as follows:

1. Browse to `http://azure.microsoft.com`.
2. Click on the **Free Trial** button.
3. This should bring you to the **Free-One Month trial** sign-up page, giving you $200 worth of spending on Azure services.
4. Log in by using a Microsoft account (formerly, Windows Live ID).
5. Complete the sign-up form details.

After successful creation of your Azure account, you should be able to log in and get access to the Microsoft Azure portal, which looks similar to the following screenshot (Don't mind the resources that are already visible in my environment. Yours should be totally empty for the moment.)



all items

NAME	TYPE	STATUS	SUBSCRIPTION	LOCATION
ppeaznl	Cloud service	✓ Running	BizSpark	West Europe
ppedevaznl	Cloud service	■ Stopped	BizSpark	West Europe
portalvhds4kmczkx1wwz0j	Storage Account	✓ Online	Pay-As-You-Go	North Europe
portalvhdsj9z6qvyzk51x9	Storage Account	✓ Online	Pay-As-You-Go	West Europe
portalvhdsw53dng2fkx26j	Storage Account	✓ Online	Pay-As-You-Go	East US
portalvhds2d52mvzkpnj25	Storage Account	✓ Online	Visual Studio Premium w...	South Central US
portalvhds8xv97bb9qvksh	Storage Account	✓ Online	Visual Studio Premium w...	West Europe
portalvhdshh22mfqblvtb9	Storage Account	✓ Online	BizSpark	North Europe
portalvhdsnjrwp3973t857	Storage Account	✓ Online	BizSpark	West Europe
ppeaznl	Storage Account	✓ Online	BizSpark	West Europe

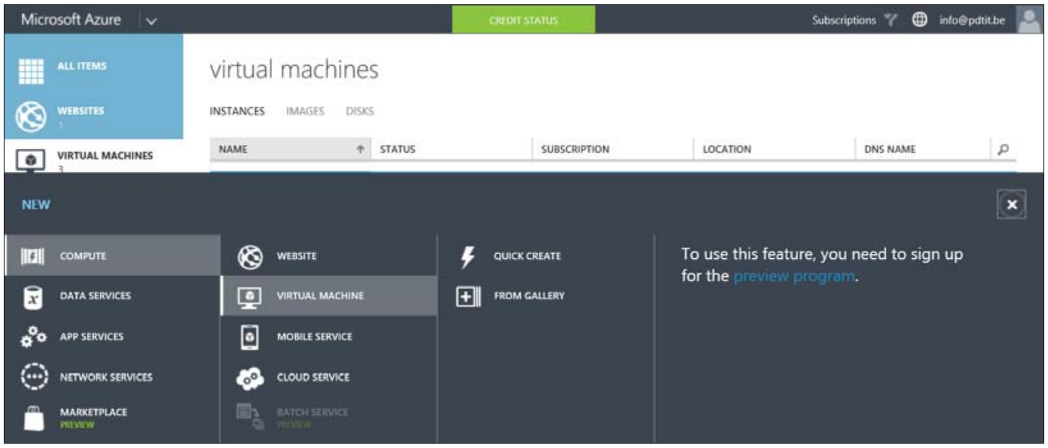
This completes the steps on how to create your Microsoft Azure account and get access to the Microsoft Azure portal.

Let's continue with actually using some of the available resources, by creating and configuring a virtual machine in the next section.

Creating and configuring virtual machines within Microsoft Azure

The steps to configure a virtual machine using Microsoft Azure are as follows:

1. From within the **Microsoft Azure** portal, select **VIRTUAL MACHINES** from the left ribbon menu; this will open up the **virtual machines** management pane.
2. Click the **NEW** button to start the new virtual machine wizard:



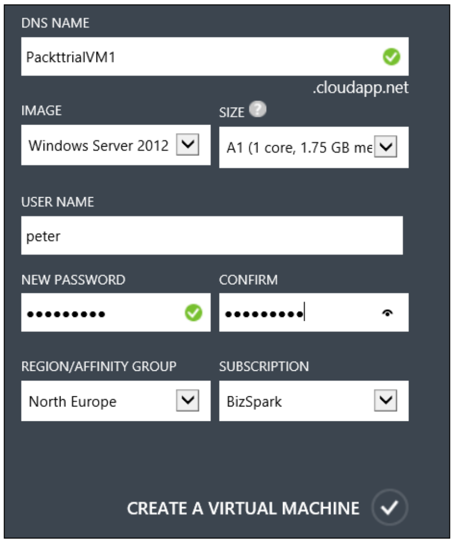
To use this feature, you need to sign up for the preview program.

Notice the two options we have available for creating a virtual machine. We can choose for a **QUICK CREATE** option or a **FROM GALLERY** option. The **QUICK CREATE** option asks just a few options around machine specs, machine name, OS flavor, and region for deployment. The **FROM GALLERY** option will provide us with an extensive list of pre-built machine configurations we can choose from, including Windows Server 2008 R2, Windows Server 2012 R2, SQL Server 2012 machines, and more. Even Linux VMs are there, as well as the technical preview of Windows Server 10.

Deploying a VM in Azure using the QUICK CREATE option

The steps to deploy a VM using the **QUICK CREATE** option are as follows:

1. Select the **QUICK CREATE** option, and complete the parameters, following the example screenshot:



DNS NAME: PackttrialVM1 ✓ .cloudapp.net

IMAGE: Windows Server 2012

SIZE: A1 (1 core, 1.75 GB me

USER NAME: peter

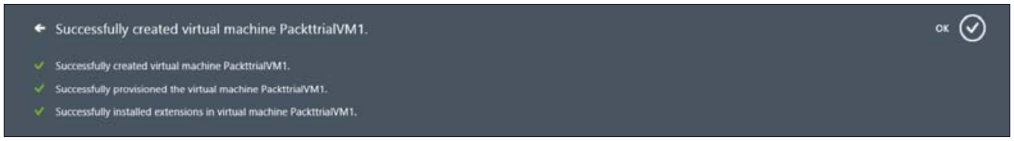
NEW PASSWORD: •••••••••

CONFIRM: •••••••••

REGION/AFFINITY GROUP: North Europe

SUBSCRIPTION: BizSpark

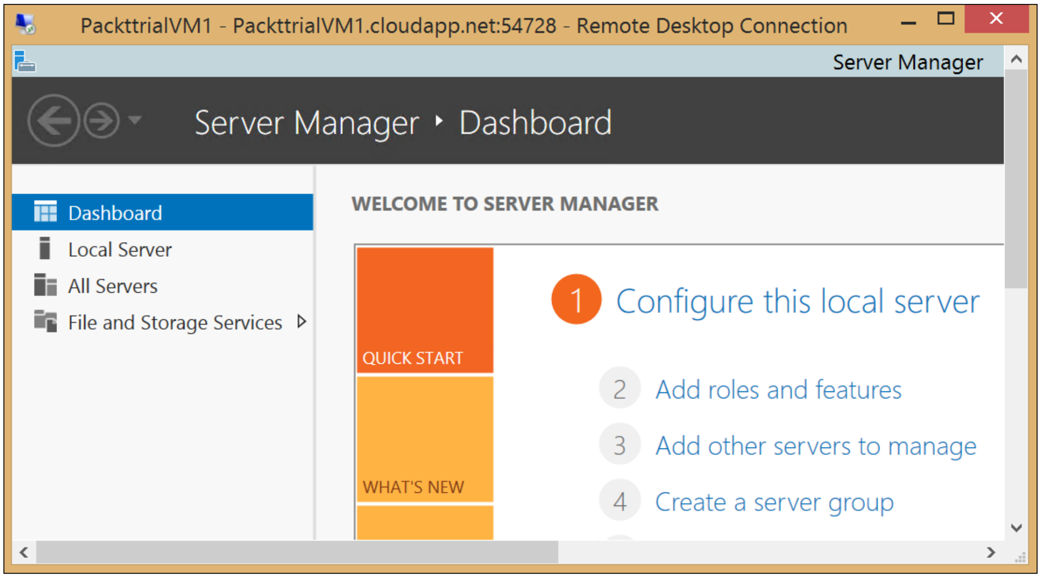
2. Click on **CREATE A VIRTUAL MACHINE** to have it created.
3. After going back to the Microsoft Azure portal, you can follow the process of the VM deployment in the status bar at the bottom, as shown in the following screenshot:



Successfully created virtual machine PackttrialVM1.

✓ Successfully created virtual machine PackttrialVM1.
✓ Successfully provisioned the virtual machine PackttrialVM1.
✓ Successfully installed extensions in virtual machine PackttrialVM1.

4. After only a few minutes, the VM provisioning is completed, after which we can log on to the virtual machine using the credentials we provided earlier in the setup wizard. To do this, click on the **Connect** button and open or save the RDP file that is presented, as shown in the following screenshot:


PackttrialVM1 - PackttrialVM1.cloudapp.net:54728 - Remote Desktop Connection
Server Manager
Server Manager ‣ Dashboard
Dashboard
Local Server
All Servers
File and Storage Services ▷
WELCOME TO SERVER MANAGER
QUICK START
WHAT'S NEW
1 Configure this local server
2 Add roles and features
3 Add other servers to manage
4 Create a server group

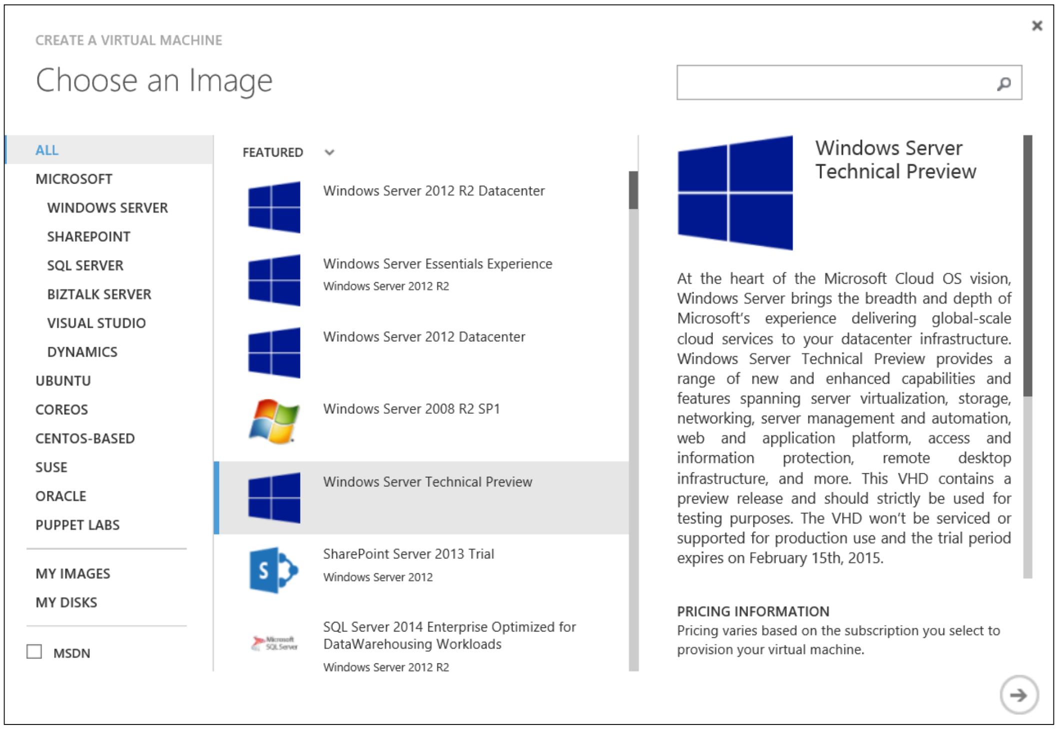
This completes the deployment of a new virtual machine in Microsoft Azure by using the **QUICK CREATE** option.

After logging on to the VM, we have full local administrative user rights, giving us the possibility to administer this server just as if we deployed it in our own data center. We can now configure this server as a domain controller, web server, application server, database server, or any kind of server we wish.

Deploying a VM using the FROM GALLERY option

The steps to deploy a VM using the **FROM GALLERY** option are as follows:

1. From the Microsoft Azure portal, select **VIRTUAL MACHINES**; click on the **NEW** button and then click on **NEW** | **VIRTUAL MACHINE** | **FROM GALLERY**.
2. This will open up the image gallery with an extensive list of pre-configured virtual machines. Select any virtual machine you want to deploy. In my example, I decided to try the **Windows Server Technical Preview**, which is the Windows 10 Server beta, as shown in the following screenshot:

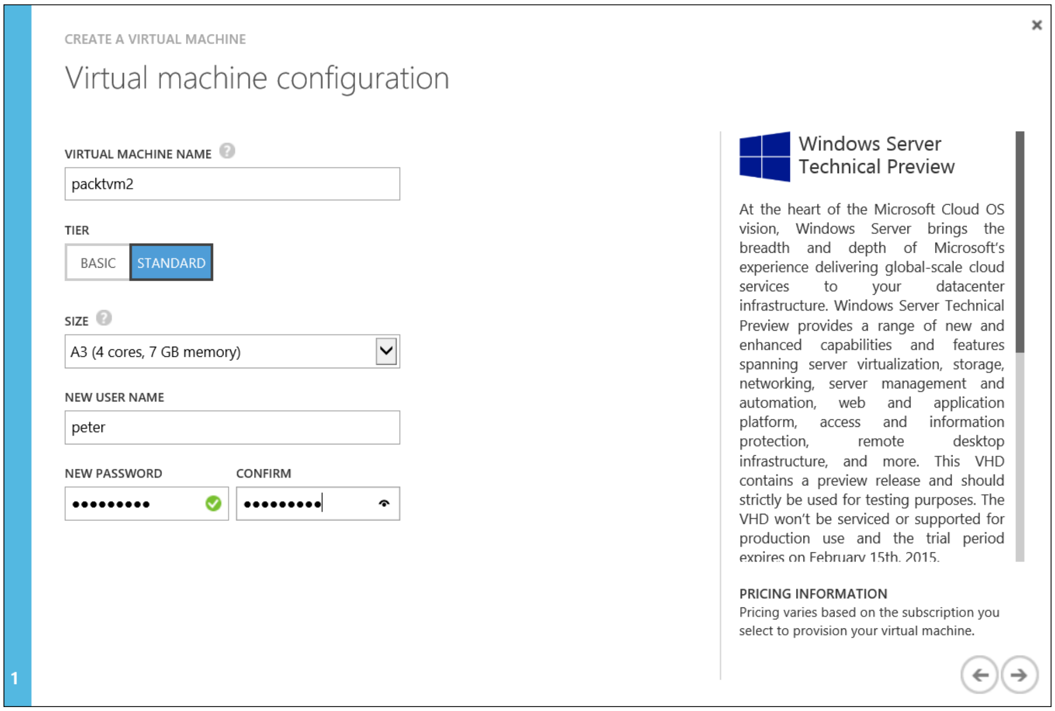

CREATE A VIRTUAL MACHINE
Choose an Image
ALL
MICROSOFT
WINDOWS SERVER
SHAREPOINT
SQL SERVER
BIZTALK SERVER
VISUAL STUDIO
DYNAMICS
UBUNTU
COREOS
CENTOS-BASED
SUSE
ORACLE
PUPPET LABS
MY IMAGES
MY DISKS
□ MSDN
FEATURED
Windows Server 2012 R2 Datacenter
Windows Server Essentials Experience
Windows Server 2012 R2
Windows Server 2012 Datacenter
Windows Server 2008 R2 SP1
Windows Server Technical Preview
SharePoint Server 2013 Trial
Windows Server 2012
SQL Server 2014 Enterprise Optimized for DataWarehousing Workloads
Windows Server 2012 R2
Windows Server Technical Preview

At the heart of the Microsoft Cloud OS vision, Windows Server brings the breadth and depth of Microsoft's experience delivering global-scale cloud services to your datacenter infrastructure. Windows Server Technical Preview provides a range of new and enhanced capabilities and features spanning server virtualization, storage, networking, server management and automation, web and application platform, access and information protection, remote desktop infrastructure, and more. This VHD contains a preview release and should strictly be used for testing purposes. The VHD won't be serviced or supported for production use and the trial period expires on February 15th, 2015.

PRICING INFORMATION

Pricing varies based on the subscription you select to provision your virtual machine.

3. Complete the remaining steps of the wizard by entering the required information in the appropriate fields to get the server virtual machine deployed, as shown in the following screenshot:



CREATE A VIRTUAL MACHINE

Virtual machine configuration

VIRTUAL MACHINE NAME: packtvm2

TIER: BASIC [✓ STANDARD]

SIZE: A3 (4 cores, 7 GB memory)

NEW USER NAME: peter

NEW PASSWORD: •••••••••

CONFIRM: •••••••••

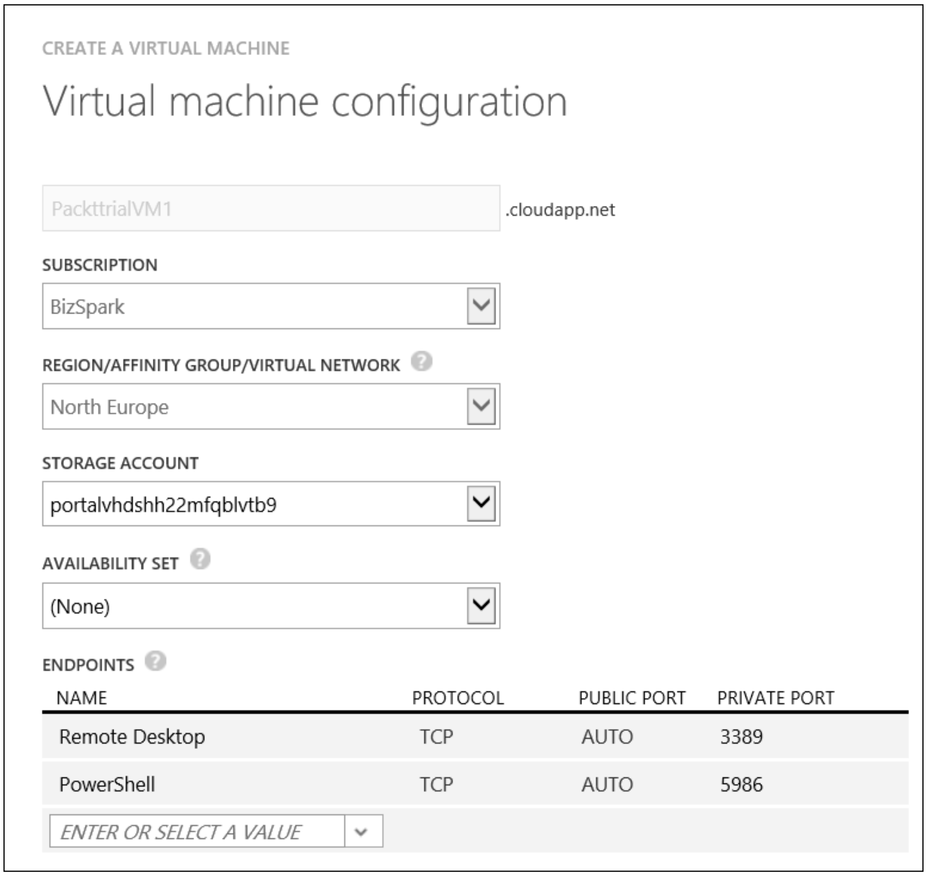
Windows Server Technical Preview

At the heart of the Microsoft Cloud OS vision, Windows Server brings the breadth and depth of Microsoft's experience delivering global-scale cloud services to your datacenter infrastructure. Windows Server Technical Preview provides a range of new and enhanced capabilities and features spanning server virtualization, storage, networking, server management and automation, web and application platform, access and information protection, remote desktop infrastructure, and more. This VHD contains a preview release and should strictly be used for testing purposes. The VHD won't be serviced or supported for production use and the trial period expires on February 15th, 2015.

PRICING INFORMATION

Pricing varies based on the subscription you select to provision your virtual machine.

4. This step is almost identical to the **QUICK CREATE** way, as you can see in the following screenshot:



CREATE A VIRTUAL MACHINE

Virtual machine configuration

PackttrialVM1 .cloudapp.net

SUBSCRIPTION

BizSpark

REGION/AFFINITY GROUP/VIRTUAL NETWORK

North Europe

STORAGE ACCOUNT

portalvhdshh22mfqblvtb9

AVAILABILITY SET

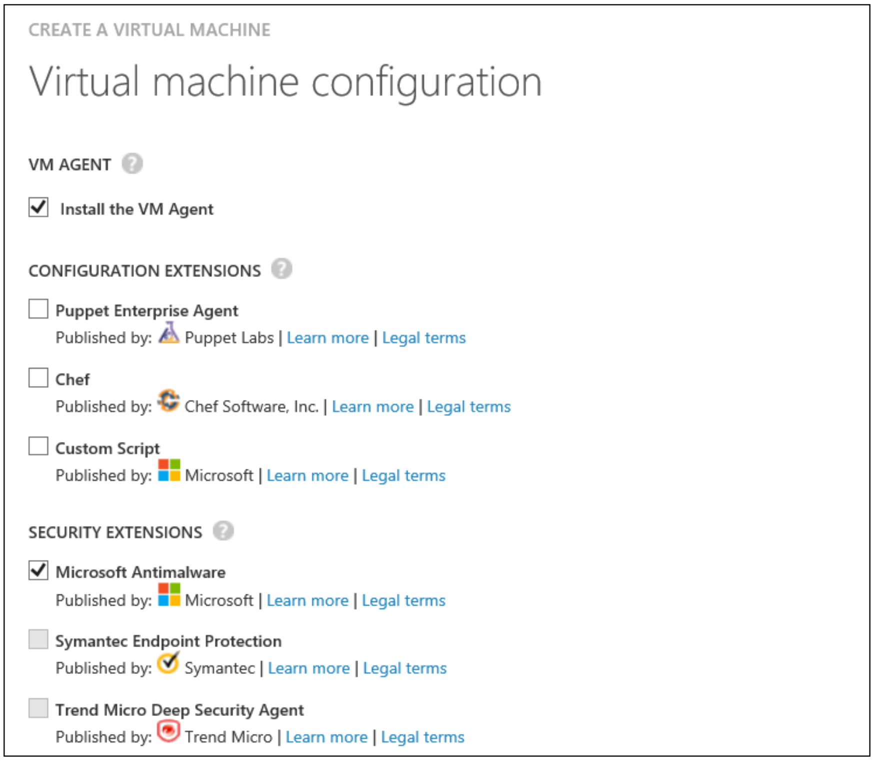
(None)

ENDPOINTS

NAME	PROTOCOL	PUBLIC PORT	PRIVATE PORT
Remote Desktop	TCP	AUTO	3389
PowerShell	TCP	AUTO	5986
ENTER OR SELECT A VALUE			

The important differences between the **QUICK CREATE** and **FROM GALLERY** options here are the options to customize the cloud service, what storage account to be used, if we want to make this VM part of an availability set, and performing manual configuration of what ports to use for remote desktop, PowerShell, and possible other services you want to connect to from the outside.

5. The following screenshot shows the final step in creating a new VM:



CREATE A VIRTUAL MACHINE

Virtual machine configuration

VM AGENT

☑ Install the VM Agent

CONFIGURATION EXTENSIONS

☐ Puppet Enterprise Agent
Published by: Puppet Labs | Learn more | Legal terms

☐ Chef
Published by: Chef Software, Inc. | Learn more | Legal terms

☐ Custom Script
Published by: Microsoft | Learn more | Legal terms

SECURITY EXTENSIONS

☑ Microsoft Antimalware
Published by: Microsoft | Learn more | Legal terms

☐ Symantec Endpoint Protection
Published by: Symantec | Learn more | Legal terms

☐ Trend Micro Deep Security Agent
Published by: Trend Micro | Learn more | Legal terms

This completes the creation of a new VM by using one of the many pre-configured templates we have available.

Uploading an on-premises custom server image to Azure

In both the previous sections, we showed you how easy it is to deploy a new VM from scratch or from the template gallery. But maybe your organization is using a custom image of the Windows Server operating system, with (for example) already pre-configured tools installed, or differing hardware specifics.

So in this next part, we will show you how to create and upload a custom VHD image to Azure.

What do you need to upload and run a custom VHD image to Azure?

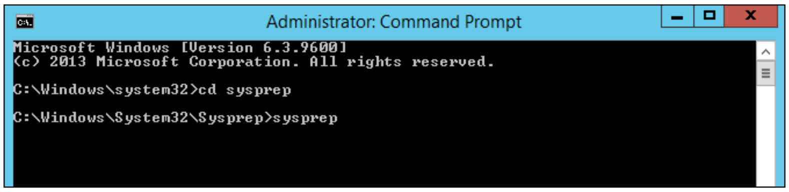
In order to upload and run a custom VHD image to Azure, you need to have the following available:

- An active Azure subscription
- Microsoft Azure PowerShell plug-in
- Your VM must run an Azure-supported operating system

Preparing your VM image

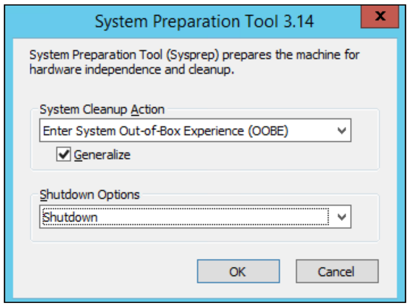
Before you can use the custom VM image in Azure as a deployment template, it needs to be "generalized". This is done by using the `sysprep` command (which is exactly the same process for creating an on-premises VM template on Hyper-V and even other hypervisors). The steps are as follows:

1. Log on to your on-premises virtual machine that you want to use as an image template, using administrative user rights.
2. Open the command prompt with administrative rights.
3. Change the directory to `%windir%\system32\sysprep`.
4. Run `sysprep.exe`, as shown in the following screenshot:



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd sysprep

C:\Windows\System32\Sysprep>sysprep
```

5. This will launch the **System Preparation Tool** program. Select the **Enter System Out-of-Box Experience (OOBE)** option, check the **Generalize** option, and then select the **Shutdown** option, as shown in the following screenshot:



System Preparation Tool (Sysprep) prepares the machine for hardware independence and cleanup.

System Cleanup Action: Enter System Out-of-Box Experience (OOBE)

☑ Generalize

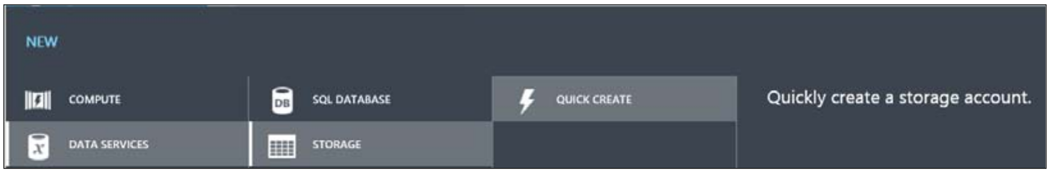
Shutdown Options: Shutdown

The **System Preparation Tool** will run for a few minutes, after which the VM will be shutdown automatically. This completes the `sysprep` step.

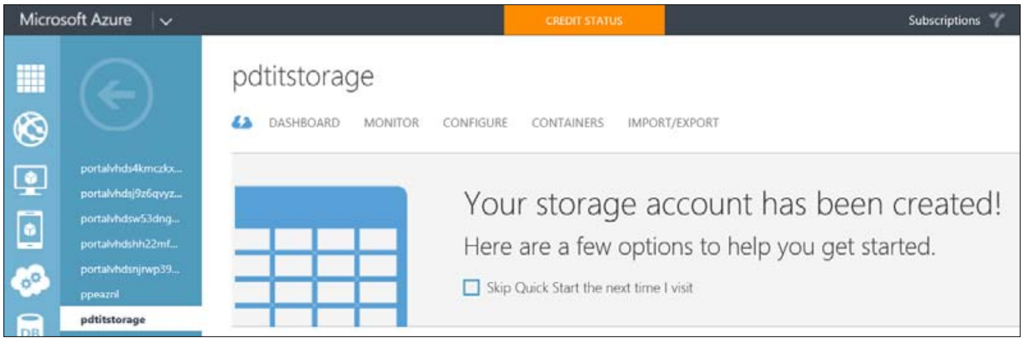
Creating an Azure storage location within your subscription

The steps are as follows:

1. Sign in to the Microsoft Azure portal.
2. Click on **NEW** | **DATA SERVICES** | **STORAGE** | **QUICK CREATE**, as shown in the following screenshot:



3. Complete the required fields. The URL is key here; choose something that is easy to remember (like in my example `pdtitstorage`) as a name. The other fields should be equally easy to fill out.
4. Once the storage has been created, select it from the storage list. This will bring you to the details pane for this specific storage component, as shown in the following screenshot:



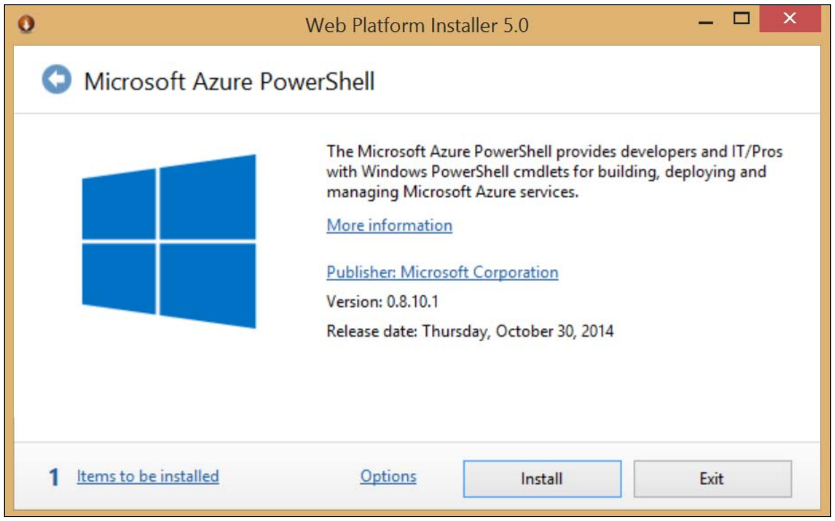
5. From here, select **Containers** | **Create Container**; give it a descriptive name (for example, `vhdstorage`) and leave **Private** as the option.

This completes the Azure storage and storage container creation step.

Connecting to Microsoft Azure using Azure PowerShell

Although the Microsoft Azure portal is very complete, certain tasks—such as uploading VMs—require Azure PowerShell. This part of the chapter will explain how to connect to Microsoft Azure using Azure PowerShell. The steps are as follows:

1. Download the Microsoft Web Platform Installer from `http://go.microsoft.com/fwlink/p/?linkid=320376&clcid=0x409` and run the setup:


Web Platform Installer 5.0
Microsoft Azure PowerShell
More information
Publisher: Microsoft Corporation
1 Items to be installed
Options
Install
Exit

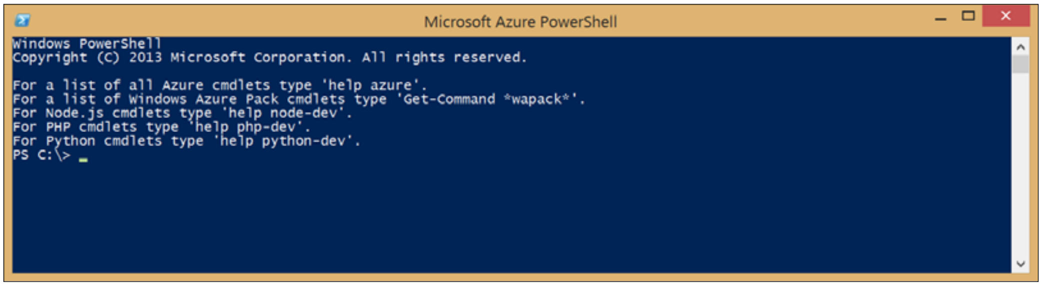
The Microsoft Azure PowerShell provides developers and IT/Pros with Windows PowerShell cmdlets for building, deploying and managing Microsoft Azure services.

Version: 0.8.10.1

Release date: Thursday, October 30, 2014

After clicking on **Install**, the following steps should be executed automatically and finishing successfully with the installation of the Microsoft Azure for PowerShell plugin.

2. Launch Microsoft Azure PowerShell from your Start screen or Start menu:

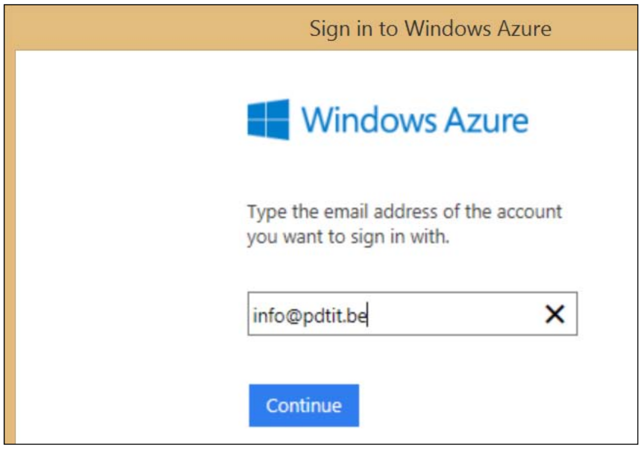


```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

For a list of all Azure cmdlets type 'help azure'.
For a list of Windows Azure Pack cmdlets type 'Get-Command *wapack*'.
For Node.js cmdlets type 'help node-dev'.
For PHP cmdlets type 'help php-dev'.
For Python cmdlets type 'help python-dev'.
PS C:\> _
```

3. Type the following command:

   ```
   Add-AzureAccount
   ```

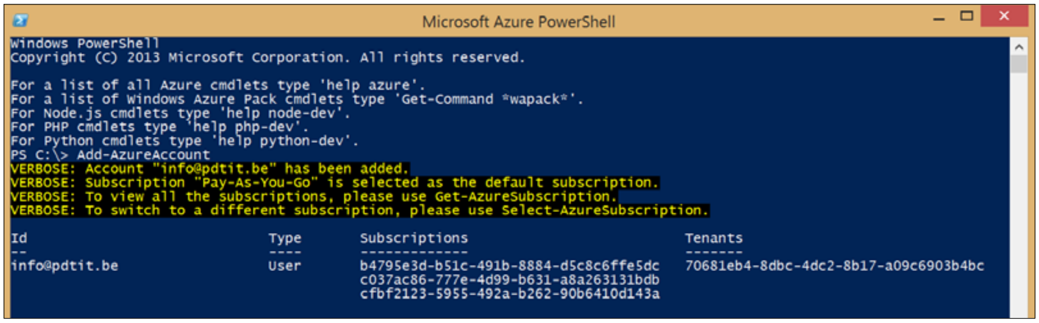
4. This will launch the Azure authentication pop-up window. Enter your Azure subscription credentials (e-mail address and password):



Type the email address of the account you want to sign in with.

info@pdtit.be

5. After successfully logging on to your Microsoft Azure subscription, the details of the active subscriptions will be visible in Azure PowerShell, as shown in the following screenshot:



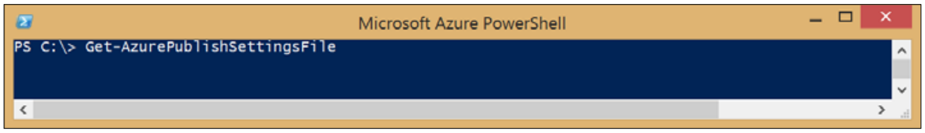
```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

For a list of all Azure cmdlets type 'help azure'.
For a list of Windows Azure Pack cmdlets type 'Get-Command *wapack*'.
For Node.js cmdlets type 'help node-dev'.
For PHP cmdlets type 'help php-dev'.
For Python cmdlets type 'help python-dev'.
PS C:\> Add-AzureAccount
VERBOSE: Account "info@pdtit.be" has been added.
VERBOSE: Subscription "Pay-As-You-Go" is selected as the default subscription.
VERBOSE: To view all the subscriptions, please use Get-AzureSubscription.
VERBOSE: To switch to a different subscription, please use Select-AzureSubscription.

Id                         Type      Subscriptions                            Tenants
--                         ----      -------------                            -------
info@pdtit.be              User      b4795e3d-b51c-491b-8884-d5c8c6ffe5dc     70681eb4-8dbc-4dc2-8b17-a09c6903b4bc
                                     c037ac86-777e-4d99-b631-a8a263131bdb
                                     cfbf2123-5955-492a-b262-90b6410d143a
```

In order to establish a secure connection every time you connect from your admin station to Azure by using Azure PowerShell, you have to present your subscription information. This is possible by using Azure AD credentials or by using a certificate. Since this is a standalone setup, we recommend using the certificate method.

6. From Azure PowerShell, run the following command:

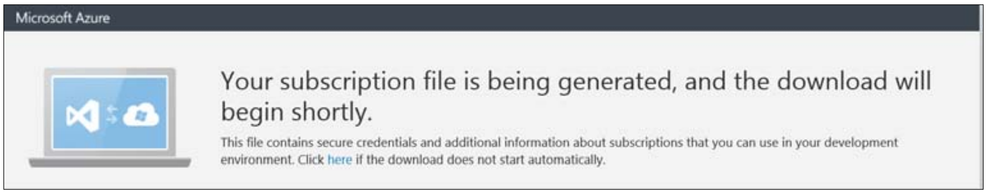
   `Get-AzurePublishSettingsFile`

   This is shown in the following screenshot:



```
PS C:\> Get-AzurePublishSettingsFile
```

   This will open up a new web page, asking you to download the certificate file. Save this file to your local hard drive.



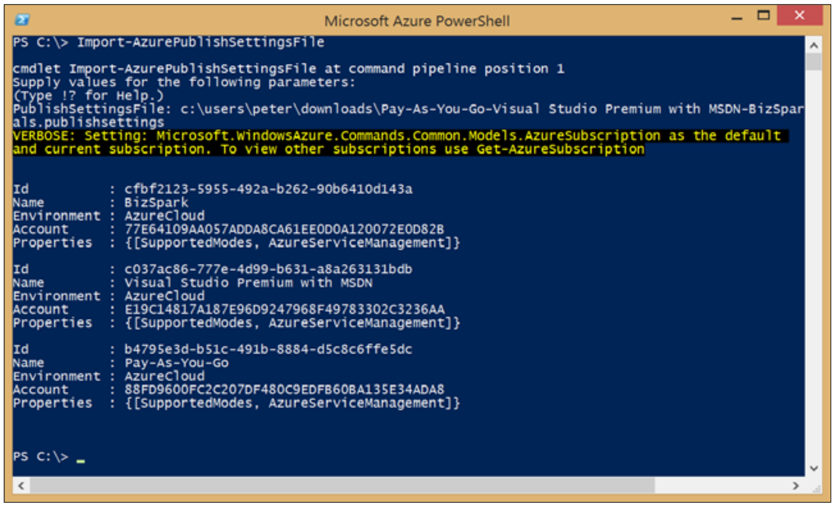
Your subscription file is being generated, and the download will begin shortly.

This file contains secure credentials and additional information about subscriptions that you can use in your development environment. Click here if the download does not start automatically.

7. From Azure PowerShell, run the following command:

   `Import-AzurePublishSettingsfile`

   This is shown in the following screenshot:



```
PS C:\> Import-AzurePublishSettingsFile

cmdlet Import-AzurePublishSettingsFile at command pipeline position 1
Supply values for the following parameters:
(Type !? for Help.)
PublishSettingsFile: c:\users\peter\downloads\Pay-As-You-Go-Visual Studio Premium with MSDN-BizSpar
als.publishsettings
VERBOSE: Setting: Microsoft.WindowsAzure.Commands.Common.Models.AzureSubscription as the default
and current subscription. To view other subscriptions use Get-AzureSubscription


Id          : cfbf2123-5955-492a-b262-90b6410d143a
Name        : BizSpark
Environment : AzureCloud
Account     : 77E64109AA057ADDA8CA61EE0D0A120072E0D82B
Properties  : {[SupportedModes, AzureServiceManagement]}

Id          : c037ac86-777e-4d99-b631-a8a263131bdb
Name        : Visual Studio Premium with MSDN
Environment : AzureCloud
Account     : E19C14817A187E96D9247968F49783302C3236AA
Properties  : {[SupportedModes, AzureServiceManagement]}

Id          : b4795e3d-b51c-491b-8884-d5c8c6ffe5dc
Name        : Pay-As-You-Go
Environment : AzureCloud
Account     : 88FD9600FC2C207DF480C9EDFB60BA135E34ADA8
Properties  : {[SupportedModes, AzureServiceManagement]}



PS C:\> _
```

This completes the configuration of the Azure PowerShell and subscription connection.

Uploading the VHD to the Azure storage container

In this last step, we will execute the command to actually upload the sysprepped VM image we created earlier to the Azure storage container we created.

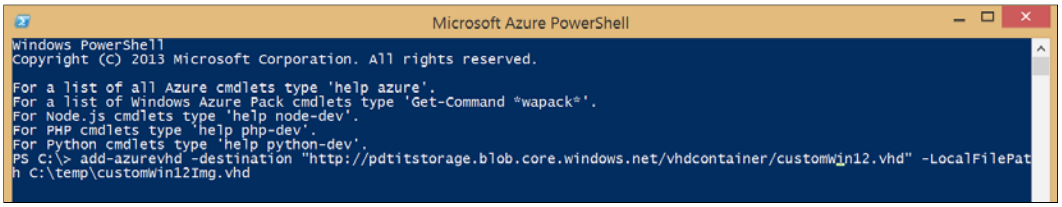
From Azure PowerShell, initiate the following cmdlet:

```
Add-AzureVHD -Destination "https://StorageURL/Container/VHDName.vhd" -
LocalFilePath "local location of the VHD-file.vhd"
```

Or more concretely, as in my setup, the command will be the following:

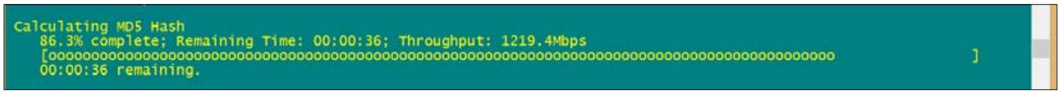
```
Add-AzureVHD -Destination "https://pdtitstorage.blob.core.windows.net/
vhdcontainer/customWin12R2.vhd" -localfilepath "c:\temp\customWin12R2Img.
vhd"
```

This is shown in the following screenshot:



```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

For a list of all Azure cmdlets type 'help azure'.
For a list of Windows Azure Pack cmdlets type 'Get-Command *wapack*'.
For Node.js cmdlets type 'help node-dev'.
For PHP cmdlets type 'help php-dev'.
For Python cmdlets type 'help python-dev'.
PS C:\> add-azurevhd -destination "http://pdtitstorage.blob.core.windows.net/vhdcontainer/customWin12.vhd" -LocalFilePat
h C:\temp\customWin12Img.vhd
```

Once the PowerShell cmdlet is fired, a progress bar is displayed showing the progress of the VHD upload:



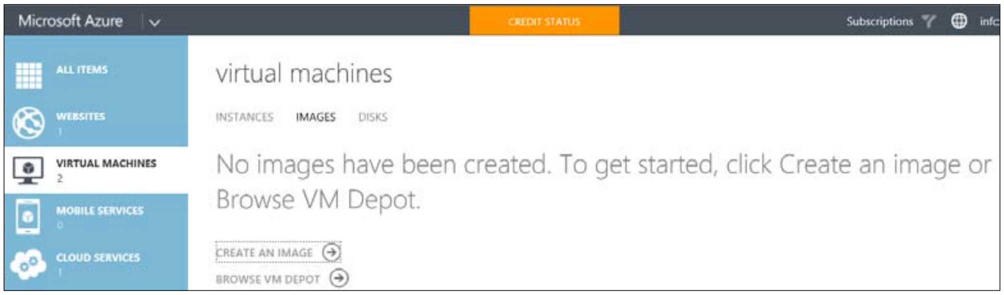
```
Calculating MD5 Hash
   86.3% complete; Remaining Time: 00:00:36; Throughput: 1219.4Mbps
   [ooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo                 ]
   00:00:36 remaining.
```

Depending on your Internet connection speed, this upload process might take several hours (in my case, using a T1 cable connection, uploading an 8GB VHD file took close to 2 hours). Do not interrupt or close the Azure PowerShell process, as this will kill your upload.

Once the upload is complete, add the uploaded VHD image to the list of images in Azure, by using the following procedure:

1. From the Azure Management portal, click on **VIRTUAL MACHINES**.
2. Click on **IMAGES** from the ribbon menu.

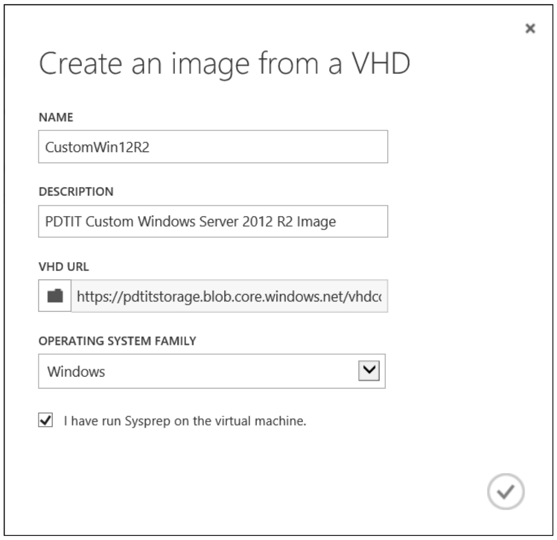
3. Click on **CREATE AN IMAGE**, as shown in the following screenshot:



virtual machines

No images have been created. To get started, click Create an image or Browse VM Depot.

4. After clicking on **CREATE AN IMAGE**, complete the fields with the required information. Note that you can browse to the exact storage container location of the VHDs we used in the previous steps, as shown in the following screenshot:



Create an image from a VHD

NAME: CustomWin12R2

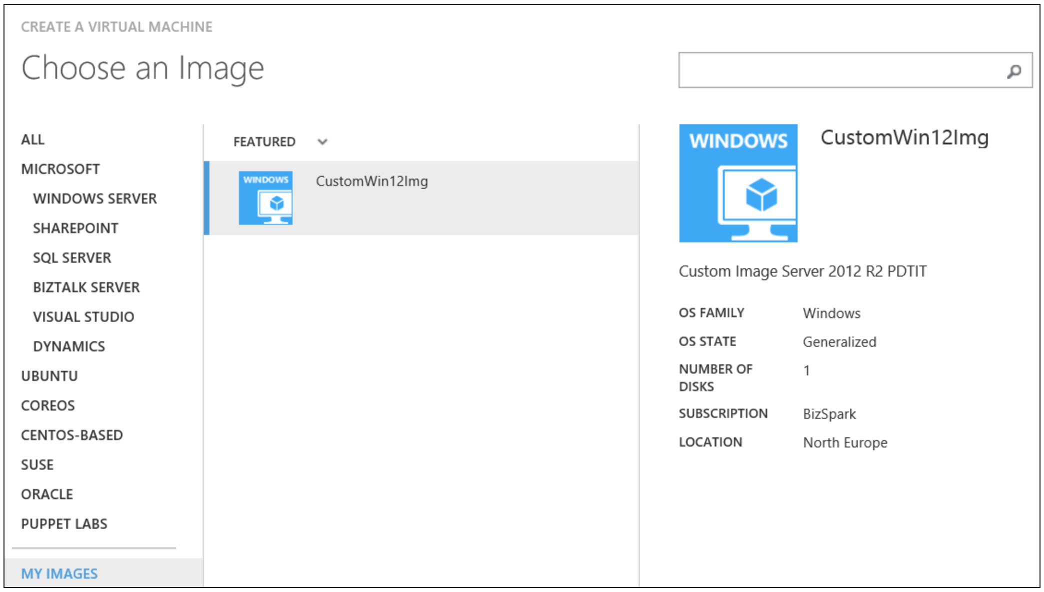
DESCRIPTION: PDTIT Custom Windows Server 2012 R2 Image

VHD URL: https://pdtitstorage.blob.core.windows.net/vhdc

OPERATING SYSTEM FAMILY: Windows

☑ I have run Sysprep on the virtual machine.

5. If you now go back to the Azure Management portal and then click on **VIRTUAL MACHINES | NEW | FROM GALLERY**, you can notice the option in the left bottom corner **MY IMAGES**, from where you can select the **CustomWin12Img.vhd** image we uploaded before:



CREATE A VIRTUAL MACHINE

Choose an Image

CustomWin12Img

Custom Image Server 2012 R2 PDTIT

OS FAMILY	Windows
OS STATE	Generalized
NUMBER OF DISKS	1
SUBSCRIPTION	BizSpark
LOCATION	North Europe

This concludes the steps that are required to upload your custom VHD image to Microsoft Azure and deploy new VMs from this image.

In all of the described actions around managing virtual machines so far, everything is happening inside Azure itself.

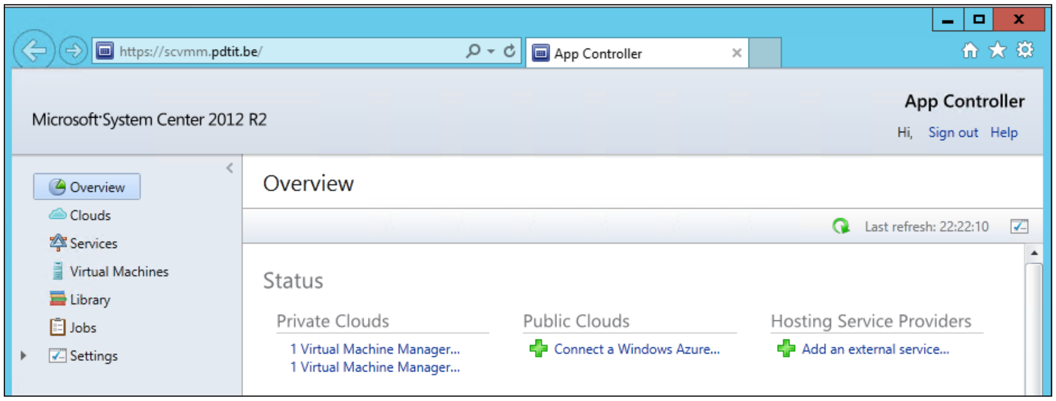
In the last section of this chapter, we will guide you through a more hybrid-based approach, in which you can manage both on-premises virtual machines and your Azure virtual machines from one single tool, System Center App Controller, which we have already used for our on-premises data center in the previous chapter.

Efficiently managing your hybrid cloud data center infrastructure using App Controller

Chapter 7, Operating Your Private Cloud Using SCVMM and App Controller, was dedicated to App Controller. But we only touched the on-premises infrastructure management in that chapter.

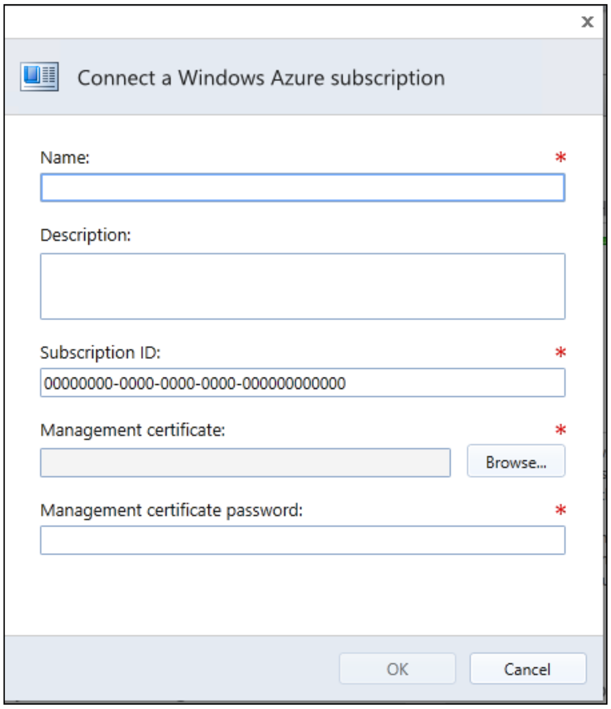
In this topic, we will show you how you can extend the same App Controller environment you deployed in the previous chapter for allowing true integration with your Azure environment. The steps are as follows:

1. Log on to the App Controller VM you created earlier. From your browser, connect to the App Controller management portal:



2. From the App Controller dashboard, select **Overview**; from the middle pane, click on the **Connect a Windows Azure subscription** option.

3. You are presented with the following parameters window:



Connect a Windows Azure subscription

Name: *

Description:

Subscription ID: *
00000000-0000-0000-0000-000000000000

Management certificate: *

Management certificate password: *

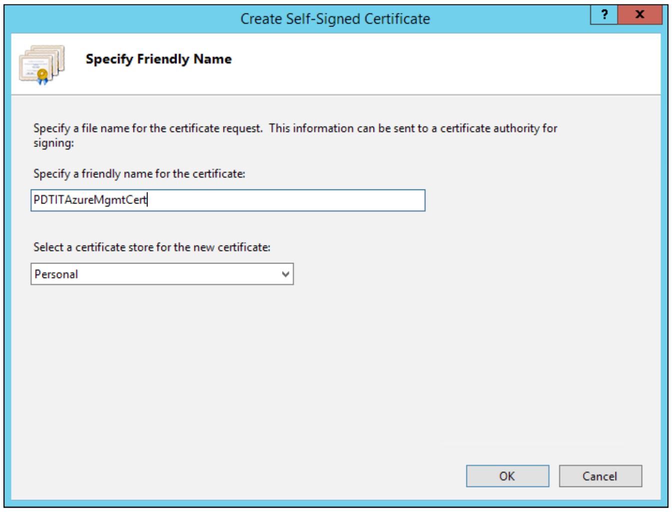
In order to be able to complete the required parameters, there are some additional steps required, as follows:

- Getting the Azure subscription ID information
- Creating an SSL certificate and uploading this to Azure, and then nominating it as a management certificate

The first step should already be a bit familiar, since we used it only a few pages back in this chapter: by using the Azure PowerShell cmdlet `get-AzureSubscription`, the Azure subscription ID is listed.

To handle the second step, having an SSL certificate available and uploading this to Azure, you have a few different ways to achieve this. You can create your own certificate from an IIS server or you can buy a public SSL certificate. As it sounds easiest to do so for the moment, let's go for the method where we create a self-signed SSL certificate from the IIS admin console on our App Controller server. The steps are as follows:

1. Open up the IIS admin console on your App Controller server (or any other IIS server you have available).
2. From within the IIS admin console, select **Certificates** in the middle pane.
3. From the **Actions** pane on the right, select the **Create Self-Signed Certificate** option:



Create Self-Signed Certificate

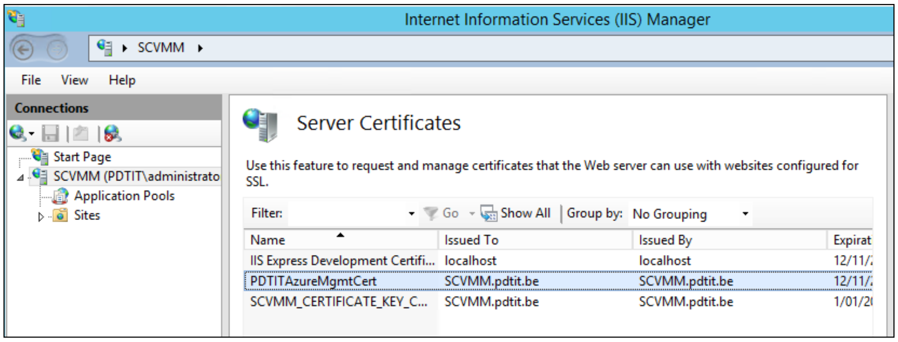
Specify Friendly Name

Specify a file name for the certificate request. This information can be sent to a certificate authority for signing:

Specify a friendly name for the certificate: PDTITAzureMgmtCert

Select a certificate store for the new certificate: Personal

4. Complete the necessary details (mention a friendly name and select **Personal** as the certificate store).



Internet Information Services (IIS) Manager

Server Certificates

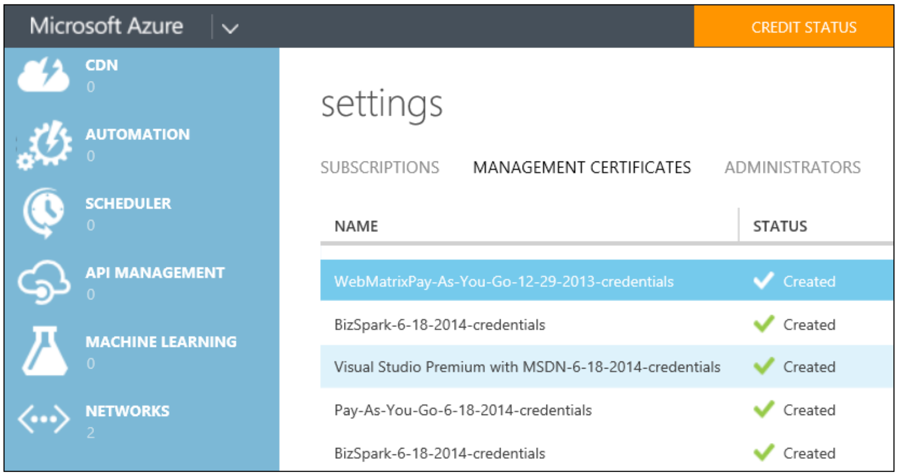
Use this feature to request and manage certificates that the Web server can use with websites configured for SSL.

Name	Issued To	Issued By	Expirat
IIS Express Development Certifi...	localhost	localhost	12/11/:
PDTITAzureMgmtCert	SCVMM.pdtit.be	SCVMM.pdtit.be	12/11/:
SCVMM_CERTIFICATE_KEY_C...	SCVMM.pdtit.be	SCVMM.pdtit.be	1/01/2(

5. The self-signed certificate will be created automatically and is visible in the list of SSL certificates.
6. Double-click on the SSL certificate to open up the details. From here, export the certificate to a PFX file, including the private key and a password, as well as a CER file (we need both certificate formats later on, so this step is really crucial!).

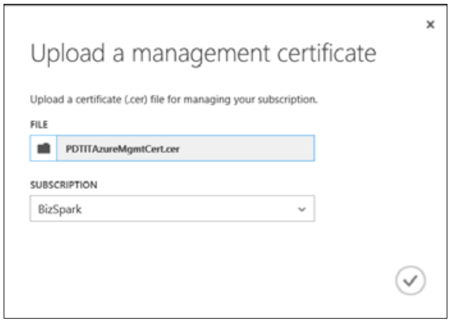
   The last step in this preparation work is uploading the SSL certificate CER file to Azure.

7. Log on to the Microsoft Azure portal; from the blue menu ribbon to your left, scroll all the way down and select **SETTINGS**.
8. From the **settings** page, select **MANAGEMENT CERTIFICATES**:



NAME	STATUS
WebMatrixPay-As-You-Go-12-29-2013-credentials	✓ Created
BizSpark-6-18-2014-credentials	✓ Created
Visual Studio Premium with MSDN-6-18-2014-credentials	✓ Created
Pay-As-You-Go-6-18-2014-credentials	✓ Created
BizSpark-6-18-2014-credentials	✓ Created

9. At the bottom of the page, select **Upload**.



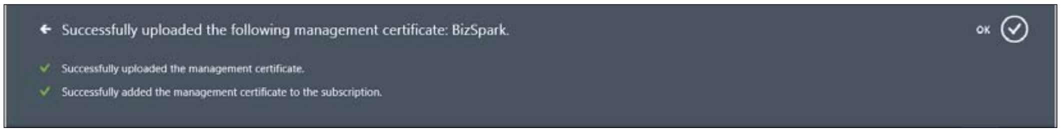
Upload a management certificate

Upload a certificate (.cer) file for managing your subscription.

FILE: PDTITAzureMgmtCert.cer

SUBSCRIPTION: BizSpark

10. Browse to the SSL Certificate CER file, and also select the subscription you want to link this certificate to. The certificate will then be uploaded:



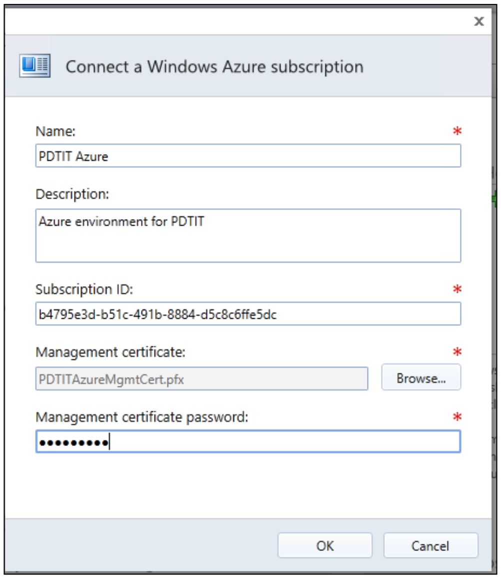
Successfully uploaded the following management certificate: BizSpark.

✓ Successfully uploaded the management certificate.

✓ Successfully added the management certificate to the subscription.

Now that all preparation work is done, we can go back to System Center App Controller, and complete the Azure Subscription connection, using the following steps:

1. Go back to App Controller and then click on **Connect Azure Subscription**, and complete all required fields, which should be pretty easy to do now. It is important to note that you need to select the SSL certificate PFX file in this case:



Connect a Windows Azure subscription

Name: * PDTIT Azure

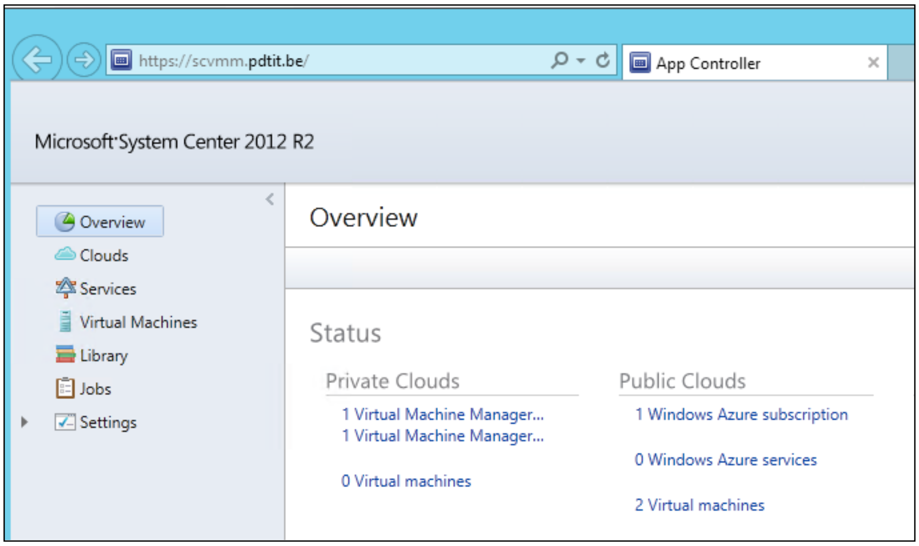
Description: Azure environment for PDTIT

Subscription ID: * b4795e3d-b51c-491b-8884-d5c8c6ffe5dc

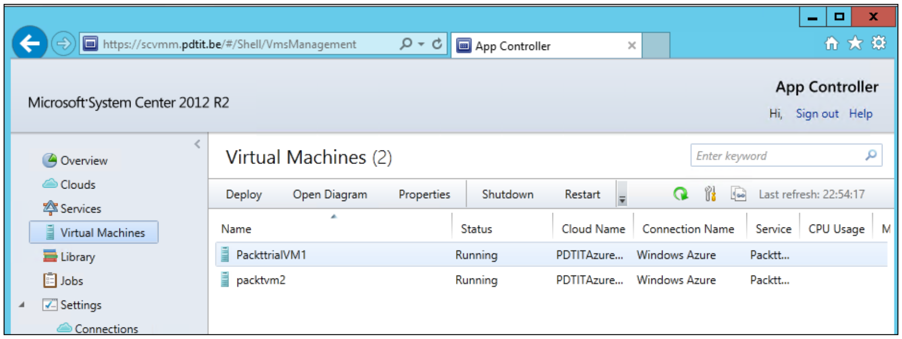
Management certificate: * PDTITAzureMgmtCert.pfx

Management certificate password: * •••••••••

2. After all the information is entered correctly, your Azure subscription should be active, showing you the real-time state of your subscription, number of virtual machines, and so on:



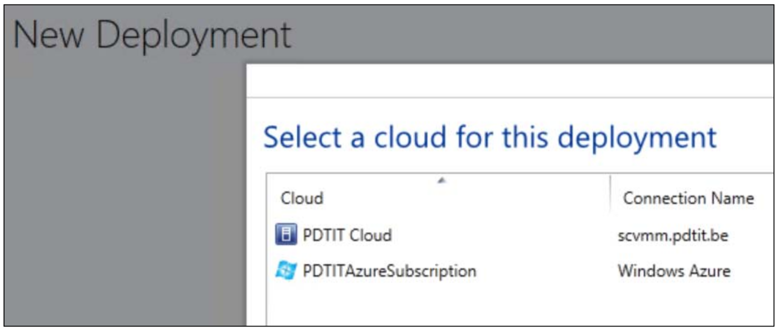
3. After selecting **Virtual Machines** from within the topic, or even from the menu on the left, the virtual machines we currently have present in Microsoft Azure should be listed there:



Name	Status	Cloud Name	Connection Name	Service	CPU Usage	M
PackttrialVM1	Running	PDTITAzure...	Windows Azure	Packtt...		
packtvm2	Running	PDTITAzure...	Windows Azure	Packtt...		

In the last part of this chapter, I will show you again how easy it is to create a new virtual machine from App Controller, but this time by deploying it in Microsoft Azure by using the virtual machine template `CustomWin12Img.VHD` we created and uploaded earlier in this chapter. The steps are as follows:

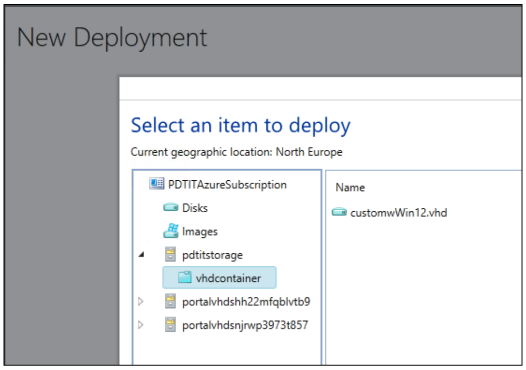
1. From within the App Controller dashboard, select **Virtual Machines** and then from the middle pane, select **Deploy**. This will launch the **New Deployment** wizard. A new item **Cloud** will show up. Click on **Configure**:



Select a cloud for this deployment

Cloud	Connection Name
PDTIT Cloud	scvmm.pdtit.be
PDTITAzureSubscription	Windows Azure

2. From the **Select a cloud for this deployment** section, select your Azure subscription.
3. The section will get extended with a **Deployment Type** option. Click on **Select an item...**:

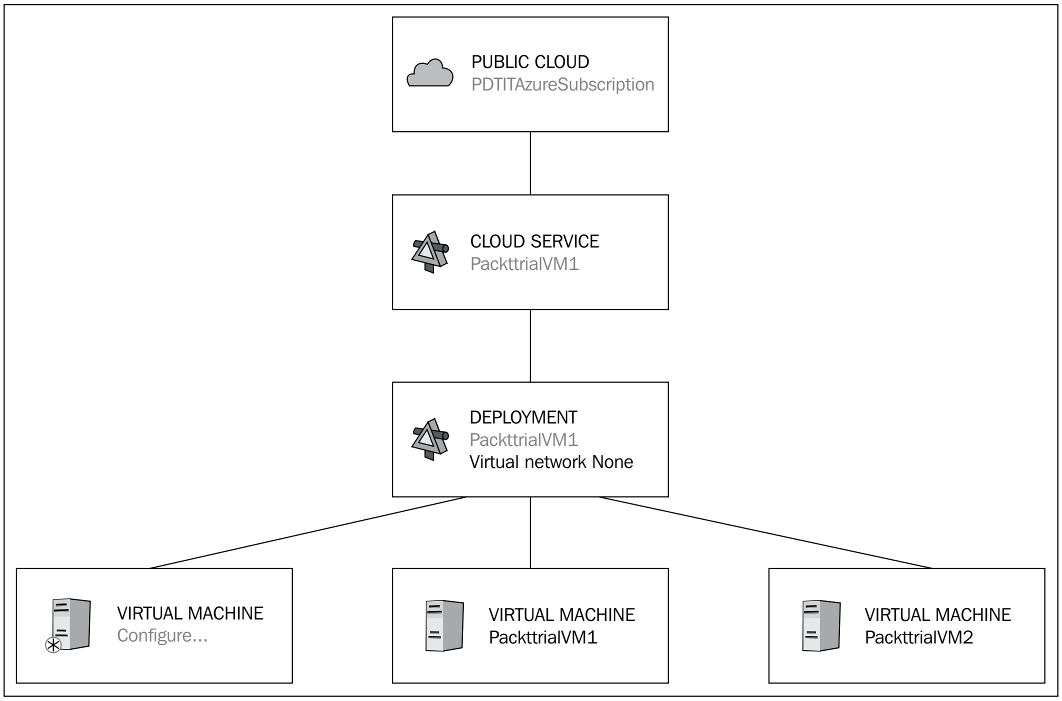


Select an item to deploy

Current geographic location: North Europe

Name
customwWin12.vhd

4. Browse to the Azure subscription/cloud storage container we created earlier, and select the **customWin12.vhd** image from there.
5. The diagram will be extended with a **Virtual Machine** option. Again, click on **Configure...** (note that the other virtual machines that are already deployed in Azure are also visible here):



6. Continue the next step of the deployment wizard, by completing the machine-specific required fields; for example, specify the affinity group (if required), create a new virtual disk for this VM, and such similar fields.
7. Once all the required parameters and steps are completed, click on **Deploy**, and see the new VM being created out of App Controller but running in Microsoft Azure. Great job, isn't it!

This concludes the walk-through on how to integrate Microsoft Azure into your on-premises data center environment, by using App Controller.

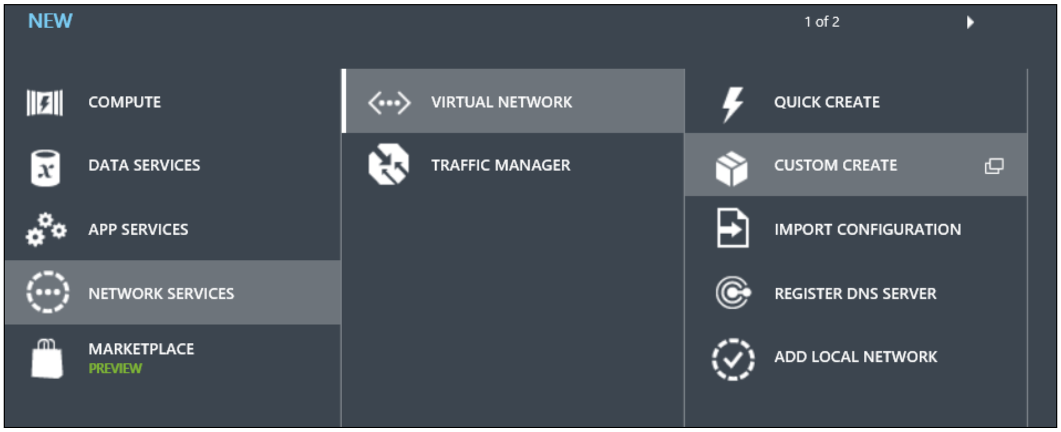
Data center integration using Microsoft Azure site-to-site VPN

From the previous topics, it should be clear that an integration can be set up between your on-premises data center and Microsoft Azure. But, from what we have showed you so far, the integration was mainly on the "management" layer. One can manage Microsoft Azure by using Azure PowerShell from within the on-premises network, or by using an integration out of App Controller.

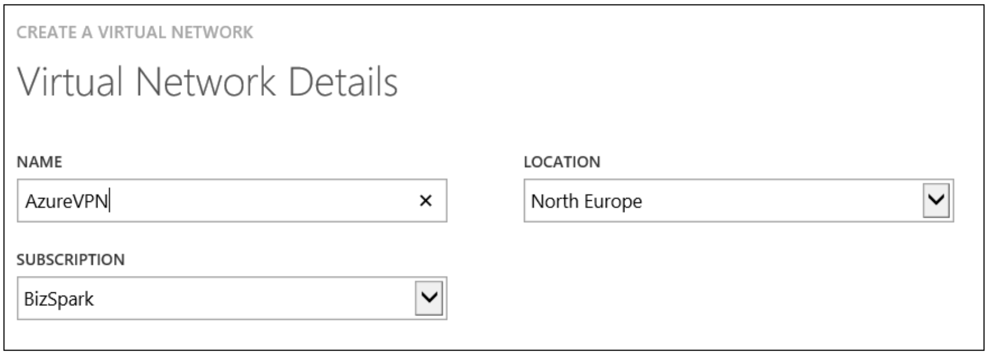
But wouldn't it be cool if we could actually "glue" the two data centers together on the network layer? Where both the on-premises and Microsoft Azure data center see each other as trusted networks, allowing for the sending of data between both networking connections? The good news is that's possible!

From within the Microsoft Azure platform, we have the possibility to establish a site-to-site VPN between both data centers. This works in the exact same way as you might already be using this mechanism today between different data center sites in different regions, or between your on-premises data center and a hosting provider's environment. The steps to configure this are as follows:

1. From the Microsoft Azure portal, select **NETWORKS** and then select **NEW**.



2. Choose **NETWORK SERVICES | VIRTUAL NETWORK | CUSTOM CREATE**. This will launch the **CREATE A VIRTUAL NETWORK** wizard:



CREATE A VIRTUAL NETWORK

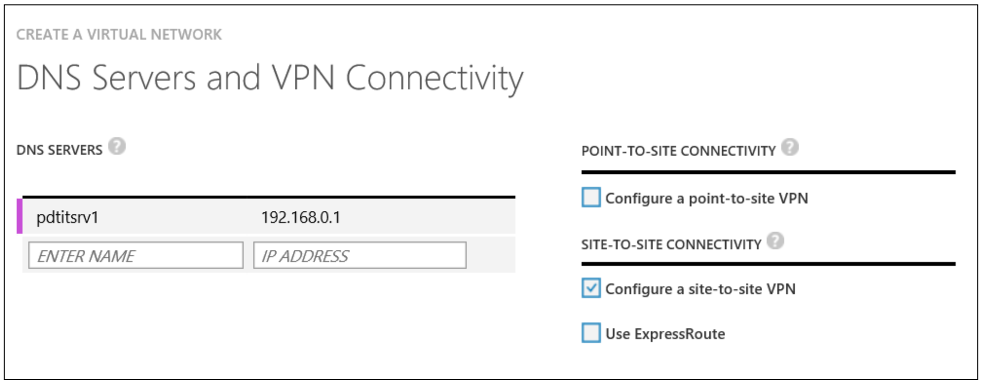
Virtual Network Details

NAME: AzureVPN

LOCATION: North Europe

SUBSCRIPTION: BizSpark

3. In the first step, provide a name for the connection, for example `AzureVPN`.



CREATE A VIRTUAL NETWORK

DNS Servers and VPN Connectivity

DNS SERVERS

pdtitsrv1	192.168.0.1
ENTER NAME	IP ADDRESS

POINT-TO-SITE CONNECTIVITY

☐ Configure a point-to-site VPN

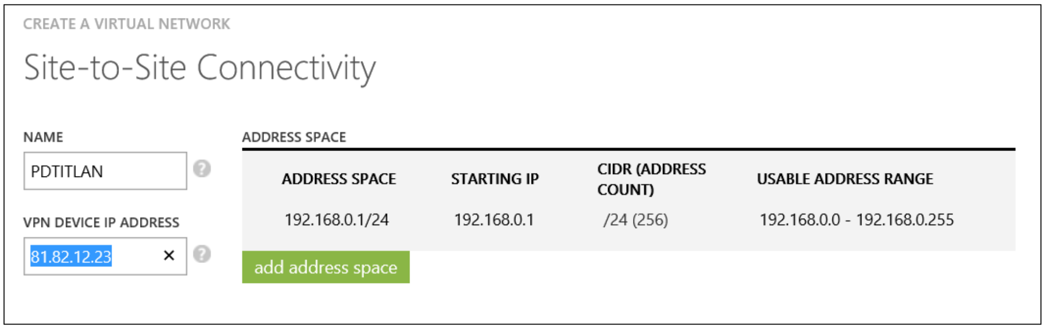
SITE-TO-SITE CONNECTIVITY

☑ Configure a site-to-site VPN

☐ Use ExpressRoute

4. In the next step, complete the DNS fields; this should refer to a DNS server that can resolve our on-premises network domain name. In my case, I added the server name and IP address of my on-premises domain controller/DNS server.

5. In the same step, check the **Configure a site-to-site VPN** option and click on **Next**.



CREATE A VIRTUAL NETWORK

Site-to-Site Connectivity

NAME

PDTITLAN

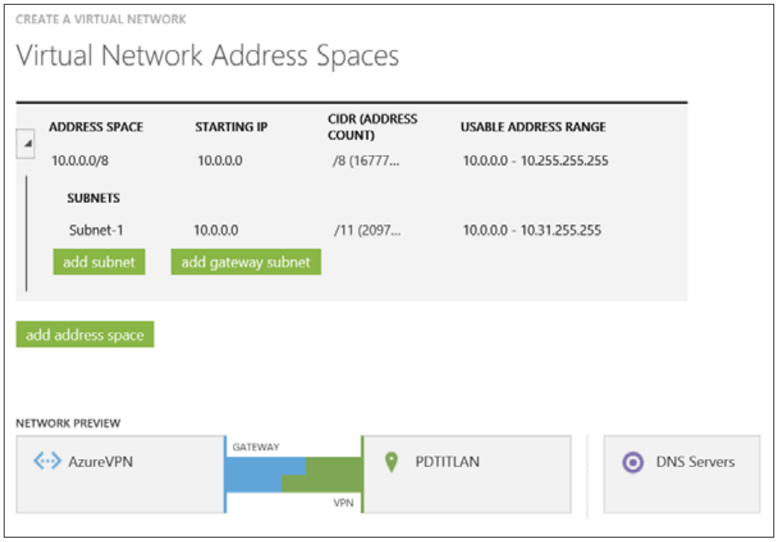
VPN DEVICE IP ADDRESS

81.82.12.23

ADDRESS SPACE

ADDRESS SPACE	STARTING IP	CIDR (ADDRESS COUNT)	USABLE ADDRESS RANGE
192.168.0.1/24	192.168.0.1	/24 (256)	192.168.0.0 - 192.168.0.255

6. In the **Site-to-Site Connectivity** step, provide the details of your LAN and WAN IP interfaces from your network (note that I'm using fake IP addresses to clearly show you the difference between internal and public IP).



CREATE A VIRTUAL NETWORK

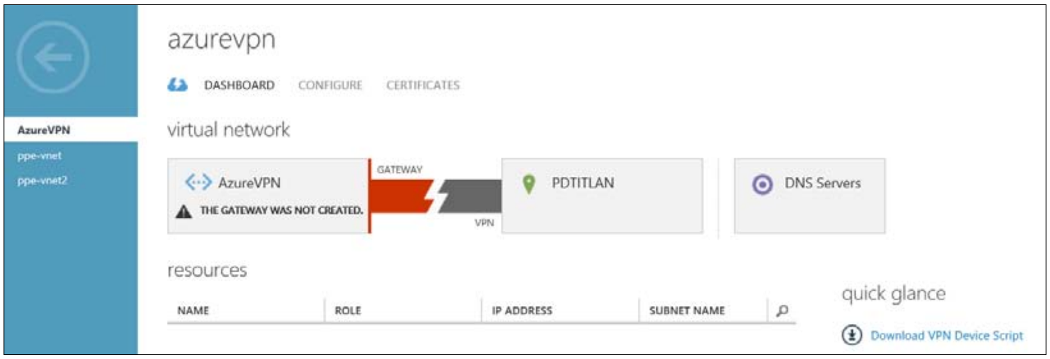
Virtual Network Address Spaces

ADDRESS SPACE	STARTING IP	CIDR (ADDRESS COUNT)	USABLE ADDRESS RANGE
10.0.0.0/8	10.0.0.0	/8 (16777...	10.0.0.0 - 10.255.255.255
SUBNETS			
Subnet-1	10.0.0.0	/11 (2097...	10.0.0.0 - 10.31.255.255

NETWORK PREVIEW

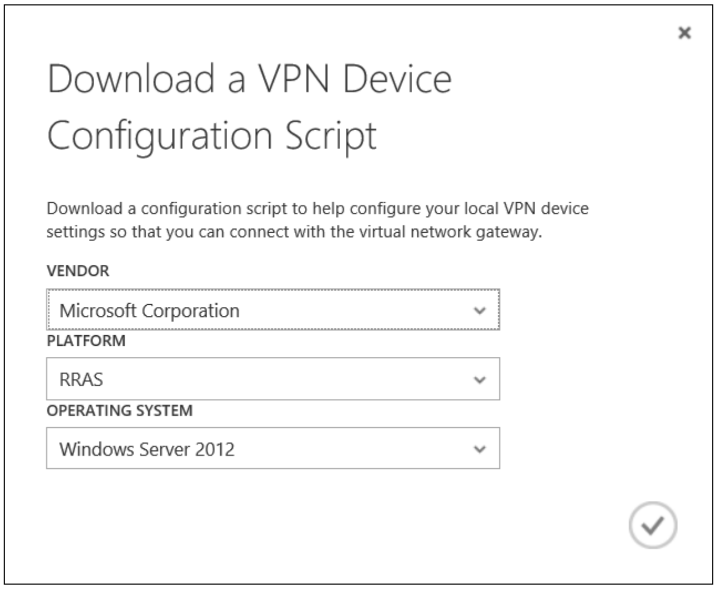
7. In the **Virtual Network Address Spaces** step, verify the IP subnets that are already there and then click on **add gateway subnet**; this is the DMZ subnet between your public interface and the LAN interface. In my demo setup, I accepted the default 10.32.0.0 IP subnet.

8. After completing all steps in the wizard, the VPN network object should be created successfully, and be listed in the **NETWORKS** section of the portal:



THE GATEWAY WAS NOT CREATED.

9. Select your VPN connection and double-click on it; this will bring you to the details window. Select **DASHBOARD** from the menu. Your screen should look like the preceding screenshot.
10. Click on **Download VPN Device Script**.



Download a VPN Device Configuration Script

Download a configuration script to help configure your local VPN device settings so that you can connect with the virtual network gateway.

VENDOR: Microsoft Corporation

PLATFORM: RRAS

OPERATING SYSTEM: Windows Server 2012

11. Select your VPN device's **Vendor** from the list (currently, certain Cisco and Juniper appliances are in the list, as well as Microsoft's own RRAS Server 2012 or 2012 R2).

Although this configuration is not 100 percent finished, it should be possible to have your VPN device configured by using the exported CFG file. For a sample blog post on how to achieve this procedure for a Cisco firewall, have a look at the TechNet blog post from the Canadian IT Pro community at `http://blogs.technet.com/b/canitpro/archive/2013/10/09/step-by-step-create-a-site-to-site-vpn-between-your-network-and-azure.aspx`.

Summary

In this prefinal chapter, we guided you through the possibilities of Microsoft Azure as a public cloud solution. We started with some history around Windows Azure, how the product has evolved enormously in the last few months, and proceeded by clearly highlighting all current features of the product.

After the introduction of Microsoft Azure, we explained to you how to configure a Microsoft Azure subscription, and how to deploy virtual machines in it and manage them. This was followed by guiding you in the process of configuring Azure PowerShell, and allowing you to upload a custom VHD image to the Azure subscription.

In the last section of this chapter, we talked about how to achieve a true integration between your on-premises data center and Microsoft Azure—first by re-using the App Controller environment we built in the previous chapter, then by showing you the detailed steps on how to establish a site-to-site VPN connection configuration file, allowing a complete network-based integration.

In the next and final chapter of this book, we will clarify a few best practices and how-tos around Microsoft licensing for Hyper-V, Windows Data Center platforms, and System Center environments. Because as we all know, Microsoft licensing is not always that easy to understand, but still very important. And if you do it right, it might even save you a lot of money too.

9
Hyper-V and System Center Licensing

Although this is a technical book, we see a lot of misconceptions and confusions in the market when talking about Hyper-V and System Center licensing with our customers. Due to the complexity of Microsoft licensing overall, this is not really a surprise.

As I assume—after reading this book—you are really interested in deploying Hyper-V and or System Center in your organization, I thought sharing my experiences in how to handle some common licensing scenarios would not hurt.

This chapter will talk about the different aspects of Hyper-V and System Center licensing, as well as guide you in making a decision, based on some real-life scenarios.

> Most of the information used in this chapter is derived from the official Microsoft source around Windows Server, Hyper-V, and System Center licensing, which can be accessed at `http://www.microsoft.com/licensing/about-licensing/windowsserver2012-R2.aspx` and `http://download.microsoft.com/download/8/9/A/89A3F8B9-94DE-4956-A56E-F6D2B215D0E6/SystemCenter2012R2_Licensing_Guide.pdf`.

Make sure you remember these web resources when needing to answer any questions about licensing for your environment or your customer's environment.

I would also like to add some sort of disclaimer, stating that this information is provided as is, out of what I understand about Microsoft licensing from my own experiences, without me being responsible for any mistakes, misconceptions, or changes in the official Microsoft licensing guides and Microsoft-specific licensing contracts and agreements your organization might have. Therefore, the preceding URLs are like the "one and only official source".

Understanding Windows Server and Hyper-V licensing basics

Let's think back to *Chapter 1*, *Introduction to Microsoft Server 2012 R2 Hyper-V*, and *Chapter 2*, *Unwrapping Hyper-V 2012 R2 Components*, where we talked about Windows Server and Hyper-V. That's where the basics of licensing your virtualized platform start.

Windows Server 2012 R2 currently exists in four editions as follows:

- **Datacenter edition**: This is recommended for highly virtualized environments
- **Standard edition**: This is recommended for non- or limited-virtualized environments
- **Essentials edition**: This is recommended for SMB customers, limited to 25 users
- **Foundation edition**: This is recommended for SMB customers, limited to 15 users

This allows for the following virtualization rights:

- **Datacenter edition**: Running unlimited number of VMs with up to two CPUs is allowed
- **Standard edition**: Running only two VMs with up to two CPUs is allowed
- **Essentials edition**: Only allowed to run the essentials server
- **Foundation edition**: This is not allowed to be run in a virtualized environment

Let us try to clarify this with a real-life scenario:

Scenario 1: Enterprise environment having four physical servers with two quad-core CPUs, 256 GB of RAM each, and running about 100 VMs.

In this scenario, it's advisable to use the Datacenter edition, as it is a highly virtualized platform, using the following calculation:

- 4 x Datacenter editions (for each physical server) x 1 (up to two CPUs) = 4 DC licenses

The 100 VMs can get their activation license "inherited" from the physical host. There is no need to buy separate Windows licenses for the VMs, if they are running on Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2, or Windows 2008.

Explained from another angle, talking about licensing cost, it would also be more beneficial to go for the Datacenter edition, using the following calculation:

Windows Server Edition	Cost per CPU (*)	Total
Datacenter edition	$ 6,000 x 4	$ 24,000
Standard edition	$ 1,000 x 50	$ 50,000

(*) Licensing cost is as shown on the Microsoft Windows Server 2012 R2 website, but rounded up for ease of calculation.

Scenario 2: Enterprise environment having four physical servers with four quad-core CPUs, 256 GB of RAM each, and running about 100 VMs.

Windows Server Edition	Cost per CPU (*)	Total
Datacenter edition	$ 6,000 x 8	$ 48,000
Standard edition	$ 1,000 x 50	$ 50,000

As you can see from this example, the physical machine CPU sizing (the number of CPUs that is) is an important component when determining your virtualization platform architecture. In certain scenarios, it's cheaper to have more CPUs per server, and in other scenarios it might be more beneficial to buy more servers having only two CPUs, instead of fewer servers with more CPUs per server.

Understanding System Center licensing basics

The next component we talked about during the initial chapters is System Center Virtual Machine Manager and System Center App Controller.

Both solutions are part of the **System Center Suite 2012 R2**. An important change—from a license perspective for sure—as of the System Center 2012 editions is the fact that you are required to buy the full suite, as it is not possible to buy individual components as was the case with System Center 2007 or 2007 R2.

For some environments, this gets interpreted in the wrong way, with some saying Microsoft only wants to make money out of it, or they only want to force you to use the full suite. Where this is probably true up to a certain extent, I see a lot of advantages in this licensing model.

First of all, it just makes licensing discussions easier, or avoids having the discussion all the way. Just as with the Windows Server, System Center licensing is provided in two ways, a Standard edition and a Datacenter edition. Next, it makes it easier for customers to try another component of the suite, without actually being worried about license compliance. And the third reason, it's cheaper buying the full suite than the products separately. In the System Center 2007 days, the trade-off was three products. Meaning if you used three out of the six components of the suite, it was already cheaper for your organization.

System Center editions

As already mentioned, System Center 2012 R2 exists in two editions, Datacenter and Standard, both containing the same features set and components.

Both editions provide the same feature set and contain the following components:

- Operations Manager
- Configuration Manager
- Data Protection Manager
- Service Manager
- Virtual Machine Manager
- Endpoint Protection
- Orchestrator
- App Controller

Both provide the following features from a licensing perspective:

Feature	**Datacenter**	**Standard**
Number of physical CPUs	2	2
Number of managed operating system environments	Unlimited	2
Includes all SC components	X	X
Allows you to run the management software and supporting SQL Server Standard edition	X	X
Manage any workload	X	X
License cost with SA	$ 3,600	$ 1,300

Just as with the Windows Server licensing, there is a correlation between CPU and licenses. Each license allows for two CPUs.

> Basically, if you have a four-CPU-based host, you should buy two Datacenter editions.

An important point to note is there are **Server Management Licenses** (**SML**) and **Client Management Licenses** (**CML**), where the CMLs are required for devices that run on a nonserver OS. At present, CMLs exist for three products:

- Configuration Manager Client ML
- Endpoint Protection subscription
- Client Management Suite ML

The Configuration Manager ML and Endpoint Protection are integrated in the Core CAL Suite or Enterprise CAL Suite. The Client Management Suite ML is available only through the Enterprise CAL Suite.

Enrolment for Core Infrastructure (ECI) licensing

The ECI is a licensing program available in an enterprise agreement and offers a three-year term, software assurance, and annual true-up. Customers can use the ECI to purchase **Core Infrastructure Server** (**CIS**) suites, as well as other Microsoft Server products.

The ECI was actually created beginning in 2012, providing enterprise customers with a **15 percent to 20 percent** discount on the bundled combination of Windows Server 2012 and System Center 2012.

It is still valid for Windows Server 2012 R2 and System Center 2012 R2 as well.

The ECI licensing has the following licensing features:

- Each license is based on a processor count
- Each license covers up to two physical processors
- CIS Suite Standard consists of Windows Server and System Center Standard edition
- CIS Suite Data center consists of Windows Server and System Center Datacenter edition
- Minimum initial purchase is 25 licenses

For more background information on the ECI Suite licensing, visit the Microsoft website article at `http://www.microsoft.com/licensing/licensing-options/enrollments.aspx`.

Extending your license agreements to Microsoft Azure

The third component we talked about in this book is Microsoft Azure, which also has a few licensing requirements, depending on what you are running or what licensing agreement you already have.

Windows Server licensing in Azure

The first interesting point to make is your Windows Server license for your VM in Azure is already *included in the per-minute cost* of your Windows VM.

The second interesting fact is you don't need Windows Server CALs for your end users or devices connecting to your Azure Windows VMs. The access rights to connect to this VM are also *included in the per-minute cost*.

The third important point to talk about is license mobility. License mobility is a license right that allows you to "move" a Microsoft Server product license (such as Exchange Server, for example) you acquired for your on-premise infrastructure to a public cloud infrastructure. It requires a software assurance as part of your licensing agreement as well.

> However, license mobility does not exist for Windows Server.

Let me explain this by giving you another example:

Scenario: A customer is running a Windows Server 2012 VM with Exchange Server 2013 as an application. The customer has the Windows Server VM licensed through a Windows Server 2012 Datacenter edition, as well as having an Exchange Standard edition 2013 license and required Windows Server CALs and Exchange Server CALs.

When moving this VM from the on-premise data center to Microsoft Azure, the Exchange Server 2013 license will persist by using the license mobility feature. The Windows Server license will be replaced by the Azure-integrated Windows Server license as part of the running cost of the VM.

As the customer was already using a Windows Server Datacenter edition, there is no real benefit of saving money. However, if the VM had been licensed through a Windows Server Standard edition, this license would become available for running another Windows Server VM on-premise, actually saving some money.

System Center licensing and Microsoft Azure

As your Windows Server VMs in Microsoft Azure can be managed by System Center, just as if they were running in your own on-premise data center, these VMs also require a System Center license.

The System Center license itself is *not* included in the Azure VM running cost.

However, the already mentioned license mobility feature can be used for System Center as well, with the following requirements:

- A System Center Standard license can be used to manage **two VM instances**
- A System Center Datacenter license can be used to manage **eight VM instances**

So, there is no "unlimited use" aspect in the Microsoft Azure scenario, as is the case for your on-premise infrastructure.

More details about the System Center and Microsoft Azure licensing combinations can be found at `http://go.microsoft.com/fwlink/?linkid=299855&clcid=0x409`.

Another interesting website that explains a lot of common FAQs around Microsoft Azure licensing is `http://azure.microsoft.com/en-us/pricing/licensing-faq/`.

Summary

This chapter covered the basics about the (still rather complex!) licensing requirements and possibilities for correctly licensing your Windows Server 2012 R2 and System Center 2012 R2 infrastructure.

We started with explaining the different Windows Server editions and what the licensing features are for each edition. This structure was repeated for the System Center Suite. We briefly touched the ECI licensing program.

To finish with, we guided you through the licensing characteristics of Windows Server VMs in Microsoft Azure, and the possibilities for licensing your Microsoft Azure VMs when managing them through System Center.

Index

Thank you for buying

Mastering Hyper-V

About Packt Publishing

Packt, pronounced 'packed', published its first book, *Mastering phpMyAdmin for Effective MySQL Management*, in April 2004, and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution-based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern yet unique publishing company that focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike.
For more information, please visit our website at `www.packtpub.com`.

About Packt Enterprise

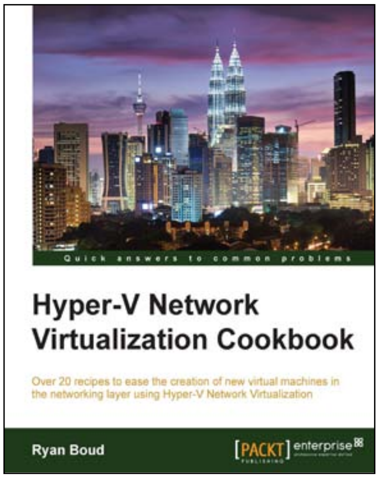
In 2010, Packt launched two new brands, Packt Enterprise and Packt Open Source, in order to continue its focus on specialization. This book is part of the Packt Enterprise brand, home to books published on enterprise software – software created by major vendors, including (but not limited to) IBM, Microsoft, and Oracle, often for use in other corporations. Its titles will offer information relevant to a range of users of this software, including administrators, developers, architects, and end users.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to `author@packtpub.com`. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, then please contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

[PACKT] enterprise
professional expertise distilled
PUBLISHING



Quick answers to common problems
Hyper-V Network Virtualization Cookbook
Over 20 recipes to ease the creation of new virtual machines in the networking layer using Hyper-V Network Virtualization
Ryan Boud
[PACKT] enterprise

Hyper-V Network Virtualization Cookbook

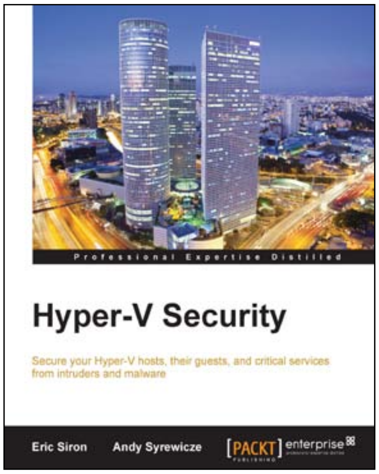
ISBN: 978-1-78217-780-7 Paperback: 228 pages

Over 20 recipes to ease the creation of new virtual machines in the networking layer using Hyper-V Network Virtualization

1. Create, configure, and administer System Center 2012 R2 virtual networks with Hyper-V.

2. Design practical solutions to optimize your network solutions.

3. Learn how to control who can access a VM on a specific port to enhance the security of your virtual machine.



Professional Expertise Distilled
Hyper-V Security
Secure your Hyper-V hosts, their guests, and critical services from intruders and malware
Eric Siron
Andy Syrewicze
[PACKT] enterprise

Hyper-V Security

ISBN: 978-1-78217-549-0 Paperback: 184 pages

Secure your Hyper-V hosts, their guests, and critical services from intruders and malware

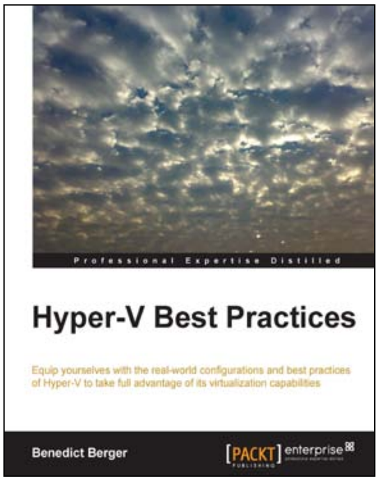
1. Discover how standard security best practices are applicable in a Hyper-V environment.

2. Acquire techniques to defend virtual machines against threats.

3. A practical guide that helps you lock down Hyper-V hosts and enhance their protection against attacks.

Please check **www.PacktPub.com** for information on our titles

[PACKT] enterprise
professional expertise distilled
PUBLISHING

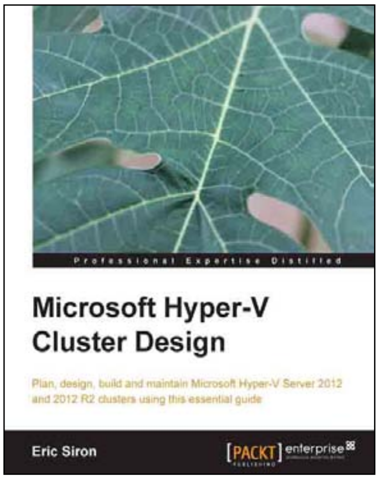

Professional Expertise Distilled
Hyper-V Best Practices
Equip yourselves with the real-world configurations and best practices of Hyper-V to take full advantage of its virtualization capabilities
Benedict Berger
[PACKT] enterprise

Hyper-V Best Practices

ISBN: 978-1-78217-609-1 Paperback: 172 pages

Equip yourselves with the real-world configurations and best practices of Hyper-V to take full advantage of its virtualization capabilities

1. Learn how to deploy standardized Hyper-V solutions quickly and effectively.
2. Leverage the complete virtualization potential of Hyper-V in production environments.
3. Use state of the art network and storage options with Hyper-V.


Professional Expertise Distilled
Microsoft Hyper-V Cluster Design
Plan, design, build and maintain Microsoft Hyper-V Server 2012 and 2012 R2 clusters using this essential guide
Eric Siron
[PACKT] enterprise

Microsoft Hyper-V Cluster Design

ISBN: 978-1-78217-768-5 Paperback: 462 pages

Plan, design, build, and maintain Microsoft Hyper-V Server 2012 and 2012 R2 clusters using this essential guide.

1. Successfully deploy a Microsoft Hyper-V Server cluster.
2. Use the right tools to build and maintain a Hyper-V cluster.
3. Master the intricacies of Hyper-V Server in a clustered environment.

Please check **www.PacktPub.com** for information on our titles