Professional Expertise Distilled

Implementing VMware Horizon View 5.2

A practical guide to designing, implementing, and administrating an optimized Virtual Desktop solution with VMware Horizon View

Jason Ventresco

[PACKT] PUBLISHING enterprise professional expertise distilled

Implementing VMware Horizon View 5.2

A practical guide to designing, implementing, and administrating an optimized Virtual Desktop solution with VMware Horizon View

Jason Ventresco

[PACKT] PUBLISHING enterprise professional expertise distilled

BIRMINGHAM - MUMBAI

Implementing VMware Horizon View 5.2

First published: May 2013

Production Reference: 1100513

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

ISBN 978-1-84968-796-6

www.packtpub.com

Cover Image by Artie Ng (artherng@yahoo.com.au)

Credits

Author
Jason Ventresco

Reviewers
Justin Paul
Mario Russo
Raymond van't Hag

Acquisition Editor
Andrew Duckworth

Lead Technical Editor
Sweny M. Sukumaran

Technical Editors
Prasad Dalvi
Amit Ramadas

Project Coordinator
Arshad Sopariwala

Proofreader
Stephen Copestake

Indexer
Monica Ajmera Mehta

Graphics
Ronak Dhruv
Abhinash Sahu

Production Coordinator
Nitesh Thakur

Cover Work
Nitesh Thakur

About the Author

Jason Ventresco is a 13-year veteran of the IT field, currently working for EMC² as a Principal Solutions Engineer. In that role he architects, builds, and tests the latest end user computing solutions to validate their performance and provide guidance to EMC² customers and partners. Jason has previously worked as a member of the Global Infrastructure team for FHI 360 and as an IT consultant for WorkSmart and Xerox Global Services.

Jason lives in Raleigh, North Carolina with his wife Christine and daughter Amanda. He holds two degrees, a Master of Science in Information Assurance from Norwich University and a Bachelor of Science in Information Technology from the University of Phoenix. He holds many certifications, some of which include VCAP-DTD, VCP5-DT, VCP, CISSP, EMCCA, and MCITP for Server 2008 and Exchange 2010. In his free time, he likes to travel and attend Carolina Hurricanes hockey and Durham Bulls baseball games.

I would like to thank my wife Christine and daughter Amanda for supporting me throughout all phases of my career, while I was attending college, and during the countless hours I spent writing this book. You are my inspiration, and I love you both.

I would also like to thank my parents, Richard and Linda Ventresco, for everything they have done for my family and me. I also thank them for helping me buy that computer when I was 13. Had that not happened, who knows what I would be doing today. I would not be where or who I am without their love and support.

About the Reviewers

Justin Paul is a Project Engineer at DP Sciences focused in the area of storage and virtualization. He has been voted one of the top virtualization-related bloggers, as well as a recipient of several vendor recognition awards including VMware's vExpert program (1 of 400) and the new EMC Elect program (1 of 75).

Besides blogging, his articles have also been published in the Dayton Technology First IT journal. He holds many certifications, some of which include VMware VCAP-DCD5, VCP5-DV, and EMCIE and EMCTA for VNX.

Justin attended the University of Dayton and majored in Computer Information Systems. He has also attended several technology-specific training classes.

At his first job, he was repeatedly pulled aside for thinking out of the box and using solutions other than what were "in the book". He still strives to break the "norm" and use innovative solutions and technologies for creative problem solving.

Mario Russo has specialized in Virtualization Solutions. He is owner of the organization Business to Virtual (`www.businesstovirtual.com`).

He is the IBM Business Partner and VMware Professional Partner.

The complete list of customers is available at `http://it.linkedin.com/in/mariorussob2v`.

I would like to thank my wife, Lina, for her coffee and endurance.

I would like to thank my daughter, Gaia, for the time that I took away from our games.

Raymond van't Hag has been working for VMware for over five years and currently holds the role of Sr. Specialist Systems Engineer in end user computing in the Netherlands. Before VMware, he worked for companies such as Dell, Symantec, and IBM. Today he is responsible for supporting larger VMware Horizon projects (such as Horizon View, Mirage, Workspace, and ThinApp), educating VMware Partners, and evangelizing VMware End-user Computing strategy via social media, and especially his own blog `http://bright-streams.com`. In 2012, Raymond also reviewed "VMware View 5 Desktop Virtualization Solutions" by Jason Langone and Andre Leibovici.

www.PacktPub.com

Support files, eBooks, discount offers, and more

You might want to visit `www.PacktPub.com` for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at `www.PacktPub.com` and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at `service@packtpub.com` for more details.

At `www.PacktPub.com`, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

`http://PacktLib.PacktPub.com`

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at `www.PacktPub.com`, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Instant Updates on New Packt Books

Get notified! Find out when new books are published by following `@PacktEnterprise` on Twitter, or the *Packt Enterprise* Facebook page.

Table of Contents

Preface

Implementing VMware Horizon View 5.2 is a hands-on guide on how to design and implement the different components of View. The examples provided in this book build upon one another, and guide the reader through the basics of View infrastructure design, and then the installation and configuration of each core View component. Using the examples provided in this book, the reader will be able to assess the basic needs of their View infrastructure, and then implement and manage their own View environment.

There are many places in this book that refer the reader to the official VMware Horizon View documentation. You are encouraged to review this documentation as it complements the material in this book, and contains additional information that can provide for a deeper understanding of the technical details and capabilities of the entire VMware Horizon View software suite.

Why Virtual Desktops?

There are a number of different reasons why an organization might decide to implement VMware View within its own environment. Many organizations are already familiar with the benefits of virtualization, such as:

- **Server consolidation**: Less physical hardware required to service the same quantity of workload
- **Simplified management**: Fewer physical resources to manage
- **More energy-efficient**: Less power and cooling required
- **Hardware independence**: Virtual machines can run on almost any hardware platform without any changes required
- **Enhanced capabilities**: Deploy new virtual servers much faster than physical ones, and with less effort

These are just a small sample of the benefits of virtualization. If you have already implemented virtualization in your organization, you likely have reasons of your own.

Virtual Desktops can provide organizations with additional advantages beyond those of virtualization itself. With VMware View, we can:

- Roll out a new Windows desktop OS across your enterprise without making any changes to the existing desktops. Those new OS pilots will be a lot less risky when users can retain access to their existing desktop.
- View desktops live in the datacenter and can be accessed from almost anywhere, from a variety of clients. View desktops, as well as the data stored on them, can't be left in airports, stolen from cars, or accidentally left on your desk at the office. Worried about people copying data from their View desktop to a USB drive on the client? Disable that feature.
- Stop caring about endpoint hardware. Use existing Windows PCs as desktops if you want, or move to a zero client and do away with common endpoint management tasks. Better yet, have users bring their own device and let them use it to access their View desktop. Worry about what's in the datacenter, not on the desk.
- Microsoft Patch Tuesday redefined. With View linked clones, you patch once and then update the desktops with a whole new base image. No more testing patches across 15 different hardware platforms. No more monitoring patch status across hundreds or thousands of desktops. The same technique can be used to roll out new applications as well.
- Less power and cooling are needed for the rest of the building. Zero and thin View clients typically require less power and generate less heat than a physical desktop.
- Stop troubleshooting one-off desktop problems. Problems with Windows? Deploy a new desktop in minutes. With features such as View Persona Management to protect user profile data, and ThinApp to automatically deploy applications not present on the Virtual Desktop base image, the individual desktop doesn't have to matter. If a problem will take more than 10 minutes to fix, deploy a new desktop instead.

These are just some of the advantages you can realize by using VMware View and vSphere to move your desktops into the datacenter. While reading this book, I encourage you to think of ways that View can change how you provide end user computing resources to your organization. For example:

- Don't just simply forklift your desktops into the datacentre as full virtual machines; consider the benefits of linked clones.

- Rather than creating large numbers of master images for different departments or worker types across your organization, create a basic image that you can layer applications on top of using ThinApp or even VMware Horizon Workspace.
- Investigate software that is optimized for Virtual Desktops, such as the vShield Endpoint antivirus platform. Software that is optimized for Virtual Desktop platforms may require less per-desktop resources, which enables you to run more desktops on a given vSphere host.
- View has features that make the individual desktop less important; use them. Use Persona Management to make your user data portable and ThinApp to make applications portable, and suddenly the individual desktop won't matter as much. The more portable everything is, the more options you will have for the types of View desktops that you deploy.

VMware Horizon View can provide you with much more than just a means of virtualizing your desktops. The more familiar you become with its features and capabilities, the more you will realize that you can rethink much of what you do concerning desktop management and delivery, and provide a higher quality experience to your end users. I certainly hope that is the case.

What this book covers

Chapter 1, *Designing a VMware Horizon View Infrastructure*, covers a number of key topics that are integral to the design of your View infrastructure. Learn about each of the different View software components, base infrastructure requirements, and how to assess Virtual Desktop resource requirements.

Chapter 2, *Implementing VMware Horizon View Connection Server*, covers Connection Server infrastructure requirements, sizing, limits, high availability, installation, configuration, backup, and recovery.

Chapter 3, *Implementing VMware Horizon View Composer*, covers Composer infrastructure requirements, installation, configuration, backup, and recovery. The benefits of linked-clone desktops are also discussed.

Chapter 4, *Implementing VMware Horizon View Transfer Server*, covers Transfer Server infrastructure requirements, installation, configuration, backup, and recovery. The benefits and capabilities of the View Local Mode desktops are also discussed.

Chapter 5, *Implementing VMware Horizon View Security Server*, covers Security Server usage, infrastructure requirements, sizing, limits, high availability, installation, configuration, backup, and recovery.

Chapter 6, *Using VMware ThinApp*, covers how to use ThinApp to virtualize applications and deploy them using View. Additional topics covered include an overview of ThinApp, benefits, limitations, and how to update applications packaged with ThinApp.

Chapter 7, *Implementing View Persona Management*, covers how to use the View Persona Management feature to manage Windows user profiles. Topics covered include requirements, features, and configuration.

Chapter 8, *Creating VMware Horizon View Desktop Pools*, covers how to create desktop pools using the View Manager Admin console. Topics covered include desktop pool options, desktop pool types, monitoring pool creation, user entitlement, and common provisioning problems.

Chapter 9, *VMware Horizon View Client Options*, covers the different types of clients available for View. Topics covered include the difference between thin and zero clients, supported operating systems and their requirements, client installation, and client command-line options.

Chapter 10, *Performing View Desktop Maintenance*, covers how to perform maintenance on View linked-clone desktops. Topics include an overview of the different maintenance operations including refresh, recompose, and rebalance, and how to manage the optional linked-clone persistent disks.

Chapter 11, *Creating a Master Virtual Desktop Image*, covers the techniques that should be used when creating a master Virtual Desktop image. Topics covered include the importance of desktop optimization, sample optimization results, and how to optimize the virtual machine hardware, Windows filesystem, Windows OS, and Windows user profile.

Chapter 12, *Managing View SSL Certificates*, covers how to replace the default SSL certificates on each of the View components including View Composer and the View Connection, Security, and Transfer Servers. Also discussed is how to create SSL certificate requests and obtain new certificates using a Microsoft Active Directory Certificate Services server.

Chapter 13, *Implementing VMware Horizon View Group Policies*, covers how to use the View Active Directory Group Policy templates to customize the different View software components. Topics covered include a detailed description of each of the different group policy template settings, an explanation of where the settings should be applied within Active Directory, the location of the Group Policy template files, and the importance of Group Policy loopback processing with View desktops.

Chapter 14, *Managing View with PowerCLI*, covers how to use vSphere PowerCLI to configure and manage the View infrastructure. All of the View PowerCLI commands are covered in detail, and examples are provided that show how they are used.

You can download this chapter from `http://www.packtpub.com/sites/default/files/downloads/7966EN_Chapter14_Managing_View_with_PowerCLI.pdf`.

Chapter 15, *VMware Horizon View Feature Pack 1*, covers the new features introduced with the release of the VMware Horizon View Feature Pack 1, including client access to View desktops over HTML5 and the Unity Touch interface. The topics covered include feature pack requirements, installation of the Feature pack components, enabling HTML access to desktops, HTML access limitations, and how to customize the Unity Touch interface.

You can download this chapter from `http://www.packtpub.com/sites/default/files/downloads/7966EN_Chapter15_VMware_Horizon_View_Feature_Pack_1.pdf`.

Appendix, *Advanced Details about Key Horizon View Features*, covers advanced information about the following View subjects: event logging options, vCenter provisioning options, Local Mode desktop policy settings, customizing View desktop names, and optimizing the PCoIP protocol.

What you need for this book

The reader should have a basic understanding of the following concepts that are integral to the implementation and management of View.

- Microsoft Windows Server
- Microsoft Active Directory
    - Certificate services
    - DNS
    - Group policies
- VMware vSphere
    - vCenter Server
    - Virtual machine snapshots
    - Virtual machine templates
    - VMware tools
    - vSphere administration

- Networking
    - DHCP
    - Protocol and port types
    - Basics of LAN and WAN networking

The following software is required to implement the solutions described in this book:

- VMware Horizon View installation media including all optional components
- VMware Horizon View Feature Pack 1
- vSphere 5.1 installation media including vCenter Server and vSphere
- Windows Server 2008 R2 installation media
- Installation media for a supported Windows desktop OS
- The installation media for the required VMware products can be obtained from the `VMware.com` website. If you do not have a current license for the products, you can register for a trail to obtain access to the software.

Who this book is for

If you are a newcomer to system administration and you wish to implement a small to midsized Horizon View environment, then this book is for you. It will also benefit individuals who wish to administrate and manage Horizon View more efficiently, or are studying for the VMware Certified Professional-Desktop (VCP5-DT).

Conventions

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

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

```
vdmimport -d -p password -f backup.LDF > decrypted.LDF
```

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Highlight the targeted View Connection Server and click on **Edit** to open the **Edit View Connection Server Settings** window."

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 may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to `feedback@packtpub.com`, and mention the book title via 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 on `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt 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 would 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 on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from `http://www.packtpub.com/support`.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, 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 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

You can contact us at `questions@packtpub.com` if you are having a problem with any aspect of the book, and we will do our best to address it.

1
Designing a VMware Horizon View Infrastructure

One task that is critical to the success of any VMware Horizon View implementation is the initial research that will shape the design of the View infrastructure. Performing this research requires not only an understanding of the individual components of View, but also an in-depth understanding of what is required to move our end user computing resources from the desk into the datacenter.

This chapter will discuss a number of topics that play a critical role in our View design. We will discuss the different components of a View installation, examine the different license levels of View, and outline the core requirements of a View infrastructure. We will also discuss how to measure the resource requirements of a desktop, and how those requirements impact all layers of our infrastructure including the storage design, network design, and virtual desktop VMware vSphere host configuration.

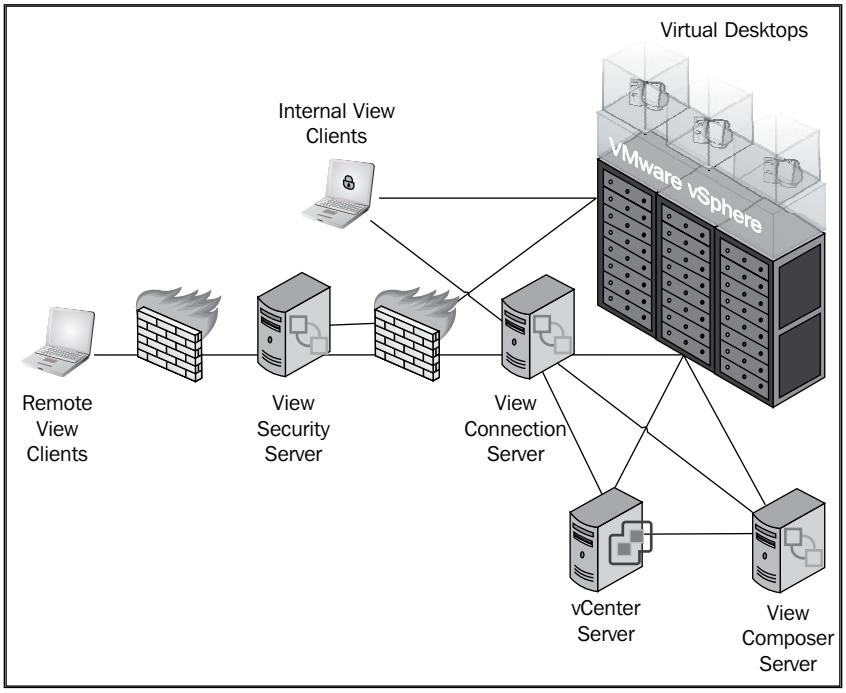
In this chapter we will learn:

- The individual components of a VMware Horizon View installation
- The role of different components of VMware Horizon View
- VMware Horizon View license options
- Core infrastructure requirements for VMware Horizon View
- Desktop operating system (OS) design considerations
- How to measure desktop resource requirements
- How to calculate the size of our virtual desktop vSphere hosts
- View client network bandwidth requirements
- The basics of running a VMware Horizon View pilot

VMware Horizon View components

A VMware Horizon View installation is comprised of a number of different components. The following section will provide a high-level overview of the function of the various components of View, not all of which may be required in your environment.

The following figure shows where each of the components of a typical View installation resides within the IT infrastructure. The only component not shown is the View Transfer server, which resides within the private network and is described in greater detail in *Chapter 4*, *Implementing VMware Horizon View Transfer Server*.



VMware Horizon View Connection Server

VMware Horizon View Connection Server is a software service that serves as the broker for View client connections. In this role, it authenticates user connection requests, verifies the desktops or Microsoft Windows Terminal Servers the user is entitled to access, and then directs the connection to the appropriate resource. View Connection Server is installed on a dedicated server that is required to be a member of an **Active Directory** (**AD**) domain that is trusted by all View clients.

View Connection Server also hosts the View Administrator console, an Adobe Flex-based web application that is used to manage the View environment and perform tasks, such as:

- Deploying virtual desktops
- Creating desktop pools
- Controlling access to desktop pools
- Examining View system events

The View Connection Server is one component that is required in every View environment due to the role it plays as the connection broker and management console.

Chapter 2, *Implementing VMware Horizon View Connection Server*, provides the information needed to install and configure a VMware Horizon View Connection Server.

VMware vSphere

VMware vSphere, also referred to as ESXi or even ESX for earlier versions, is a Type 1 hypervisor that is the virtualization platform used for the vSphere suite of products. Type 1 hypervisors are designed to run directly on the host hardware, whereas Type 2 hypervisors run within a conventional operating system environment.

vSphere is the only hypervisor that is fully supported for hosting View virtual desktops, as it fully integrates with View for full desktop lifecycle management. All of the primary desktop provisioning and maintenance tasks are performed using the View Manager Admin console; the vSphere Client is not used. View supports multiple versions of vSphere, but vSphere 5.0 and newer are required to leverage many of the latest features of the platform. Refer to the *vCenter Server requirements* section for examples of some View features that require a specific version of both vSphere and vCenter Server.

VMware vCenter Server

VMware vCenter Server is a software service that provides a central administration point for VMware vSphere hosts as well as other components of the vSphere suite. vCenter Server performs the actual creation and management of virtual desktops, based on instructions received from the View Connection Server and the View Composer Server.

VMware Horizon View Composer

VMware Horizon View Composer is a software service that works alongside the VMware vCenter and View Connection Servers to deploy and manage linked-clone desktops. View Composer can be installed directly on the vCenter Server, or on a dedicated server.

View Composer is only required if linked-clone desktops will be deployed. *Chapter 3*, *Implementing VMware Horizon View Composer*, provides the information needed to install and configure View Composer.

VMware Horizon View Transfer Server

VMware Horizon View Transfer Server is a software service that controls data transfers for virtual desktops that are checked out for use directly on the **View Client with Local Mode**. The View Client with Local Mode is used in scenarios where access to a virtual desktop is required during times where no network access is available. View Transfer Server is installed on a dedicated server.

Local Mode desktops require a full Windows XP, Vista, Windows 7, or Windows 8 based-client, and run on a Type 2 hypervisor that is installed with the View Client with Local Mode installation package.

VMware Horizon View Agent

VMware Horizon View Agent is a software service that is installed on the systems that will be managed by View. This includes not only a virtual desktop image that will be deployed using View, but any physical desktops or Microsoft Terminal Servers as well.

The View agent provides services including, but not limited to, support for connecting the virtual desktop to View's client-attached USB devices, client connection monitoring, Virtual Printing, single sign-on, and View Persona Management.

VMware Horizon View Client

VMware Horizon View Client is a software application that is used to communicate with a View Connection Server, and initiate connections to desktops and Microsoft Windows Terminal Servers.

The View Client is available for multiple software platforms, including Microsoft Windows, Apple OS X, Android, iOS, and Ubuntu Linux. In addition, there are a number of Thin and Zero clients that come preloaded with View-compatible clients.

The VMware Horizon View Client with Local Mode, described previously in this chapter, can also be used to connect to desktops and laptops remotely. *Chapter 9*, *VMware Horizon View Client Options*, provides more information about the various View Client options.

VMware Horizon View Persona Management

VMware Horizon View Persona Management is an optional component of the View Agent that enables an alternative means of managing end user Windows profile data and application settings.

View Persona Management can be used in place of traditional Microsoft Windows roaming profiles, while also providing additional benefits such as:

- User profile data is loaded only as required, speeding up the user desktop login process
- User profile updates can be synced back to the remote persona management repository at predefined internals, enabling quicker logoff times compared to traditional Windows roaming profiles
- View Persona Management settings are controlled through Microsoft Active Directory (AD) Group Policies rather than through individual Active Directory user objects

Chapter 7, *Implementing View Persona Management*, provides information about how to implement and administer View Persona Management.

VMware ThinApp

VMware ThinApp is an application virtualization platform that integrates with View to provide users with rapid access to new or upgraded applications without having to perform any changes to the virtual desktops.

Applications that have been packaged with ThinApp are delivered as a single executable file that runs completely isolated from both other ThinApp packaged applications as well as applications that are installed on the desktop itself. If required, ThinApp packages can be configured to communicate with one another using a feature known as **ThinApp AppLink**.

ThinApp provides View customers with a number of powerful capabilities. The following list details three popular scenarios where ThinApp can benefit an organization:

- Reduce the number of applications that need to be installed on the master virtual desktop image, which reduces the need to deploy and maintain a large number of images for different user bases
- Eliminate application conflicts that can occur when specific programs are installed together within the desktop image
- Virtualize legacy applications to ensure that they will continue to function regardless of the underlying Windows OS

Chapter 6, *Using VMware ThinApp*, provides information about how to use VMware ThinApp to virtualize applications and deliver them using View.

VMware Horizon View licensing

VMware Horizon View offers three different license levels: **Bundle**, **Add-on**, and **Add-on to Bundle Upgrade**. The license levels are differentiated by whether or not they include licenses for vCenter Server and the vSphere hosts. The licenses are sold in 10 and 100 packs.

- The Bundle license includes all the features of VMware Horizon View, including the licenses needed for the vSphere desktop hosts and vCenter Server. The version of vSphere included with this license is known as **vSphere Desktop**.
- The Add-on license includes all the features of VMware Horizon View, but you must provide your own licenses for the vSphere desktop hosts and vCenter.
- The Add-on to Bundle Upgrade license is for customers who already have Add-on licenses, but wish to upgrade them to the Bundle license level.

The advantage of using vSphere Desktop is that it is licensed on a per-desktop basis, while a traditional vSphere license is licensed on a per-socket basis. This provides View customers with maximum flexibility when considering what servers they will use when deploying their View infrastructure, as it removes the per-socket licensing costs as a deciding factor in server selection.

Visit the VMware Horizon View website (`http://www.vmware.com/products/view/overview.html`) for the most recent information concerning licensing options and their costs.

VMware Horizon View core infrastructure requirements

There are a number of requirements to consider even before the infrastructure needs of the virtual desktops themselves are considered. These include, but are not limited to:

- Operating system requirements for both vSphere and View components
- Database requirements for vCenter Server, View Composer, and View Connection Server
- Required Microsoft infrastructure services and components

The online **VMware Compatibility Guide** (`http://www.vmware.com/resources/compatibility/search.php`) and **Product Interoperability Matrixes** (`http://partnerweb.vmware.com/comp_guide2/sim/interop_matrix.php`) maintain an up-to-date listing of supported operating systems, hardware platforms, and product compatibility for all VMware products.

Microsoft infrastructure requirements

VMware Horizon View requires **Microsoft Active Directory** to support the virtual desktop infrastructure. VMware Horizon View supports both Windows 2003 and Windows 2008 Active Directory.

View also requires **Domain Name System** (**DNS**) servers that can resolve requests for the standard Microsoft Active Directory **Service Record** (**SRV**) and **Resource Record** (**RR**) DNS entries. Microsoft domain-integrated DNS servers typically store these DNS entries by default. Incomplete or inaccurate DNS entries can lead to issues with tasks, such as virtual desktop deployment and user authentication.

Dynamic Host Configuration Protocol (**DHCP**) servers are required in the View environment to provide **Internet Protocol** (**IP**) addresses to the virtual desktops. In situations where the virtual desktops cannot self-register the IP addresses they have been assigned, the DHCP server should be configured to register the entries with a DNS server that is accessible by the View Connection server.

Operating system requirements

The following table shows which Microsoft Windows Server Operating Systems (OSs) are supported for the each of the different software packages that comprise a View infrastructure. Unless otherwise noted, the software packages support the Standard, Enterprise, and Datacenter versions of the Microsoft Server version listed.

Operating System	vCenter Server 5.1	Horizon View Connection Server, Security Server, Transfer Server, and Composer
Windows Server 2003 SP2 64-bit	Supported	Not supported
Windows Server 2003 R2 64-bit (any service pack)	Supported	Not supported
Windows Server 2008 64-bit (both SP1 and SP2)	Supported	Not supported
Windows Server 2008 R2 (No SP or SP1 installed)	Supported	Supported for Standard and Enterprise versions only

As View Composer supports only Windows Server 2008 R2, any View installation that plans on deploying linked-clone desktops and installing Composer directly on the vCenter Server will need to choose that specific version of Windows.

Visit the VMware Horizon View Installation guide (`http://www.vmware.com/support/pubs/view_pubs.html`) for updated information about which OSs are supported.

Database requirements

The following table shows which database types are supported for the components of a View infrastructure. Unless otherwise noted, both 32-bit and 64-bit versions of the specified database platform are supported. In addition, unless otherwise noted the ONE, Standard, and Enterprise versions of the Oracle database platforms are supported.

Database Platform	vCenter (all databases)	View Composer	View Event Log
IBM DB2 10 Enterprise	Supported	Not supported	Not supported
IBM DB2 Enterprise 9.7.2	Supported	Not supported	Not supported

Database Platform	vCenter (all databases)	View Composer	View Event Log
Microsoft SQL Server 2005 Standard, Enterprise, and Datacenter editions (SP4)	Supported	Supported	Supported
Microsoft SQL Server 2008 Standard and Enterprise editions (SP2, SP3)	Supported	Supported	Supported
Microsoft SQL Server 2008 Datacenter edition (SP2)	Supported	Supported	Supported
Microsoft SQL Server 2008 R2 Express (64-bit only), Standard, and Enterprise editions (SP1)	Supported	Supported; Express supported only for vCenter 5.0 U1 and newer	Supported; Express supported only for vCenter 5.0 U1 and newer
Oracle 10*g* (Release 2)	Supported	Supported	Supported
Oracle 11*g* (Release 1 and 2)	Supported	Supported (Release 2 with Patch 5 only)	Supported (Release 2 with Patch 5 only)

Visit the VMware Horizon View Installation guide for updated information on which databases are supported.

vCenter Server requirements

VMware Horizon View supports multiple versions of vSphere and vCenter Servers. The purchase of Bundle or Add-on to Bundle Upgrade licenses entitles users to use the latest supported version of both vSphere and vCenter Servers.

The following versions of vSphere are supported by VMware Horizon View:

- vSphere 5.1
- vSphere 5.0, 5.0 U1, and 5.0 U2
- vSphere (ESX/ESXi) 4.1 , 4.1 U1, 4.1 U2, and 4.1 U3
- vSphere (ESX/ESXi) 4.0 U3 and 4.0 U4

The following versions of vCenter Server are supported by VMware Horizon View:

- VMware vCenter Server 5.1
- VMware vCenter Server 5.0, 5.0 U1, and 5.0 U2
- VMware vCenter Server 4.1 U1, 4.1 U2, and 4.1 U3
- VMware vCenter Server 4.0 U3 and 4.0 U4

Visit the VMware Product Interoperability Matrixes for an updated listed of the supported versions of vSphere and vCenter Servers.

Supporting earlier versions of vSphere and vCenter Servers is important for customers who are already running earlier versions of either software platform, and cannot or will not upgrade for some reason. Even with this support, it is recommended to use dedicated vSphere hosts and vCenter Servers for your View environment to ensure that all the latest View features are supported.

There are multiple View features that are supported only if certain other prerequisites are met. Some examples of these requirements are:

- View Composer requires Windows Server 2008 R2 as the host operating system, which became available after vSphere 4.0 was launched. Customers running vSphere 4.0 may need to upgrade their Windows OS to gain support for View Composer.
- View Storage Accelerator requires vSphere 5.0 or newer. Customers who wish to leverage this feature will need to upgrade their vSphere desktop hosts.
- Space Reclamation requires **Space-Efficient** (**SE**) Sparse format virtual hard disks, which is only available in vSphere 5.1 or newer.
- vSphere 5.0 or newer is required to enable View support for vSphere clusters with up to 32 hosts.

A complete list of View features that require specific versions of vSphere or vCenter Server can reviewed in either the official VMware Horizon View Installation guide or the View Release Notes that accompany each release of the View platform.

VMware Horizon View Agent supported operating systems

The VMware Horizon View Agent supports multiple versions of the Microsoft Windows desktop operating system and Microsoft Windows Terminal Server. The following table outlines which version of Windows is supported, based on what type of View-brokered service we wish to provide.

Windows Version	View Desktop or Terminal Service session	View Local Mode Desktop
Windows XP Professional 32-bit (SP3)	Supported	Supported
Windows Vista Business and Enterprise 32-bit (SP1 or SP2)	Supported	Not supported
Windows 7 Enterprise or Professional, 64-bit and 32-bit (No SP or SP1)	Supported	Supported
Windows 8 Enterprise or Professional, 64-bit and 32-bit	Supported	Supported
Windows 2008 Terminal Server 64-bit (SP2)	Supported	Not applicable
Windows 2008 R2 Terminal Server 64-bit (SP1)	Supported	Not applicable

To obtain current information about which desktop operating systems and Microsoft Terminal Services servers are supported, please refer to the online VMware Product Interoperability Matrixes.

Windows 7 virtual machines require vSphere 4.0 U4 (ESX or ESXi) or later, 4.1 U2 (ESX or ESXi) or later, 5.0 U1 or later, or 5.1 or later. Windows 8 virtual machines require vSphere 5.1 or later.

Measuring Virtual Desktop resource requirements

One of the most important aspects of any View design is ensuring that an infrastructure has adequate compute, storage, and network resources to host the required number of virtual desktops. Were it not for troublesome things such as budgets, we could simply purchase an excess of all three of those resources and rest easy at night. For this exercise, our goal is to build an infrastructure that is robust enough to support our average user workload, with some capacity in reserve for growth or maintenance purposes.

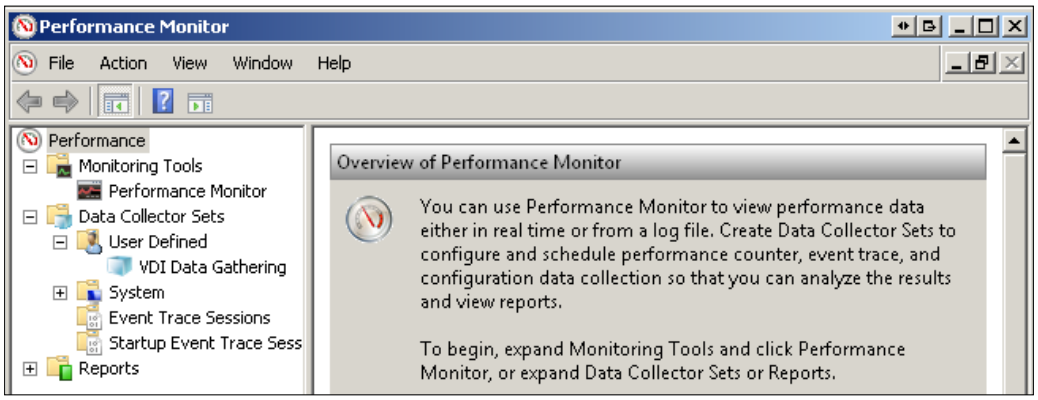
Determining the resource requirements of a View environment is a complicated task. Companies such as Liquidware Labs (`http://www.liquidwarelabs.com`) have created tools that can assist you in determining virtual desktop resource needs, while other companies such as Login VSI (`http://www.loginvsi.com/`) have created tools that can be used to test the performance of your View infrastructure. This section will focus on how to use the tools that you already have available with you, but you may wish to research the Liquidware Labs and Login VSI tools further to determine if they are something you would want to use when designing and testing your View infrastructure.

Using Performance Monitor to gather Windows counters

One of the most accurate means of measuring desktop resource usage is to gather performance data during a typical user session. The Microsoft Performance Monitor tool is built into every Windows operating system, and can be used to gather the required performance data.

Configuring Performance Monitor

The examples provided for this section will use the Windows 7 performance monitoring tool, which can be initiated from the Windows Start menu by running the command `perfmon`. The tool can also be found in the Windows Start menu under **All Programs** | **Administrative Tools** | **Performance Monitor**. The following screenshot shows the default view of the **Performance Monitor** window:



To determine the average core resource requirements of the virtual desktop, we will be gathering the following counters:

- **Network Adapter Bytes Total/sec**: This counter represents the total network throughput of the desktop. The average of this value will help us calculate the network requirements of each virtual desktop vSphere host.
- **PhysicalDisk**: This counter gives you read and write bytes per second. The disk read and write bytes of a desktop provide the basis for sizing the storage network connection that will connect the vSphere host to the storage infrastructure.
- **PhysicalDisk**: This counter gives you read and write operations per second. The number of disk reads and writes of a desktop provide the basis for sizing the virtual desktop storage platform. The storage design is impacted not only by the total amount of disk input/output (I/O), but by the ratio of reads to writes.
- **% Processor Time**: This counter measures the percentage of time the processor was busy during the interval. The average of this value will influence the number of virtual desktop processors we can host per vSphere server CPU core.
- **Memory Committed Bytes**: This counter represents the total number of bytes allocated by Windows processes, including any that were paged to physical disk. The average of this value will help us determine how much memory should be allocated to the virtual desktop master image, and by extension how much memory will be required in each virtual desktop vSphere host.

To gather the performance data for analysis, we need to create a user-defined Data Collector Set. To create the Data Collector set complete the following steps:

1. Expand the `Data Collector Sets` folder in the left column of the **Performance Monitor** window, right-click on the `User Defined` folder, and select **New | Data Collector Set**.
2. When prompted, provide a name for the Data Collector Set, select the option to **Create manually (Advanced)**, and select **Next**. This will create a blank Data Collector Set that we will populate with the performance metrics required to perform this analysis.
3. On the next screen select the **Create data logs** radio button, the check box next to **Performance counter**, and click on **Next**.

4. Add the following counters to the Data Collector Set by selecting **Add** to open the **Available counters** window. Highlight the indicated counter from the following list and selecting **Add >>** to add it to the **Added counters** list. Repeat this process for the remaining counters, and select **OK** when all have been added.
    - **Memory – Committed Bytes**
    - **Network Adapter – Bytes Total/sec – Network Adapter**
    - **PhysicalDisk – Disk Read Bytes/sec – 0 C:**
    - **PhysicalDisk – Disk Reads/sec – 0 C:**
    - **PhysicalDisk – Disk Write Bytes/sec – 0 C:**
    - **PhysicalDisk – Disk Writes/sec – 0 C:**
    - **Processor - % Processor Time - _Total**

The counters for the **Network Adapter** and **PhysicalDisk** objects are specific to each system. Please select all applicable network adapters and local system drives, using the *Ctrl* key if it is necessary to gather counters for multiple objects.

5. Accept the default sample interval of 15 seconds and select **Next**.
6. Select a destination directory for the counter data that is not being monitored by Performance Monitor, such as a network drive. This ensures that the disk activity associated with the Performance Monitor data-gathering process is not counted among the rest of the disk statistics. Select **Next** when finished with this step.
7. Accept the default **Run as:** settings or provide different service credentials if required. Select the **Save and close** radio button and then **Finish**.
8. In the left column of the **Performance Monitor** window, within the `User Defined` folder, highlight the Data Collector Set that has been created. In the right column, right-click on the Data Collector and select **Properties**. This Data Collector should be named `DataCollector01`.
9. In the **Log format:** drop-down menu, select **Comma Separated** and then **OK**. This will save the data in a format that can be imported into a spreadsheet program for more advanced analysis.

Performance Monitor is now configured to gather the required counters. To start gathering data, right-click on the `Data Collector Set` in the left column of the Performance Monitor application and select Start. To stop the data collection process, right-click on the same `Data Collector Set` and select Stop, or simply restart or power down the computer.

A default installation of a Microsoft Windows desktop operating system runs a number of processes and scheduled tasks that are typically not required in a virtual desktop environment. Performing desktop counter gathering or performance analysis without considering the impact of these services or tasks may lead to an overestimation of virtual desktop resource requirements. *Chapter 11*, *Creating a Master Virtual Desktop Image*, provides information about what changes should be made to a virtual desktop master image, and information that can also be applied to the sample desktop during data collection to understand the impact of the proposed changes on desktop performance.

The value of the Performance Monitor data gathered from a single desktop is dependent on a number of factors. It is very likely that in order to determine the resource requirements of our View infrastructure, we will need to monitor and analyze multiple types of users. A common way of classifying user types is to break them down into three distinct groups:

- **Task Workers**: They run a limited set of applications that typically have lower resource requirements. Examples include factory-floor computers and web-based data entry.
- **Knowledge Workers**: They run multiple, often concurrent, standalone Office applications, web-based tools, and other similar applications that have medium resource requirements. Examples include many types of office workers or other individuals who use a computer for most if not all aspects of their job.
- **Power Users**: They run resource-intensive applications that are more likely to require the advanced features of a desktop, including multiple vCPUs. Examples include workers that use programs such as SAP or Oracle clients, AutoCAD, or that require 64-bit desktop operating systems.

These classifications of user types are just one example. You will need to research the types of users you have within your own organization to determine if there is a more suitable way of classifying user resource requirements.

To accurately gauge the resource needs of our View infrastructure, we should gather desktop performance data from multiple users that fall within each user classification. The more sources of sample data we have, the less likely it is that any analysis will be influenced by anomalies from any one sample.

Using Performance Monitor to properly size the infrastructure

Once we have gathered desktop Performance Monitor data, we need to perform some data analysis to determine how to size our View infrastructure. This section will outline the processes used to take raw Performance Monitor data, and use it to determine our View infrastructure requirements.

Basics of sizing a View infrastructure

One of the most accurate ways to determine our infrastructure requirements is to take an average of each of the Performance Monitor counter values we have gathered, which should provide us with a per-desktop figure for the amount of resources that a given desktop type requires.

The first thing that must be taken into consideration is whether or not we plan on separating our virtual desktops based on any sort of metric or other user classification. In the previous section, we broke down users into one of three different groups: Task Workers, Knowledge Workers, and Power Users. Each group has different desktop performance expectations, and as their expected performance requirements increase, their tolerance for events that impact that performance decreases. Each user base is different, of course, but when designing our View infrastructure we should consider whether or not we should provide unique storage, network, or compute resources for each of our own user classes. The following provides an example of how that might be accomplished:

- **Task Workers**:
    - Higher desktop consolidation ratios per vSphere host
    - Lower tier storage
- **Knowledge Workers**:
    - Average desktop consolidation rations per vSphere host
    - Medium tier storage
- **Power Users**:
    - Low desktop consolidation ratios per vSphere host
    - High performing storage
    - Network QOS to guarantee desktop bandwidth availability

The analysis done in this section assumes that we are sizing a View infrastructure for one classification of user, and not multiple user classifications that may have different performance requirements. As we discussed earlier, our final design may allocate unique resources to each user classification in order to provide the expected level of performance.

Interpreting Performance Monitor data

The following screenshot shows a portion of the **Performance Monitor** data collected from a sample Windows desktop. This data was imported from the CSV file created by the **Performance Monitor** application.

	A	B	C	D
1	(Eastern I	\\Win7\Memory\Committed Bytes	\\Win7\Network Adapter(Network Adapter)\Bytes Total/sec	\\Win7\PhysicalDisk(0 C:)\Disk Read Bytes/sec
2	34:54.1	3310661632	1195.716736	11531.64825
3	35:09.1	3307077632	1829.222542	2184.40103
4	35:24.1	3318734848	29287.10973	22752.72172
5	35:39.1	3321143296	604.2023803	404211.2592
6	35:54.1	3323736064	490.9373652	6280.584913
7	36:09.1	3312582656	525.7930616	2183.673896
8	36:24.1	3317223424	466.2654823	4369.680344
9	36:39.1	3317805056	634.9735711	546.2826606

Column `A` displays a time reference showing that the data was gathered in 15-second intervals, as configured in the previous section. Row `1` displays the counter names, which are arranged by default in alphabetical order.

The following table shows the average value of each of the **Performance Monitor** counters from our sample desktop. To make the results easier to read, the data recorded in Bytes was converted to Megabytes.

Performance Monitor Counter	Average Value
Memory Committed Megabytes per second	2,443.4 Megabytes
Network Total Megabytes per second	0.75 Megabytes
Disk Reads per second	7.25 Reads
Disk Read Megabytes per second	0.109 Megabytes
Disk Writes per second	10.09 Writes
Disk Write Megabytes per second	0.120 Megabytes
% Processor Time	13.80 percent Processor Time

This data provides the starting point for determining the amount of resources we need to provide for each virtual desktop, and by extension how many desktops we can run on each vSphere host.

Storage for our virtual desktops can be provided using a number of different solutions that include both server-based (local) storage, and shared storage arrays. The Performance Monitor data we have collected includes counters for the number of Disk Reads and Disk Writes per second, which is the basis for properly sizing whichever storage solution we plan to use.

Regardless of which storage protocol your vSphere hosts uses, there will be some overhead involved. After you have measured your baseline disk bandwidth (Disk Read or Write Megabytes per second) or IO (Disk Reads or Writes per second) from your reference desktop, add 15 percent to the value recorded prior to calculating your overall resource requirements. The sample calculations in this chapter involving Disk Reads, Disk Writes, Disk Read Megabytes per second, and Disk Write Megabytes per second assume that you have already added the 15 percent overhead.

Server processor configurations are a good starting point for determining how many desktops we can run per vSphere server. While most server types can accommodate a number of different memory configurations, they support a fixed number of processors, and each of those processors comes with a specific number of CPU cores. For the purpose of this exercise, we will assume that we have existing servers that we want to use for our View infrastructure.

Server Resource	Quantity
Physical Processor Count	2
Cores Per Processor	8
Memory	144 GB
Network Interfaces	10 GB—2 interfaces
Fiber Channel Interface	4 GB (800 MB)—2 interfaces

Using these specifications, we can determine exactly how many View desktops we should be able to host on this server. The goal is to determine which resource is the limiting factor, based on the average values obtained during our **Performance Monitor** data collection. To determine the number of desktops supported, we divide the aggregate quantity of server resources by the average usage of that resource as determined by our analysis of the performance monitor data. View supports up to 16 virtual desktop CPUs per physical processor core, but your own environment may support less based on the average desktop CPU utilization.

With regard to virtual desktop memory, it is important to remember that the amount of memory we assign to the desktop should be at least 25 percent higher than the average value obtained from our Performance Monitor Memory Committed Megabytes counter data. The reason for this is that we want to prevent the desktop from having to utilize the Windows paging file during spikes in memory utilization, which would increase the amount of I/O that the storage infrastructure must service, and potentially impact virtual desktop performance.

Storage technologies such as all-flash storage arrays and flash storage installed directly within the vSphere hosts can lessen the performance impact of virtual desktop memory swapping. Just be sure to assign the virtual desktops the minimum memory required by the OS vendor to ensure that your configuration will be supported.

The previous table shows our four core resources: server processor power measured in number of cores, server memory, server network bandwidth, and storage network bandwidth. To calculate the number of desktops supported, we use the following calculations:

- **Processor**: (Number of server cores * 100) / % Processor Time of reference desktop:
    - (16 * 100) / 13.8 = 115.94 desktops
- **Memory**: Total server memory in MB / (Memory Committed MB per second of reference desktop * 1.25):
    - 1,47,456 / (2,443.4 * 1.25) = 48.28 desktops

The value obtained when you multiply the desktop Memory Committed MB per second times 1.25 (2,443.4 * 1.25 = 3,054.25 MB) indicates that each desktop requires 3 GB of memory, which should provide sufficient free memory, and in turn reduce likelihood of having to use the Windows paging file.

- **Network**: Total server network bandwidth in MB / Network Total MB per second of reference desktop:
    - 2,560 / 0.75 = 3,413 desktops.

Remember to convert the network adapter line speeds from megabit to megabyte to match the output format of the **Performance Monitor** data. The following formula is used to perform the conversion:

Value in megabits / 8 = Value in megabytes

- **Storage Network**: Total server storage network bandwidth in MB / (Disk Read MB per second + Disk Write MB per second) of the reference desktop:
    - 1,600 / (0.109 + 0.120) = 6,987 desktops.

These calculations assume that we are using a dedicated storage network to connect our vSphere servers to a storage array, in this case Fibre Channel. In the event that our storage network will utilize the same network connections as our virtual machine traffic, we will need to combine both the values observed for Network Total MB per second, Disk Read MB per second, and Disk Write MB per second when determining how many desktops our vSphere host can accommodate.

Using the numbers from the previous example, we will calculate the maximum number of desktops the server could host, assuming the server has only the two 10 GB connections for all network traffic, and no Fibre Channel storage network exists:

- **Network**: Total server network bandwidth in MB / (Network Total MB per second of reference desktop + Disk Read MB per second + Disk Write MB per second) of the reference desktop:
    - 2,560 / (0.75 + 0.109 + 0.120) = 2,614 desktops

To determine the minimum specifications for the storage solution we will use to host our virtual desktops, we need to take the average number of Disk Reads and Writes per second from our Performance Monitor data and multiply that number by the number of desktops we wish to host. The following calculation shows an example of how we would calculate the required I/O per second, also known as IOPS, that our storage solution is required to service:

- Data used for calculations:
    - Performance Monitor Disk Reads per second: 7.25
    - Performance Monitor Disk Writes per second: 10.09
    - Number of desktops to size the storage solution for: 500
    - Average IOPS for one 15K RPM SAS disk drive: 175 IOPS
- (Disk Reads per second + Disk Writes per second) * Total number of desktops = Total IOPS required by the virtual desktop storage solution.
    - (7.25 + 10.09) * 500 = 8,670 IOPS

- Our calculations tell us that our storage array will need to service at least 8,670 IOPS, which would require at least fifty 175 IOPS, 15K RPM SAS disk drives. Our calculations are based on the raw IOPS capabilities of the disk drives, and do not take into account the overhead required to implement a **redundant array of inexpensive disks** (**RAID**) using a large quantity of disks. The actual number of disks required to service 8,670 IOPS will be more than fifty; how much more is dependent on the architecture of our storage array.

The following table summarizes the number of desktops that the server could host, based on the current hardware configuration. The server referenced in this table has distinct networks for storage and virtual machine networking.

Server Total Resources	**Desktop Average Utilization**	**Number of View Desktops Supported**
16 processor cores	13.80 percent (of one core)	115.94
144 GB memory	2,443.4 MB	48.28
20 GB network bandwidth	0.75 MB	3,413
8 GB storage network bandwidth	0.23 MB	35,772

Based on the data that was gathered from the **Performance Monitor** session and the specifications of the servers, we can host a maximum of 60 desktops on these servers as they are currently configured. As the table indicates, our limiting resource is server memory. If the server supported it, and we wanted to maximize the number of virtual desktops the server could host, we could increase the amount of memory in the server and host up to 115 desktops, which is the maximum supported based on the processor configuration. To determine how much memory would be required, we would apply the calculation used earlier, in reverse:

- Memory Committed MB per second of reference desktop * 1.25 * Number of desktops we want to host = Amount of memory required in MB:
    - 2,443.4 MB * 1.25 * 115 = 351,239 MB = 343 GB

In this example, if we were to increase the memory in our server to at least 343 GB, we could host 115 desktops.

In both the examples, the number of desktops that the servers could support, based on network bandwidth, may seem rather high. One of the reasons for this is that we are basing our calculated maximum on the combined capacity of our two 10 GB connections, which not every server may have. The second, and more common reason, is that our **Performance Monitor** data did not generate a significant amount of disk or network traffic. This is why it is critical that we gather **Performance Monitor** data from as many desktops as is feasible, across multiple job or user types within our organization.

The resources required by the virtual desktops are the most important part of determining vSphere capacity, but not the only factor we must take into consideration. The next section will discuss how virtual machine overhead and planning for vSphere failure or maintenance affects our sizing calculations.

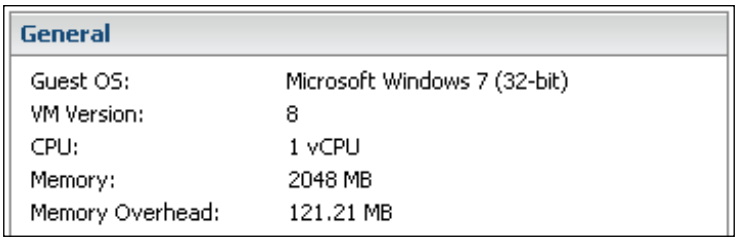
Virtual Desktop overhead and vSphere reserve capacity

When determining the absolute maximum number of desktops we can host on a given vSphere host, we must take into account topics such as virtual machine overhead, accommodating vSphere host failures or maintenance, and other similar factors.

Calculating virtual machine overhead

vSphere requires a certain amount of memory and a small amount of CPU resources to manage the hosted virtual machines, including the ability to power them on. The amount of resources required for overhead is typically minimal compared to the resources required by the virtual machines themselves, but it is important not to determine capacity solely by using the calculations from the previous section.

The vSphere console can provide an estimate of the expected amount of memory overhead required for a given virtual machine. The following screenshot shows where the memory overhead is displayed in the **Summary** tab of the virtual machine properties:



The following table shows the expected amount of memory required to support a virtual machine of several different memory and processor configurations. This information is useful for determining how much memory should be available on a given host in order to properly manage the guest virtual machines.

Virtual Machine Memory	1 vCPU	2 vCPUs
1024 MB	101.06 MB	123.54 MB
2048 MB	121.21 MB	146.71 MB
3096 MB	141.35 MB	169.88 MB
4096 MB	161.50 MB	193.04 MB

The overhead associated with an individual virtual machine is subject to change during the operation of the virtual machine. There is no definitive way to calculate what this overhead will be, but these figures are considered a reasonable estimation.

The figures listed in the previous table represent not only the per virtual machine memory overhead, but also the amount of memory that must be available in order to power on the virtual machine. If our vSphere host refuses to power on a virtual machine, lack of available memory is the likely cause.

The calculations we performed in the previous section revealed that we should assign 3 GB of memory to each of our sample Virtual Desktops. For the purpose of this calculation, we will assume that these Virtual Desktops require only one **virtual CPU** (**vCPU**), which is common for all but the heaviest of Power User desktops. Based on the vCPU and memory configuration, each Virtual Desktop would require an additional 141.35 MB of memory to be available for vSphere overhead, which would require another 16 GB be added to our previous server configuration, bringing the total to 359 GB of memory required to host 115 desktops.

These examples assume that we intend to operate our vSphere hosts at 100 percent capacity at all times. In many cases, including the one we will discuss next, this will not necessarily be the case. Just remember that operating any individual component of our View infrastructure at 100 percent capacity can lead to problems if our initial design does not take into account all possible contingencies, including spikes in usage, hardware failure, and other similar events.

The need for vSphere reserve capacity

A second reason to not fully populate a vSphere host is to be able to accommodate all the desktops in the event of a vSphere host failure or host maintenance operation. Consider a vSphere cluster with eight vSphere servers hosting 128 desktops each (1024 total desktops):

- Desktop requirements:

Desktop requirements will vary from one environment to the next; these figures are just an example.

    - Each single vCPU desktop requires 10 percent of one vSphere server CPU core (average % Processor Time)
    - Each desktop requires 2,048 MB of memory (average Memory Committed Megabytes)

- Eight vSphere hosts, each running 12.5 percent of the total number of Virtual Desktops:
    - 1024 desktops / 8 vSphere hosts = 128 desktops per host
- To continue to run all the desktops in the event one vSphere host were to become unavailable, we would need to be able to accommodate 18.29 desktops on each of the remaining seven hosts.
    - 128 desktops / 7 remaining vSphere hosts = 18.29 desktops per each vSphere host
- To continue to run all desktops without any degradation in the quality of service, each server needs to have an excess of capacity that is sufficient to host 18-19 desktops. This is:
    - 19 desktops * 10% of a CPU core = 1.9 available CPU cores required
    - 19 desktops * 2,048 MB of memory = 38,912 MB or 38 GB of available memory required
    - 19 desktops * 121.21 MB of memory for virtual machine overhead = 2303 MB or 2.3 GB of additional available memory required
    - 19 desktops * 0.75 MB network bandwidth = 14.25 MB of available network bandwidth required
    - 19 desktops * 0.23 MB storage network bandwidth = 4.37 MB of available storage network bandwidth required

These calculations assume that we want to protect the ability to provide resources for 100 percent of our desktops at all times, which is a very conservative yet valid approach to building a View infrastructure.

The final configuration of the vSphere servers should take into account not only what percentage of desktops are actually in use at a given time, but also the cost of purchasing the additional capacity needed to support vSphere host failures or other events that require downtime.

Always take into consideration the growth of our View environment. Purchasing equipment that has limited ability to scale may save money today, but could cost you dearly when we need to expand. If a piece of equipment you plan to buy for your View infrastructure just barely meets your needs, look into the next larger model or even a competing product, if necessary.

View Client network bandwidth requirements

One of the easiest things to overlook when designing our View infrastructure is how much network bandwidth is required to support the View Client connections. The preferred protocol for VMware Horizon View is **PC-over-IP**, commonly known as **PCoIP**. View also supports the use of **Microsoft Remote Desktop Protocol** (**RDP**).

PCoIP is a display protocol provided by VMware for use in the View product suite. The PCoIP protocol has multiple features that make it ideal for connecting to View desktops:

- Capable of adapting to varying levels of connection latency and bandwidth
- Has multiple built-in techniques for optimizing and accelerating connections over a **wide area network** (**WAN**)
- Able to achieve compression ratios of up to 100:1 for images and audio
- Uses multiple codecs that enable more efficient encoding and decoding of content between the Virtual Desktop and the remote Client
- Based on **User Datagram Protocol** (**UDP**), which eliminates the need for latency-inducing handshakes used in **Transmission Control Protocol** (**TCP**)-based display protocols

Microsoft RDP is a TCP-based display protocol that does not have many of the WAN optimization and acceleration techniques that are found in PCoIP. In addition, VMware Horizon View includes Microsoft GPO templates that enable a very granular control over PCoIP connection characteristics. *Chapter 13*, *Implementing VMware Horizon View Group Policies*, provides information about how to use the View GPO templates to control the settings of the PCoIP connections.

Client bandwidth estimates

The VMware Horizon View Architecture Planning guide (`http://www.vmware.com/support/pubs/view_pubs.html`) provides estimates for PCoIP bandwidth utilization based on the application workload of the client. The following table is built upon that information:

User type	Workload characteristics	Bandwidth in Kbps
Task Worker	2D display and single monitor. Web and limited Office applications.	50-100 Kbps
Knowledge Worker (2D)	2D display and single monitor. Office Applications.	100-150 Kbps
Knowledge Worker (3D)	3D display (Windows Aero) and multiple monitors. Office Applications.	400-600 Kbps
Knowledge Worker (3D)—High Use	3D display (Windows Aero) and multiple monitors. Office Applications. Frequent display changes.	500 Kbps-1 Mbps
Power User	3D display (Windows Aero) and multiple monitors. 480P video and images frequent screen changes.	2 Mbps

Bandwidth utilization is heavily dependent on a number of factors, many of which can be controlled with the View PCoIP GPO settings or even Windows Virtual Desktop settings. Actual bandwidth utilization will vary based on usage and PCoIP settings.

The PCoIP protocol was invented by a company called Teradici. For additional information about how the protocol works, visit the Teradici PCoIP technology page (`http://www.teradici.com/pcoip-technology.php`).

Even with a careful analysis of user desktop usage patterns, it is important to remember that there will be spikes in usage from time to time. A Knowledge or Task Worker who has a need to use an application with a large amount of screen changes, such as viewing images in succession or watching a video, may cause a brief bandwidth spike of between 500 Kbps and 1 Mbps or more. Preparing for these spikes in bandwidth utilization is important in order to preserve the quality of service for all of the View Client connections.

Refer to *Chapter 11*, *Creating a Master Virtual Desktop Image* for information about optimal settings for a Windows desktop, and *Chapter 13*, *Implementing VMware Horizon View Group Policies* for information about how to configure PCoIP settings to reduce bandwidth needs. The information contained in both of these chapters is critical to optimizing View Client connections and Virtual Desktop performance in general.

The importance of a VMware Horizon View pilot

Up until now, this chapter has been about introducing us to a variety of different concepts that form the basis of architecting our View infrastructure. If we learn anything from this chapter, it is that our goal is to obtain the resources we need to provide an acceptable end user computing experience.

Classifying our end users and measuring their resource requirements is a valuable exercise that will help us understand what will be required to transition our end user computing resources from the desktop to the datacenter. That being said, no amount of planning can possibly replace a properly run pilot that validates not only the configuration of our master Virtual Desktop image, but also the performance of the View infrastructure and the quality of the experience from an end user perspective.

Our View pilot should involve the same types of users as our user analysis did, but not necessarily the same users within each group. The following list includes a number of goals that our View pilot should attempt to achieve:

- Include multiple users from each user classification: Task Worker, Knowledge Worker, and Power User
- Include fully remote users as well as WAN-connected users at other company sites

- Perform additional performance analysis at all layers of the View infrastructure including:
    - Storage
    - Network
    - vSphere host
    - Guest operating system
- Measure the impact of common View scenarios, such as:
    - User logon storms: Large numbers of users logging on within a short time frame.
    - Steady state user load: Measure View infrastructure performance during a period of steady desktop usage by a significant number of users.
    - Antivirus platform performance: Measure the impact of common antivirus platform tasks, such as on demand scans and pattern file updates.
    - View refresh or recompose: Measure the impact of these common View linked clone desktop maintenance operations, described in detail in *Chapter 10*, *Performing View Desktop Maintenance*.
    - A fully populated vSphere host: Measure host performance with higher than normal workloads, such as simulating an outage or an other period of higher than usual utilization.

Performance deficiencies at any layer of the View infrastructure can lead to a poor end user experience, usually in the form of longer than anticipated application response times. This is why it is critical to involve a large cross-section of our users in the pilot process, and to seek their opinion throughout the program.

The performance data that we collect during the pilot program can be used to measure the average of the actual resource utilization, which can then be compared to the estimated average resource utilization from the initial physical desktop analysis. Ideally, the numbers would be rather close to one another, but if they are not we will want to work to identify the cause. Now that we can measure performance at all layers of the View infrastructure, it should be easy to determine where the higher than expected utilization originates from. Some potential issues to look for include:

- The earlier analysis of the users did not include a sufficient number or a wide enough cross-section of users.

- The Virtual Desktop master image was not properly optimized. Refer to *Chapter 11*, *Creating a Master Virtual Desktop Image* for details on how to optimize the master desktop image.
- A component of the View infrastructure was improperly configured. This problem can affect any number of components of the infrastructure.
- The pilot program is occurring during a period of higher than normal user workload, for example a recurring event unique to the organization such as financial reporting.

In summary, the View pilot is your best time to learn about how View will perform within your environment, both from a performance perspective and in terms of user acceptance. Use the pilot program to identity any potential barriers to a successful rollout, and make any changes that are needed in order to minimize the risk of failure as the project moves forward.

Summary

In this chapter, we have been introduced to the different components that comprise a VMware Horizon View infrastructure, including the licensing and core infrastructure requirements. Later chapters will discuss how to install and configure each of these components.

We have also been introduced to the basics of what level of research is required even before the first virtual desktop is deployed, including assessing our existing physical desktops, calculating bandwidth requirements for remote users, and adjusting our design to accommodate vSphere host maintenance or failure.

We concluded this chapter by learning the basics of running a View pilot, which is critical as it will either validate or invalidate all of the research that we did in the early phases of our View design.

In the next chapter, we will begin the installation of our VMware Horizon View infrastructure, beginning with the View Connection Server.

2
Implementing VMware Horizon View Connection Server

The VMware Horizon View Connection Server is a key component of the VMware Horizon View infrastructure. In addition to acting as a connection broker between View clients and View desktops or Microsoft Windows Terminal Servers, it also hosts the View Manager Admin console.

This chapter will discuss multiple topics surrounding the design, deployment, configuration, backup, and recovery of a View Connection Server. We will also discuss how to install additional View Connection Servers, which are known as replicas.

In this chapter we will learn:

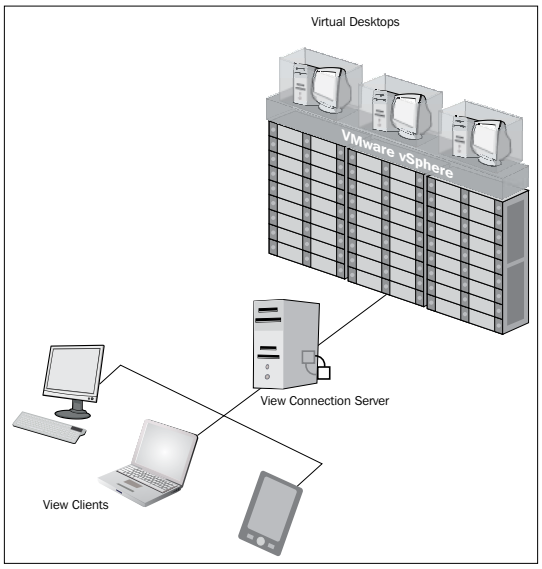
- An overview of View Connection Server
- The hardware and operating system requirements of View Connection Server
- The connection limits of a View Connection Server
- How to determine the number of View Connection Servers required
- The vCenter requirements of a View infrastructure
- View Connection Server network protocol and port usage
- View Connection Server pre-installation tasks
- How to install the first View Connection Server
- The initial configuration of a View Connection Server
- How to install additional View Replica Connection Servers
- How to back up components of the View Connection Server and which components to back up
- How to restore a View Connection Server from backups

Overview of VMware Horizon View Connection Server

In *Chapter 1*, *Designing a VMware Horizon View Infrastructure*, we discussed many roles of the View Connection Server. These roles include:

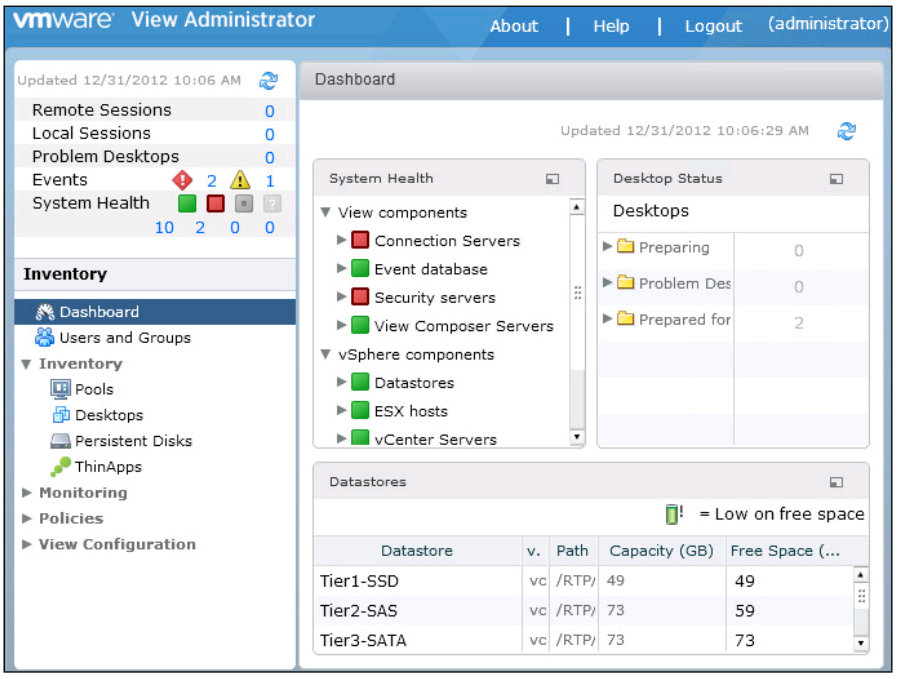
- Managing connections between View clients and View-managed resources including View desktops and Microsoft Windows Terminal Servers
- Authenticating user connection requests and providing access to assigned resources
- Hosting the View Manager Admin console for the VMware Horizon View infrastructure
- Working in tandem with VMware vCenter and View Composer to manage, deploy, and maintain virtual desktops

The following diagram shows the placement of the View Connection Server in a simple View environment. For now, only the View Connection Server is displayed; later chapters will add additional View Servers to this diagram.



The View Connection Server authenticates the clients and provides them with access to the appropriate resources. Depending on the configuration of the View Connection Server, the clients may connect directly to the remote resource, or the connection may be tunneled through the View Connection Server. The behavior of a View client connection will be discussed further in this chapter, and again in *Chapter 5, Implementing VMware Horizon View Security Server*, when discussing the View Security Server.

The View Connection Server also hosts the VMware Horizon View Manager Admin console, the central management point for View resources. The following screenshot shows the dashboard presented to a View administrator upon login:



While virtual desktops appear in VMware vCenter in the same format as a typical virtual machine, it is important that all changes to those virtual desktops are done only through the View Manager Admin console. View maintains configuration information about the virtual desktops within a **Microsoft Active Directory Lightweight Directory Services** (**MS LDS**) database installed on each View Connection Server, and any changes made to those desktops within the vCenter console may lead to problems that prevent the desktop from being managed by the View or accepting client connections.

VMware Horizon View Connection Server requirements

Like many other software services, View Connection Server requires a minimum server configuration to ensure adequate performance. To properly support View Connection Server, the VMware vCenter Server must also be assigned adequate resources and configured appropriately, based on the projected size of the View infrastructure.

Hardware requirements

The View Connection Server software has specific requirements with regard to the hardware specifications and host operating system. In addition, all View Connection Servers, regardless of the type, should be installed on a dedicated virtual or physical server.

The first View Connection Server you install will be referred to as a View Standard Server during the installation process. To meet scalability and high availability needs, up to six additional View Connection Servers can be installed per View deployment; each of these is referred to as a View Replica Server during the installation process. The limits of a View Connection Server are discussed later in this section.

The following table outlines both the minimum and recommended hardware specifications of a View Connection Server. The same specifications apply to both View Replica Servers and View Security Servers.

Hardware component	Minimum requirement	Recommended
Processor	Pentium IV 2.0GHz or higher	4 CPUs
Memory	4 GB	10 GB minimum for deployments of 50 or more desktops
Networking	One or more 10/100 MBps **network interface cards** (**NICs**)	1 GBps NICs
Hard disk capacity	40 GB	40 GB

Like most other software platforms, the recommended guidelines should be followed to ensure that the View Connection Server performs optimally.

Software requirements

Each of the four different View Connection Server types, which include Standard, Replica, Security, and Transfer, support only Windows Server 2008 R2 64-bit as the host operating system. Both Standard and Enterprise editions of Windows Server 2008 R2 are supported, with or without SP1 installed.

Limits of a View Connection Server

The limits of a View Connection Server are defined by the maximum number of simultaneous client connections it can support, which varies based on what type of View-managed resource is being accessed. The following table shows the limitations of a single View Connection Server for each of the different types of connections a View Connection Server can broker, including View virtual desktops, physical PCs, and Microsoft Windows Terminal Servers.

Connection type	Maximum simultaneous connections
To virtual desktop:	2,000
PCoIP or RDP direct connection	
RDP tunnelled connection	
PCoIP tunnelled connection through View Security Server	
Unified Access to physical desktop PCs	100
Unified Access to Microsoft Windows Terminal Server	200

The term "Unified Access" refers to View-brokered connections to resources other than View-deployed virtual desktops.

Each View installation supports a maximum of seven View Connection Servers in a 5+2 configuration, supporting a maximum of 10,000 desktops. This means that five of the servers are considered active, and two are held in reserve as spares. The spare View Connection Servers maintain an up-to-date copy of the View AD LDS database so that they can serve clients immediately in the event they are needed.

Not every View environment will require seven View Connection Servers but at the very least they should have two. Even if the number of desktops to be deployed is less than 2,000, having two View Connection Servers ensures that sufficient capacity exists in the event of a server outage or maintenance event. Always build in extra capacity to accommodate events that can impact availability.

In environments where more than 10,000 virtual desktops are required, you will be required to deploy a second Standard Connection Server. This will create a unique View AD LDS database, rather than a replica of the one in your first View installation, enabling support for additional View Connection Servers and an additional 10,000 desktops.

Regardless of whether you choose to build one large View environment or multiple small ones, each is typically referred to as a **pod**. The term pod implies that the View environment is managed as one entity, based on the limitations of the Connection Server. A single-pod design is suitable for most smaller single-site View environments as it is the easiest to manage. If you plan to install View Connection Servers in multiple sites, it is recommended you create a pod in each site, due to the amount of communication traffic generated by Connection Server replication.

Load balancing Connection Servers

VMware Horizon View provides no native method to load balance client requests to View Connection Servers. It is recommended you implement some sort of load balancing method to help balance the client connections across all the View Connection Servers in your infrastructure. Different methods that can be used to load balance View Connection Server requests include **Round Robin DNS**, Microsoft Windows **Network Load Balancing** (**NLB**), and a physical or virtual load-balancing appliance. The following three sections explain the benefits and shortcomings of each of these options.

Round Robin DNS

Round Robin DNS provides View clients with a rotating list of IP addresses, one for each Connection Server, when they perform a DNS lookup of the Connection Server's **fully qualified domain name** (**FQDN**).

While simple to configure, Round Robin DNS has multiple shortcomings that make it a less than optimal solution:

- It does not alter behavior based on server availability, meaning that, if a server is down, its IP address will still be provided to clients and they will not be able to connect.
- Client caching of DNS records means that clients will connect directly to the most recent View Connection Server they received an IP address for, rather than the optimal IP as determined by the DNS server. This results in a View client load imbalance.

Refer to the documentation for your DNS server software for information on how to configure Round Robin DNS.

Windows Server 2008 R2 Domain Controllers that run DNS have Round Robin DNS enabled by default. Create multiple host records with the same hostname and different IP addresses, and the DNS server will automatically balance the traffic across each.

Microsoft Windows Network Load Balancing

Microsoft NLB is a feature of the Windows Server operating system that enables native server clustering. Microsoft NLB clusters distribute traffic evenly among the cluster members and redirect connections as needed in the event a cluster member becomes unavailable.

While effective, implementing Microsoft NLB clusters does require a certain amount of resources on each cluster member. In addition, using a dedicated load-balancing appliance may offer features that native Microsoft NLB clusters cannot.

Refer to the VMware document *Implementing Microsoft Network Load Balancing in a Virtualized Environment* (`http://www.vmware.com/files/pdf/implmenting_ms_network_load_balancing.pdf`) for information about how to implement Microsoft NLB when using virtual servers.

Load-balancing appliances

Dedicated load-balancing appliances are available both in physical and virtual formats. The difference between the physical and virtual versions will differ from one vendor to another, so it is important to understand the impact of choosing one over the other.

Dedicated load balancers often have the capability to balance client connections based on a number of different factors. The following load-balancing metrics are not from any specific vendor but are simply examples of the metrics that various solutions use:

- Server load
- Recent server response times
- Server up/down status
- Number of active connections
- Geographic location
- How much traffic has been assigned to a given host

In addition to any advanced features, dedicated load-balancing appliances do not require additional resources on the servers they are balancing traffic for, which is one additional advantage they have over using Microsoft NLB.

Not all dedicated load-balancing solutions are commercial products. Numerous open source products exist as either a dedicated virtual appliance or as a software package that is installed within an existing operating system. You will need to research all of your options when determining what the ideal solution is for your environment.

VMware provides no guidance as to when one load-balancing solution should be chosen over another. The decision rests with the View architect and other involved parties within an organization, who must weigh the costs of a dedicated commercial load-balancing solution with alternatives from the open source community and native features such as Microsoft NLB and Round Robin DNS.

vCenter Server requirements

You should use a dedicated VMware vCenter Server for VMware Horizon View wherever possible, particularly if you are using View Composer to deploy linked-clone desktops. This will ensure that your vCenter Server is configured based on the exact requirements of your View infrastructure, ensuring optimal performance and providing maximum flexibility when future upgrades or updates are required.

Using a dedicated vCenter Server for the View provides a number of benefits over using an existing vCenter Server. These benefits include:

- A new View deployment that plans to use an existing vCenter Server may require a version of vCenter that is currently not in use, necessitating an upgrade that may not be possible based on environmental or licensing constraints.
- If a View upgrade or patch requires an accompanying vCenter Server upgrade or patch, this operation will not affect the existing vCenter server.
- The existing vCenter Server may not be properly sized to handle the planned number of virtual desktops, and may require changes to CPU, memory, the operating system version, or vCenter Server settings.
- For organizations that plan to deploy linked-clone desktops using View Composer, the vCenter Server will be placed under a significant load during various Composer Operations. Isolating these operations to a dedicated vCenter Server ensures that this load does not affect the management of other, non-View related, vSphere hosts.

Should you still choose to leverage an existing vCenter Server, deploying VMware Horizon View requires no specific changes, assuming that the vCenter was appropriately sized based on the number of virtual machines and vSphere hosts it will manage. This section will outline those vCenter options, which may need to be updated during the installation process.

The installation and configuration of VMware vCenter can proceed as outlined in the *VMware vSphere Installation and Setup Guide* (`http://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-pubs.html`). The following options, presented during the vCenter installation process, should be configured as required, based on the number of virtual desktops the vCenter Server will manage. The best time to configure these settings is during the vCenter installation process, so consider future virtual desktop deployments wherever possible when performing the installation. The vCenter Server installation options that directly apply to View include:

- **Ephemeral ports**: Increase the number of ephemeral ports if the vCenter Server will ever power on more than 2,000 virtual machines simultaneously. VMware currently supports a maximum of 10,000 desktops per vCenter host, so increase this number if required.

- **Inventory size**: The number of virtual machines the vCenter Server will host affects the JVM heap settings for the services that include the VMware VirtualCenter Management Webservices, VMware vCenter Inventory Service, and VMware vSphere Profile-Drive Storage Service.
    - **Small**: Fewer than 100 vSphere hosts or 1,000 virtual machines
    - **Medium**: 100 to 400 vSphere hosts or 1,000 to 4,000 virtual machines
    - **Large**: More than 400 vSphere hosts or 4,000 virtual machines

In addition to these settings, the vCenter Server should have the recommended amount of CPUs and memory, as outlined in the *VMware vSphere Installation and Setup Guide*. These requirements vary based on which vCenter Server components are installed, as well as the number of virtual desktops and vSphere hosts the vCenter server will manage. View can utilize multiple vCenter Servers for managing desktops, although this is not required as a single vCenter Server can manage up to the per-pod limit of 10,000 desktops.

View Connection Server networking

VMware Horizon View utilizes multiple ports in order to broker a connection between a View client and a View desktop or Microsoft Windows Terminal Server. In addition, how the connection is made from the View client to the desktop or terminal server differs based on the tunneling configuration of the View Connection Server.

Tunneling versus direct client connections

The default configuration of a View Connection Server treats PCoIP and RDP connections differently:

- Upon successful login, View client connections that utilize PCoIP will connect directly to the View desktop
- Upon successful login, View client connections that utilize RDP will be tunneled through the View Connection Server to the View desktop or terminal server

There are multiple reasons why you should tunnel internal View client RDP connections through a View Connection Server—the default setting—even in cases where those connections are made internally.

- Tunneled RDP connections will be secured using SSL encryption

- Clients can access multiple View desktops or terminal services over a single HTTPS connection, which requires less protocol overhead compared to using multiple RDP connections
- The View Connection Server manages the HTTPS connection, increasing reliability by automatically re-establishing View client connections and the RDP session in the event of an interruption.

PCoIP offers similar benefits without the need to tunnel the internal View client connections through the View Connection Server. These direct client connections support the features that include:

- The use of third-party VPN clients
- Session reliability features that can automatically re-establish client connections after an interruption
- PCoIP connections are encrypted using **Advanced Encryption Standard** (**AES**) by default
- Hardware-based implementations of PCoIP utilize both AES and **IP Security** (**IPsec**) encryption

While the features of a direct PCoIP connection likely meet the need of most View environments, it is possible that some organizations will wish to implement even more strict security for internal PCoIP connections. In cases such as this, you would simply enable tunneling for internal PCoIP connections.

Connections that are made through a View Security Server are required to be tunnelled. *Chapter 5*, *Implementing VMware Horizon View Security Server* will discuss the usage and configuration of a View Security Server, and how it differs from a design that utilizes only View Connection Servers.

Introduction to View communication protocols

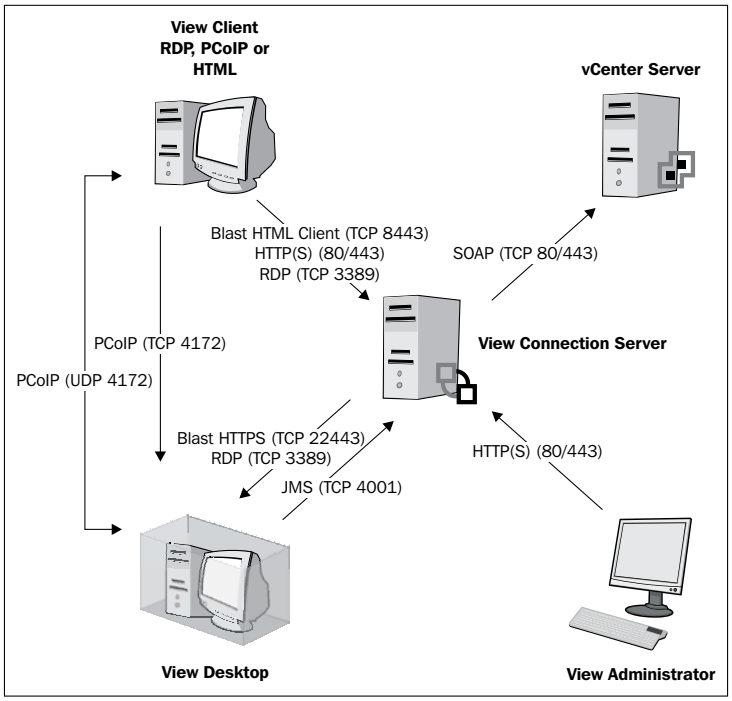
The following diagram illustrates how the primary protocols used by the View work when configured using the default settings. The diagram shows the following components of a View infrastructure:

- A tunneled RDP client connection
- A tunneled HTML connection
- A direct PCoIP client connection

- Communication between the View Connection Server and the vCenter Server
- Communication between the View Administrator accessing the View Manager Admin console on the View Connection Server

VMware Horizon View Feature Pack1, which enables access to desktops over HTML5-capable web browsers, is an optional add-on for View. Consult *Chapter 15*, *VMware Horizon View Feature Pack 1* for information about the installation, configuration, and usage of all features introduced with the release of the Feature Pack..

Additional protocols are used but they are optional, whereas the protocols shown in the diagram are integral to View communications. The additional protocols are described in detail in the table that follows the diagram. The arrows indicate the direction in which each protocol travels, assuming the default settings are used.



The diagram shows only the core components of a View infrastructure; as new components are introduced in later chapters, the placement of those components within the infrastructure will be shown in an updated diagram. These components include:

- View Composer (*Chapter 3*, *Implementing VMware Horizon View Composer*)
- View Transfer Server (*Chapter 4*, *Implementing VMware Horizon View Transfer Server*)
- View Security Server (*Chapter 5*, *Implementing VMware Horizon View Security Server*)

A full list of the protocols and ports that are used within a VMware Horizon View infrastructure is outlined in the following table:

Protocol or service	Port	Notes
AJP13 (Apache Tomcat Connector)	TCP 8009	Used only when a View Security Server is present
HTTP/HTTPS	TCP 80/443	
JMS (Java Messaging Service)	TCP 4001	
HTTP	TCP 80	
HTTPS	TCP 443	
HTTPS	TCP 8443	Used for clients accessing desktops using the Blast HTML protocol
HTTPS	TCP 22443	Used for connections from the Connection Server to the Blast agent on the desktop
MMR (Multi-Media Redirection)	TCP 9427	Used alongside RDP; uses clients rather than server resources to render DirectShow-based media and codecs
PCoIP	TCP/UDP 4172	
RDP	TCP 3389	
SOAP (Simple Object Access Protocol)	TCP 80 or 443	
USB Redirection for PCoIP and RDP	TCP 32111	TCP 32111 is used to support USB redirection to View clients

MMR and USB redirection are only used when the indicated services are required. In instances where they are in use, they work alongside their associated PCoIP or RDP protocols using the same communication path as indicated in the earlier protocol diagram.

Installation and configuration

The installation and configuration process for the first View Connection Server, referred to as a View Standard Server during installation, requires some amount of preparation. This section will outline what is required prior to beginning the installation.

Installation prerequisites

There are a number of prerequisites that should be addressed prior to installing the first VMware Horizon View Connection Server. The previous chapter discussed the infrastructure requirements. These included:

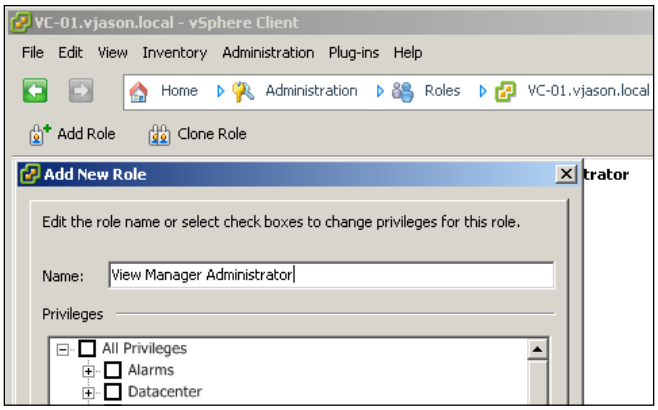
- Windows 2003 or 2008 Active Directory
- An Active Directory user account or security group that will be granted full View administrator access
- A Static IP address for the View Connection Server
- Local administrator access on the host server
- A View license key
- A View event log database

In addition to the items described in *Chapter 1*, *Designing a VMware Horizon View Infrastructure*, the following items should be prepared in advance for the installation.

A vCenter user account

VMware Horizon View requires access to the vCenter Server in order to perform tasks related to the creation or management of virtual desktops. To facilitate this access, we first need to create an AD user account that the View Connection Server will use to access the vCenter Server. To make it easier to change the View Connection Server AD account in the future, we will create a vCenter role that includes all the required privileges. Once created, the role can be quickly applied to AD user accounts. Perform the following steps to create the role in vCenter:

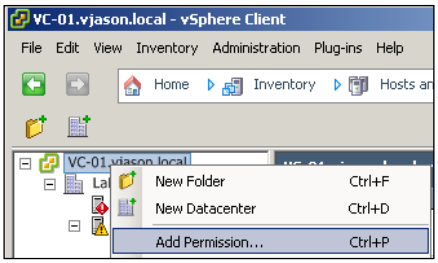
1. In vSphere Client, navigate to **Home** | **Roles** | **Add Role** and enter a role name such as `View Manager Administrator` as shown in the following screenshot:



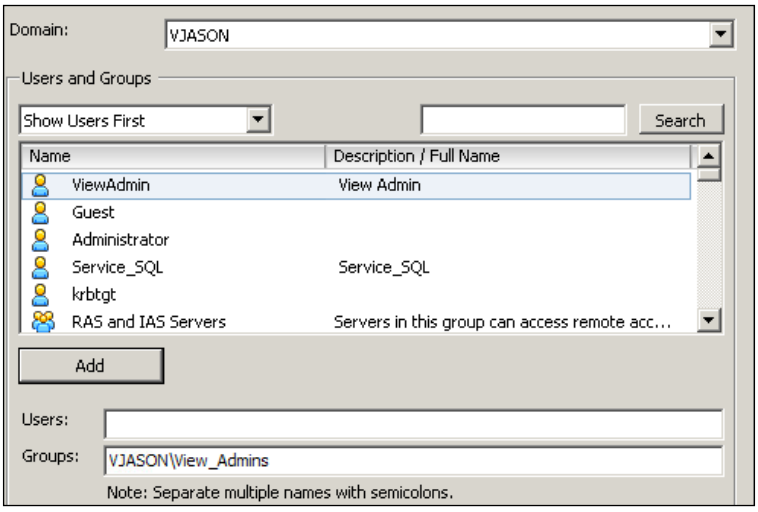
2. From within the **Add New Role** window, expand each privilege group listed in the following table below and check the required privilege items. All listed privileges must be checked in order for the View Connection Server to function properly. Click on **OK** when finished with creating the role.

vCenter privilege group	Privilege subsection	Privilege
Folder		Create a folder
		Delete a folder
Global		Act as vCenter Server
		Only required if vSphere host caching will be used
Resource		Assign virtual machine to resource pool
Virtual Machine	Configuration	Add or remove a device
		This is an advanced privilege
		Modify device settings
	Interaction	Power off
		Power on
		Reset
		Suspend
	Inventory	Create new
		Remove
	Provisioning	Customize
		Deploy template
		Read customization specifications

3. In the **vSphere Client** page, click on **Home** | **Hosts and Clusters**, right-click on the vCenter Server at the top level of the inventory, and click on **Add Permission** as shown in the following screenshot. This will open the **Assign Permissions** window.



4. In the **Assign Permissions** window, click on the **Add** button to open the **Select Users and Groups** window.
5. In the **Domain** drop-down menu, select the **AD** domain that contains the View administrator user or security group. In our example, the domain is named `VJASON`.
6. In the **Users and Groups** list, select either the View Administrator AD account or the View Administrator AD security group, choosing whichever is applicable for your environment. For our sample environment, we have chosen the security group titled **ViewAdmin**. Once selected, click on the **Add** button as shown in the following screenshot. Click on **OK**, and then click on **OK** again to close the **Assign Permissions** window and complete the action.



In addition to the vCenter privileges, the AD account used for vCenter access also requires local administrator rights on each View Connection Server. *Chapter 3*, *Implementing VMware Horizon View Composer* will discuss the additional vCenter and AD permissions required for View Composer, while *Chapter 4*, *Implementing VMware Horizon View Transfer Server* will discuss the additional permissions required for View Transfer Server.

View event database

View Manager Server requires an external database in order to record in-depth details about any events that occur. Without an event database, only minimal information is recorded within the log files on the View Connection Server. It is recommended you create a database, both to retain historical data about View events, as well as to support any troubleshooting that may need to be done.

Chapter 1, *Designing a VMware Horizon View Infrastructure* outlines different database types that can be used to record View Connection Server events. In addition to using a supported database platform, the following database configurations must be done:

- For SQL Server:
    - View requires an account on the database server with SQL Server authentication
    - The View database user account must have permission to read from, write to, and create tables and views within the database
- For Oracle:
    - View requires an Oracle database user account
    - The Oracle database user account must have permission to read from, write to, and create tables, views, triggers, and sequences

The database schema will be installed when the database is configured within the View Manager Admin console.

Deploying the first View Connection Server

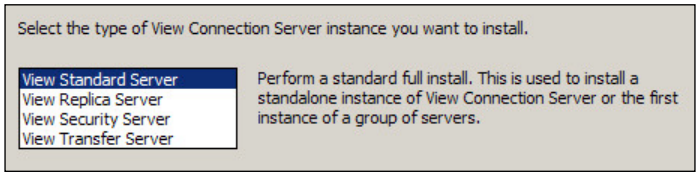
Deploying the first View Connection Server is broken down into two stages; the installation of the Connection Server software and the final setup using the View Manager Admin console.

Installing the first View Connection Server

The View Connection Server software is delivered as a single executable (EXE) file, named in a format similar to `VMware-viewconnectionserver-x86_64-x.x.x-yyyyyy.exe`. This installer is used for all four View Connection Server types, which include Standard, Replica, Transfer, and Security. The following steps outline the installation process:

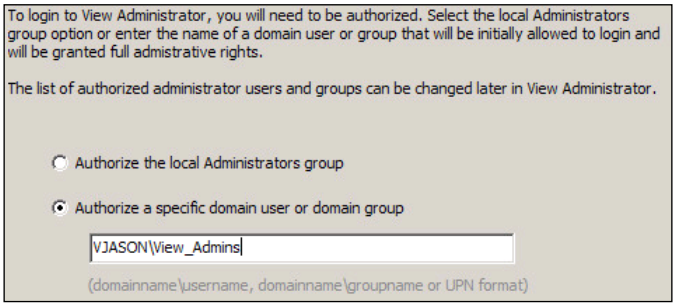
If the View Connection Server has less than 10 GB of RAM, during the installation process the Java heap size of the server will be set to a reduced value, which may cause problems during periods of heavy use. This value cannot be changed without removing and reinstalling the View Connection Server software, so it is recommended you allocate at least 10 GB of RAM to the server prior to installation.

1. Double-click on the View Connection Server installer EXE file to launch the installer.
2. In the **Welcome to the Installation Wizard for VMware View Connection Server** window, click on **Next**.
3. Review the **VMware End User License Agreement** section, select the **I accept the terms in the license agreement** radio button, and click on **Next**.
4. Select the installation directory and click on **Next**.
5. Highlight the View Connection Server instance type. Since this is the first View Connection Server in the environment, we will choose **View Standard Server**, as shown in the following screenshot, and then click on **Next**.



6. Enter a data recovery password and an optional password reminder and click on **Next**. Record this password, as it will be required for any future restores of the **View Connection Server** configuration.
7. Select either the **Configure Windows Firewall automatically** or **Do not configure Windows Firewall** radio button and click on **Next**. If the option **Do not configure Windows Firewall** was selected, configure the firewall manually using the settings provided earlier in this chapter.

8. Select the **Authorize a specific domain user or domain group** radio button, enter either a single AD account or AD security group that is designated as the View Administrator, and click on **Next**. As shown in the example provided in the following screenshot, we will use an AD security group titled `view_admins` in the domain `vjason` in the format `domain-name\group-name`. Click on **Next** when finished with this step.



9. Configure the drop-down menus for the View User Experience Improvement Program, or uncheck the **Participate anonymously in the user experience improvement program** checkbox, and click on **Next**.
10. Review the final installation screen to ensure that the installation directory is correct. If changes are required, click on the **Back** button to reach the necessary configuration screen and make the required changes. Assuming that the settings are correct, click on **Install** to begin the automated installation process
11. Click on **Finish** when prompted at the completion of the installation process.

The installation process will install all the components required for a Standard View Connection Server, including the Connection Server software and the AD LDS database used to store configuration information. The final configuration steps will be completed in the View Manager Admin console, and will be detailed in the next section.

Configuring the first View Connection Server

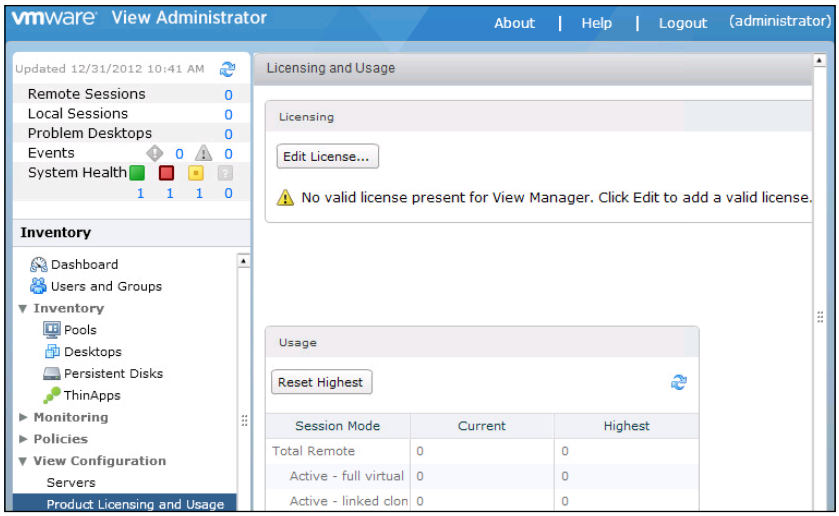
Once the installation of the first View Connection Server has completed, we need to log in to the View Manager Admin Console. The console URL will be in the format `https://Connection Server FQDN/admin`. You must use lowercase letters when typing `admin`.

By default, the installation process creates a self-signed SSL certificate to encrypt connections to the View Connection Server. While the server will function with this default self-signed certificate installed, it is recommended you replace the default certificate with one from a trusted internal or commercial certificate authority. Until the default certificate is replaced, View clients will be notified about the untrusted certificate each time they connect to View. In addition, the View Manager Admin console will display an informational error. *Chapter 12, Managing View SSL Certificates* will provide the process used to replace the default SSL certificates for all View components.

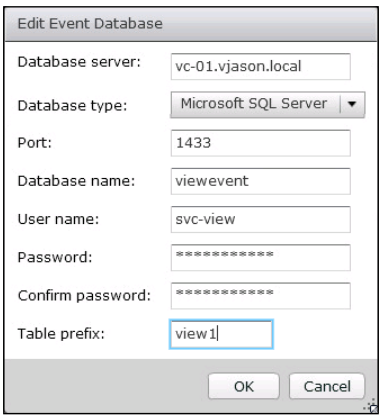
Complete the following steps to configure the first View Connection Server:

1. Log in to the View Manager Admin console using an AD account that was granted administrative permissions during the installation process. The following screenshot shows the login page for our sample View Connection Server, which was accessed at the URL `https://viewmgr-01.vjason.local/admin`.

2. Upon successful login to the first View Connection Server that was installed, the dashboard will open to the **Licensing and Usage** window in the **View Configuration** page as shown in the following screenshot. Click on the **Edit License** button to open the **Edit License** window.



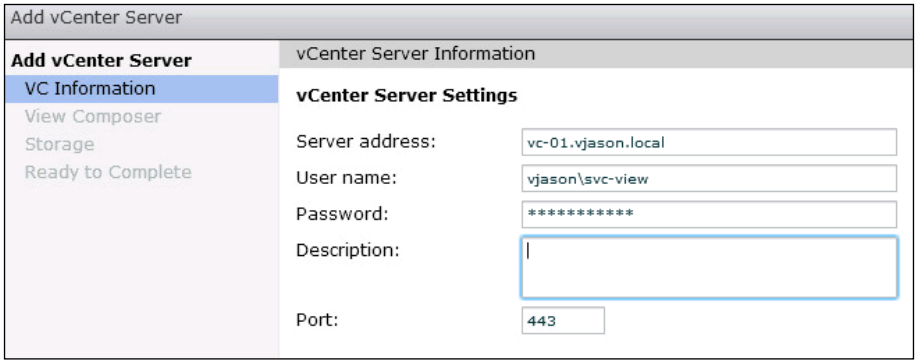
3. Enter the 25-character license key, including the dashes, and click on **OK** to complete the license entry.
4. Open the **Event Configuration** window in the **View Configuration** page within the console. Click on **Edit** in the **Event Configuration** page to open the **Edit Event Database** window.
5. Fill in the information for the View event database configuration including database server name, database type, port, database name, database user name, password, and table prefix. The **Table prefix** field is used to identify this View installation within the database itself. The following screenshot shows the database configuration for our sample server. Click on **OK** once all the information has been provided, and View will complete the database configuration assuming the necessary privileges have been granted as outlined earlier in this chapter.



The console's **Event Configuration** page also allows us to customize how events are displayed in the View Manager Admin console. The *Appendix* provides more information about these custom settings, which can be changed at any time.

6. Go to **View Configuration** | **Servers** to bring up the **vCenter Servers** tab of the **Servers** window. Click on **Add** to open the **Add vCenter Server** window.
7. In the **vCenter Server Information** window, fill in the information required to link the vCenter Server that we will be using for desktops to our View Connection Server. The information required includes the vCenter Server address (FQDN), the View vCenter AD user name that we created earlier in this chapter in the format `domain-name\user-name`, and the password for the account. Click on **Next** when the required information has been provided. You may also link additional vCenter Servers at this time if required. The following screenshot shows the completed screen for our sample server:

The **vCenter Server Information** window also allows us to specify advanced options for increasing the number of concurrent vCenter tasks our View Connection Server can initiate. The *Appendix* provides more information about these advanced settings, which can be changed at any time.

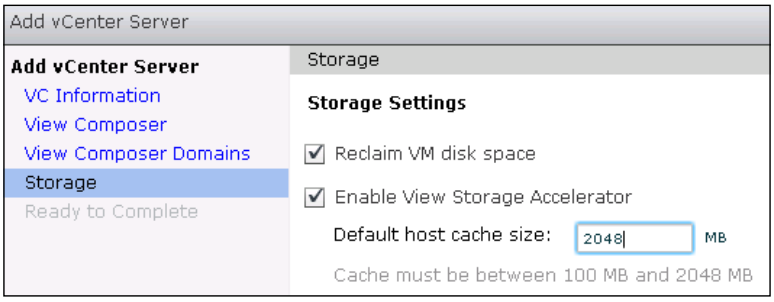


8. View Connection Server will attempt to verify the SSL certificate of the vCenter Server as part of the linking process. If the vCenter Server is still using the default self-signed certificate, a window will open and announce that the identity of the specified vCenter Server cannot be verified. If this happens, click on **View Certificate** to open the **Certificate Information** window, verify that the certificate information is correct, and click on **Accept** to move to the next window, **View Composer**.

9. *Chapter 3*, *Implementing VMware Horizon View Composer*, will outline how to deploy VMware Horizon View Composer. Since Composer is not yet installed, select the **Do not use View Composer** radio button and click on **Next** to open the **Storage** window. View Composer can be enabled within the View Manager Admin console at any time after it has been installed.
10. If vSphere 5.0 or newer is being used, the option to enable View Storage Accelerator will be available. If vSphere 5.1 or newer is being used, the option to reclaim disk space will be available. These options will be grayed out if the vSphere version does not support the specific feature. To enable host caching, check the **Enable View Storage Accelerator** checkbox and change the **Default host cache size** field's value to `2048` MB, which is the maximum amount supported, as shown in the following screenshot. To enable linked-clone disk space reclamation, check the **Reclaim VM disk space** checkbox. Click on **Next** to move on to the **Ready to Complete** window.

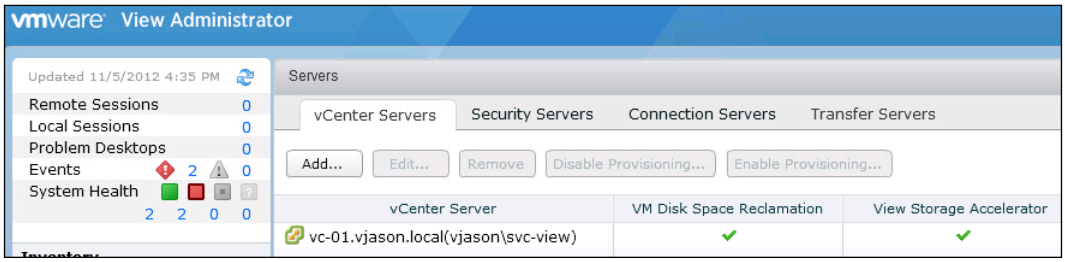
Additional information about View Storage Accelerator and disk space reclamation is available in the *Appendix*.

By default, View enables View Storage Accelerator for all hosts managed by the vCenter Server, using the same cache size. To edit the settings for an individual host, highlight the desired host and click on the **Edit cache size** button to override the default settings.



11. The **Ready to Complete** window will provide a summary of the options chosen in the previous steps. Review the summary and click on **Finish** to complete the process and link the vCenter Server to the View Connection Server.

The vCenter Server will now be displayed in the Servers window in the **View Administrator** page, as shown in the following screenshot:



At this point, the basic configuration of the View Connection Server has been completed; if we had a virtual desktop image, we could deploy full clone virtual desktops. However, before moving forward to the next step it is advisable to set up a second View Connection Server; this one will be a replica of our first one. The next section will outline how to install a View Replica Connection Server.

Deploying a View Replica Connection Server

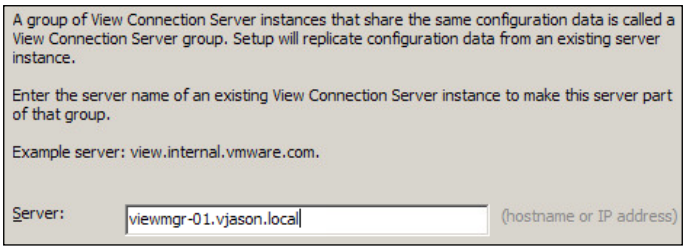
The installation process for a View Replica Connection Server is straightforward as the configuration information is copied from the View Standard Connection Server that was installed in the previous section. This section will explain how the installation process differs with regard to options that are chosen during installation.

Installing a View Replica Connection Server

The steps required to install a View Replica Connection Server are largely the same as those for a Standard Connection Server; only a few steps differ. The following list outlines where the installation process differs, using the numbered steps provided within the *Installing the first View Connection Server* section as a reference.

- **Step 5 – Installation options**: Choose **View Replica Server** as the instance type and click on **Next**.

- **Step 6 – source server**:
    - Provide the FQDN for the View Standard Connection Server deployed in the previous section and click on **Next**. For our sample server, the FQDN is `viewmgr-01.vjason.local`. Refer to the following screenshot as an example:



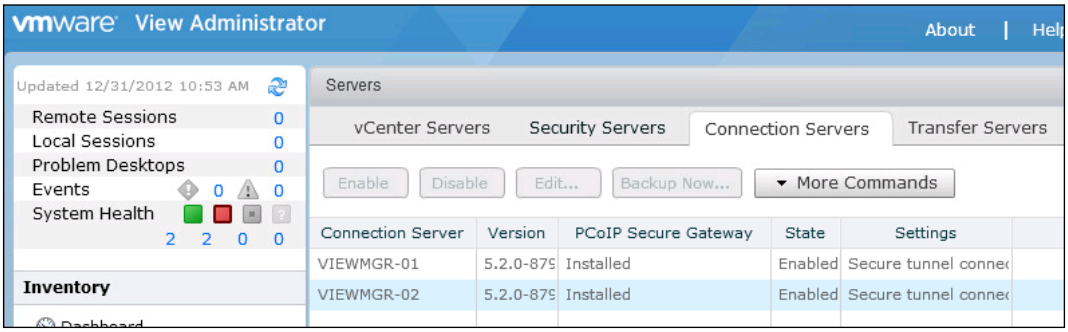
    - This installation step is unique to the View Replica Connection Server installation.

- Step 8 – Ready to install the program:
    - Verify the installation directory and click on **Install**.
    - This is the final installation step for a View Replica Connection Server installation.

During the installation process, the View Replica Connection Server will install all the same components as a Standard Connection Server but the AD LDS database will be a replica of the Standard Connection Server AD LDS database. The AD LDS database holds the configuration data for the View environment, so no additional configuration is needed beyond the installation itself.

View Replica Connection Servers should only be installed in the same site as the View Standard Connection Server that they are a replica of. View Connection Server AD LDS replication requires continual high network throughput and low latency connections and any interruption or degradation may lead to inconsistencies in the AD LDS configuration, which prevents the users from accessing their desktops.

Once the installation has successfully completed, the View Replica Connection Server will be displayed in the **Connection Servers** column in the **Servers** window in the **View Administrator** page as shown in the following screenshot:



At this point, the View Connection Server would be available to broker connections to whatever resources have been configured within the View Manager Admin console.

Once installed, a View Replica Connection Server is a full peer of all other View Standard Connection Servers and View Replica Connection Servers within the installation. Unlike some software platforms, which have a master-slave or split role architecture, all View Standard Connection Servers and View Replica Connection Servers share the same responsibilities and capabilities.

As previously mentioned, no native load-balancing exists within View. As a result, unless a load-balancing solution of some sort is deployed, clients will need to manually choose which of the two View Connection Servers they want to connect to. Since this is likely to lead to an unbalanced distribution of client resources, it is recommended you consider one of the previously mentioned options for load-balancing the View client connections.

View Connection Server backup

The information required to restore a View Connection Server is stored in two different databases:

- View Manager AD LDS database
- vCenter Server database

View Composer, described in *Chapter 3*, *Implementing VMware Horizon View* `Composer`, also uses a database that contains critical View configuration information. View installations that utilize View Composer will also need to back up the View Composer database as part of their backup and recovery plan.

Backing up the vCenter Server database

The vCenter Server database should be backed up using whatever method is available within your environment. This includes options such as:

- Native backups based on the capabilities of the database platform, such as Microsoft SQL Server backup or Oracle **Recovery Manager** (**RMAN**) backup
- Third-party database backup solutions

There are no specific requirements with regard to database backup methodologies, so long as you can recover the database to a previous state; either to the same database server or an alternative. Consult with your database server or backup software documentation if additional information about performing database backups is required.

Consult the *VMware vSphere Installation and Setup Guide* for additional information about how to back up the components of a vSphere installation.

Backing up the AD LDS database

The VMware Horizon View Connection Server AD LDS database contains key View configuration information and should be backed up on a regular basis. By default, a View Standard Connection Server or a View Replica Connection Server will perform a nightly backup of the AD LDS database at midnight (12:00 AM).

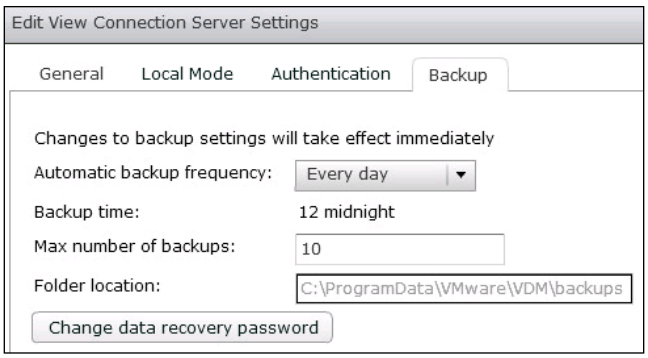
A limited number of changes to the View Connection Server backup policy can be made within the View Manager Admin console. These changes include:

- Backup frequency
- Number of backups to retain
- Data recovery password

These options can be configured using the following steps:

1. Navigate to **View Configuration** | **Servers** | **Connection Servers**.
2. Highlight the targeted View Connection Server and click on **Edit** to open the **Edit View Connection Server Settings** window.

3. Click on the **Backup** tab and make the desired changes; refer to the following screenshot for an example.



View Connection Server recovery

The process to restore a View Connection Server varies based on the scenario that necessitated the restore.

If you are restoring all of your View Connection Servers from scratch, the View Composer database will need to be restored as part of the recovery process as its contents are tied to those of the View AD LDS database. Refer to the *Restoring the View Composer database* section in *Chapter 3*, *Implementing VMware Horizon View Composer,* for additional information about restoring the View Composer database.

In situations where it is required to restore multiple View-related databases at once, you should use backups that were taken as closely together as possible when performing the restore. Ideally, backup plans for all View components should coincide as closely as possible to ensure that the data shared between them is consistent. The further apart the backups are taken, the less likely it is that the contents will match when a restore is required. This could lead to issues that require the assistance of VMware Support as items within one or more of the databases may need to be removed or edited.

Restoring a single View Connection Server

View configuration settings are stored in the local AD LDS database on each View Connection Server. If the View Connection Server software becomes corrupt, you can simply uninstall and reinstall it without having to perform any additional configuration.

If a View Connection Server cannot be accessed due to a hardware or software failure, and you need to replace it, you must remove it from the AD LDS replication set first. The following steps should be executed from an existing View Connection Server. In this example, we will remove the server named `viewmgr-02.vjason.local` from the AD LDS replication set using the `vdmadmin.exe` utility.

1. Go to the `Program Files\VMware\VMware View\Server\tools\bin` directory.
2. Execute the following command:

```
vdmadmin.exe -S -r -s viewmgr-02.vjason.local
```

The failed View Connection Server `viewmgr-02.vjason.local` has been removed from the AD LDS replica set and can now be replaced.

To learn about all the options available for the `vdmadmin.exe` command, run `vdmadmin.exe /?`.

To restore or replace the server, simply reinstall the View Connection Server software on a new server, selecting the **View Replica Server** option when prompted; the software will be installed and the server will be joined to the existing View installation.

Removing a View Connection Server

The same `vdmadmin.exe` command should be used when you want to remove an existing View Connection Server. To remove a View Connection Server from your environment, perform the following steps:

1. Open the **Windows Programs and Features** control panel on the target server.
2. Uninstall the **VMware View Connection Server** software.
3. Uninstall the **AD LDS Instance VMwareVDMDS** software.
4. On any remaining View Connection Server, execute the `vdmadmin.exe` command using the syntax provided in the previous section. Supply the name of the server you wish to remove from the command text.

Restoring the vCenter database

The vCenter database is restored using the same tools used to perform the backup. Consult with the database server or backup software documentation for information about how to perform the restore operation.

Consult the *VMware vSphere Installation and Setup Guide* for additional information about how to restore the components of a vSphere installation.

Restoring the View Connection Server AD LDS database

The View Connection Server AD LDS database can be restored using the `vdmimport` utility, located in the VMware Connection Server installation drive in the `Program Files\VMware\VMware View\Server\tools\bin` directory. The utility is a command-line tool and is executed from a Windows command prompt. The utility requires administrative access to the View infrastructure, so the AD user account used to run it must have administrative rights. If the AD account does not have sufficient rights, errors will be displayed during the restore operation.

To learn about all the options available for the `vdmadmin.exe` command, run `vdmimport.exe /?`.

The AD LDS database is not usually restored unless all the View Connection Servers are lost, or the AD LDS database is found to be corrupt. If you are having problems with just a single View Connection Server, you should refer to the *Restoring a single View Connection Server* section in this chapter for instructions on how to restore or remove just that server.

The restore operation requires two commands:

1. Decrypt the AD LDS database backup titled `backup.LDF` to a file titled `decrypted.LDF`. Replace the password with the data recovery password specified during the installation of the first View Connection Server.

   ```
   vdmimport -d -p password -f backup.LDF > decrypted.LDF
   ```

2. Restore the decrypted backup.

   ```
   Vdmimport –f decrypted.LDF
   ```

Once the restore is complete, the remaining View Connection Servers will replicate the restored data into their local AD LDS databases.

In the event that vSphere snapshots are used to restore a VMware Horizon View Connection Server to a previous state, the Connection Server will stop replicating its AD LDS database to other Connection Servers. In the event you wish to retain the configuration contained within the AD LDS database on the server where the snapshot was reverted, all other Connection Servers within the same View installation will need their View Connection Server software and AD LDS database uninstalled and reinstalled from scratch. Refer to the *VMware Horizon View Upgrades Guide* (`http://www.vmware.com/support/pubs/view_pubs.html`) for additional information about what is required if a View Connection Server is reverted to an earlier vSphere snapshot.

Summary

In this chapter, we have been introduced to the key component of a VMware Horizon View installation: the View Connection Server. We learned what is required to deploy a View Connection Server, what the limits of a Connection Server are, how to perform the installation and initial configuration, and how to install additional Replica Connection Servers.

We also discussed how to back up to the View Connection Server configuration, including the vCenter databases.

We concluded this chapter by discussing how to restore the View Connection Server configuration, and how that process varies based on the recovery scenario.

In the next chapter we will implement View Composer; the component of a View installation that enables the deployment of linked-clone virtual desktops.

3
Implementing VMware Horizon View Composer

VMware Horizon View Composer is a feature of View that enables the rapid provisioning of linked-clone virtual desktops. A pool of linked-clone desktops shares the same master image and writes any changes to a dedicated virtual hard disk, also known as a delta disk. This drastically reduces the amount of per-virtual desktop storage required compared to full clone virtual desktops, as each of those requires its own copy of the master image. In addition, linked-clone desktops can be provisioned much more quickly than full clone desktops, which is beneficial in and of itself and also enables new ways of managing desktops throughout their lifecycle.

This chapter will discuss benefits, installation, configuration, backup, and recovery of the View Composer component of VMware Horizon View.

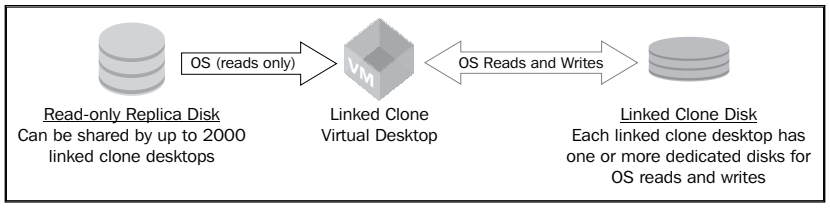
In this chapter, we will learn:

- An overview of the capabilities and benefits of View Composer
- The hardware and operating system requirements of View Composer
- View Composer network protocol and port usage
- View Composer pre-installation tasks
- How to install View Composer
- How to configure View Composer
- How to back up components of View Composer and what components to backup
- How to restore View Composer from backups

View Composer requires Microsoft **Key Management Services** (**KMS**) in order to support and properly activate Windows 7 and newer linked-clone desktops. Do not use **Multiple Active Key** (**MAK**) licenses with linked clones as the frequent need to reactivate Windows will exhaust the MAK activation limit.

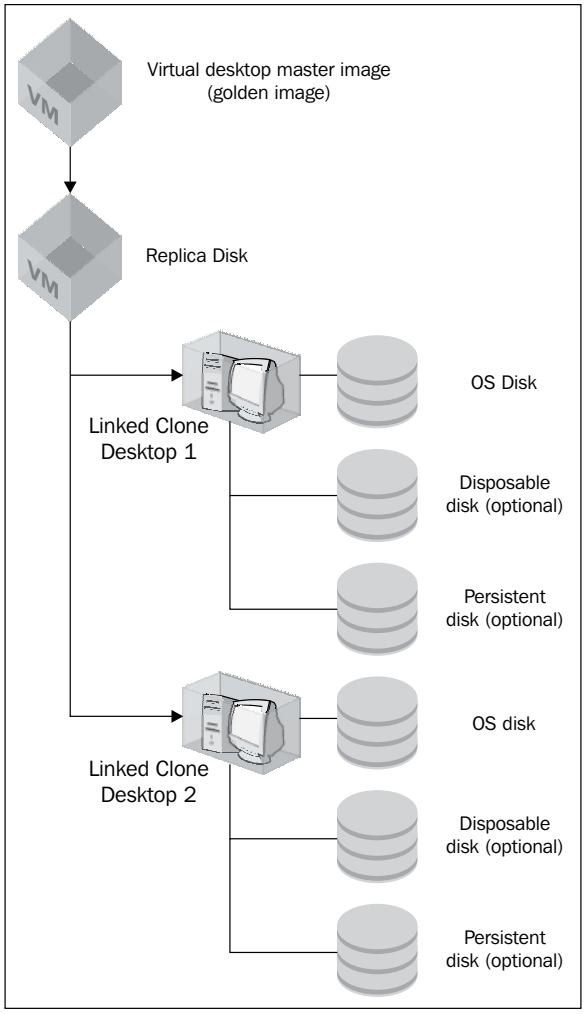
Overview of VMware Horizon View Composer

View Composer is used to provision linked-clone virtual desktops, which are a type of virtual machine that shares a common virtual desktop master image, also known as a golden image. View and vSphere support up to 2,000 desktops for each single replica of the virtual desktop master image, which enables significant storage savings over traditional full clone virtual desktops. The concept behind a linked-clone desktop is demonstrated in the following diagram, which shows the relationship between the master read-only replica disk and the linked-clone disk:



When a pool of linked clone desktops is provisioned, a replica of the virtual desktop master image is copied to storage accessible by the View vSphere hosts. This replica will be used as a read-only copy of the virtual desktop master image; all writes are redirected to unique disks that are attached to each linked clone. Linked clones are provisioned using thin virtual disks, the configuration of which varies depending on the desktop pool settings.

The following diagram shows the configuration of two linked-clone virtual desktops that share the same virtual desktop master image:



In this diagram, each linked-clone desktop is configured with optional disposable disks and persistent disks. *Chapter 8*, *Creating and Managing View Desktop Pools* explains the different options that affect the configuration of these and other linked-clone disks.

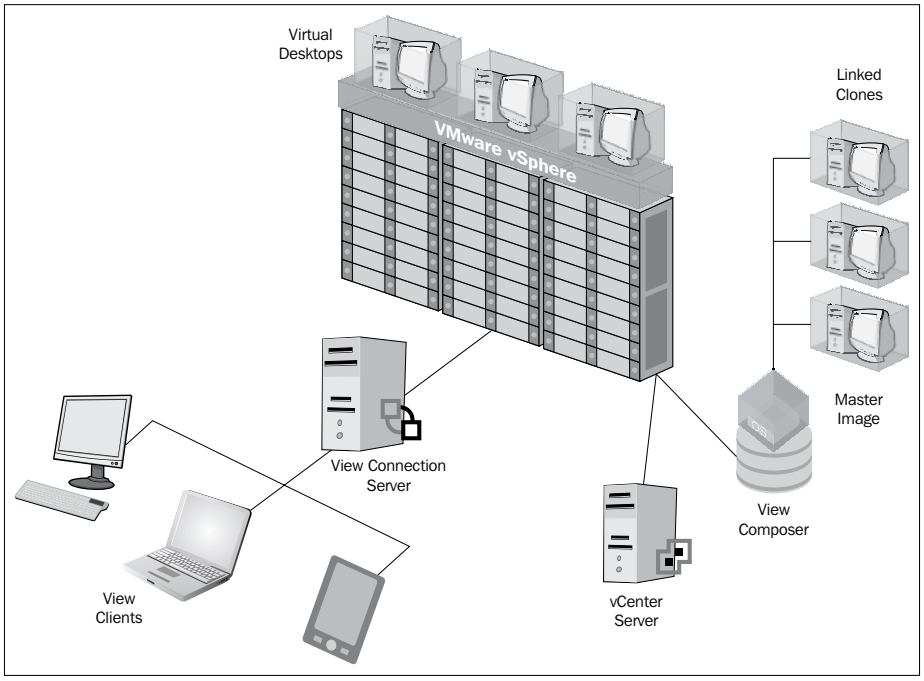
When the linked-clone desktops are powered on for the first time, they will begin redirecting any writes to the linked clone disks and will also create virtual swap (vswp) file of a size equal to the amount of unreserved virtual **RAM** (**vRAM**) allocated to the virtual machine.

Controlling the growth of a linked clone desktop requires an understanding of multiple topics, all of which will be described within this book. The following chapters contain valuable information that can help you control the storage utilization of linked clone desktops:

- *Chapter 7*, Implementing *View Persona Management*
- *Chapter 8*, *Creating and Managing View Desktop Pools*
- *Chapter 10*, *Performing View Desktop Maintenance*
- *Chapter 11*, *Creating a Master Virtual Desktop Image*

View Composer works at the direction of the View Connection Server to provision and manage linked clone desktops. View Composer uses the vSphere **application program interface** (**API**) to initiate whatever tasks are required based on the operation that is being performed. To achieve this, View Composer requires specific permissions within vCenter Server, as well as permissions within Microsoft Active Directory, both of which will be described later in this chapter.

View Clients do not come into contact with View Composer; its role as an orchestration tool is to perform actions based on the configuration of the View desktop pool. As such, View Composer is depicted in the following architectural diagram as a standalone component that works directly with the vSphere and Horizon View infrastructure.



The benefits of View Composer do not start and end with gains in storage efficiency. The following features are just some of the ways with which you can leverage View Composer to change how you manage desktop maintenance.

Recomposing linked clone desktops

A virtual desktop **recompose** operation is used when you need to update the master replica image. One of the key methods of maintaining the storage efficiencies of linked clone desktops is to control the amount of data that is written to the linked clone disks. Traditional per-desktop administrative tasks such as installing applications and applying patches can quickly increase linked clone storage utilization, as the writes would be made to the linked clone disk of each desktop that was the target of the action. In addition, these operations would need to be repeated on each desktop, which can require significant management overhead.

A recompose operation replaces the linked clone replica disk with an updated version that uses the same operating system. Recomposing to a completely different operating system is not supported. During the recompose process, all virtual desktops will be provisioned again and linked to the new replica disk. If the desktop was configured with a persistent disk for storing user profile data—an option for dedicated assignment desktop pools—that disk would be retained and attached to the recomposed desktop.

The recompose process is described in detail in *Chapter 10*, *Performing View Desktop Maintenance*.

When a desktop is recomposed, it will maintain the same computer name but will obtain a new DHCP lease. Your DHCP server must have a large enough pool of addresses available to handle these requests. In addition, it is recommended you use a short address lease period so that the now unused leases are quickly removed and the associated addresses are made available for use.

Refreshing linked clone desktops

The changes made to a linked clone desktop can be discarded on demand or upon user logoff. This operation is called a **refresh**, as it refreshes the desktop back to the exact configuration as it was when it was first provisioned.

View also supports deleting the desktop as an additional option. After the desktop has been deleted, a new desktop is provisioned to replace it.

By refreshing desktops you maintain tight control over their storage utilization as all writes to the dedicated linked clone virtual disk are discarded. In addition, when all changes are discarded, the desktop is returned to the same state it was in when it was provisioned, which enables tight control over the end-user computing experience.

The refresh process is described in detail in *Chapter 10*, *Performing View Desktop Maintenance*.

Refer to *Chapter 7*, *Implementing View Persona Management*, and *Chapter 8*, *Creating and Managing View Desktop Pools*, for more information about how to preserve user data during recompose or refresh operations.

View Composer requirements

View Composer requires a minimum server configuration to ensure adequate performance and to properly support linked clone provisioning and maintenance operations.

Hardware requirements

The View Composer software has specific requirements with regard to the hardware specifications and host operating system. View Composer may be installed on a dedicated virtual or physical server, or on an existing vCenter virtual or physical server. The vCenter Server Linux-based virtual appliance also supports View Composer but requires that it be installed on a dedicated server.

The following table outlines both the minimum and recommended hardware specifications of a dedicated View Connection Server.

Hardware component	Minimum requirement	Recommended
Processor	2 CPUs (1.4 GHz 64-bit or faster)	4 CPUs (2.0 GHz or faster)
Memory	4 GB	8 GB minimum for deployments of 50 or more desktops
Networking	One or more 10/100 MBps network interface cards (NICs)	1 Gbps NICs
Hard disk capacity	40 GB	60 GB

If View Composer is installed on the same host server as vCenter Server, the only requirement is that vCenter meets the established requirements for the number of virtual desktops and vSphere hosts that it will manage. The *VMware vSphere Installation and Setup Guide* (`http://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-pubs.html`) provides detailed information about the hardware requirements of vCenter Server; in particular, how those requirements change as the number of virtual machines or vSphere hosts changes.

Due to the critical importance of vCenter Server and View Composer within the View environment, it is recommended you use a dedicated server for View Composer rather than installing it on the same server as vCenter. Separating the two components ensures that the maximum performance will be obtained with regard to linked clone provisioning and maintenance operations. In addition, separating the two components ensures that maintenance operations that involve either View Composer or vCenter Server do not affect the availability of either component, be it due to downtime or other issues that may occur.

Like most other software platforms, the recommended guidelines should be followed to ensure that the View Composer performs optimally.

Software requirements

View Composer requires either the Standard or Enterprise edition of Windows Server 2008 R2, with or without SP1 installed. This requirement applies to the vCenter Server as well in scenarios where View Composer will be installed directly on the vCenter Server.

Limits of View Composer

A single View Composer instance can support only one vCenter Server. Starting with the combination of Horizon View 5.2 and vSphere 5.1, a single vCenter Server can now support up to 10,000 desktops, which is the maximum number of desktops supported in a single View installation. While View can support multiple vCenter Servers, each with its own dedicated View Composer instance, only one of each is actually required to manage up to the maximum number of desktops supported by View.

If you still wish to have multiple vCenter Servers in your View environment, you will need multiple instances of View Composer as a single instance can service only one vCenter Server. Each additional instance of View Composer will also require its own database, as the databases cannot be shared between View Composer instances.

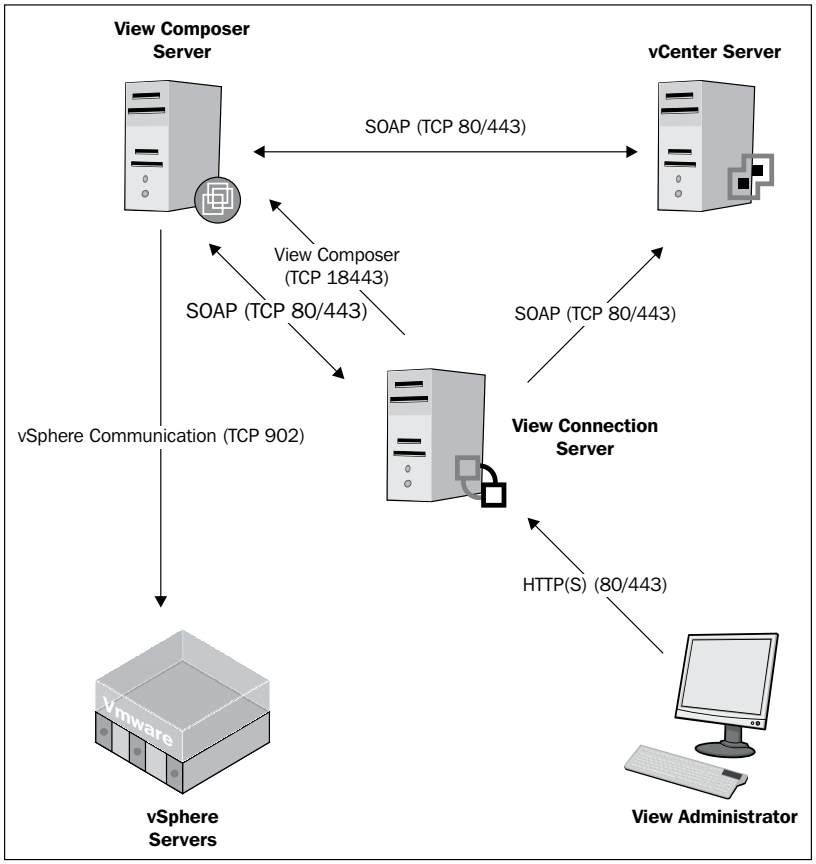
View Composer networking

The following diagram illustrates how the primary protocols used by View Composer work with other components of the View infrastructure. The diagram shows the following components of a View infrastructure:

- Communication between View Composer and the vCenter Server
- Communication between View Composer and the vSphere virtual desktop hosts
- Communication between the View Connection Server and the vCenter Server
- Communication between the View Administrator and the View Manager Admin console on the View Connection Server

The arrows indicate the direction in which each protocol travels, assuming the default settings are used.

This chapter assumes that you are using a dedicated server for the host View Composer. This does not change how View Composer operates, only where it is installed.



The diagram shows only the core components of a View infrastructure that includes View Composer; as new components are introduced in later chapters, the placement of those components within the infrastructure will be shown in an updated diagram. These components include:

- View Transfer Server (discussed in *Chapter 4*, *Implementing VMware Horizon View Transfer Server*)
- View Security Server (discussed in *Chapter 5*, *Implementing VMware Horizon View Security Server*)

This list of ports used by the core components is outlined in the following table. Refer to *Chapter 2*, *Implementing VMware Horizon View Connection Server*, for a full list of protocols used including those dedicated to client connections.

Protocol or service	Port	Notes
HTTP/HTTPS	TCP 80 or 443	
SOAP (Simple Object Access Protocol)	TCP 80 or 443	
vSphere Communication	TCP 902	TCP 902 is a core port of the vSphere infrastructure used for multiple functions
View Composer Communication	TCP 18443	TCP 18443 is the default port for communication with View Composer

Installation prerequisites

There are a number of prerequisites that should be addressed prior to installing View Composer:

- At least one configured View Connection Server with a license key installed
- An Active Directory user account or security group that will be granted the necessary permissions within Microsoft Active Directory and the vCenter Server
- A dedicated Windows 2008 R2 host server for View Composer or an existing vCenter Server that has 2008 R2 as the host OS
- A Static IP address for the dedicated View Composer host server (required only if you are using a dedicated server)
- Local administrator access on the host server
- A supported View Composer database as referenced in *Chapter 1*, *Designing a VMware Horizon View Infrastructure*
- A 64-bit ODBC connection to the View Composer database configured on the View Composer host server

In addition to the items described in *Chapter 1*, *Designing a VMware Horizon View Infrastructure* the following items should be prepared in advance of the installation.

View Composer user account

View Composer requires access to the vCenter Server in order to perform tasks related to the creation and management of virtual desktops. To facilitate this access, you can either grant additional permissions to the View Communication Server vCenter user account created in *Chapter 2*, *Implementing VMware Horizon View Connection Server*, or create a dedicated AD user account that View Composer will use to access the vCenter Server and Active Directory.

If you install View Composer directly on the vCenter Server, you have no option but to use the View Connection Server vCenter AD account for View Composer. If this is the case, you will need to grant that account or security group the additional vCenter and AD permissions outlined in this section.

This section assumes that you will use a dedicated AD account and a standalone instance of View Composer.

Configuring View Composer vCenter permissions

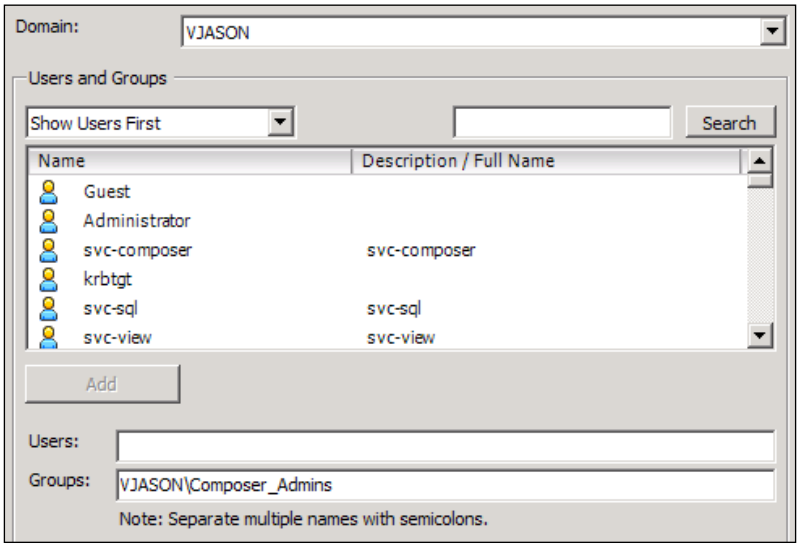
To make it easier to change the View Composer AD account in the future, you will create a vCenter role that includes all the required privileges. Once created, the role can be quickly applied to AD user accounts or security groups. Use the following steps to create the role in vCenter.

1. In vSphere Client, go to **Home** | **Roles** | **Add Role** and enter a role name such as `View Composer Administrator`.

2. From within the **Add New Role** window, expand each privilege group listed in the following table and check the required privilege items as listed in the table. All listed privileges must be checked in order for View Composer to function properly. If vSphere host caching is not to be used, the **Act as vCenter Server** permission under **Global** is not required. Click on **OK** when finished with creating the role.

vCenter privilege group	Privilege subsection	Privilege
Folder		Create a Folder
		Delete a Folder
Datastore		Allocate space
		Browse datastore
		Low level file operations
Global		Act as vCenter Server
		Disable methods
		Enable methods
		System tag
Network		All
Resource		Assign virtual machine to resource pool
		Migrate the powered off virtual machine
Virtual Machine	Configuration	All
	Interaction	Power Off
		Power On
		Reset
		Suspend
	Inventory	All
	Provisioning	Allow disk access
		Clone virtual machine
		Customize
		Deploy template
		Read customization specifications
	State	All

3. In the vSphere Client, go to **Home | Hosts and Clusters**, right-click on the vCenter Server at the top level of the inventory, and click on **Add**. This will open the **Assign Permissions** window.
4. In the **Assign Permissions** window, click on the **Add** button to open the **Select Users and Groups** window.
5. In the **Domain** drop-down menu, select the AD domain that contains the View Administrator user or security group. In our example, the domain is named `VJASON`.
6. In the **Users and Groups** list, select either the View Composer AD account or the View Composer AD security group; choose whichever is applicable for your environment. For our sample environment, we have chosen the security group titled `Composer_Admins`. Once selected, click on the **Add** button as shown in the following screenshot. Click on **OK** to close the **Select Users and Groups** window.



7. From the **Assigned Role** drop-down menu, select the View Composer Administrator role. Click on **OK** again to close the **Assign Permissions** window and complete the action.

In addition to the vCenter privileges, the AD account used for vCenter Access also requires specific permissions within AD in order to manage the linked clone computer accounts. These permissions are outlined in the next section.

Active Directory permissions

The View Composer AD account requires permission to manage the AD Computer objects for the virtual desktops that it creates. As there is some risk associated with granting accounts direct access to AD in order to create and delete computer objects, it is important to minimize the access the View Composer account has.
To minimize risk, the following guidelines are recommended:

- Create an AD organizational unit that will be used only to store linked clone virtual desktops using View Composer, and grant permissions only to the objects contained within this OU
- Grant the View Composer AD account the minimum permissions required in order to manage the View Composer user accounts

To grant the necessary permissions, you need as a minimum full control over the OU which will contain your AD computer accounts. This gives you the ability to not only delegate the required permissions for View Composer, but also to create additional child OUs to enable additional control over the various desktop pools that you provision.

Chapter 8, *Creating and Managing View Desktop Pools*, and *Chapter 13*, *Implementing VMware Horizon View Group Policies*, outline some of the reasons why you might want to divide virtual desktop computer accounts into different OUs.

The following steps outline the process used to delegate the minimum permissions required for View Composer. In our example, we will be granting to the AD security group `Composer_Admins` the necessary permissions for the `View_Desktops` OU.

1. From the Windows Start menu, select **Administrative Tools** | **Active Directory Users and Computers**.
2. In the **Active Directory Users and Computers** console, select the **View** menu and then select **Advanced Features**.
3. Right-click on the parent OU that will contain the virtual desktops created using View Composer and select **Properties** to open the **Properties** window. In our example, the OU is named `View_Desktops`.
4. In the **View_Desktops Properties** window, select the **Security** tab and then click on **Advanced** to open the **Advanced Security Settings for View_Desktops** window.

5. Click on **Add**, add the `Composer_Admins` security group to the list of accounts with permissions, and click **OK** to open the **Permission Entry for View_Desktops** window.
6. Click on the **Object** tab. In the **Permissions** list, select **Allow** for the following permissions:
    - **Create Computer Objects**
    - **Delete Computer Objects**
7. Verify that **This object and all descendant objects** is selected in the **Apply to** drop-down menu.
8. Click on the **Properties** tab. In the **Apply onto** list, select **Descendant Computer objects**.
9. In the **Permissions** list, select **Allow** for the following permissions:
    - **Read all properties**
    - **Write all properties**
10. Click on **OK** to apply the changes and close the window.

In addition to the permissions within AD, the View Composer AD account or the `Composer_Admins` security group requires local Windows administrator permissions on the vCenter Server that it is linked to, as well as the Composer host server itself.

View Composer database

View Composer requires an external database in order to store information about vCenter Server connections, AD connections, and linked clone desktops and their associated virtual hard disks.

Chapter 1, *Designing a VMware Horizon View Infrastructure* outlines the different database types that are supported by View Composer. In addition to using a supported database platform, the following database configuration items must be included:

Using View Composer with Microsoft SQL Server databases:

- **Local SQL instance**: View supports Windows NT authentication
- **Remote SQL instance**: View requires an account on the database server with SQL Server authentication
- **The View database SQL Server user account**: This must have database owner permissions

Using View Composer with Oracle databases:

- The database should be created with the **General Purpose** or **Transaction Processing** template using Database Configuration Assistant
- View requires an Oracle database user account with the following permissions:
    - Connect
    - Create materialized view
    - Create procedure
    - Create sequence
    - Create table
    - Create view
    - Execute on `dbms_job`
    - Execute on `dbms_lock`
    - Resource
    - Unlimited tablespace

The database schema will be installed during the installation of View Composer.

Deploying View Composer

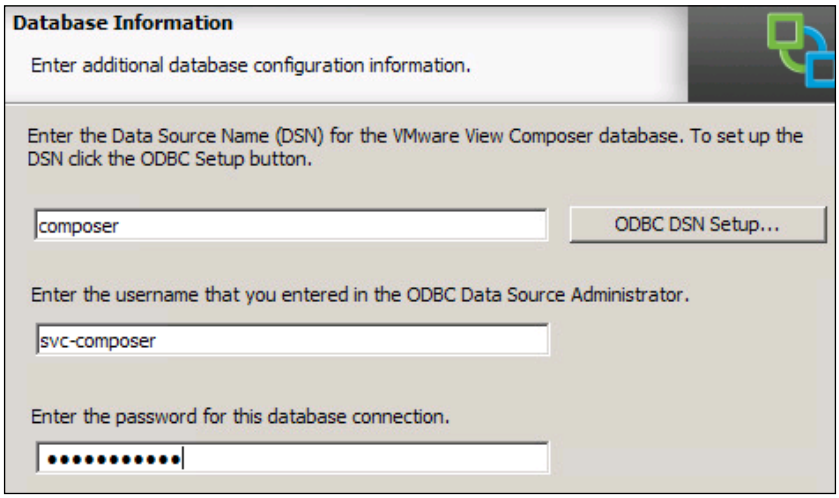
Deploying View Composer is broken down into two stages: the installation of the View Composer software, and the final setup using the View Manager Admin console.

Installing View Composer

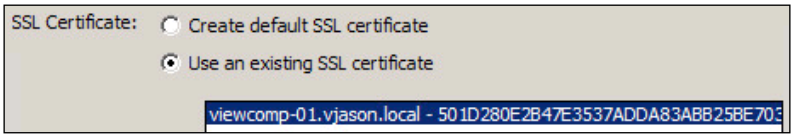
The View Composer software is delivered as a single executable (EXE) file, named in a format similar to `VMware-viewcomposer-x.x.x-yyyyyy.exe`. Prior to installing the View Composer software, create a **Data Source Name** (**DSN**) connection to the View Composer database. Consult the MSDN article Lesson 1: *Creating a SQL Server ODBC Data Source* for instructions on how to create a DSN using the Windows **ODBC Data Source Administrator** (`http://msdn.microsoft.com/en-us/library/cc879308(v=sql.105).aspx`). Name the DSN connection `composer`. The following steps outline the installation process:

If you plan to use a custom SSL certificate for View Composer, install that certificate prior to installing View Composer. If you install the certificate now, you can select it during the installation process. Refer to *Chapter 12, Managing VMware Horizon View SSL Certificates* for instructions on how to obtain and deploy a new SSL certificate for View Composer.

1. Double-click on the View Connection Server installer EXE file to launch the installer.
2. In the **Welcome to the Installation Wizard for VMware Horizon View Composer** window, click on **Next**.
3. Review the VMware End User License Agreement. Then select **the I accept the terms in the license agreement** radio button and click on **Next**.
4. Select the installation directory and click on **Next**.
5. Provide the name of the View Composer database, **Data Source Name** (**DSN**), the View Composer database user, and the password. The following screenshot shows the required information for our sample environment. Click on **Next** to move on to the next step.



6. If no SSL certificates were previously installed, select **Next** to accept the default port settings and then select the **Create default SSL certificate** radio button. If a SSL certificate was preinstalled, select the **Use an existing SSL certificate** radio button and highlight the desired certificate from the list as shown in the following screenshot. Click on **Next** to move on to the next step.



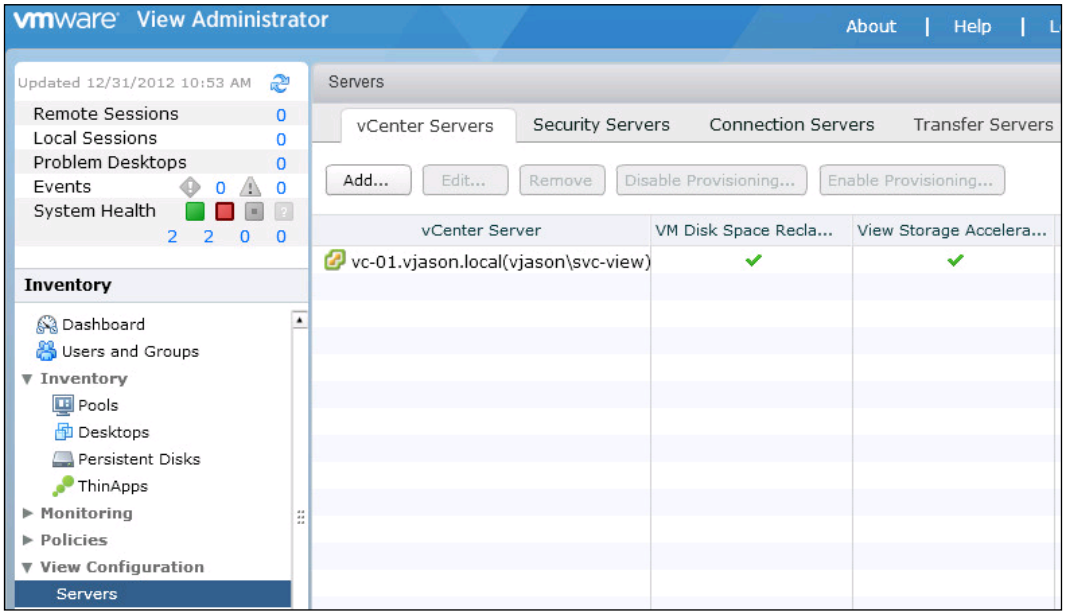
7. Click on **Install** to initiate the installation process and reboot the server as requested when the installation is complete.

View Composer is now ready to be configured, which is done using the View Manager Admin console.

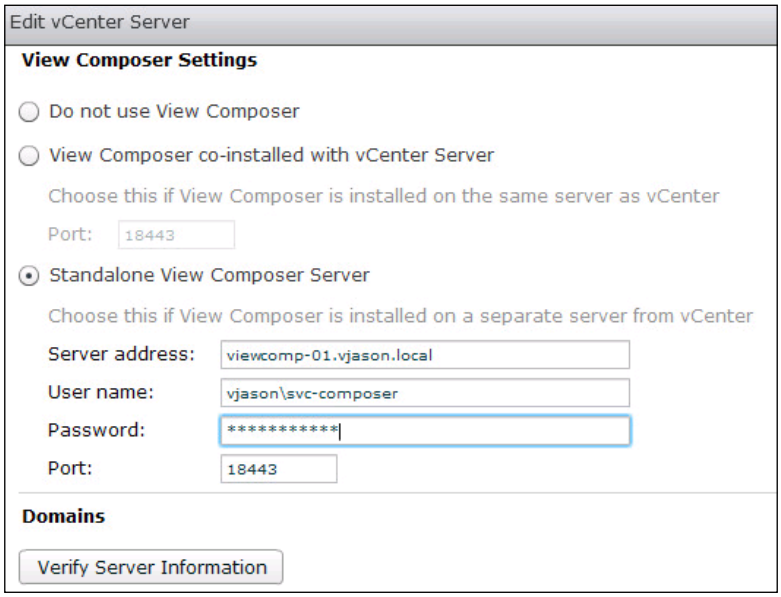
Configuring View Composer

Once the installation of View Connection has completed, you need to log in to the View Manager Admin console. The console URL will be in the format `https://Connection Server FQDN/admin`. The following steps outline the configuration process:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Go to **View Configuration** | **Servers** within the console.
3. Select the **vCenter Servers** tab in the **Servers** window, highlight the vCenter Server you wish to enable for View Composer, and click on **Edit** to open the **Edit vCenter Server** window. In the following screenshot, we edit the **vc-01.vjason.local** vCenter Server. Note the appearance of the green and yellow vCenter icon to the left of the vCenter Server name; that icon will change once View Composer is enabled.

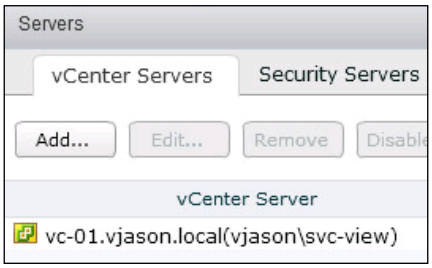


4. If View Composer was installed on the vCenter Server, select the **View Composer co-installed with vCenter Server** radio button and proceed to the next step. If using a dedicated server to host View Composer, check the **Standalone View Composer Server** radio button and complete the following tasks:
    1. Populate the **Server address** field with the FQDN of the dedicated View Composer host server.
    2. Provide **User name** and **Password** for the dedicated View Composer AD account. Use the format `domain-name\user-name` for the username.
    3. Since we used the default port when installing View Composer, we do not need to change the **Port** value.
    4. Click on **Verify Server Information** to verify View Composer access to the AD domain and update the window with the domain information.



Any errors that occur during this step are likely related to inadequate permissions to either AD or vCenter. If errors occur, review the *View Composer user account* section and verify that the necessary permissions have been granted.

5. Click **Ok** to complete the View Composer configuration.
6. Review the vCenter icon to the left of the vCenter Server name in the **View Configuration | Servers** page within the console; as View Composer is now enabled, that icon will be displayed within a yellow square as shown in the following screenshot:



Backing up View Composer

The information required to restore View Composer is stored in two different locations:

- View Composer database
- View Composer SSL certificates or RSA key container

The View Composer database should be backed up as part of a larger backup plan that includes the vCenter database, the View Connection Server MS LDS database, and the View Event database. By default, each View Connection Server backs up both the ADAM database and the Composer database to a folder on the Connection Server.

Backing up the View Composer database

The View Connection Server backs up the View Composer database as part of its own native backup process. This is the preferred method of backing up the View Composer database as it will be backed up at the same time as the View Connection Server ADAM database. As these databases contain related information, it is critical that they are backed up at the same time.

The default location for the View Connection Server database backups is on any View Connection Server in the `C:\ProgramData\VMware\VDM\backups` folder.

The View Composer database backups will have an SVI extension and include the name of the View Composer host server in the filename. In our example, the most recent View Composer database backup file is titled `Backup-2012-1017000010-viewcomp-01_vjason_local.SVI`. `Viewcomp-01` is the name of the dedicated View Composer host server in our sample environment.

The View Composer database may also be backed up using native database backup tools. This includes options such as:

- Native backups based on the capabilities of the database platform, such as Microsoft SQL Server backup or Oracle RMAN backup
- Third-party database backup solutions

Remember that, while native database backups will work, they may not be usable for restore purposes if they were not performed at the same time as the View Connection Server ADAM database backup. It is recommended you use the backups performed by the View Connection Server, if you need to restore the View Composer database.

Refer to your database server or backup software documentation if additional information about performing database backups is required.

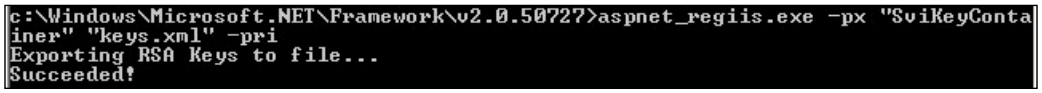
Backing up the View Composer SSL certificates

The process used to back up the default View Composer SSL certificate requires Microsoft .NET Framework to be installed on the View Composer host server. The following steps explain how to back up the SSL certificates:

1. From the command prompt on the View Composer host server, navigate to the `c:\Windows\Microsoft.NET\Framework\v2.0xxxxx` directory.
2. Execute the following command to export the View Composer RSA key container to a local file named `keys.xml`:

```
aspnet_regiis -px "SviKeyContainer" "keys.xml" -pri
```

The following screenshot shows the expected output if the command was successful.



The `keys.xml` file should be backed up to an alternative location to be used in the event that the View Composer software needs to be installed on a new server.

Chapter 12, *Managing View SSL Certificates* outlines the process by which you can obtain new certificates for all View components. During this process, you obtain a copy of the SSL certificate that can be used when restoring a View Connection server from backups. If you choose to use a custom SSL certificate, you do not need to use the `aspnet_regiis` command to export the View Composter RSA key container. During the creation of your custom SSL certificate, you should have been given a copy of it with the private key intact, which is what is required to perform a restore.

View Composer recovery

The same process is used to recover or move View Composer to a new host server. To retain the current settings, all that needs to be restored is the View Composer database and the RSA key container or custom SSL certificate.

Restoring the View Composer database

The View Composer database should be restored using the native View command-line tool `sviconfig.exe`. This tool is located within the `install` directory of View Composer, which is at the following location on our sample server:

```
D:\Program Files (x86)\VMware\VMware View Composer
```

You will need the following information to restore the database:

- **Data Source Name (DSN)**: The name of your DSN connection on the View Connection Server. On our sample server, the name is `composer`.
- **View Composer database username**. On our sample server, the name is `svc-composer`.
- **View Composer database password**. On our sample server, the password is `Password123`.

- **Backup file path**: This is the location of the `Backup-2012-1017000010-viewcomp-01_vjason_local.SVI` file referenced in the *Backing up the View Composer database* section. On our sample server, the file is located in `C:\Temp`.

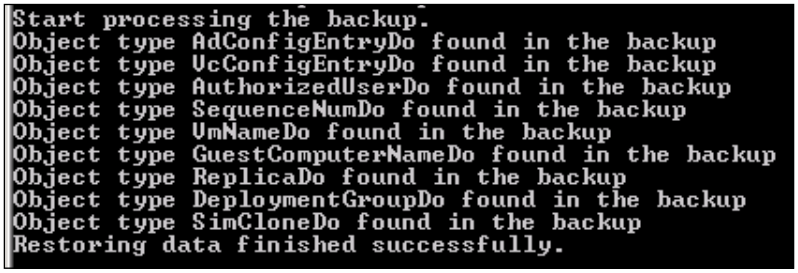
The View Composer database and View Connection Server MS LDS databases contain related data. If one is being restored, the other should also be restored, using the restore data from the same backup set. Failure to adhere to this rule can lead to database inconsistencies that will require the assistance of VMware Support in order to fix.

The following steps outline the process used to restore the database, using the information from our sample server:

1. Stop the VMware View Composer service.
2. From the command prompt, navigate to the View Composer installation directory.
3. Execute the following command to restore the View Composer database backup:

```
sviconfig -operation=restoredata -dsnname=composer
-username=svc-composer -password=Password123
-backupfilepath="C:\Temp\Backup-2012-1017000010-viewcomp-01_
vjason_local.SVI"
```

4. The restore process should output several lines of status information. The last few lines of the output are shown in the following screenshot, the last of which indicates that the restore was successful.



5. Start the VMware View Composer service.

View Composer is now operating with the restored database.

Restoring the View Composer SSL certificates

The process to restore View Composer SSL certificates varies depending on the scenario. The following sections explain the procedure you should use based on whether or not you plan to re-use an existing SSL certificate. Both of these procedures assume that you have already restored your View Composer database and configured an ODBC connection to that database on your View Composer host server.

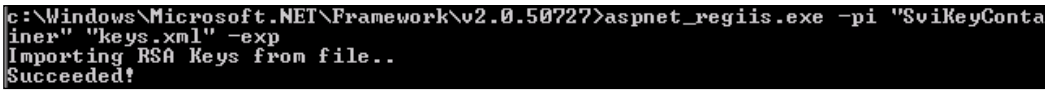
Restoring View Composer with a new default SSL certificate

Prior to installing the View Composer software, restore the RSA key container that was backed up in the section *Backing up the View Composer SSL certificates*. The following steps outline the full restore process:

1. Copy the `keys.xml` backup file to a location on the new View Composer host server. In our example, the file has been placed within the folder from which we will be executing the restore command.
2. From the command prompt on the new View Composer host server, navigate to the `c:\Windows\Microsoft.NET\Framework\v2.0xxxxx` directory.
3. Execute the following command to import the View Composer RSA key container:

   ```
   aspnet_regiis -pi "SviKeyContainer" "keys.xml" -exp
   ```

   The following screenshot shows the expected output if the command was successful:



4. Reinstall View Composer using the steps provided in the section *Installing View Composer*. Since this is a new server, View Composer will note that no SSL certificates are available and will create a new one.

View Composer is now ready to be linked to the View Connection server using the steps provided in the section *Configuring View Composer*.

Restoring View Composer with a custom SSL certificate

The process used to restore a View Connection server with a custom SSL certificate is straightforward as all the steps are handled within either the Microsoft Certificates MMC Snap-in or during the installation of View Composer. The following steps outline the full restore process:

Since we are re-using the same SSL certificate, it is important to remember that the new View Composer host server needs to have the same computer name as the old one.

1. Install the custom SSL certificate on the new View Connection server using the procedure outlined in *Chapter 12*, *Managing View SSL Certificates*.
2. Reinstall View Composer using the steps provided in the section *Installing View Composer*. Since the SSL certificate has already been installed, select the option **Use an existing SSL certificate**, and select the designated certificate.
3. Complete the installation and reboot the View Composer host server.

View Composer is now ready to be linked to the View Connection server using the steps provided in the section *Configuring View Composer*.

Summary

In this chapter, you have been introduced to an important and powerful component of the View installation: VMware Horizon View Composer. You learned what is required to deploy View Composer, what the limits of Composer are, and how to perform its installation and configuration.

We also discussed how to back up the View Composer configuration, which includes the RSA key container and View Composer database.

We concluded this chapter by discussing how to restore the View Composer database and RSA key container or SSL certificate, and how that process varies based on the recovery scenario.

In the next chapter, we will implement View Transfer Server, the component of a View installation that enables the deployment of virtual desktops that can be downloaded to and run from a View client computer, enabling total virtual desktop mobility.

4

Implementing VMware Horizon View Transfer Server

VMware Horizon View Transfer Server enables Virtual Desktops to be downloaded and run from a computer that is using the View Client with Local Mode option. Local Mode desktops provide a fully mobile Virtual Desktop, even when a network connection is not available. A Local Mode desktop shares many of the same features of a desktop hosted in the datacenter as well as a few additional ones, all of which will be discussed in this chapter.

This chapter will discuss benefits, installation, configuration, backup, and recovery of the View Local Mode feature of VMware Horizon View.

In this chapter, we will learn:

- An overview of the capabilities and benefits of View Local Mode
- The hardware and operating system requirements of a View Transfer Server
- View Transfer Server network protocol and port usage
- View Transfer Server preinstallation tasks
- How to install a View Transfer Server
- How to configure a View Transfer Server
- How and what components of Transfer Server to backup
- How to restore Transfer Server from backups

VMware Horizon View Transfer Server overview

A View Transfer Server is a specialized View Connection Server that enables the use of Local Mode desktops, which are Virtual Desktops that are downloaded and run on the View Client computer. A View Client that wishes to use Local Mode desktops needs a specific version of the View Client software, and must meet minimum hardware requirements as outlined in *Chapter 9*, *VMware Horizon View Client Options*.

View Local Mode desktops retain many of the benefits of traditional View Virtual Desktops, and add some additional ones as well. Some of these benefits include:

- Local Mode desktops are delivered using traditional View-centralized provisioning methods
- Local Mode desktops support traditional IT policy controls, such as those delivered using VMware Horizon View or Microsoft group policies, as well as Local Mode-specific policies set within the View Manager Admin console
- Local Mode desktops are stored in an encrypted format on the View client
- Once configured, Local Mode desktops do not require a persistent network connection
- Local Mode desktops can synchronize their changes back to the datacenter at regular intervals or when a connection becomes available, either on demand or according to a schedule set by the View administrator
- Local Mode desktops utilize the resource of the client system, including support for up to four processor cores, preserving datacenter resources

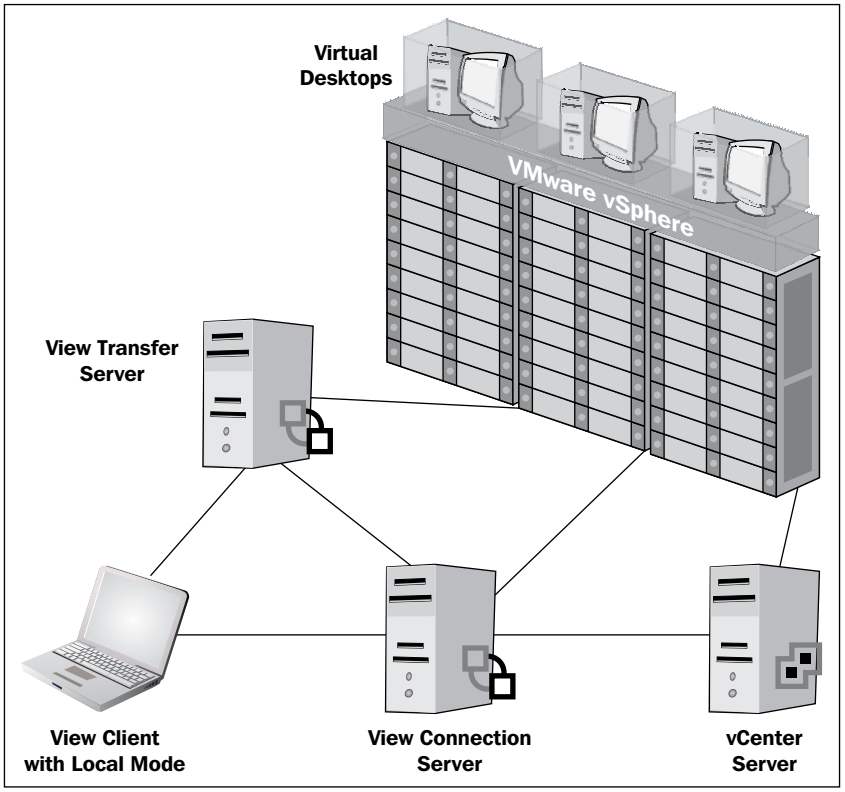
These are just some of the ways that a View Transfer Server and Local Mode desktops provide organizations with additional flexibility for their end user computing infrastructure.

A View Transfer Server works alongside a View Connection Server to support the movement of Virtual Desktop data between the View Local Mode client and the source Virtual Desktops running on the vSphere Servers. The Transfer Server maintains a copy of the Virtual Desktop base image that will be used for Local Mode desktops; when View Clients check out a desktop for the first time, they download this image directly from the Transfer Server, rather than from the vCenter Server itself. By offloading Virtual Desktop base image download operations to the Transfer Server, View lessens the impact of these operations on the vCenter Server.

The View Client with Local Mode includes a Type 2 hypervisor that it installs alongside the traditional View Client software; this hypervisor is used to host the virtual desktop on the client.

The Type 2 hypervisor included with View Client with Local Mode is not directly user accessible, and works only in tandem with the View Client. In addition, you cannot install VMware Workstation or VMware Player on a computer that has the View Client with Local Mode installed.

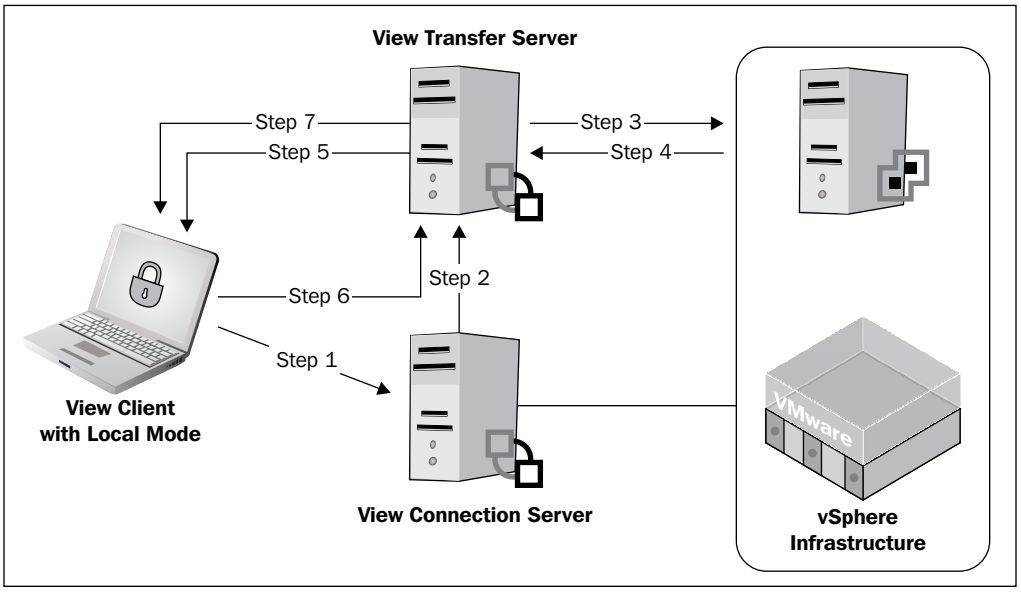
The View Connection Server acts as a session broker for operations involving View Transfer Server tasks only; all transfers of data take place between the View Client and the Transfer Server directly. The following figure shows the relationship between components of a View infrastructure that includes a Transfer Server:



The following diagram explains what occurs when a client makes a request to download a Local Mode desktop. For the purpose of this explanation, we will assume that the client is running the View Client with Local Mode option, which includes the hypervisor software needed to host the desktop.

1. The View Client logs into the View environment and requests to check out a Virtual Desktop from a View Connection Server.
2. The View Connection Server asks a View Transfer Server to mount the shared storage that contains the **virtual machine disk** (**VMDK**) files.
3. The View Transfer Server mounts the VMDK files.
4. The vSphere server sends the address of the shared storage to the View Transfer Server.
5. The View Transfer Server relays the address of the vSphere Server to the View Client.
6. The View Client requests the offline desktop using the address that was provided by the View Transfer Server.
7. The offline desktop is sent to the View Client through the View Transfer Server.

The process is illustrated in the following diagram:



The View Client also connects to the Transfer Server for other reasons, such as replicating or backing up Local Mode desktop data back to the source desktop in the datacenter, or checking in desktop images when they are no longer needed or so that they can be recomposed.

View Transfer Server requirements

View Transfer Server requires a minimum server configuration to ensure adequate performance and properly support the movement of Local Mode desktop data between the View client and the View Transfer Server. This section will outline these requirements.

Hardware requirements

A View Transfer Server has specific requirements with regard to the hardware specifications. View Transfer Server uses multiple virtual SCSI adapters in order to mount multiple VMDK files at once when transferring data to and from the View Client with Local Mode. This functionality requires that the View Transfer Server be installed on a virtual machine only.

The following table outlines the minimum hardware specifications of a View Transfer Server:

Hardware Component	Minimum Required
Processor	Two 64-bit CPUs
Memory	4 GB
Networking	1 Gbps NICs
Hard disk capacity	40 GB

The View Transfer Server has additional requirements that are unique to the Transfer Server role. These include the following:

- The View Transfer Server must be a virtual machine, installed on a vSphere host that is managed by a vCenter Server that is linked to the View Connection Server.
- The vSphere host of the View Transfer Server must have access to all data stores that contain the View desktop VMDK files.

- If linked clone desktop pools will be used for the source of Local Mode desktops, sufficient shared or local Transfer Server storage must be available in order to store all required View Composer base images.
- Shared storage is preferred in multi-Transfer Server environments as it can be used by both Transfer Servers.

When a View Transfer Server is linked to a View Connection Server, multiple configuration changes are made to the virtual machine. The vSphere **Distributed Resources Scheduler** (**DRS**) is set to manual for the host virtual machine, and four additional SCSI controllers are added to support the mounting of multiple desktop VHD files during transfer operations. Avoid making any changes to these settings as they are required for the Transfer Server to function properly.

Software requirements

View Transfer Server requires either the Standard or Enterprise edition of Windows Server 2008 R2, with or without SP1 installed.

As with the View Connection Server, when installed the View Transfer Server is configured with a self-signed SSL certificate, which will not be trusted by View Clients. It is recommended to replace the self-signed certificate with one issued from an internal or commercial certificate authority that the View Clients will trust.

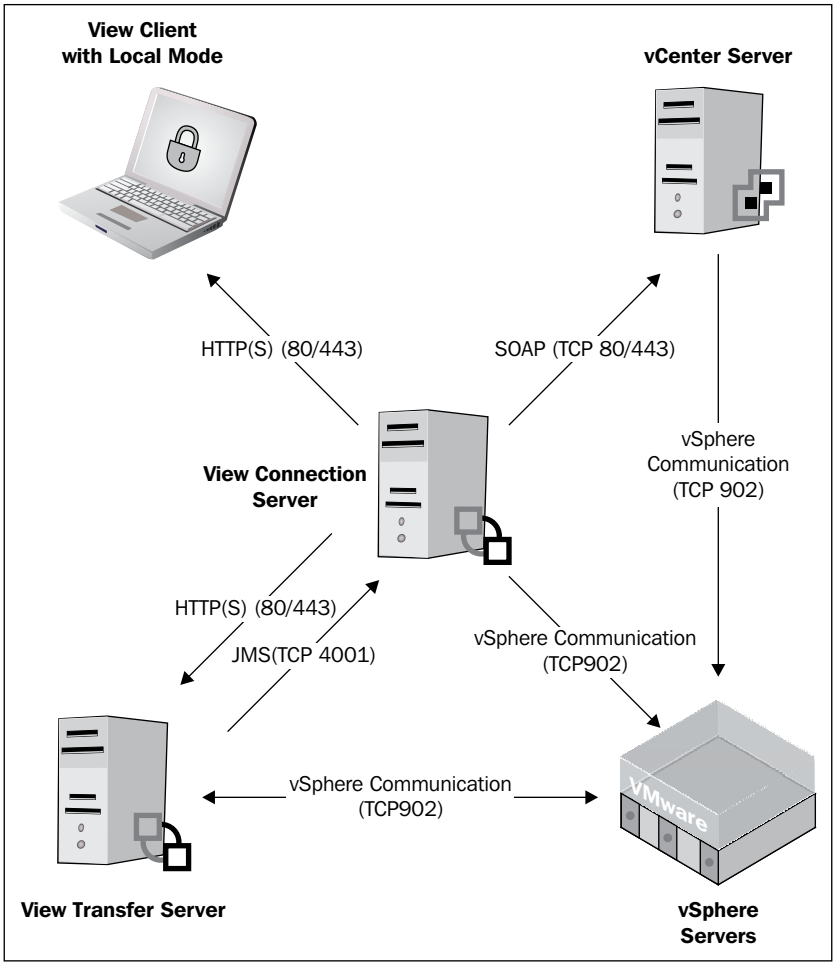
A single View Transfer Server should support no more than 20 concurrent disk transfers at a time. Additional View Transfer Servers can be installed to scale the number of concurrent transfers, or to simply enable high availability for the Transfer Server role.

View Transfer Server networking

The following diagram illustrates the how the primary protocols used by the View Transfer Server work with other components of the View infrastructure. The diagram shows the following components of a View infrastructure:

- Communication between the View Transfer Server and the vSphere servers
- Communication between the View Transfer Server and the vCenter Server
- Communication between the View Administrator accessing the View Manager Admin console on the View Connection Server

The arrows indicate the direction in which each protocol travels, assuming that the default settings are used.



The diagram shows only the core components of a View infrastructure that includes a View Transfer Server; as new components are introduced in later chapters, the placement of those components within the infrastructure will be shown in an updated diagram. The last component of the View infrastructure that will be discussed is the View Security Server, in *Chapter 5*, *Implementing VMware Horizon View Security Server*.

This list of ports used by the core components is outlined in the following table. Refer to *Chapter 2*, *Implementing VMware Horizon View Connection Server*, for a full list of protocols used including those dedicated to client connections.

Protocol or Service	Port	Notes
HTTP/HTTPS	TCP 80/443	
JMS (Java Messaging Service)	TCP 4001	
SOAP (Simple Object Access Protocol)	TCP 80 or 443	
vSphere Communication	TCP 902	TCP 902 is a core port of a vSphere infrastructure used for multiple functions.

View Transfer Server installation prerequisites

There are a number of prerequisites that should be addressed prior to installing a View Transfer Server:

- At least one configured View Connection Server with a View license key installed
- An Active Directory user account or security group that will be granted the necessary permissions to the View Transfer Server repository
- A dedicated Windows 2008 R2 server to host the View Transfer Server role.
- A static IP address for the Transfer Server host
- Local administrator access on the host serve

In addition to the items described in *Chapter 1*, *Designing a VMware Horizon View Infrastructure*, the following items should be prepared in advance of the installation.

The View Transfer Server repository and user account

The View Transfer Server requires a repository for storing the base images of linked-clone desktops that will run in Local Mode. There are two options for configuring the View Transfer Server repository:

- A folder on the Transfer Server
- A **Common Internet File System** (**CIFS**) file share

Either of the two options is suitable for storing the base images; however, if you plan to deploy multiple View Transfer Servers the file share option provides the maximum amount of flexibility. Some of the advantages of using a file share for your Transfer Server repository include:

- One file share can serve as the repository for multiple Transfer Servers
- Individual Transfer Servers can be taken offline for maintenance without interrupting access to the repository
- When the linked-clone image is updated, only a single repository will need to be updated
- Storing the repository on a separate server eliminates the need to back up the Transfer Server

To complete the configuration of the View Transfer Server, an AD user account will be required that has read/write access to the remote Transfer Server repository; no other permissions are required. This account will be used when specifying the repository location within the View Manager Admin console.

A Transfer Server repository is only required if linked-clone desktops are going to be used with local mode.

Deploying a View Transfer Server

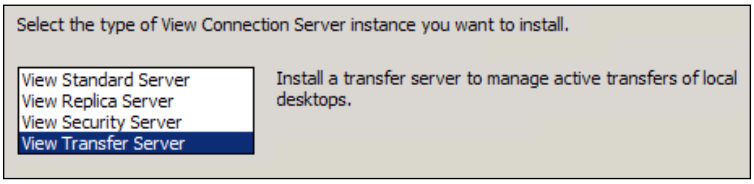
Deploying a View Transfer Server is broken down into three stages: the installation of the View Transfer Server software, creation of the Transfer Server repository, and the final setup using the View Manager Admin console.

Installing a View Transfer Server

The View Transfer Server software is delivered as a single executable (EXE) file, named in a format similar to `VMware-viewconnectionserver-x86_64-x.x.x-yyyyyy.exe`. This installer is used for all four View Connection Server types which include Standard, Replica, Transfer, and Security. The following steps outline the installation process:

1. Double-click the View Connection Server installer EXE file to launch the installer.
2. In the **Welcome to the Installation Wizard for VMware View Connection Server** window, click on **Next**.

3. Review the VMware End User License Agreement, select the **I accept the terms in the license agreement** radio button, and click on **Next**.
4. Select the installation directory and click on **Next**.
5. Select **View Transfer Server** as shown in the following screenshot, and then click on **Next**:



6. Update the **Transfer Server Configuration** fields if required with the **Network Domain**, **Server Name**, and **Administrator's Email Address**. The following screenshot shows the information as entered in our sample environment:

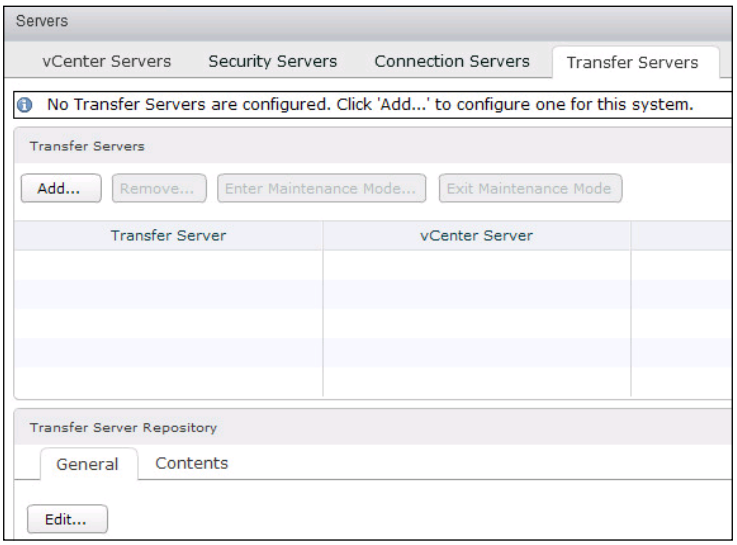
7. Select either the **Configure Windows Firewall automatically** or **Do not configure Windows Firewall** radio button and click on **Next**. If the option **Do not configure Windows Firewall** was selected, configure the firewall manually using the settings provided earlier in the chapter.
8. Review the final installation screen to ensure that the installation directory is correct. If changes are needed, click on the **Back** button to reach the necessary configuration screen and make the required changes. Assuming that the settings are correct, click on **Install** to begin the automated installation process.
9. Click on **Finish** when prompted at the completion of the installation process.
10. The installation process will install all the components required for the View Transfer Server. The final configuration steps will be completed in the View Manager Admin console and will be detailed in the next section.

Configuring the View Transfer Server repository

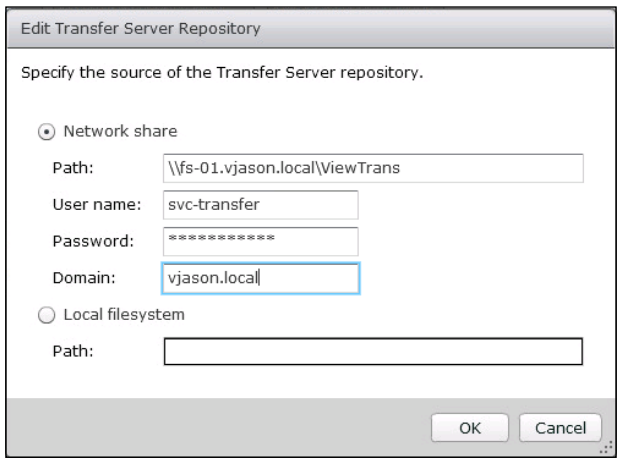
Once the installation of the View Transfer Server has been completed, we need to log on to the View Manager Admin Console to specify the Transfer Server repository location. The following steps outline the process used to specify the repository location:

It is not explicitly required to add the repository prior to linking the View Transfer Server to the View environment. If you link the Transfer Server first, you will need to place it into maintenance mode prior to adding the repository location.

1. Log on to the View Manager Admin Console using an AD account that has administrative permissions within View.
2. Open the **View Configuration** | **Servers page** within the console.
3. Select the **Transfer Servers** tab in the **Servers** window.
4. Under the **Transfer Server Repository** section of the **Transfer Servers** tab, click on the **Edit** button as shown in the following screenshot to open the **Edit Transfer Server Repository** window:



5. In the **Edit Transfer Server Repository** window, click on the **Network share** radio button and populate the fields as shown in the following screenshot. Our example uses a file share as the repository; if a local folder will be used, it is specified in this window as well. In the **Path** field, provide the **Universal Naming Convention** (**UNC**) path to the share that will be used as the repository. Populate the **User name**, **Password**, and **Domain** fields, using the AD account that has read/write access to the share. Click on **OK** to complete the process and add the repository.



If you plan to use a folder located on the Transfer Server as the repository, click on the **Local filesystem** radio button and provide the path to the folder on the Transfer Server, such as `D:\Repository`.

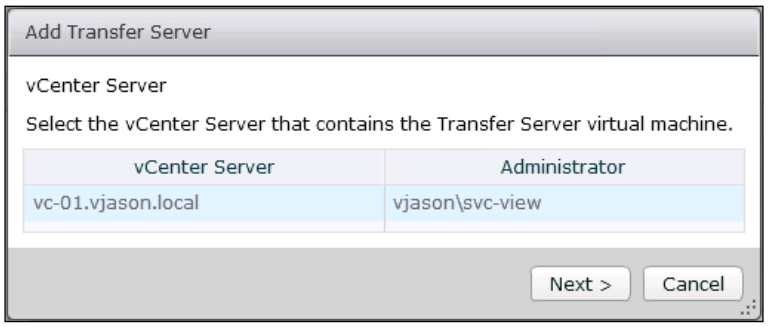
The Transfer Server repository should now be displayed in the **Transfer Server Repository** section of the **Transfer Servers** tab in the View Manager Admin console. The final configuration of the View Transfer Server repository will occur when you link the Transfer Server to the View environment, which we will do in the next section.

Linking the View Transfer Server

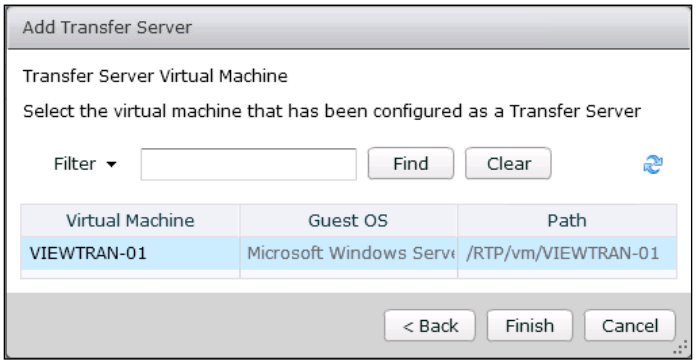
Once the Transfer Server repository has been added in the View Manager Admin console, we need to link the Transfer Server to the View environment. The following steps outline the configuration process:

1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.

2. Open the **View Configuration** | **Servers** page within the console.
3. Select the **Transfer Servers** tab in the **Servers** window.
4. Under the **Transfer Servers** section of the **Transfer Servers** tab, click on the **Add** button to open the **Add Transfer Server** window. Refer to step 4 in the *Configuring the View Transfer Server repository* section for an example of the layout of the **Transfer Servers** tab.
5. In the **Add Transfer Server** window, select the vCenter Server that manages the vSphere host where the Transfer Server virtual machine resides and click on **Next**. In the following screenshot we have selected the **vc-01.vjason.local** vCenter Server:

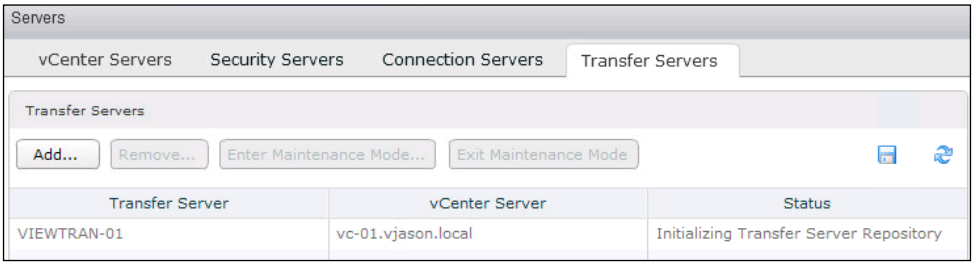


6. Select the Transfer Server from the list of virtual machines and click on **Finish** to return to the **Transfer Servers** tab of the **Server** window. In the following screenshot we have selected the Transfer Server named **VIEWTRAN-01**.



7. The transfer server should now be listed in the **Transfer Servers** section of the **Transfer Servers** tab in the View Manager Admin console.

   Once a Transfer Server is linked to the View environment it will attempt to initialize the Transfer Server repository. As this happens, the **Status** of the Transfer Server will transition from **Pending**, to **Initializing Transfer Server Repository**, and finally to **Ready**. This status can be viewed in the **Transfer Servers** section of the **Transfer Servers** tab in the View Manager Admin console, as shown in the following screenshot:



   Once the **Ready** status has been achieved, the Transfer Server is ready for use.

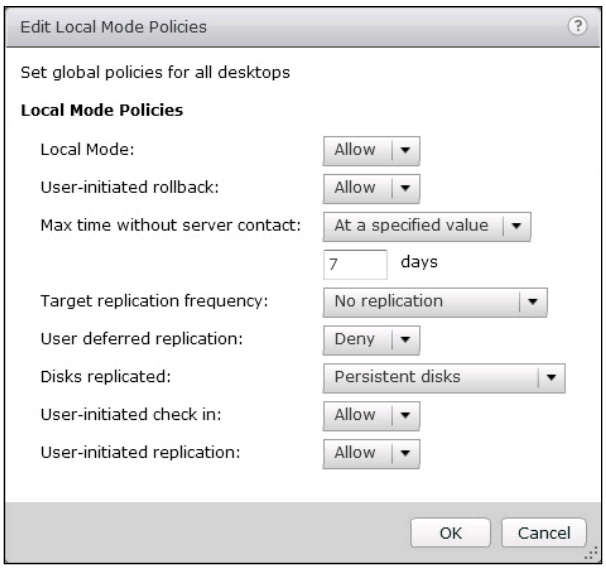
Any errors that occur during the initialization process are likely related to permissions, DNS name resolution, or file share configuration. Refer to the the VMware document VMware Horizon View Administration (`http://www.vmware.com/support/pubs/view_pubs.html`) for a full list of Transfer Server states and their description.

Enabling Local Mode

The final step in the configuration process of the View Transfer Server is to enable Local Mode within the View Manager Admin console. By default, Local Mode is disabled and View Clients are not presented with the option to check out Local Mode desktops. The following steps outline how to enable Local Mode:

1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Policies** | **Global Policies** page within the console.
3. Under the **Local Mode Policies** section of the **Global Policies** page, click on the **Edit Policies** button to open the **Edit Local Mode Policies** window.

4. In the **Edit Local Mode Policies** window, click on the drop-down menu for **Local Mode:** and select **Allow** as shown in the following screenshot:



5. Click on **OK** to close the **Edit Local Mode Policies** window and enable Local Mode. The **Global Policies** page should now show the policy if **Local Mode** is set to **Allow**.

View Clients with the Local Mode option will now be given the option to check out desktops upon login to View.

The screenshot displayed in step 4 of *Enabling Local Mode* shows all of the options that apply to Local Mode clients. For information about each of these policies refer to the *Appendix, VMware View Design and Maintenance*.

View Transfer Server backup

The View Transfer Server has minimal backup requirements. The following components should be backed up in order to preserve the ability to fully restore the Transfer Server:

- View Transfer Server SSL certificate
- View Transfer Server repository

Details on how to back up these items will be discussed in the next two sections.

Backing up the View Transfer Server SSL certificate

The View Transfer Server SSL certificates are stored in PKCS#12 format within the Transfer Server installation folder. The default filenames are `server.crt` and `server.key`, and are stored in the `Program Files\VMware\VMware View\Server\httpd\conf` directory.

There is no reason to backup the default self-signed certificates that are installed with the View Transfer Server unless measures were taken to get the View clients to trust them, such as importing them into the client trusted root certificate store. If custom certificates were installed using the process described in *Chapter 12, Managing View SSL Certificates*, the corresponding `.crt` and `.key` files should be backed up as part of the Transfer Server backup plan.

Backing up the View Transfer Server repository

The Transfer Server repository should be backed up in its entirety, at the root level of the repository folder. The repository should be backed up as often as is needed to ensure that the backup contains the current linked-clone image. As mentioned previously, if full desktops are being used for Local Mode the repository is not used.

If the View infrastructure uses multiple Transfer Servers, and each has its own local repository, each repository should be backed up.

View Transfer Server recovery

The settings for the View Transfer Server and the Transfer Server repository are stored independently within the View Connection Server AD LDS database; as such they are restored separately, and only if required. The restore process for both components is described in the next two sections.

Restoring the View Transfer Server SSL certificate

The View Transfer Server SSL certificates can be restored as part of the installation of a replacement View Transfer Server. The following steps outline the process used to restore the SSL certificates:

1. If required, reinstall the View Transfer Server software on a server with the same name as the previous Transfer Server.

2. Stop the **VMware View Transfer Server** service.
3. Restore the `server.crt` and `server.key` files to the `Program Files\VMware\VMware View\Server\httpd\conf` directory.
4. Start the **VMware View Transfer Server** service.
5. Use a web browser to access the Transfer Server web page over https and verify that the expected SSL certificate is being presented.

If you are restoring a custom SSL certificate, and your `.crt` and `.key` files are named something other than `server.crt` and `server.key`, you will also need to update the `mod_vprov.conf` file located in the `Program FIles\VMware\VMware View\Server\httpd\conf` directory with the correct filenames. The file should be modified prior to starting the **VMware View Transfer Server** service.

Restoring the View Transfer Server repository

The View Transfer Server repository can be restored using a traditional file restore process. To preserve the existing repository configuration, it is important to restore the repository using a backup set that contains the current repository contents and to the expected file share location, as configured in the View Manager Admin console.

Summary

In this chapter, we have been introduced to the VMware Horizon View Transfer Server, a feature of View that provides organizations with even more options and use cases for deploying Virtual Desktops. We have learned what is required to deploy and configure a View Transfer Server, what the limits of a Transfer Server are, and where the Transfer Server fits in within the View Infrastructure.

We also discussed which components of the Transfer Server need to be backed up, namely the files related to the SSL certificates and the Transfer Server repository.

We concluded this chapter by discussing how to restore both the View Transfer Server and Transfer Server repository.

In the next chapter, we will implement View Security Server, the component of a View installation that enables secure remote access to View desktops.

5
Implementing VMware Horizon View Security Server

VMware Horizon View Security Server is a core feature of the View platform that enables secure remote access to desktops, without the need to use a **virtual private network** (**VPN**) connection or provide direct access from the Internet to the View Connection Server. The View Security Server is a specialized installation of View Connection Server that serves as the connection point between remote View Clients and View desktops hosted on a private network.

This chapter will discuss the installation, configuration, backup, and recovery of the View Security Server.

By the end of this chapter, we will learn:

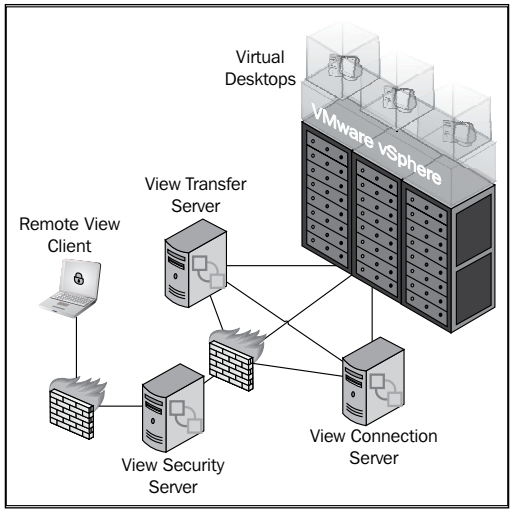
- An overview of View Security Server
- The hardware and operating system requirements of View Security Server
- The connection limits of a View Security Server
- How to determine the number of View Security Servers required
- View Security Server network protocol and port usage
- View Security Server preinstallation tasks
- How to install a View Security Server
- How and what components of the View Security Server to backup
- How to restore a View Security Server from backups

VMware Horizon View Security Server overview

The View Security Server is a type of View Connection Server that is designed to add an additional layer of security between remote View Clients and View resources that are located on a private network. Rather than provide remote View Clients with direct access to the View Connection Server, organizations can deploy a View Security Server within a DMZ or other secure network to provide secure remote access to View-managed resources. Some of the functions and features of the View Security Server include:

- Provides remote View Clients with their own dedicated View connection broker, ensuring an optimal user experience
- Brokers connections between remote View Clients and internal View-managed resources
- Authenticates user connection requests
- Supports RSA SecurID and RADIUS for enabling optional two-factor user authentication
- Can be placed in a DMZ to further isolate the Security Server from the private network
- Does not need to be a member of an Active Directory domain

The following diagram shows the placement of a View Security Server in a simple View environment. The View Security Server brokers access to a number of different components of the private View infrastructure, each of which is shown in the diagram:



The View Security Server authenticates the clients by contacting the View Connection Server, and then provides them with access to the entitled resources including View desktops or even the View Transfer Server, if Local Mode desktops are being used. View Clients with Local Mode have the same access to their desktops over a Security Server as they do when logging on while on the private network.

View Security Server requirements

View Security Server requires the same hardware and software configuration as the View Connection Server, described in the *VMware Horizon View Connection Server requirements* section of *Chapter 2*, *Implementing VMware Horizon View Connection Server*.

Security Server limits

The View Security Server shares the same limits as the View Connection Server.

Connection Type	Maximum Simultaneous Connections
To virtual desktop: • RDP-tunneled connection • PCoIP-tunneled connection through View Security Server	2,000

Each View Security Server is a standalone instance; therefore, there is no specific guidance with regard to how many can be deployed. A View Security Server can only be paired with one View Connection Server while a Connection Server can be paired with multiple Security Servers. When implementing View Security Servers, multiple instances should be deployed to meet capacity and availability requirements.

Security Server additional considerations

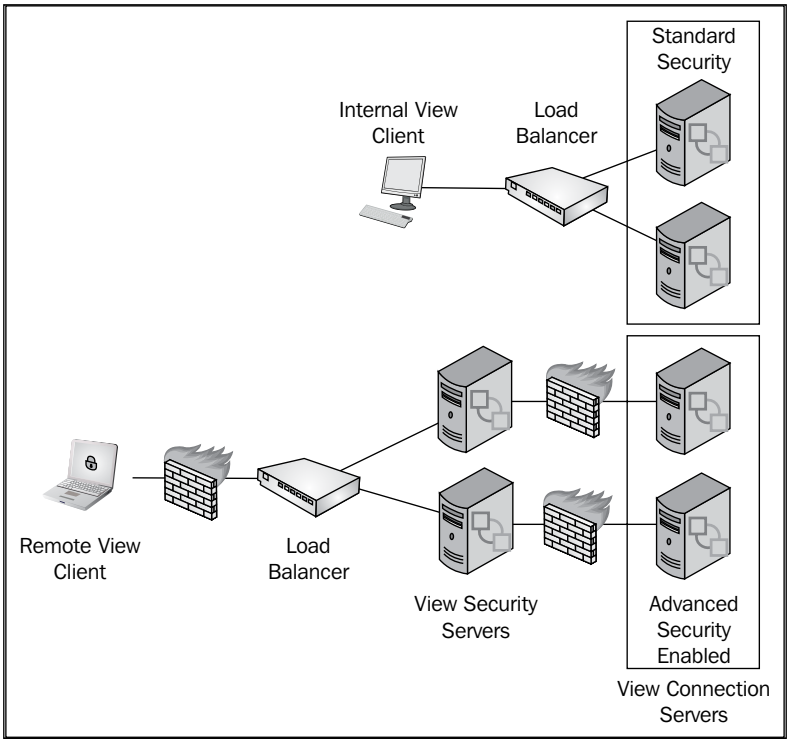
The following are additional considerations that should be kept in mind when deploying a View Security Server:

- If you require Windows IPsec encryption to be applied to the network traffic between the View Security Server and the View Connection Server, the Windows firewall service must be enabled for both hosts in order for View to create the required Windows IPsec policies. The firewall service is enabled by default; if it was disabled, visit the Microsoft TechNet article *Managing Windows Firewall with Advanced Security and IPsec* (`http://technet.microsoft.com/en-us/library/cc732283(v=ws.10).aspx`) for information about how to enable it. It is recommended to enable the firewall service prior to the installation of any View software component, as the installer will then automatically configure the appropriate settings.
- Like View Connection Servers, View Security Servers have no native load balancing functionality. It is recommended to implement some sort of load balancing functionality to help balance the client connections across all the View Security Servers in your infrastructure. Refer to the *Load Balancing Connection Servers* section in *Chapter 2*, *Implementing VMware Horizon View Connection Server* for information about load balancing options.
- When installed, the View Security Server is configured with a self-signed SSL certificate which will not be trusted by View Clients. It is recommended to replace the self-signed certificate with one issued from an internal or commercial certificate authority that the View Clients will trust. *Chapter 12*, *Managing View SSL Certificates* will provide the process used to replace the default SSL certificates for all View components.
- Options such as tunneling connections and two-factor authentication are set on a per-Connection Server basis. If either of these options is going to be used, and you do not want to subject internal View Clients to the additional security measures, you are required to deploy additional Connection Servers with these settings enabled to be used solely with the View Security Servers.

The following diagram illustrates a View infrastructure that meets the following three requirements:

- Internal View Clients using load-balanced connections to Connection Servers
- Remote View Clients using load-balanced connections to Security Servers
- Security Servers installed in a DMZ
- Two-factor authentication or connection tunneling policies that apply only to remote View Clients

The diagram does not show the connections to the View desktops; it is only meant to illustrate the placement of load balancing appliances and how true high-availability might be achieved in an environment that includes multiple View Security Servers. In addition, it shows that additional Connection Servers are being used for internal clients, as these connections do not require the same security settings as the remote clients do.



This View architecture ensures that View Client connections will be maintained if either of these two scenarios were to occur:

- Failure of any one of the four Connection Servers shown in the diagram
- Failure of any one of the Security Servers

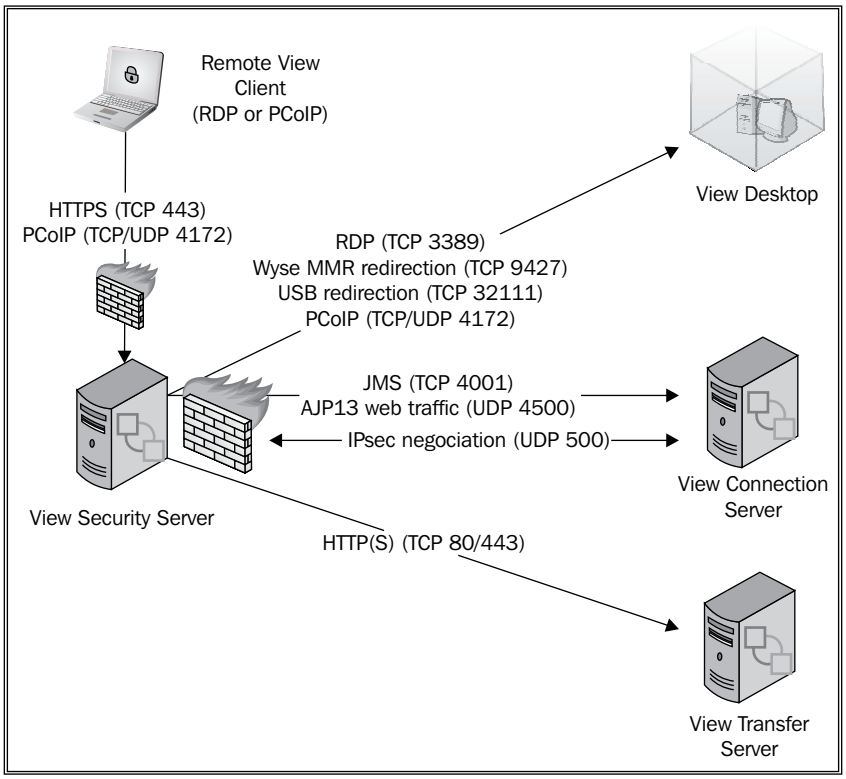
As a single View Security Server cannot be paired with more than one Connection Server, there is no need to place a load balancer between the Security Servers and the Connection Servers. Load balancing the Security Servers ensures that the View Client connection will be maintained regardless of which server fails, be it a Security Server or the Connection Server that it is paired to.

View Security Server networking

The following diagram illustrates how the primary protocols used by the View Security Server work with other components of the View infrastructure. The diagram shows the following components of a View infrastructure:

- Communication between the View Security Server and the View desktops
- Communication between the View Security Server and the View Transfer Server
- Communication between the View Security Server and the View Connection Server

The arrows indicate the direction in which each protocol travels, assuming that the default settings are used.



The diagram shows only the core components of a View infrastructure that includes a View Security Server.

This list of ports used by the core components are outlined in the following table. Refer to *Chapter 2, Implementing VMware Horizon View Connection Server* for a full list of protocols used including those dedicated to client connections.

Protocol or Service	Port	Notes
AJP13 (Apache Tomcat Connector)	TCP 8009	Not used if IPsec is enabled and the DMZ backend firewall uses one-way or two-way NAT.
HTTP/HTTPS	TCP 80/443	
JMS (Java Messaging Service)	TCP 4001	
MMR (Multimedia redirection)	TCP 9427	Used alongside RDP; uses client rather than server resources to render DirectShow-based media and codecs.
PCoIP	TCP/UDP 4172	
RDP	TCP 3389	
IPsec	UDP 500	Used for IPsec negotiation.
USB Redirection for PCoIP and RDP	TCP 32111	TCP 32111 is used to support USB redirection to View clients.

If the DMZ backend firewall uses one-way or two-way NAT, and IPsec is enabled, UDP port 4500 must be allowed in each direction between the Security Server and the View Connection Server. This is used in place of the AJP13 protocol mentioned in the preceding table.

Installing and configuring View Security Server

The installation and configuration process for the View Security Server requires some amount of preparation. This section will outline what is required prior to beginning the installation.

Installation prerequisites

There are a number of prerequisites that should be addressed prior to installing a View Security Server:

- At least one configured View Connection Server with a license key installed
- A dedicated Windows 2008 R2 server to host the View Security Server role

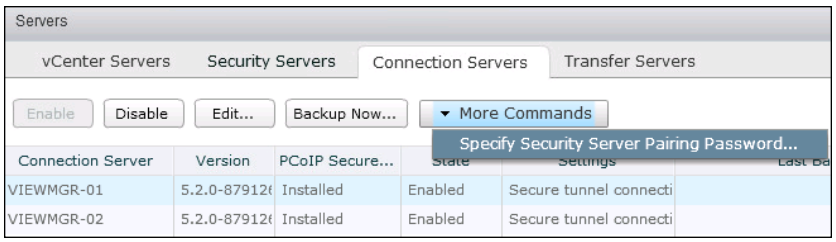
- A static IP address for the Security Server host
- The second network controller and static IP address (optional), so that the host server can be configured with dedicated connections for both public facing (external) and private (internal) networks
- The Security Server host should be able to resolve the FQDN of the Security Server it will pair with, either using DNS or the local host's file
- A valid View Connection Server pairing password
- Firewall access between the View Security Server and the necessary View components on the private network
- Firewall access between the Internet and the View Security Server
- A resolvable public URL that will be used for accessing the View Security Server
- Local administrator access on the host server

In addition to the items described in *Chapter 1*, *Designing a VMware Horizon View Infrastructure*, the following items should be prepared in advance of the installation.

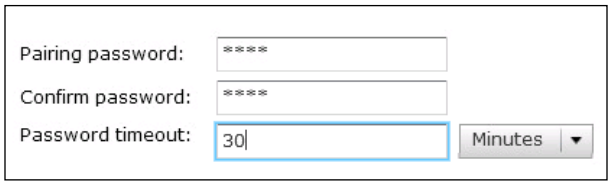
View Security Server Pairing Password

The View Security Server is paired to a View Connection Server using a password that is specified in the View Manager Admin console. This password is entered during the installation of the Security Server, and enables secure communication between it and the Connection Server on the private network. The following steps outline how to generate the password:

1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **View Configuration** | **Servers page** within the console.
3. Select the **Connection Servers** tab in the **Servers** window.
4. Highlight the Connection Server that you wish to pair with the Security Server, click on the **More Commands** button, and select **Specify Security Server Pairing Password**. In the following screenshot, we have highlighted the **VIEWMGR-01** Connection Server:



5. In the **Specify Security Server Pairing Password** window, specify a password and the amount of time it will be valid for. Click on **OK** when finished. The following screenshot shows the **Pairing password:**, Confirm **password:**, and **Password timeout:** fields:



With the password specified, the installation of the Security Server can now proceed. In the event that the installation cannot be completed prior to the password expiring, simply generate a new password.

Deploying a View Security Server

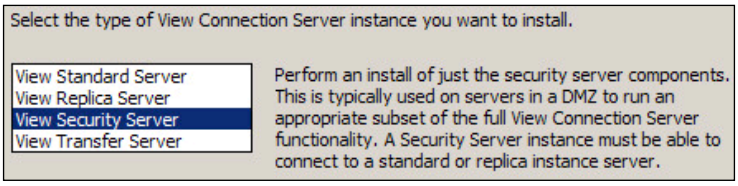
Deploying a View Security Server is broken down into two stages: the installation of the View Security Server software, and the final setup using the View Manager Admin console.

Installing a View Security Server

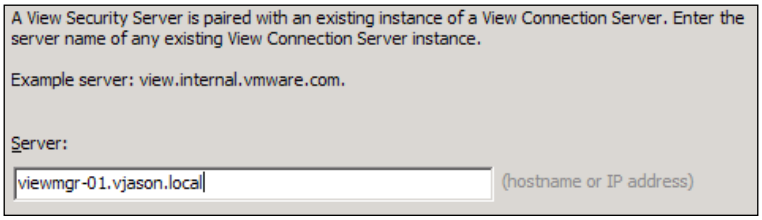
The View Security Server software is delivered as a single executable (EXE) file, named in a format similar to `VMware-viewconnectionserver-x86_64-x.x.x-yyyyyy.exe`. The following steps outline the installation process:

1. Double-click the View Connection Server installer EXE file to launch the installer.
2. In the **Welcome to the Installation Wizard for VMware View Connection Server** window, click on **Next**.
3. Review the VMware End User License Agreement, select the **I accept the terms in the license agreement** radio button, and click on **Next**.
4. Select the installation directory and click on **Next**.

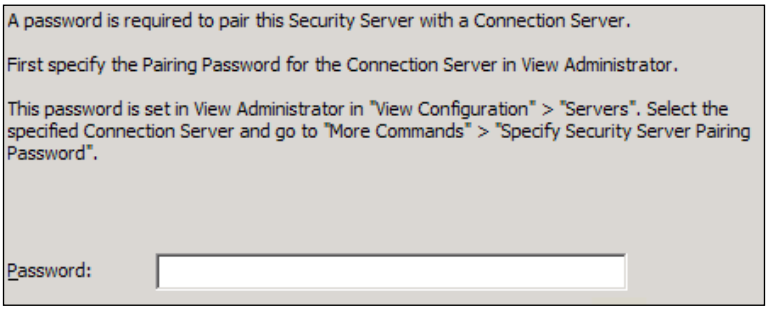
5. Select **View Security Server** as shown in the following screenshot, and then click on **Next**.



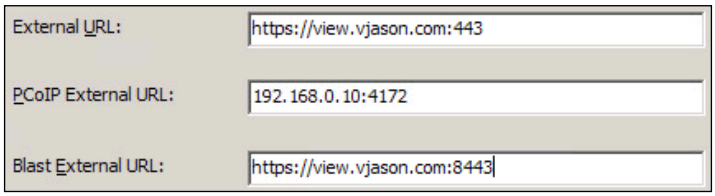
6. Enter the name of the View Connection Server that the View Security Server should be paired with in the **Server:** field and click on **Next**.



7. In the **Password:** field, enter the Security Server Pairing Password that was specified earlier and click on **Next**.



8. Enter each of the publicly resolvable URLs as requested in the **External URL:**, **PCoIP External URL:**, and **Blast External URL:** fields. Click on **Next** when complete.



- The PCoIP External URL must be entered as an IP address that remote clients will use to access the View Security Server.
- The Blast External URL is only used if you have installed the VMware Horizon View Feature Pack 1 and configured your desktop pools to allow HTML client connections. The feature pack is discussed in *Chapter 15*, *VMware Horizon View Feature Pack 1* (available for free download from the Packt Publishing website).

9. Select either the **Configure Windows Firewall automatically** or **Do not configure Windows Firewall** radio button and click on **Next**. If the option **Do not configure Windows Firewall** was selected, configure the firewall manually using the settings provided earlier in the chapter.
10. Review the final installation screen to ensure that the installation directory is correct. If changes are needed, click on the **Back** button to reach the necessary configuration screen and make the required changes. Assuming that the settings are correct, click on **Install** to begin the automated installation process.
11. Click on **Finish** when prompted at the completed of the installation process.
12. The installation process will install all the components required for View Security Server. The same process can be used to install additional View Security Servers, although a new Security Server Pairing Password would need to be generated as each is only valid one time.
13. The final configuration steps will be completed in the View Manager Admin console, and will be detailed in the next section.

Updating the View Security Server settings

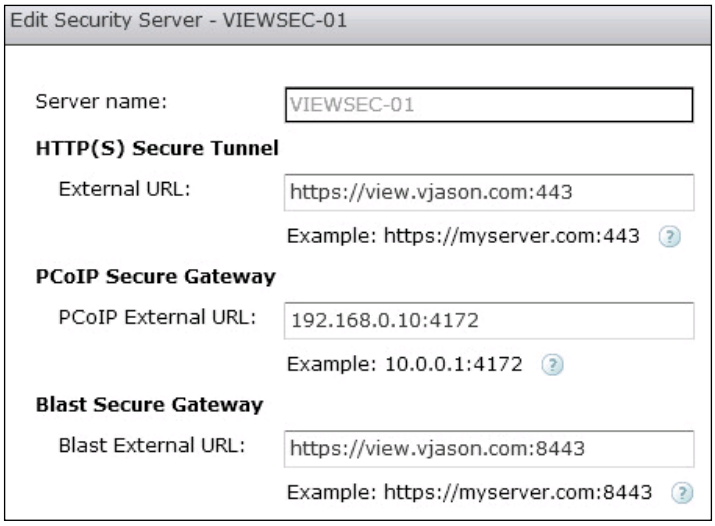
Once paired to a View Connection Server, the Security Server settings can be changed using the View Manager Admin console. The following sections illustrate where within the Admin console you update the Security Server Settings.

Security Server options

The following steps outline how to verify or update the Security Server options.

1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **View Configuration** | **Servers** page within the console.
3. Select the **Security Servers** tab in the **Servers** window.

4. Highlight the Security Server you wish to update and click on **Edit** to open the **Edit Security Server** window as shown in the following screenshot:



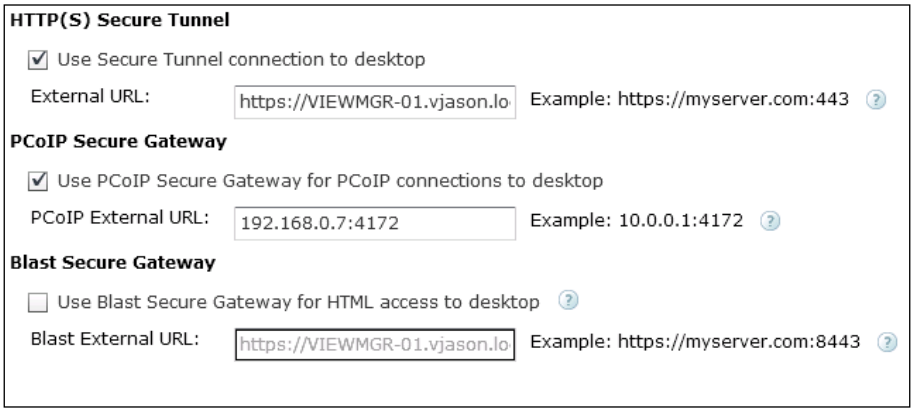
In the event that either the external URL or IP address of the Security Server is changed, it should be changed in this page to ensure that View will function properly.

Enabling Secure Tunnel and Secure Gateway

Remote View clients do not have direct access to their desktops; thus, in order for those clients to gain access, the View Connection Server must be configured with the appropriate PCoIP Secure Gateway and HTTP(S) Secure Tunnel settings. These settings are configured during the pairing of the Security Server, and should be verified once the installation is complete. The following steps outline how to verify or update the Connection Server settings:

1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **View Configuration** | **Servers** page within the console.
3. Select the **Connection Servers** tab in the **Servers** window.

4. Select the Connection Server to which the Security Server is paired and click on **Edit** to open the **Edit View Connect Server Settings** window as shown in the following screenshot:



5. The checkboxes for **HTTP(S) Secure Tunnel** and **PCoIP Secure Gateway** should be checked, and the appropriate information for the Connection Server should be populated as shown in the preceding screenshot. As with the Security Server, the **PCoIP Secure Gateway** field should be populated with the IP address and not the FQDN of the Connection Server.

View Security Server backup

A View Security Server contains no information about the configuration of the View installation, and therefore has no backup requirements. Assuming that the self-signed SSL certificate was replaced with one from a trusted internal or commercial certificate authority, it is important to maintain a backup of that certificate that includes the private key.

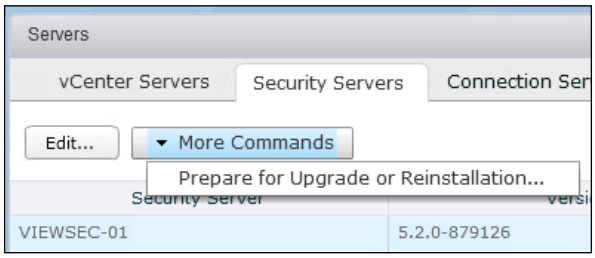
As the View Security Server is a publicly accessible server, you may wish to back up the log files on a regular basis. These files are located in the same folder on every type View Connection Server: `%ALLUSERSPROFILE%\Application Data\VMware\VDM\logs`

Chapter 13, *Implementing VMware Horizon View Group Policies*, outlines how to change the log configuration settings using the View group policy templates.

View Security Server recovery

The simplest way to restore a View Security Server is to simply reinstall the software using the steps provided earlier in this chapter, and re-pair the Security Server with the View Connection Server. The following steps outline how to restore a View Security Server in a scenario where the previous one is unavailable:

1. Configure a new Security Server host using the same server name and IP address.
2. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
3. Open the **View Configuration** | **Servers** page within the console.
4. Select the **Security Servers** tab in the **Servers** window.
5. Highlight the Security Server you wish to remove, click on the **More Commands** option, and then the **Prepare for Upgrade or Reinstallation** button as shown in the following screenshot:



6. Click on **OK** in the **Warning** window to dissociate the Security Server from the Connection Server.

This process permanently dissociates the Security Server from the View Connection Server. Once selected, the only way to restore the connection is to reinstall the View Security Server software.

7. Generate a new Security Server Pairing Password using the process outlined earlier in this chapter.
8. Install the Security Server software using the process outlined earlier in this chapter.
9. Verify the Security Server settings in the View Manager Admin console and test remote View client connections.

Summary

In this chapter, we have been introduced to the VMware Horizon View Security Server, a feature of View that provides organizations with the ability to provide secure remote access to View desktops. We have learned what is required to deploy and configure a View Security Server, what the limits of a Security Server are, and where the Security Server fits in within the View infrastructure.

We also discussed what components of the Security Server need to be backed up, which are custom SSL certificates as well as the Security Server logs.

We concluded this chapter by discussing how to restore a View Security Server.

In the next chapter, we will discuss VMware ThinApp, a software package included with View that enables the virtualization of desktop applications into single executable files that are isolated from one another, can be deployed within View, and can be easily updated.

6
Using VMware ThinApp

VMware ThinApp is an application virtualization platform that provides organizations with an alternative means of application deployment, maintenance and migration. Applications being deployed using ThinApp run isolated from one another and the host operating system, eliminating common application conflicts and adding a layer of security between the application and the host operating system.

This chapter will discuss how to use VMware ThinApp to capture, package, and deliver an application.

In this chapter we will learn:

- An overview of VMware ThinApp
- The benefits and limitations of ThinApp
- ThinApp requirements
- How to capture an application with ThinApp
- How to deploy ThinApp packages within VMware Horizon View
- How to update an application that was packaged with ThinApp

VMware ThinApp is a very powerful utility that can be used for a number of different purposes, as well as with other VMware products such as Horizon Workspace. The *VMware ThinApp User's Guide* is 100 pages long by itself—a testament to how much there is to learn about the product.

This chapter will focus on common ThinApp use cases and administrative tasks that even the smallest View environment may find useful. To learn about the full capabilities of the ThinApp suite, refer to the following documents available at the *VMware ThinApp Documentation* page (`http://www.vmware.com/support/pubs/thinapp_pubs.html`):

- *Streaming Execution Mode – Application Streaming with VMware ThinApp*
- *ThinApp Package.ini Parameters Reference Guide*
- *ThinApp User's Guide*

An overview of VMware ThinApp

VMware ThinApp virtualizes an application by capturing all the changes that occur when that application is installed, and packaging those files and Windows registry settings into a self-contained package. This process is described in more details in the *Capturing an Application with ThinApp* section of this chapter. The resulting ThinApp application contains not only this application data, but also the software used to create the virtualization layer that isolates the application from the host operating system and any other applications that are in use. These are some advantages of using ThinApp to package and deliver applications:

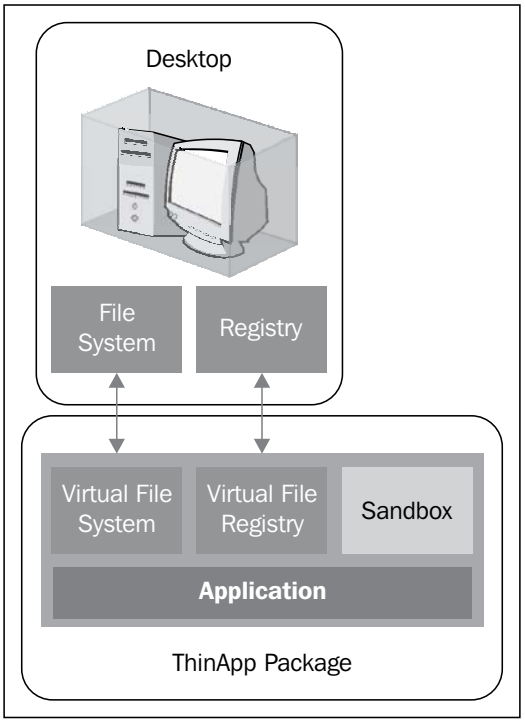
- Each application packaged using ThinApp runs within its own virtualization layer, eliminating the potential for conflict with other applications and the host operating system.
- Deploying applications using ThinApp reduces the size of the master virtual desktop image, as well as the need to maintain that application within the context of that image. Rather than updating the application within the master image, or within the already deployed virtual desktop, you can simply update the ThinApp package.
- With ThinApp, users can run different versions of the same application, such as Internet Explorer or even the Microsoft Office suite.
- Applications deployed using ThinApp are completely portable and require no additional application software to run. Once packaged, ThinApp applications can be used anywhere including physical desktops, streamed over the network, or even run from a portable USB drive.
- ThinApp can be used to capture older versions of Internet Explorer, such as IE6, to be used on newer versions of Windows. Refer to the *Virtualizing Internet Explorer 6 with ThinApp 4.6 (1026565)* article (`http://kb.vmware.com/kb/1026565`) for instructions on how to use ThinApp to capture IE6.

There are some restrictions on how ThinApp can be used, or what types of applications can be captured. The following list represents some of the restrictions of the ThinApp platform:

- Where possible, for vendor support reasons, ThinApp should be used to package only applications that have native support for the host Windows operating system.

- Using ThinApp does not alter the vendor application licensing requirements or operation. Due to the way some applications handle license activation, it is possible that they will not be suitable for ThinApp. Many applications such as Microsoft Office and Adobe Acrobat support volume licensing, which often uses different activation methods that are more suitable when using ThinApp. Consult your application vendor for more information about software license and activation options.
- Application vendors may not support the use of ThinApp to virtualize their applications, regardless of whether or not it appears to work.
- ThinApp cannot be used to virtualize the applications that include Windows kernel-mode drivers. These drivers can however be installed directly to the desktop that the ThinApp-delivered application can access.
    - ODBC drivers are user mode drivers, and therefore are supported by ThinApp.
    - Examples of applications that use kernel-mode drivers include antivirus software, firewall software, and some VPN clients.

The following diagram shows the relationship between the application packaged using ThinApp, and the desktop that is executing the application:



The ThinApp-delivered application runs entirely self-contained, with its own virtualized versions of the filesystem and Windows registry. This isolation is what prevents these applications from interfering with other applications or the host OS, as those changes occur only within the ThinApp virtualization layer.

ThinApp-supported Windows operating systems

ThinApp supports a number of different Microsoft Windows operating systems. The full list of operating systems supported by ThinApp is as follows:

- **32-bit platforms**: Windows 8, Windows 2000, Windows XP, Windows XPE, Windows 2003 Server, Windows Vista, Windows Server 2008, and Windows 7
- **64-bit platforms**: Windows 8, Windows XP 64 bit, Windows 2003 64 bit, Windows Vista 64 bit, Windows Server 2008 64 bit, Windows Server 2008 R2 64 bit, and Windows 7 64 bit

While ThinApp supports a number of different operating systems, there are specific limitations on application support. These limitations include:

- 16-bit applications will run only on a 32-bit Windows operating system
- 32-bit applications will run on either a 32-bit or 64-bit Windows operating system
- Technologies such as Citrix XenDesktop or Microsoft Terminal Server are supported, enabling applications packaged using ThinApp to be used with more than just View desktops

Refer to the *VMware ThinApp Licensing information (1006248)* article (`http://kb.vmware.com/kb/1006248`) for more information about how to license ThinApp in environments that feature more than just View desktops.

In addition, ThinApp supports Japanese applications assuming that they are captured on, and run from, a Japanese version of the Windows operating system.

ThinApp does not currently support the capturing of 64-bit Windows applications.

ThinApp recommendations

There are a number of recommendations when working with ThinApp to ensure that the applications packaged with ThinApp function properly on the target computer.

Version of the operating system

In the event that the applications packaged with ThinApp will be executed on multiple versions of Windows, it is recommended you capture the application on the earliest version of Windows that is required. For example, if our environment has both Windows 7 and Windows 8 desktops, we would capture the application under a clean installation of Windows 7.

Regardless of which Windows OS version is used to capture an application using ThinApp, once you have packaged it using ThinApp, you should test it on all OS versions that are present within your environment to ensure that it works as expected.

ThinApp capture desktop

The computer that will be used to capture applications using ThinApp should be as clean as is possible, meaning that it should have only the base operating system installed. Ideally, this desktop should be representative of the master virtual desktop image that will be used to deploy the virtual desktops.

It is suggested that the ThinApp capture desktop should be a virtual machine, as you can use virtual machine snapshots to quickly roll back any changes that were made during the application capture process. ThinApp includes a license for VMware Workstation, which enables you to run the capture desktop as a virtual machine on top of an existing Windows host while also supporting virtual machine snapshots. Additional information about VMware Workstation requirements and features can be found at `http://www.vmware.com/products/workstation/`.

One example of why this is important would be the Microsoft .Net Framework. If the capture computer has the .Net Framework installed, and the application requires it, the resulting captured application will only work on computers that have it installed as well.

It is also possible to link together two ThinApp applications using a ThinApp feature called Application Link, which can be used to address application dependencies without needing to install additional software in the master desktop image. This advanced ThinApp topic is discussed in detail in the *VMware ThinApp User's Guide* available at `http://www.vmware.com/support/pubs/thinapp_pubs.html`.

The ThinApp Administrative Workstation

The ThinApp capture process generates not only the self-contained version of the target application, but also a collection of files and folders called a ThinApp project that the administrator can use to further customize or update the ThinApp package. It is recommended you install ThinApp on a second workstation that can be used to process any changes or updates to the ThinApp packages.

The ThinApp administrative workstation can also host the ThinApp capture desktop, assuming that it meets the requirements needed to run VMware Workstation. Consult the *VMware Workstation System Requirements* guide for a full list of system requirements (`http://www.vmware.com/products/workstation/support.html`).

ThinApp common terms

ThinApp uses a number of different terms that are in no way related to View itself. The following is a list of key ThinApp terms that will be used throughout this chapter:

Term	Definition
Application template	Used to assign one or more ThinApp packages to an individual desktop or desktop pool.
`build.bat`	The `build.bat` file is used by the ThinApp administrator to rebuild the ThinApp package to integrate any changes or updates that were made.
Deployed execution mode	Deployed execution mode ThinApps are copied to, and executed from, the local desktop. The deployed execution mode should not be used with linked-clone desktops, as it will cause each desktop to increase in size by the amount of space needed to install the application.

Term	Definition
Entry point	Points of entry into a ThinApp application; typically an `.exe` or `.com` file. When building a ThinApp package, you have to specify what entry points you want to enable.
Merged isolation mode	This grants ThinApp applications permissions to read and write from local nonsystem directories on the desktop. All other changes will be stored in the sandbox. This option is recommended for Windows Logo Certified applications such as Microsoft Office.
`Package.ini`	The `package.ini` file is located in the root of a ThinApp project folder, and is used to edit a number of different options related to the configuration of a ThinApp package.
Project	A project folder is a folder that contains the results of a ThinApp capture, and can be used to rebuild the ThinApp package.
Sandbox	This is a component of a ThinApp where user changes or customizations are stored. The sandbox location is specified during the creation of the ThinApp package, and can reside within the user's Windows profile, on a remote CIFS share, or in the same directory as the ThinApp application itself, such as when it is executed from a USB device.
`Sbmerge.exe`	`Sbmerge.exe` is a command-line utility used to perform maintenance of ThinApp projects. The most common use of this utility is to update a ThinApp project folder after the installation of application updates.
Streaming execution mode	Streaming execution mode ThinApps are typically run from a centralized file share and stream application data to the desktop only when it is needed, on a block-by-block basis directly into memory. Streaming execution mode is optimal for linked-clone desktops, where the goal is to reduce the amount of per-desktop storage required.
ThinApp registration	ThinApp registration is used to make a ThinApp application appear within the desktop similar to a traditionally installed application, adding common shortcuts and other traditional application features.
ThinApp repository	This is a CIFS-based file share used to store applications packaged with ThinApp. Once identified within the View Manager Admin console, the ThinApp packages within the repository can be assigned to individual desktops or View desktop pools.
ThinDirect plugin	The VMware ThinDirect Internet Explorer plugin is installed directly on the virtual desktop and it redirects specified URLs to be loaded only through an alternative browser that has been packaged with ThinApp.

Term	Definition
`ThinReg.exe`	`ThinReg.exe` is a command-line utility that is used to associate file extensions with specific applications. ThinApp administrators may use ThinReg to customize file associations in situations where a given file could be opened by multiple applications on the desktop, including applications packaged with ThinApp.
WriteCopy isolation mode	This mode prevents ThinApp applications from writing to all but the `My Documents` and `Desktop` folders on the desktop. This option is recommended for legacy or untrusted applications.

Installing ThinApp

The installation process for the ThinApp software is straightforward. Obtain the ThinApp license key, and complete the following steps to install the ThinApp software:

1. Download the ThinApp installer to the previously configured capture desktop or administrative workstation.
2. Double-click on the ThinApp executable file.
3. In the **Patent Lists** dialog box, click on **Next**.
4. Accept the license, type the serial number, and type a license display name. The license display name will be displayed when you launch ThinApp packaged applications.
5. Click on **Install** to complete the installation process.

The ThinApp software has now been installed and is ready to perform the capture of an application installation.

Capturing an application with ThinApp

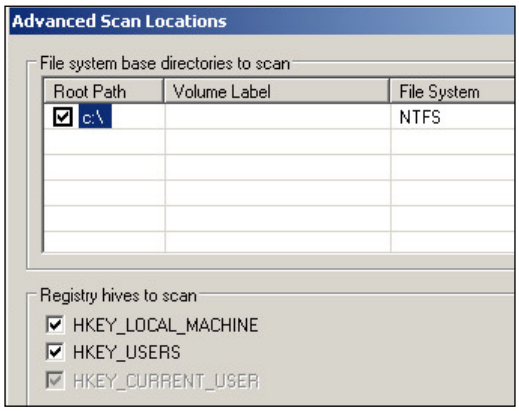
The ThinApp application capture process is straightforward. Simply put, the ThinApp platform records the initial condition of the capture desktop, prompts for the application to be installed, and once the installation is complete, the condition of the desktop is examined again. The changes that occurred during the installation form the basis of what is used to build the ThinApp package.

The ThinApp Packaging Community on the VMware Communities website is a public forum that ThinApp administrators can use to post information about the applications that they have packaged. This community is a great resource for learning more about what was required to use ThinApp to capture a specific software package. The ThinApp Packaging Community is located at `http://communities.vmware.com/thinap.jspa`.

The following steps outline the process used to capture the installation of an application and package it with ThinApp:

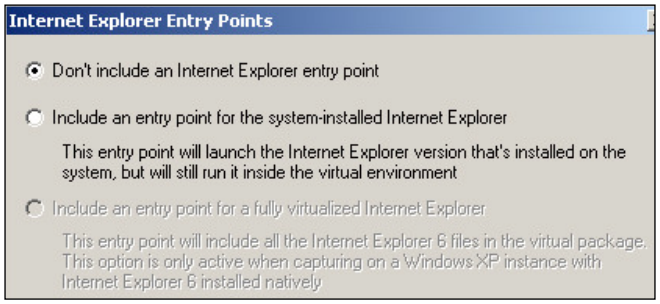
1. Take a VMware snapshot of the ThinApp capture desktop.
2. Log in to the ThinApp capture desktop.
3. From the Windows Start menu, select **All Programs** | **VMware** | **ThinApp Setup Capture**.
4. In the **Setup Capture – Welcome** window, click on **Next**.
5. In the **Setup Capture – Ready to Prescan** window, click on **Prescan**. ThinApp will now scan and record the current configuration of the capture desktop.

   The **Advanced Scan Locations** feature is an optional feature displayed in the following screenshot. This feature provides the ability to exclude certain directories or registry hives from the prescan. Changes to these settings are not usually required.



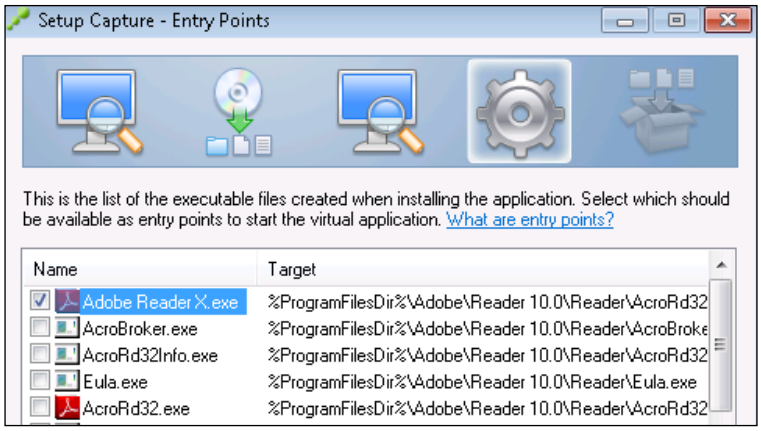
6. Once the prescan has completed, the **Setup Capture – Install Application** window will be displayed, indicating that ThinApp is ready for the target application to be installed.

   The **Internet Explorer Entry Points** feature is an optional feature displayed in the following screenshot. This feature can be used to add an entry point to the ThinApp package for the locally installed Internet Explorer. This enables the locally installed Internet Explorer to be launched within the ThinApp virtual environment, providing the ability to use ThinApp virtualization with the locally installed browser.



7. Install the application using the normal installation process. Complete any additional configuration tasks such as:
   - Running the application once and accepting any license agreements
   - Performing application license entry or software activation tasks
   - Disabling any built-in automatic update capabilities
   - Installing additional components that are required, such as web browser plugins

8. Once the application has been installed and the configuration is completed, click on the **Postscan** button. When prompted, confirm that all installation tasks have been completed. ThinApp will now scan the capture desktop and gather all the changes that occurred during the installation of the application.
9. In the **Setup Capture – Entry Points** window, deselect any entry points that you do not want to be available to the application user. The following screenshot shows some of the entry points that were detected after an installation of Adobe Acrobat Reader. In this example, we want users to have access only to the Adobe Reader executable, so we have unchecked all other potential entry points. Click on **Next** when the desired selections have been made.

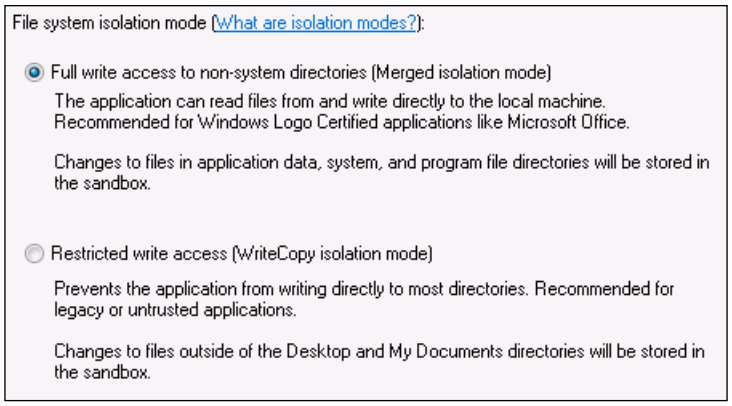


10. If the application being captured is a web browser, the **Setup Capture – Native Browser Redirection** window shown in the following screenshot will be displayed next. This feature works in tandem with a VMware ThinDirect Internet Explorer plugin, and the URLs specified here will load only through the ThinApp delivered browser, and not through the local instance of Internet Explorer. Make any required changes and click on **Next**.

    The ThinDirect plugin and accompanying group policy template can be found in the `Program Files\VMware\VMware ThinApp` directory on any computer with ThinApp installed.

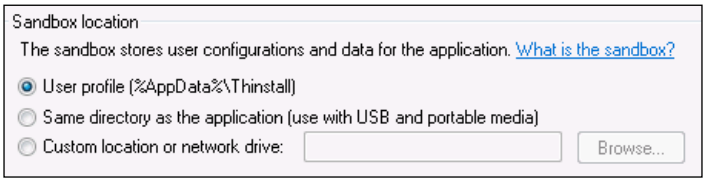
    The `ThinDirect.adm` group policy template can also be used to configure the browser redirection settings, which are configured on a per-URL basis.

11. In the **Setup Capture – Manage with Horizon** window make any required changes. No changes are made if this ThinApp is going to be used only with View.
12. In the **Setup Capture – Groups** window, apply any application restrictions that may be required. By default, all users can use the application; if restrictions are required, AD security groups can be used to grant explicit access. The capture desktop must be a member of the domain to assign access based on AD security groups. By assigning explicit permissions during this step, you ensure that the target group will be able to execute the application regardless of the permissions on the View desktop.
13. In the **Setup Capture – Isolation** window, choose the filesystem isolation mode from one of the two options as described in the *ThinApp common terms* section. Click on either the **Merged isolation mode** or **WriteCopy isolation mode** radio button, as shown in the following screenshot, and click on **Next**.



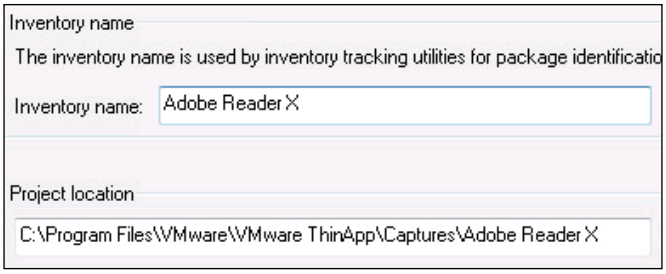
14. In the **Setup Capture – Sandbox** window, choose the Sandbox location from one of the three options, as shown in the following screenshot. The description of each is described in the *ThinApp Common Terms* section.

    The **User profile** option is the default, and acceptable in most cases.



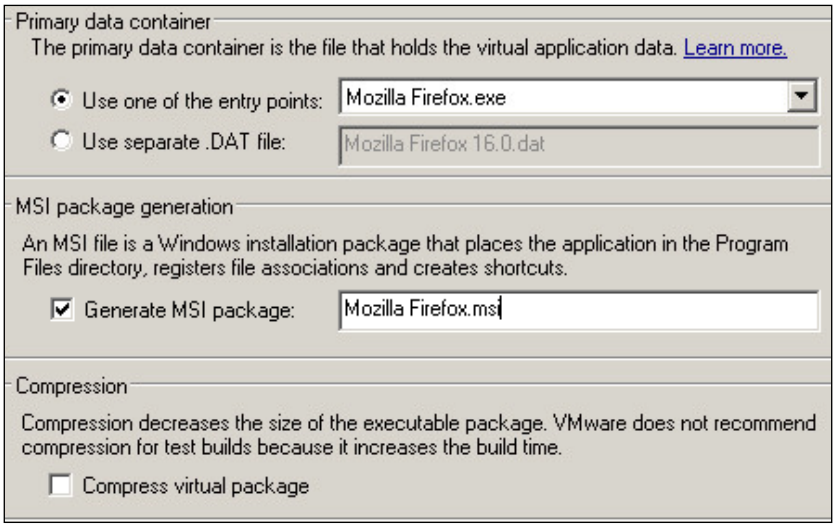
If you plan to use View Persona Management, described in *Chapter 7*, *Implementing View Persona Management*, the Thinstall directory should be set to background download rather than download on demand. The default Persona Management setting will only download this data as needed, which can lead to performance issues with applications delivered using ThinApp. This setting is configured in **VMware View Agent Configuration** | **Persona Management** | **Roaming & Synchronization** | **Folders to background download**. This recommendation also applies to other profile management platforms that include similar functionality.

15. In the **Setup Capture – Quality Assurance Statistics** window, select your preference for sharing ThinApp statistics with VMware.
16. In the **Setup Capture – Project Settings** window, edit **Inventory name** and specify **Project location** as shown in the following screenshot:



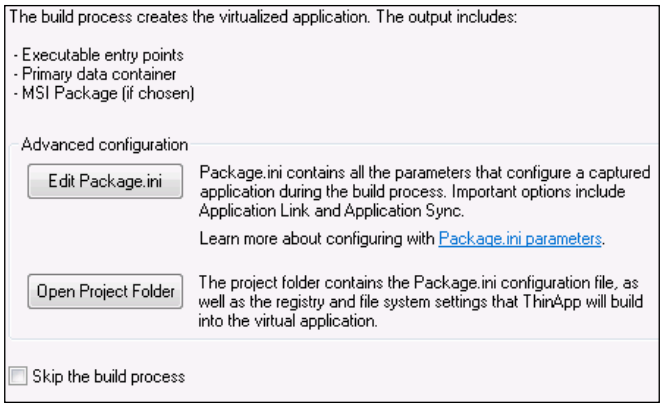
Consider removing any Software version or build numbers from the **Inventory name**, as shown in the previous screenshot. When a ThinApp is updated, it typically retains the old version number, which can lead to confusion as people identify the software version by the inventory name, and not by looking within the application itself to identify the actual version number.

17. In the **Setup Capture – Package Settings** window, make any necessary changes to the default settings. To enable ThinApp registration, you must enable MSI package generation by checking the **Generate MSI package** checkbox as shown in the following screenshot. Optionally, check the **Compress virtual package** checkbox to compress the resulting executable package. Click on **Save** when finished to generate the ThinApp project directory.

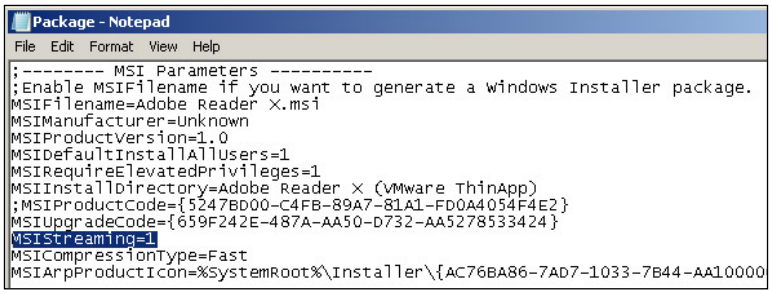


The default options for **Primary data container** are usually the optimal choice. ThinApp will choose the preferred option based on the application, and any changes can affect the display of the application icon.

18. In the **Setup Capture – Ready to Build** window, make any required changes. In our example, we will click on the **Edit Package.ini** button, as shown in the following screenshot, in order to enable streaming execution mode, which is disabled by default.



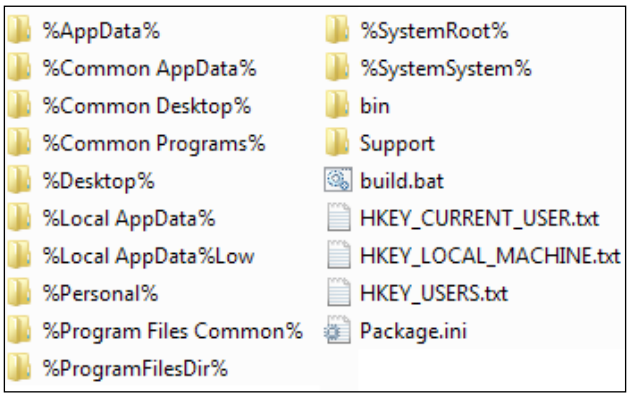
19. To enable the streaming execution mode, edit the `Package.ini` file as shown in the following screenshot, updating the `MSIStreaming=0` configuration parameter to read `MSIStreaming=1`. Once the required changes have been made, save the changes and close the text editor.



One additional configuration parameter of interest is `RemoveSandboxOnExit=1`, which instructs ThinApp to retain the sandbox contents when the application is closed. By default, the setting is disabled; to enable it, remove the `;` character from in front of the parameter and save the `settings.ini` file. When this parameter is enabled, any changes to the application will be discarded when Sandbox is closed. Common reasons for discarding the Sandbox contents include the desire to tightly control the configuration of the application or scenarios where the application will be used with non-persistent desktops, which often discards any application changes upon logoff anyway.

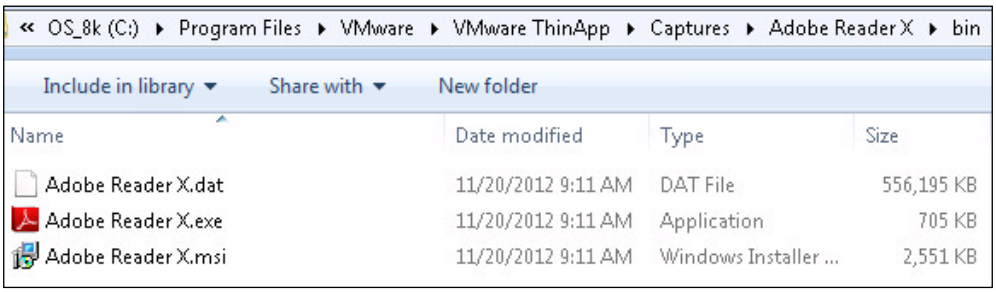
Consult the VMware document *ThinApp Package.ini Parameters Reference Guide* for a full description of all the parameters within the `settings.ini` file.

20. In the **Setup Capture – Ready to Build** window, click on **Build** to assemble the ThinApp package.
21. Test the operation of the new ThinApp package, including verifying that the expected updates were applied and that the configuration settings were retained.
22. If the ThinApp project directory was saved on the capture desktop, copy it to an alternate location for archival purposes. The entire project directory should be copied, as it includes the files needed to repackage the application if needed. The contents of a sample project directory are shown in the following screenshot:



23. Revert the ThinApp capture desktop to the saved VMware snapshot to discard the changes that were made during the application capture process.

The completed ThinApp version of the application will be placed in the `bin` folder within the ThinApp project directory. The following screenshot shows contents of the ThinApp project's `bin` folder for Adobe Acrobat, which includes the `.exe`, `.dat`, and `.msi` files.



These files are what are required by the end user to be able to use the ThinApp application. To make them available for use, copy them to a CIFS-based file share with the appropriate permissions, which are usually read only. The only reason a ThinApp application would ever need to be installed in a directory with write permissions is that the Sandbox is configured to be stored with the ThinApp application itself, which is typically only used when delivering the application in a portable format, such as on a removable USB drive.

The ThinApp project folders should be backed up at regular intervals. The contents of the project folder can be used to rebuild the application, which is commonly done after updating the application or altering the configuration.

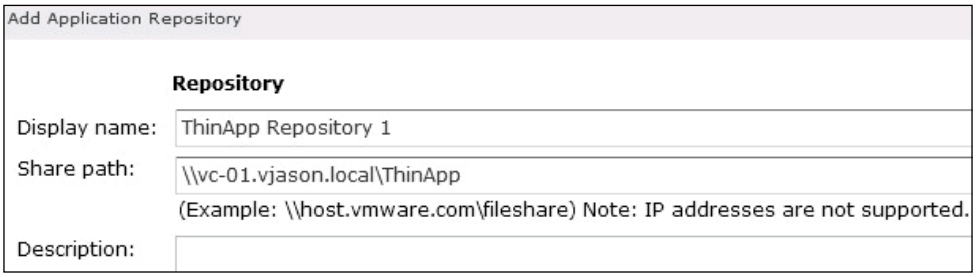
Deploying ThinApps in View

One of the benefits of using ThinApps with View is that View can automatically publish applications to desktops, either at the individual desktop or desktop pool level. This section will show how to configure a ThinApp repository within View and then discover and publish the applications within that repository.

Configuring the View ThinApp repository

The ThinApp repository is configured within the View Manager Admin console. The following steps outline how to add a ThinApp repository to View:

1. Navigate to **View Configuration** | **ThinApp Configuration** within the View Manager Admin console. Click on the **Add Repository** button in the **ThinApp Configuration** page to open the **Add Application Repository** window.
2. Provide a **Display name** for the repository as well as the **Share path** to the CIFS share that will host the applications. A sample configuration is shown in the following screenshot. Click on **Save** when finished with adding the repository and then close the window.

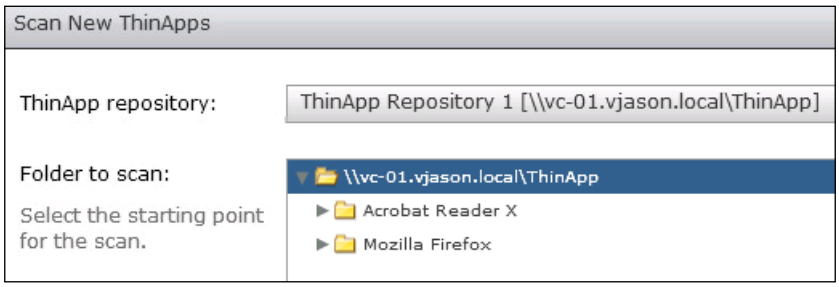


The ThinApp repository should now be displayed in the **ThinApp Configuration** page. The next step is to scan the repository for ThinApp packages; we will perform this in the next section.

Scanning for ThinApp packages

Once we have added the ThinApp repository in the View Admin console, we must scan it and indicate which applications we wish to assign from within View. The following steps outline how to scan the repository for ThinApp packages.

1. Go to **Inventory** | **ThinApps** within the View Manager Admin console. Click on the **Scan New ThinApps** button in the **ThinApps** page to open the **Scan New ThinApps** window.
2. In the **ThinApp repository** drop-down menu, select the repository you wish to scan. By default, the scan will begin in the root of the repository; if you wish to restrict what folders are scanned, expand the repository folder structure in the **Folder to scan** field and select the desired folder to start the scan from. An example of this field is shown in the following screenshot. Click on **Next** to initiate the scan.



Only ThinApps that have MSI files will be detected during the scan. If you did not check the checkbox during the capture process to generate a MSI package with ThinApp, you will need to enable the option and rebuild the ThinApp.

3. In the results window, select the MSI files you wish to scan. In the following example, we have highlighted both MSI files that were found. Click on **Scan** to initiate the MSI scan.

Select the MSI files to scan

Select the MSI files to scan and add to View Administrator:

MSI File	Path
Adobe Reader X.msi	\\vc-01.vjason.local\ThinApp\Acrobat Reader X
Mozilla Firefox.msi	\\vc-01.vjason.local\ThinApp\Mozilla Firefox

4. The scanning will complete and the results will be displayed in the window as shown in the following screenshot. Click on **Finish** to close the window.

Scanning: Scanning completed. 2 ThinApps added.

Name	Path	MSI File	Status
Mozilla Firefox	\Mozilla Firefox	Mozilla Firefox.msi	Added
Adobe Reader X	\Acrobat Reader X	Adobe Reader X.msi	Added

The **ThinApps** page will now display the applications that were detected during the scanning process. These applications are now available to be individually assigned to a desktop or desktop pool, or to be added to a template that enables the assignment of multiple applications at once. The next section outlines how to assign applications within View.

Assigning ThinApp applications

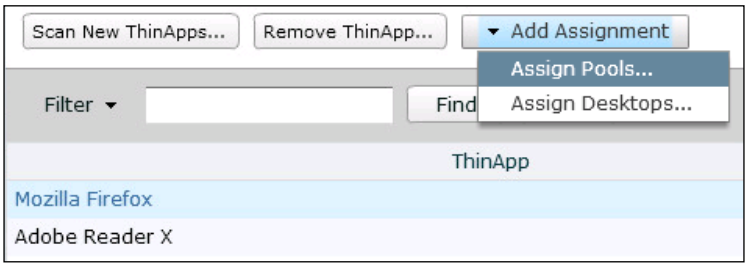
In this section, we will assign applications to a desktop pool. The same process is used to assign desktops to an individual desktop, so only one example will be shown.

ThinApp cannot place shortcuts on the desktop if that folder is being directed using folder redirection. If folder redirection is being used, the application shortcuts must be copied to the folder manually.

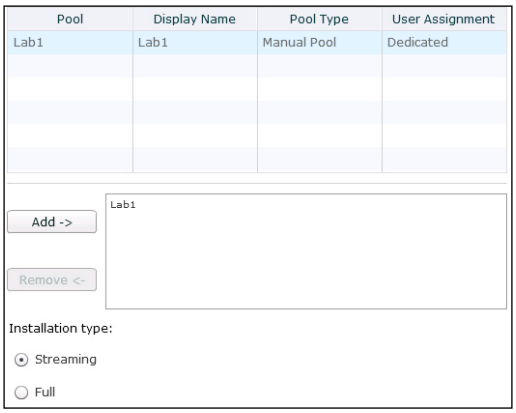
Assigning an application directly

The following steps outline how to assign an individual application to a desktop or desktop pool:

1. Go to **Inventory | ThinApps** within the View Manager Admin console. Highlight the application you wish to assign to a desktop, click on the **Add Assignment** drop-down menu, and select **Assign Pools...** as shown in the following screenshot:



2. In the **Add Pool Assignment** window shown in the following screenshot, select the desktop pool you wish to assign the application to and click on **Add**. Click on either the **Streaming** (executed from the remote ThinApp repository; optimal for linked-clone desktops) or **Full** (Copied to and executed directly on the View desktop) radio button to set the application installation type. Click on **OK** to complete the application assignment.



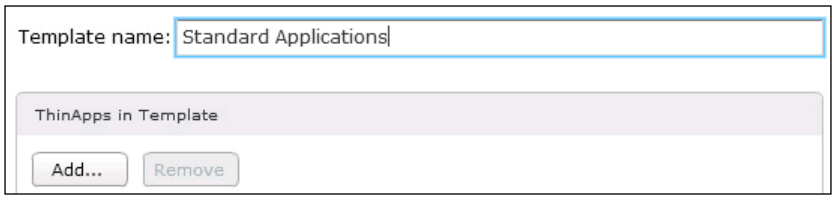
The application is now assigned to the desktop pool. If the **Full** application type was selected, the application will be downloaded to the desktop during the user login process. As an MSI file was used to deploy the ThinApp, the default application shortcuts will be placed on the desktop and in the Start menu of the desktop.

By default, the installation type will be set to **Streaming**. If the application was not configured to support streaming mode, the application installation type would default to **Full**. As described in the section *Capturing an Application with ThinApp*, streaming must be enabled during the application build process as it is not enabled by default.

Assigning applications using a template

A ThinApp template can be used to assign multiple applications to a desktop or desktop pool at once. If large numbers of desktops require the same set of applications, a template can be used to simplify the deployment process. The following steps outline how to create and assign a ThinApp template:

1. Go to **Inventory | ThinApps** within the View Manager Admin console. Click on the **New Template** button to open the **New ThinApp Template** window.
2. In the **Template name** field, specify a name for the template as shown in the following screenshot. Click on the **Add...** button to select the applications you wish to assign to the template.



3. Highlight the applications you wish add to the template, click on the **Add...** button, and click on **OK** to return to the previous window.
4. Click on **OK** to close the **New ThinApp Template** window, create the template, and return to the **ThinApps** page within the View Manager Admin console.

The template and the ThinApps it contains will now be displayed in the **ThinApps** page as shown in the following screenshot:

Template	ThinApps
Standard Applications	Adobe Reader X, Mozilla Firefox

The template is assigned using the same steps used to assign a single ThinApp application, including selecting the application installation type. Refer to the section *Assigning application directly* for the steps used to assign the template.

Removing ThinApp assignments

There are multiple reasons why a View Administrator might need to remove ThinApp assignments from a desktop or desktop pool. These reasons include:

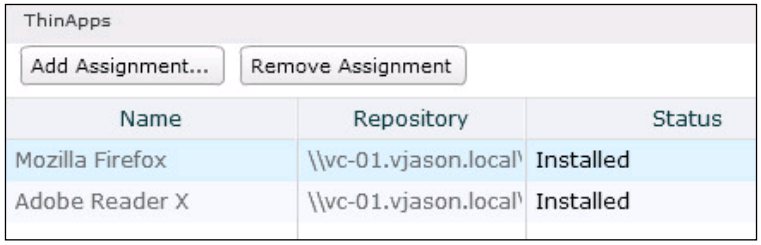
- To remove a ThinApp from View, it must be unassigned from all desktops or desktop pools first
- The application is an application that is no longer required
- The application is an application that is being replaced with a newer version

The process used to remove a ThinApp assignment is straightforward for both desktops and desktop pools. The following sections outline how to remove a ThinApp assignment from both items.

Removing a ThinApp assignment from a desktop

The following steps outline how to remove a ThinApp assignment from a single View desktop:

1. Navigate to **Inventory** | **Desktops** within the View Manager Admin console and search for the target desktop.
2. Double-click on the target desktop to bring up the desktop's **Summary** page.
3. In the **ThinApps** window of the **Summary** page, highlight the target ThinApp and click on the **Remove Assignment** button as shown in the following screenshot:

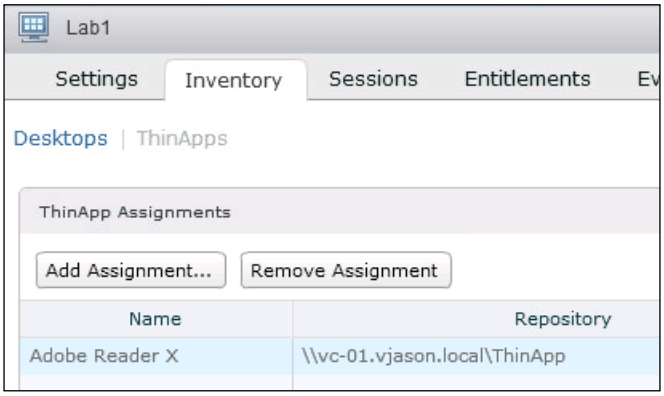


The ThinApp has now been unassigned from the desktop and will automatically be removed by the View agent installed on the desktop.

Removing a ThinApp assignment from a desktop pool

The following steps outline how to remove a ThinApp assignment from a View desktop pool:

1. Navigate to **Inventory** | **Pools** within the View Manager Admin console and double-click on the target pool.
2. Click on the **Inventory** tab, and then select ThinApps as shown in the following screenshot:



3. Highlight the target ThinApp and click on the **Remove Assignment** button.

The ThinApp has now been unassigned from the desktop pool, and will automatically be removed by the View agent installed on each desktop within the pool.

Updating ThinApp packages

There are multiple ways to update applications that have been packaged using ThinApp. The simplest is to simply repackage the updated application and to redeploy it using whatever method you wish. This can even include using View to assign the newer version of the application and unassign the old version.

This section will focus on using the native features of ThinApp to update existing ThinApp projects and repackage them, which preserves any previous application's customization and prevents the need to repeat the entire capture process.

Do not attempt to update ThinApp packages when performing a major version upgrade of an application. It is best to capture a clean installation of the newer version of the application rather than attempting an in-place upgrade. By performing a clean installation of the target application, you eliminate the possibility that any settings or files associated with the previous version will affect the behavior of the newer version.

Using built-in application updaters

Applications such as Mozilla Firefox have the ability to download and install updates within the Firefox application itself. This section will outline how we use that functionality to apply updates to Firefox and integrate those changes into our master ThinApp project. Firefox is just used as an example for this section; the process is the same for other applications whose native update features are supported by ThinApp. The following steps outline how to repackage the Firefox application after updates have been applied:

1. Make a backup of the ThinApp project that is going to be updated.
2. Using the ThinApp capture desktop, attach the CIFS share that contains the ThinApp project folder.
3. Verify that the `AppData\Roaming\Thinstall` directory within the logged in user's Windows profile on the ThinApp capture desktop is empty. If not, delete the contents.

When updating an application, it is important to discard any previous Sandbox data as we want to control what is included in the repackaged ThinApp. Emptying the `Thinstall` folder deletes all existing Sandbox data on the ThinApp capture desktop and ensures that only data related to the update is included in the ThinApp project folder.

4. Launch the ThinApp from within the `bin` folder of the ThinApp project and update the application using the native application update functionality.
    1. For Firefox, this is performed by going to **Help** | **About Firefox** | **Check for Updates**.

Not every application can be updated in this way. For applications where this native update process fails, or where patches must be installed outside of the application runtime, you must use an alternate entry point to execute the update. The next section outlines how to enable a Windows command prompt as an alternate endpoint into a ThinApp package.

2. Once the update has completed, relaunch the application and verify that the desired settings are configured. Close the application when completed.
3. From a command prompt on the ThinApp capture desktop, navigate to the `Program Files\VMware\VMware ThinApp` directory.
4. Execute the following command to merge the application updates into the ThinApp project. In this example, the Firefox ThinApp project is located in `T:\Mozilla Firefox`.

```
Sbmerge apply -ProjectDir "t:\Mozilla Firefox"
```

5. The `sbmerge` utility will update the ThinApp project with all the files and configuration changes that occurred when Firefox was updated. Once `sbmerge` has completed, execute the `build.bat` file from within the ThinApp project directory. In our example, the `build.bat` file is located at `T:\Mozilla Firefox\build.bat`. The `build.bat` file will rebuild the ThinApp package, and overwrite the existing ThinApp application in the `bin` directory.
6. Test the operation of the updated ThinApp package, including verifying that the expected updates were applied and that the configuration settings were retained.

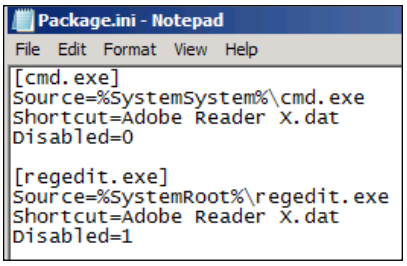
Once the ThinApp has been tested and the configuration is verified, it can be deployed.

Using alternate entry points

Due to the way that the ThinApp technology works, it is not always possible to update an application using the native update features. In addition, updates that are installed outside of the application runtime, such as a patch for Microsoft Office, cannot by default be launched within the context of the ThinApp container.

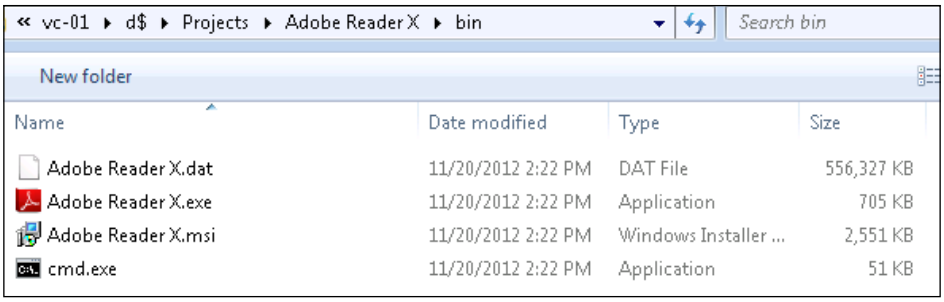
To update these applications, we need an alternate entry point into the application. Specifically, we need access to a Windows command prompt that executes in the context of the ThinApp. By enabling the Windows command prompt within the ThinApp, we gain the ability to install those updates that cannot be applied within the application runtime itself. The following steps outline how to enable the alternate endpoints, apply the required patches, and repackage the application. Adobe Reader X is just used as the example in this section; the process is the same for other applications.

1. Make a backup of the ThinApp project that is going to be updated.
2. Using the ThinApp capture desktop, attach the CIFS share that contains the ThinApp project folder.
3. Verify that the `AppData\Roaming\Thinstall` directory within the logged-in user's Windows profile on the ThinApp capture desktop is empty. If not, delete the contents.
4. Open the `Settings.ini` file from within the ThinApp project folder.
5. Scroll down to the bottom of the `Settings.ini` file, and under the `[cmd.exe]` section update the `Disabled=1` setting to `Disabled=0` as shown in the following screenshot. This setting enables `cmd.exe` as an additional entry point into the ThinApp. Save the `Settings.ini` file and close the text editor.



In addition to the `cmd.exe` entry point, we can also enable the `regedit` Windows registry editing tool by making the same change in the `[regedit.exe]` section of the `Settings.ini` file.

6. Execute the `build.bat` file from within the ThinApp project directory. In our example, the `build.bat` file is located at `T:\Mozilla Firefox`. The `build.bat` file will rebuild the ThinApp package and overwrite the existing application in the `bin` directory. Once the `build.bat` completes, the `bin` folder will now contain an entry point for `cmd.exe`, which we will use to install our patches. The following example shows the new `bin` folder with the new `cmd.exe` entry point.



7. Launch the `cmd.exe` ThinApp application and use the command window to launch the installation of the software patches. If multiple patches are required, simply install them in succession. If the installation of the patch requires a reboot in order to proceed with the installation of additional patches, simply close and reopen the `cmd.exe` ThinApp rather than rebooting the ThinApp capture desktop.
8. Verify that the patched application is working as intended, and reconfigure any application's settings, if required.
9. Undo the previous changes to the `Settings.ini` file so that the `cmd.exe` entry point is removed when the application is repackaged using the `sbmerge` utility. This is accomplished by changing the `Disabled=0` setting back to `Disabled=1` under the `[cmd.exe]` section of `Settings.ini`.
10. Execute the `sbmerge` command to merge the application updates into the ThinApp project folder. In this example, the Adobe Reader X ThinApp project is located in `T:\Adobe Reader X`.

    ```
    Sbmerge apply -ProjectDir "t:\Adobe Reader X"
    ```

11. The `sbmerge` utility will update the ThinApp project with all the files and configuration changes that occurred when Firefox was updated. Once `sbmerge` has completed, execute the `build.bat` file from within the ThinApp project directory. In our example, the `build.bat` file is located at `T:\Adobe Reader X`. The `build.bat` file will rebuild the ThinApp package and overwrite the existing application in the `bin` directory. In addition, the `cmd.exe` ThinApp file will be removed during the execution of the `build.bat` script.
12. Test the operation of the updated ThinApp package, including verifying that the expected updates were applied, and that the configuration settings were retained.

Once the ThinApp has been tested and the configuration is verified, it can be deployed.

Advanced ThinApp topics

There are multiple topics that may be of interest to anyone who desires a more advanced understanding of how to deploy and manage ThinApp applications within their environment.

Refer to the *VMware ThinApp User's Guide* (`http://www.vmware.com/support/pubs/thinapp_pubs.html`) for more information about the following topics:

- **Application Sync**: This is a web-based service that is used to keep ThinApp applications up to date, regardless of where they are executed from.
- **Application Link**: This is used to connect dependant applications together at runtime. One example would be an application that also requires the .Net Framework. Application Link enables ThinApp administrators to capture both pieces of software separately, and link the ThinApp packages together.
- **ThinReg**: This is located in the VMware ThinApp `install` directory and used to customize application file associations.

Consult the VMware Horizon Workspace website (`http://www.vmware.com/products/desktop_virtualization/horizon-workspace/overview.html`) to learn about additional ways that your organization can leverage ThinApp for application delivery. Among other features, Horizon Workspace provides access to ThinApp applications from anywhere using a HTML5-based browser or device.

Summary

In this chapter, we have been introduced to VMware ThinApp, a powerful application virtualization platform that integrates with VMware Horizon View to enable the automated deployment of applications packaged with ThinApp.

We discussed a number of common ThinApp usage scenarios, and showed how to deploy ThinApp applications to desktops and desktop pools within View.

In the next chapter, we will discuss VMware Horizon View Persona Management, a feature of View that improves the performance and functionality of traditional Windows roaming profiles to enable user portability.

7
Implementing View Persona Management

VMware Horizon View Persona Management is a feature of View that preserves user profile data and dynamically synchronizes it with a remote profile repository. View Persona Management improves upon traditional Microsoft roaming profiles by only loading user profile data as needed, synchronizing user profile changes on an ongoing basis, enabling the persistence of all application settings and data, eliminating the need for a Virtual Desktop persistent data disk, and other features described throughout the chapter.

This chapter will discuss how to implement and administer View Persona Management.

In this chapter, we will learn:

- An overview of View Persona Management
- The features of View Persona Management
- Persona Management requirements
- Enabling View Persona Management
- Persona Management Group Policy settings

View Persona Management overview

View Persona Management was first introduced in View 5.0 to provide View Administrators with an additional option for managing user profile data. When a user with Windows roaming profiles enabled logs into or out of a desktop, the entire contents of the profile must be transferred between the desktop and the profile repository, regardless of how much of that profile data will be required during the session. This method of profile management is resource intensive; particularly when the profile is large.

View Persona Management provides a much more efficient means of managing user profile data using multiple techniques that are not supported by Windows roaming profiles. The following are just some of the features of View Persona Management:

- User profile data is loaded only when needed, minimizing the network traffic and IO required to support a user logon session
- Changes to user profile data are synchronized to the profile repository at configurable intervals, providing enhanced data protection and enabling faster logoffs, as less data will need to be saved upon logoff
- Implementing Persona Management requires no changes to the user Active Directory accounts; all configuration options are configured using Group Policies that are applied to the View desktop Active Directory computer objects
- Eliminates the need to use persistent data disks with linked-clone desktops
- Enables persistent user customization even when using non-persistent or floating assignment View desktops
- Compatible with existing roaming profile repositories and can be enabled quickly with minimal Group Policy settings
- Provides additional capabilities not available with Windows roaming profiles, such as Persona Management specific informational **user interface** (**UI**) elements, expanded folder redirection support, and advanced logging capabilities
- Support for importing existing user profiles into the Persona Management repository by installing the standalone View Persona Management service on the source desktop
- Provides tools to migrate existing Windows XP **V1** format profiles to the **V2** version used by Windows 7 and newer Microsoft OSs. This feature is discussed in the VMware document *VMware Horizon View User Profile Migration* (`http://www.vmware.com/support/pubs/view_pubs.html`)

While Windows roaming profiles provide some of the same functionality as View Persona Management, it is not optimized for Virtual Desktop environments and thus, requires more resources when processing user logon and logoff requests.

Understanding View Persona Management

View Persona Management is one of the few components of View that are not administered using the View Manager Admin console. To enable Persona Management three things are required:

- A CIFS-based file share to store the profile data
- The `ViewPM.adm` Group Policy template, available on any machine with the View Agent or View Connection Server software installed
- A desktop that has the View Agent installed, including the View Persona Management option, or a physical desktop that has the standalone View Persona Management service installed

View Persona Management is configured using the `ViewPM.adm` Group Policy template, which can be applied using multiple methods:

- Applied to the Virtual Desktop master image prior to deployment of the Virtual Desktops
- Applied using Active Directory to an OU that contains the target computer objects

Once configured, Persona Management will be used for all desktop profile management on the desktops where it is enabled, even if Windows roaming profiles are enabled within the user's Active Directory account.

Persona Management works by making it appear to the desktop operating system that all the profile data is already present on the local hard disk. Only when the desktop attempts to access any required files, application data, or registry entries is that data downloaded to the desktop from the remote profile repository. In addition, Persona Management replicates any profile updates back to the profile repository at a configurable interval, rather than uploading those changes all at once when a user logs off. These features of Persona Management minimize the impact of a user logon or logoff request on the View infrastructure, which in turn reduces the time required for these events to complete.

The *Deploying View Persona Management* section later in this chapter will provide detailed information about how to configure each of the items required to fully implement Persona Management.

Features of View Persona Management

View Persona Management includes a number of options to further customize how the user profile data is stored and accessed. Some of these features include:

- Profile folder redirection for a number of key profile directories
- The ability to download specific folders in the background
- The ability to exclude specific folders from roaming
- The ability to exclude the files from specific processes from roaming, such as antivirus application data
- The ability to remove the local persona when the user logs off the desktop

Details on how to configure the various options for View Persona Management will be provided in the *Advanced Persona Management options* section later in this chapter.

Deploying View Persona Management

This section will detail the steps that are needed to deploy and enable View Persona Management within a View environment. This section will focus on the minimum steps required to implement Persona Management. To learn about the many other configuration options that are present within the `ViewPM.adm` Group Policy template, refer to the *Advanced Persona Management options* section later on in the chapter.

Infrastructure requirements

To ensure optimal performance, Persona Management recommends that the components of the View infrastructure meet certain minimum requirements. These recommendations include:

- One file server with 8 GB of ram for every 1,000 users:
    - If a virtual server is used, it is recommended that the filesystem that hosts the Persona Management repository should be striped across four virtual disks, each with its own SCSI controller
- 1 Gbps connectivity between the desktops and the servers that host the Persona Management repository

While these recommendations are very specific, each View environment will be different and you may find that the final design requires either more or fewer resources to host the Persona Management repository. Factors that impact file server and infrastructure requirements include average user profile size, how often the profile is accessed, storage capacity and performance, and network speed and latency.

As with any critical infrastructure component, the performance of the file server that hosts the Persona Management repository should be monitored to ensure that it is performing adequately. Consult your operating system or **network attached storage** (**NAS**) vendor documentation for information about how to monitor the performance of your file server, and perform any optimizations that may be required.

View Persona Management uses the Microsoft Volume Shadow Service to back up profile data to the Persona Management repository. Do not back up profile data using client-based utilities that also use this feature, as it may corrupt the profile.

Rather than using desktop antivirus tools to scan local profile data, scan the profile repositories using antivirus tools on the file server itself. This will reduce the desktop IO requirements. You may also scan profile data using antivirus platforms that are optimized for use in virtual environments, such as the VMware vShield Endpoint. Information about vShield Endpoint can be found on the VMware website at `http://www.vmware.com/ap/products/datacenter-virtualization/vsphere/endpoint.html`.

Persona Management repository

View Persona Management requires a CIFS-based file share with a specific minimum permission set for both the folder and the share.

NTFS permissions for the Persona Management parent folder are as follows:

User account	Minimum permissions required
Administrator	None
Creator/Owner	Full control; subfolders and files Only
Everyone	No permissions
Local System	Full control; this folder, Subfolders, and Files
Active Directory Security Group that contains View Persona Management users	List folder/read data, create folders/ append data—this folder only

These settings may need to be adjusted to ensure that the folder can be backed up. Consult your backup vendor for the minimum permissions required by your backup program.

Share permissions for the Persona Management share are:

User account	Minimum permissions required
Everyone	No permissions
Active Directory Security Group that contains View Persona Management users	Full Control

As with Windows roaming profiles, View Persona Management will automatically create individual folders for each user that logs into a desktop with Persona Management enabled.

If the users will use both traditional desktops and View desktops concurrently, it is recommended to standardize on one solution for managing user profiles rather than attempting to use roaming profiles and View Persona Management at the same time. As discussed in this chapter, View Persona Management is the optimal solution for managing user profile data. Should you choose to use View Persona Management, you will need to use the standalone View Persona Management installer to install the service on the physical desktops.

If View Persona Management is replacing Windows roaming profiles, the same profile repository can be used, as the format is the same. However, if the profile directory name format changes when switching to View Persona Management, a new blank profile folder will be created and used.

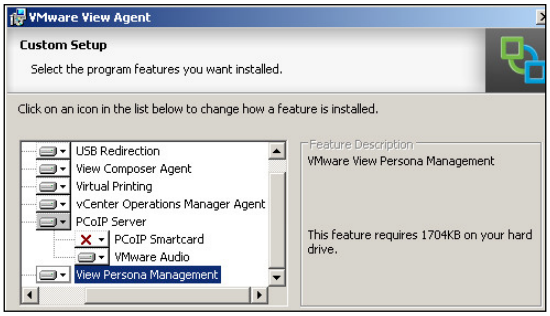
View Agent Persona Management component

Assuming that the default installation options were used to install the View Agent in the Virtual Desktop master image, the Persona Management option should already be installed. To verify that the option is installed, look for the **VMware View Persona Management** service in the services control panel on the Virtual Desktop. By default, this service is installed and configured as disabled; it will be enabled automatically when Persona Management is required. Do not make manual changes to the Persona Management service start up configuration, as this is not supported.

View Persona Management is not supported on Microsoft Terminal Servers or View desktops that will be run in Local Mode.

If the Persona Management component was not selected when the View Agent was installed, there are two options for enabling it after the fact:

- Install the **Persona Management** option using the standalone installer; this installer will have a name similar to `VMware-personamanagement-x.x.x-yyyyyy.exe`.
- Remove and reinstall the View Agent software, verifying that the **View Persona Management** option is selected as shown in the following screenshot:



Persona Management requires specific Windows services in order to function. As these services are sometimes disabled during the optimization of a Virtual Desktop master image, it is important that they be set to startup type **Automatic** in order for Persona Management to function properly. The services are:

- Microsoft Software Shadow Copy Provider
- Volume Shadow Copy

These services can be enabled using the following Windows PowerShell commands:

```
Powershell Set-Service 'VSS' -startuptype "automatic"
Powershell Set-Service 'swprv' -startuptype "automatic"
```

If Persona Management will be used with a Windows XP desktop, the **Microsoft User Profile Hive Cleanup** (**UPHClean**) service should be installed on the Virtual Desktop master image.

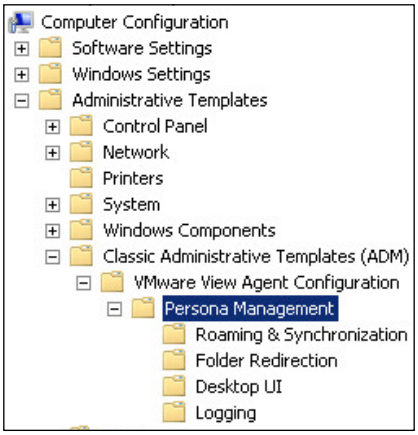
Using the Persona Management Group Policies

The Persona Management Group Policies are contained within a Group Policy template filename `ViewPM.adm`. This file is located in the `Program Files\VMware\VMware View\Server\extras\GroupPolicyFiles` folder on any View Connection Server, or in the `Program Files\VMware\VMware View\Agent\bin` folder on a desktop with the View Agent software installed.

The process used to enable Persona Management is the same regardless of how the Persona Management group policies are deployed. In this section, we will be implementing the settings directly in the Virtual Desktop master image by launching the local Group Policy manager tool `gpedit.msc`.

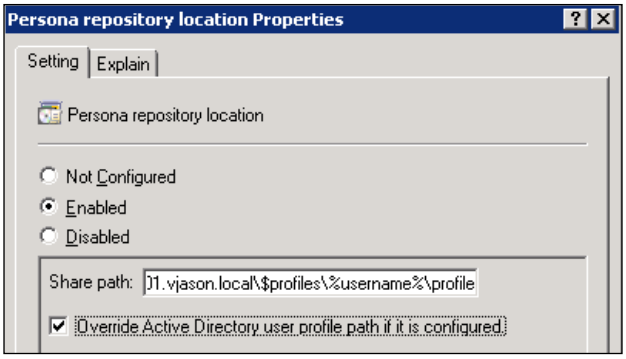
The following steps outline how to enable Persona Management using the `ViewPM.adm` Group Policy template. These steps were performed on a desktop running the Windows 7 operating system:

1. Copy the `ViewPM.adm` Group Policy template to a location accessible to the Virtual Desktop master image.
2. Execute `gpedit.msc` on the Virtual Desktop master image to launch the **Local Group Policy Editor**.
3. Right-click on the **Local Computer Policy** | **Computer Configuration** | **Administrative Templates** folder and select **Add/Remove Templates**.
4. In the **Add/Remove Templates** window, click on the **Add** button.
5. Browse to the location of the `ViewPM.adm` Group Policy template, highlight the template, click on **Open** to import the template, and return to the **Add/Remove Templates** window.
6. Click on the close button to close the **Add/Remove Templates** window and return to the **Local Group Policy Editor**.
7. Expand the **Local Computer Policy** | **Computer Configuration** | **Administrative Templates** | **Classic Administrative Templates (ADM)** | **VMware View Agent Configuration** | **Persona Management** folder to display all the Persona Management policy folders, as shown in the following screenshot:



8. Select the `Roaming & Synchronization` folder.
9. Double-click the **Manage user persona** Group Policy setting in the right pane to open the policy configuration window.
10. Click on the **Enabled** radio button to enable Persona Management as shown in the following screenshot. The **Profile upload interval** can also be changed in this window; by default it is set to upload profile changes every 10 minutes. Click on **OK** when completed with the configuration:

11. Double-click the **Persona repository location** Group Policy setting in the right pane to open the policy configuration window.
12. Click on the **Enabled** radio button and populate the Persona **Share path** with the path to the Persona repository. In the following screenshot, the **Share path:** field is set to `\\dc-01.vjason.local\$profiles\%username%\profile`. Click on **OK** when completed with the configuration.

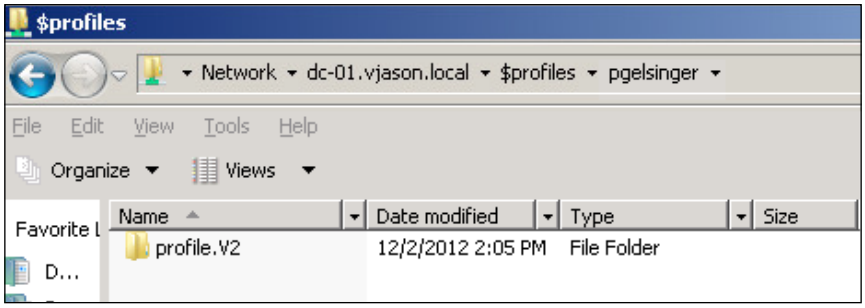


By default, Persona Management will use the Active Directory user profile path if that setting has been configured within the users Active Directory account, or if it is set using Group Policies. As mentioned previously, the same repository should not be used if a user is using both Windows roaming profiles and View Persona Management. To override any existing Windows user profile path settings, click on the **Override Active Directory user profile path if it is configured** checkbox in the **Persona repository location** Group Policy setting. Enabling this option does not affect the operating of Windows roaming profiles, only View Persona Management.

View Persona Management has now been enabled and will manage the Windows profiles of users who log on to Virtual Desktops based on this master image. If the policies were applied using Active Directory Group Policies, they will apply to any Virtual Desktops contained within the OU to which the policy was linked.

In step 12 when specifying the profile repository **Share path:**, we used the `%username%` variable in the path to the profile directory. Rather than create the profile directory in the root of the share, this creates it within a folder named after the user's login ID. By doing this, we can re-use that folder later for features such as folder redirection, which is discussed in the *Folder redirection* section later in the chapter. Were we simply to create the profile directory within the root of the share, we would have needed to create another folder to store any profile folders we wanted to redirect.

The following screenshot shows a profile directory that was created upon login on a desktop with Persona Management enabled.:



The next section will provide information about all of the additional Group Policy settings that can be used to further customize Persona Management.

Advanced Persona Management options

The Persona Management `ViewPM.adm` Group Policy template contains settings that are broken down into the following four categories:

- `Roaming & Synchronization`
- `Folder Redirection`
- `Desktop UI`
- `Logging`

The purpose and usage of these settings will be explained in this section.

Roaming and Synchronization

The Roaming & Synchronization policies are the key policies used to enable and configure View Persona Management. Unless otherwise indicated, all settings are disabled or blank by default.

- **Manage user persona**: This enables View Persona Management to manage user profiles. This setting must be enabled to use Persona Management.
    - **Profile upload interval**: A subsetting of the **Manage user persona** setting. This setting is measured in minutes and determines how often changes to the profile are uploaded to the Persona Management repository. The default value is `10`; acceptable values range from `1` to `9999`.
- **Persona repository location**: This setting specifies the Persona Management repository location.
    - The **Share path** must be the UNC path to a network share configured with the appropriate share and folder permissions for Persona Management.
    - The **Override Active Directory user profile path if it is configured** checkbox is used to override any existing Active Directory settings that set the user profile path, and to ensure that Persona Management is using the share indicated in the **Share path:** field.

- **Remove local persona at log off**: This setting deletes the locally stored user profile upon logout. This setting is typically used to reduce desktop storage requirements when desktops are shared between users. This setting is generally not used for dedicated desktops, as these users will benefit by having the frequently used data already cached to their desktop. This setting should not be used when a desktop pool is configured to refresh or delete the desktop upon logoff, as the persona will be deleted during that operation anyway.
    - **Delete 'Local Settings' or 'AppData\Local' when persona is removed**: A subsetting of the **Remove local persona at log off** setting that is enabled by default. This setting enables the removal of the `Local Settings` or `AppData\Local` folders upon logout, which contain application data that is retained by Persona Management by default.
- **Roam local settings folders**: This setting enables the synchronization of the `Local Settings` or `AppData\Local` folders of the user persona.
    - The `Local Settings` or `AppData\Local` folders contain application data for applications such as Outlook, and should be synchronized when these settings must be shared between multiple desktops.
    - This option can place additional load on Persona Management as some applications, such as Outlook, may place large amounts of data in these folders. Due to this, it is not recommended to roam these folders when using dedicated desktops.
- **Files and folders to preload**: This setting is used to specify files and folders that will be fully downloaded at login.
    - This setting is typically used to enhance desktop performance by preloading specified folders or files.
    - This setting will delay the logon process while the files and folders are being downloaded.
    - By default, the list includes **<Start Menu>\Programs\Startup**, which cannot be removed.
    - This setting may increase logon times as the files are being downloaded. An additional setting, **Folders to background download**, provides a similar functionality without impacting logon times.
    - Files and folder names are specified using paths relative to the root of the local profile, such as `My Documents\Preload`.

- **Files and folders to preload (exceptions)**: This setting is used to exclude files or folders from preload that are located within the folders specified in the **Files and folders to preload** policy.
    - Using the example from the Files and folders to preload policy, to exclude a folder within the `My Documents\Preload` folder, the format would be `My Documents\Preload\Exclude`.
- **Windows roaming profiles synchronization**: This setting forces the specified files and folders to be managed by standard Windows roaming profiles.
    - With this policy, the specified files are folders are fully downloaded at login and fully uploaded only during logout.
    - By default the list includes **<Roamed Application Data>\Microsoft\Windows\Themes\Custom.theme** and **<Roamed Application Data>\Microsoft\SystemCertificates**, which cannot be removed.
    - Files and folder names are specified using paths relative to the root of the local profile.
- **Windows roaming profiles synchronization (exceptions)**: This setting is used to exclude files or folders from preload that are located within the folders specified in the **Windows roaming profiles synchronization** policy.
- **Files and folders excluded from roaming**: This setting is used to exclude files and folders on the local system from being copied to the Persona Management repository.
    - By default, the list includes the `temp` folder for the profile, the `ThinApp cache` folder, and the `cache` folders for the Internet Explorer, Mozilla Firefox, Google Chrome, and Opera browsers.
    - This setting is typically used to exclude application subfolders that hold cache data that is not required to roam with the user.
    - Files and folder names are specified using paths relative to the root of the local profile.
- **Files and folders excluded from roaming (exceptions)**: This setting is used to exclude files or folders from preload that are located within the folders specified in the **Files and folders excluded from roaming** policy.
- **Folders to background download**: This setting enables the background download of specified folders upon user logon, while still allowing the logon to complete.

- This setting is typically used in addition to, or in place of, the **Files and folders to preload** setting in scenarios where the folders are needed locally, but not immediately.
- This setting does not delay the user logon process.
- This setting is commonly used to background-download the `ThinApp Sandbox` directory. This is recommended to prevent the Sandbox corruption that can occur if the ThinApp application is executed without the entire contents of the Sandbox present on the local disk.
- Folder names are specified using paths relative to the root of the local profile.

- **Folders to background download (exceptions)**: This setting is used to exclude files or folders from preload that are located within the folders specified in the **Folders to background download** policy.

Folder redirection

View Persona Management supports folder redirection in addition to copying profile data between the remote Profile Repository and the desktop. Folder redirection is the use of a network folder in place of a local profile folder, bypassing the use of the local profile folder entirely.

There are a number of cases where using folder redirection in tandem with Persona Management may be the ideal solution for managing user data. Some possible reasons include:

- The user is required to use a combination of physical desktops with roaming profiles and Virtual Desktops with Persona Management, which should not share the same profile repository
- The user will be using a combination of Windows 7 and Windows XP desktops, which cannot share the same profile
- The Persona Management repository is located in close proximity (from an infrastructure perspective) to the Virtual Desktops, enabling fast transfers of profile data

In the *Using the Persona Management Group Policies* section, we created a Group Policy that will create a folder structure dedicated to each user. In the example shown, a folder has been created with the path `\\dc-01-vjason.local\$profiles\pgelsinger`. As discussed in that section, we will use that folder to provide an example of how to use the Persona Management folder redirection policies.

The following folders can be redirected using the Persona Management Group Policy template:

- `Application Data (Roaming)`
- `Contacts`
- `Cookies`
- `Desktop`
- `Downloads`
- `Favorites`
- `History`
- `Links`
- `My Documents`
- `My Music`
- `My Pictures`
- `My Videos`
- `Network Neighborhood (My Network Places)`
- `Printer Neighborhood`
- `Recent Items`
- `Saved Games`
- `Searches`
- `Send To`
- `Start Menu`
- `Startup Items`
- `Templates`
- `Temporary Internet Files (Internet Cache)`

Two additional folder redirection policies exist to enable a more granular control over what data is redirected. These policies include:

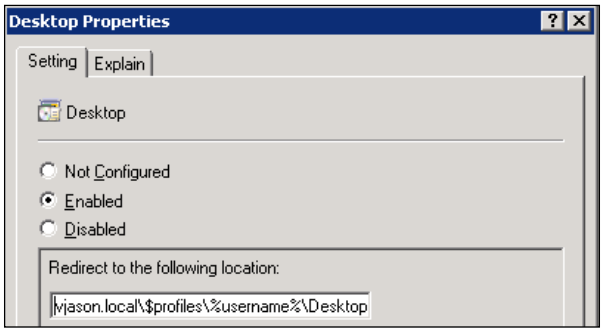
- **Folders excluded from Folder Redirection**: This setting is used to exclude subfolders of redirected folders on the local system from being redirected:
    - Files and folder names are specified using paths relative to the root of the local profile
    - A sample folder exclusion would be `Desktop\DoNotRedirect`

- **Folders excluded from Folder Redirection (exceptions)**: This setting is used to enable redirection for folders contained within other folders that were marked excluded by the **Folders excluded from Folder Redirection** policy. Using the previous example, a sample exception would be `Desktop\DoNotRedirect\DoRedirect`.

There are restrictions on what folders can be redirected if you are attempting to share the folders between multiple desktops:

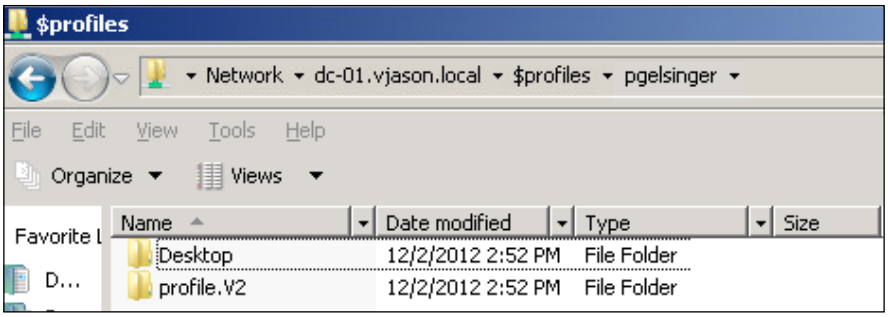
- If sharing files between Windows XP and newer versions of Windows, only the following folders can be redirected: `Desktop`, `My Documents`, and `My Pictures`
- The `Temporary Internet Files (Internet Cache)` folder cannot be shared between desktops of any type

To redirect a folder, a UNC path to an alternate location must be provided. In this example, we will redirect the `Desktop` folder to the location `\\dc-01.vjason.local\$profiles\%username%\Desktop` using the policy setting shown in the following screenshot:



When the user logs in, a folder titled `Desktop` will be created within their dedicated folder in the Persona Management share. Any files present in the existing `Desktop` folder will be copied to this new folder, the local folder will be deleted, and this folder will be used as the active `Desktop` folder while using any desktop with Persona Management enabled. This process can be repeated to redirect any other folders that Persona Management supports, assuming that the path used in the **Redirect to the following location** setting uses a unique folder for each one that is being redirected.

The following screenshot shows a Persona Management folder that contains both the user profile (`profile.V2`) and the redirected `Desktop` folder. By configuring our policies to place each of these folders within another dedicated folder, they will remain together which often simplifies backup and restore operations.



Desktop UI

The Desktop UI policies provide control over various settings related to user notifications and file icons.

- **Hide local offline file icon**: This setting controls whether or not the file icons display their status.
    - By default the setting is enabled, the offline icon is hidden, and all file icons in a user profile appear as normal icons.
    - When disabled, files that have not been downloaded from the Persona Management repository will appear with the offline symbol.
    - This setting is commonly disabled when troubleshooting Persona Management.
- **Show progress when downloading large files**: This setting enables a progress window on the desktop during any downloads of large files from the Persona Management repository.
    - By default this setting is disabled.
    - **Minimum file size to show progress window (MB)**: A subsetting of the **Show progress when downloading large files** setting, and is used to specify the smallest file size that will display a download progress window. The default value is `50 MB`, and values ranging from `0` to `4294967295 MB` are supported.

- **Show critical errors to users via tray icon alerts**: This setting displays critical errors using alert icons in the user's desktop tray.
    - This setting is disabled by default.
    - This setting is commonly enabled when troubleshooting Persona Management.

In most cases, the default settings for the Desktop UI policies will be sufficient. Altering any of these policies will change the end user experience, which should only be done after careful consideration.

Logging

The Logging policies control how and where information about Persona Management is logged. Persona Management logging is enabled for error and informational messages by default, placing the log data in the `ProgramData\VMware\VDM\logs` folder. The Persona Management logs files start with the characters `VMWVvp`. The following policies provide full control over Persona Management logging:

- **Logging filename**: This setting controls the name and location of the Persona Management log file. While this setting can be used to relocate the log file, it is recommended to leave it in the default location.
- **Logging destination**: This setting controls where the Persona Management log messages will be sent.
    - By default all log messages are sent to the log file.
    - Log messages can also be sent to the debug port.
    - This setting is commonly used when performing advanced troubleshooting.
- **Logging flags**: This setting controls what types of data are recorded in the log files.
    - By default **Log error messages** and **Log information messages** are enabled.
    - **Log debug messages** can also be enabled, which is commonly used when troubleshooting Persona Management issues.

- **Debug flags**: This setting enables additional debug logging.
    - By default this setting is disabled.
    - Debug flags that can be enabled include **error**, **information**, **registry**, **IRQL**, **port**, and **process** messages.
    - This setting is commonly used when performing advanced Persona Management troubleshooting.

In most cases, the Persona Management logging settings will only need to be changed to perform advanced troubleshooting, often while working with VMware Technical Support.

Summary

In this chapter, we have been introduced to View Persona Management, which enables robust control over user profile data while requiring fewer resources than traditional Windows roaming profiles.

We discussed what is required to implement Persona Management, how to implement it using the Persona Management Group Policies, and how to use the remainder of those policies to further control the Persona Management implementation.

In the next chapter, we will discuss how to create and manage View desktop pools, which are the most frequent and important tasks that a View Administrator will undertake.

8
Creating VMware Horizon View Desktop Pools

A **View desktop pool** is a collection of desktops that users select when they log in using the View client. A pool can be created based on a subset of users, such as finance, but this is not explicitly required unless you will be deploying multiple Virtual Desktop master images. The pool can be thought of as a central point of desktop management within View; from it you create, manage, and entitle access to View desktops. This chapter will discuss how to create a desktop pool using the View Manager Admin console, an important administrative task.

In this chapter, we will learn:

- An overview of View desktop pools
- Desktop pool common terms
- How to create two different types of desktop pools
- How to monitor the provisioning of a View desktop pool
- Common problems encountered when provisioning a desktop pool
- How to entitle access to a desktop pool

A number of storage array vendors have given their products the ability to provision desktops outside the View environment and then register them directly within View. If you choose to use this method to create your Virtual Desktops, you should review the vendor documentation carefully to understand the impact it has on managing your View environment. For example, you will need to know how Virtual Desktop maintenance, discussed in *Chapter 10*, *Performing View Desktop Maintenance*, differs if the desktops were not deployed using View.

VMware desktop pool overview

Creating a View desktop pool is commonly the final step in the process of deploying Virtual Desktops. In most cases, when you are ready to deploy your first desktop pool you have done at least the following:

- Created the necessary infrastructure services needed for your Virtual Desktop deployment, such as DHCP, DNS, Active Directory, and so on
- Deployed or identified the vCenter Server you will use with View
- Deployed at least one vSphere Server to host your Virtual Desktops
- Deployed and configured at least one View Connection Server
- If linked clones will be used, you will have deployed View Composer either on the View vCenter Server or on a dedicated server
- Created a Virtual Desktop master image, discussed in *Chapter 11*, *Creating a Master Virtual Desktop Image*.
- Each of these items will be required in order to deploy your first desktop pool. Prior to placing your View environment into production, it is important to verify that you have deployed sufficient resources to meet your View scalability and availability needs.

This chapter will focus on using the View Manager Admin console to create your desktop pools. You also have the option of using PowerShell to create and manage your desktop pools; this is discussed in *Chapter 14*, *Managing View with PowerCLI*. (this can be downloaded from the Packt Publishing website)

Desktop pool common terms

There are a number of options that must be selected when creating a View desktop pool. Understanding what these options mean is important, as they will impact not only how the desktops are deployed, but also what options users will have when they attempt to access those desktops.

Later on in the chapter, we will go through the deployment of two different desktop pools, but prior to that it is important to familiarize yourself with some of the terminologies that will come up during that process. Once you understand these terms, you will be able to create any type of desktop pool, something which we cannot demonstrate in this book due to the sheer number of options that exist.

Term	Definition
Adobe Flash settings for Remote sessions	Pool settings that control Adobe Flash quality and frame rate for View Clients, both of which affect connection bandwidth utilization.
Automated assignment	Used with Dedicated desktop assignment; if a user does not already have a desktop assigned, they will automatically be assigned a free one from the desktop pool.
Automated Pool	A pool that uses desktops provisioned using View.
Blackout times	Used to set when the View Storage Accelerator and VM disk space reclamation will not run.
Connection Server restrictions	Used to restrict what Connection Servers can be used to access a desktop pool; commonly used to assign pools to a Connect Server that has specific security settings.
Dedicated assignment	A desktop from the pool is assigned to a user, and from then on it is available only to that user unless the assignment is manually removed.
Display name	The desktop pool name that will be displayed in the View Client login window.
Disposable file redirection	Used with a linked-clone desktop to redirect disposable files to a non-persistent disk rather than the OS disk.
Floating assignment	Desktops are not assigned to any one user; if not in use, they are available to anyone with permissions to access the desktop pool.
Full virtual machine desktops	Full virtual machine clones of a vCenter template created from the Virtual Desktop master image.
Guest customization	The process of preparing a View desktop for placement within an Active Directory domain.
Desktop pool ID	The unique identifier for a desktop pool within a View organization.
Manual Pool	A pool that uses desktops that already exist within vCenter, such as those configured using storage array-based cloning technologies.
Naming pattern	Used by View to generate names for the virtual machines that it creates.
Non-persistent disk	Optional part of a View Composer linked-clone desktop; used to store disposable files that will be deleted automatically when the user's session ends.
Persistent disk	Optional part of a View Composer linked-clone desktop; used to retain user profile data during a View Composer refresh, recompose, or rebalance operation.

Term	Definition
QuickPrep	Similar to Microsoft Sysprep; optional method offered by View for customizing and then joining linked-clone desktops to an Active Directory domain.
Reclaim VM disk space	Reclaims blocks that are no longer being used by the virtual machine operating system.
Replica disk	Part of a View Composer linked-clone desktop; the replica disk is a read-only copy of the Virtual Desktop master image virtual hard disk that is shared among the desktops in the pool.
Storage Overcommit	Determines how View places new VMs on selected datastores. The more aggressive the setting, the more VMs View will place on the datastore while reserving less space for sparse disk growth.
Terminal Services pool	A pool that provides Microsoft Terminal Services sessions to servers as desktops.
Use native NFS snapshots (VAAI)	This feature eliminates the need for the vSphere hosts to read and write data during the creation of View Composer linked clones by using the built-in capabilities of the storage array to create the virtual machines.
View Composer linked clone	Clones of a Virtual Desktop master image that share the same base disk. Changes to linked-clone desktops are written to a dedicated virtual hard disk attached to the linked-clone virtual machine.
View folder	Used to organize desktop pools within a View organization, for reasons such as delegated administration.
View Storage Accelerator	Uses up to 2 GB of vSphere host RAM to cache frequently used blocks of Virtual Desktop data.
VM folder location	The location where View will place the virtual machines it creates in the vCenter VMs and Templates view.

Desktop pool considerations

There are multiple decisions that must be made prior to configuring your first desktop pool. These choices will impact your infrastructure and how your Virtual Desktops work, which is why they must be considered in advance.

Linked clone versus full clone

View can provision two different desktop types: View Composer linked clones and full clones. Before we discuss the differences, let's consider what it is about them that is the same:

- From the perspective of an end user, a linked-clone and full-clone desktop look exactly the same.
- The master image for both is prepared using the same tuning techniques, discussed in *Chapter 11*, *Creating a Master Virtual Desktop Image*.
- Deciding on what type of clone type to use is not an easy task. While linked clones have some definite advantages, which I will describe in detail, to maintain that advantage you must adopt new ways of performing desktop maintenance.

Full Clone desktops

Full clone View desktops are created using a virtual desktop master image that has been converted to the vSphere template format. A full clone is an independent copy of that template, managed separately from other desktops and the template on which it was based. Aside from the fact that it was created from a template, from a conceptual standpoint it is very similar to the physical desktop that it may have replaced. As a result, the lifecycle of the full clone desktop is typically managed using the same techniques used with a physical desktop.

View Composer linked clones

View Composer linked clone desktops are created from a Virtual Desktop master image. While a full clone is created from a vSphere template, creating linked clones requires a Virtual Desktop master image that is in the standard virtual machine format. Once the image is ready for deployment, the only requirement is that it be powered down and a snapshot of the image is created. A snapshot is required so that View Composer can create the replica of the Virtual Desktop master image. This replica will be used as the base image for all of the linked-clone desktops in the pool.

In addition, as the Virtual Desktop master image has a snapshot the View administrator will be able to power it on and make changes to it, while still retaining the ability to deploy additional desktops based on the condition of the desktop when the snapshot was taken. When it is time to deploy the updated image, you would simply take a second snapshot and recompose the desktops using the techniques described in *Chapter 10*, *Performing View Desktop Maintenance*. Assuming you left the initial snapshot in place, you could even recompose the desktops to that snapshot as well as the second one that you created.

A linked-clone desktop has a number of advantages over a full-clone desktop. Some of these advantages include:

- Linked-clone desktops share the same parent virtual disk for read operations; therefore the amount of disk space they require is greatly reduced.
- Linked-clone desktops can be recomposed, which is a process where their replica disks are replaced with an updated version that has software updates or other changes applied. Rather than apply updates to individual desktops, you can update the master image once, and then use a recompose operation to update the replica disks, which applies those changes to the entire desktop pool. The recompose process is described in *Chapter 10*, *Performing View Desktop Maintenance*.

Using a recompose to upgrade the operating system version is not supported.

- Linked-clone desktops can be refreshed, a process that deletes the contents of the linked-clone OS and disposable data disks, which returns them to the same state they were in when initially deployed. This enables you to discard any changes that were made after the desktop was deployed, allowing for tight control over the end user experience. Desktops can be refreshed at a specific time, when a specified amount of disk space has been used, or even after every logoff. The refresh process is described in *Chapter 10*, *Performing View Desktop Maintenance*.
- A linked-clone desktop pool can be rebalanced, which redistributes linked-clone storage evenly across datastores. Individual linked-clone disk utilization will vary over time, leading to an imbalance in storage utilization across all the datastores. A rebalance operation addresses this by redistributing the linked-clone desktops evenly across the datastores. The rebalance process is described in *Chapter 10*, *Performing View Desktop Maintenance*.

Storage vMotion is not supported with linked-clone desktops. You must use a rebalance operation to relocate or rebalance linked-clone desktop storage.

While a full-clone desktop requires only one virtual hard disk, a linked-clone desktop requires up to four virtual hard disks, as well as the replica disk that is shared among the desktops in the pool. These disks were first discussed in the *Overview of VMware Horizon View Composer* section of *Chapter 3*, *Implementing VMware Horizon View Composer*, and include:

- **Replica disk**: When a desktop pool is created, a clone of the Virtual Desktop master image hard disk is created on each datastore that will contain linked clones. These clones are referred to as **replica disks**; each is shared among the linked clones on their respective datastore. Alternatively, a replica disk can also be created on a dedicated datastore, which results in only one replica disk being created rather than one for each linked-clone datastore. The replica disks are read-only; all changes are written to the individual linked-clone virtual hard disks. The replica disk is what is replaced when a recompose operation occurs.
- **OS disk**: Stores the system data that associates the linked clone with the base image and functions as a unique desktop. The contents of the OS disk are not retained during a refresh, recompose, or rebalance operation.
- **Persistent disk**: Optional disk used in dedicated assignment pools only. This disk can be used to store user profile data, and the contents are retained during a refresh, recompose, or rebalance operation. Persistent disks can be stored in a different location from the other linked clone disks, and can be attached to other linked clones if required. If a persistent disk is not used, the user profile data is stored in the OS disk and will be lost during refresh or recompose operations.

Placing the persistent disks in a separate datastore may be useful for backup purposes, as that is the only disk in a linked clone whose contents are retained throughout the life of the desktop. By placing these disks within the same datastore, it may make it easier for you to protect them using backup utilities, or with the snapshot features present in many storage arrays.

- **Disposable data disk**: Optional disk used to store the OS paging and temporary files. The contents of the disposable data disk are discarded when the desktop is powered off and during refresh and recompose operations. If a disposable data disk is not used, the page file and temporary files are stored in the OS disk.
- **QuickPrep configuration data disk**: This disk stores QuickPrep and other OS-related data that must be preserved during refresh and recompose operations. Despite the name, this disk is also created if Sysprep was used to customize the desktops.

Due to how a linked-clone desktop works, it is important to remember that they should not be managed using the same techniques as a typical virtual machine. Were you to manage linked clones and their guest operation system using typical means, you may find that the advantages of the linked-clone desktop start to disappear. Some examples of this include:

- If you were to apply software patches to linked clones individually, rather than using a recompose operation, the linked-clone virtual hard disks will grow significantly over time. This defeats the storage efficiencies that are one of the primary reasons for choosing linked clones.
- Recompose, refresh, and rebalance operations all change the state of the linked-clone Virtual Desktop, which can affect utilities such as indexing programs. If these operations lead to resource-intensive operations, such as a file index, every time they occur, it may be that they need to be disabled or their behavior altered. This topic is discussed further in *Chapter 11*, *Creating a Master Virtual Desktop Image*.

Generally speaking, you should approach managing linked clones from the master Virtual Desktop image wherever possible as this enables you to realize most of the advantages of linked clones. If a proposed change can be done on the master Virtual Desktop image, and then rolled out to users using a recompose operation, that is the preferred method of working with linked clones. *Chapter 10*, *Performing View Desktop Maintenance* will provide more information about how to manage linked-clone desktops.

QuickPrep versus Sysprep

Windows Sysprep is a utility included with the Windows operating system that is used to personalize a Windows image. Rather than install Windows on each machine individually, organizations can apply a preconfigured image to a machine and then use Sysprep to generate the identifiers that make that installation of Windows unique.

When deploying linked-clone desktops you have the option of using Sysprep or the included VMware QuickPrep tool to customize the operating system. The tools do not perform all of the same tasks, so it is important to understand what differs when you choose one over the other.

The following table details the differences between Sysprep and QuickPrep:

Task	QuickPrep	Sysprep
Change security identifiers on the parent image	No	Yes
Change the computer name	Yes	Yes
Join the new virtual machine to the domain	Yes	Yes
Remove local accounts	No	Yes
Remove parent image from the domain	No	Yes
Re-use pre-existing AD computer accounts	Yes	Yes
Generate a new SID	No	Yes
Update language, regional, data, and time settings	No	Yes
Reboots required	0	1
Requires a configuration file and Sysprep utility	No	Yes

It is important to consider the differences between Sysprep and QuickPrep when determining which method to choose. In some environments, it may be that QuickPrep cannot be used because it does not generate a new SID for the guest operating system. In other environments, it may be that there are no issues with using QuickPrep. Generally speaking, QuickPrep enables faster desktop deployment, which affects not only desktop pool creation but recompose operations as well. Regardless of which method you choose, it is important to monitor the behavior of the desktop during a pilot program to ensure that the desktops are functioning as expected.

You should use **Microsoft Key Management Services** (**KMS**) to license and activate your Virtual Desktops. This is particularly important when using linked clones, as nearly every View Composer maintenance task will initiate a license activation request, regardless of whether QuickPrep or Sysprep was used. These requests would quickly exhaust ordinary **Microsoft Multiple Activation Key** (**MAK**) license keys. Microsoft typically provides both KMS and MAK keys to organizations that purchase volume licensed versions of their Windows and Office products.

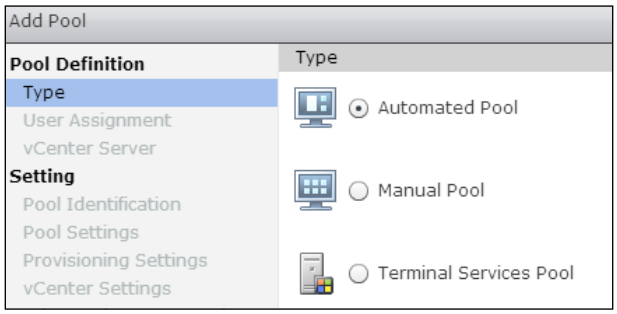
Creating a View desktop pool

This section will provide an example of how to create two different View dedicated assignment desktop pools, one based on View Composer linked clones and another based on full clones. Also discussed will be how to use the View Manager Admin console and the vSphere client to monitor the provisioning process.

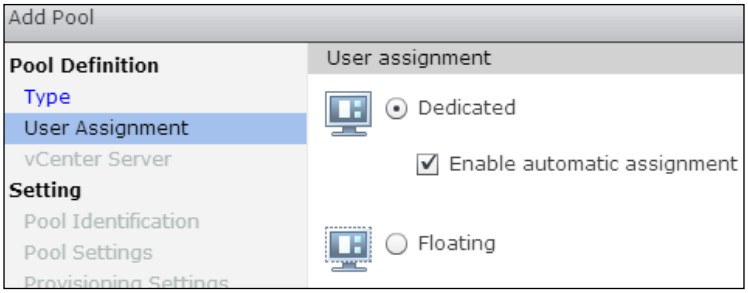
Creating a pool using View Composer linked clones

The following steps outline how to use the View Manager Admin console to create a dedicated assignment desktop pool using View Composer linked clones. As discussed previously, it is assumed that you already have a Virtual Desktop master image that you have created a snapshot of. During each stage of the pool creation process, a description of many of the settings is displayed in the right-hand side of the **Add Pool** window. In addition, a question mark appears next to some of the settings; click on it to read important information about the specified setting.

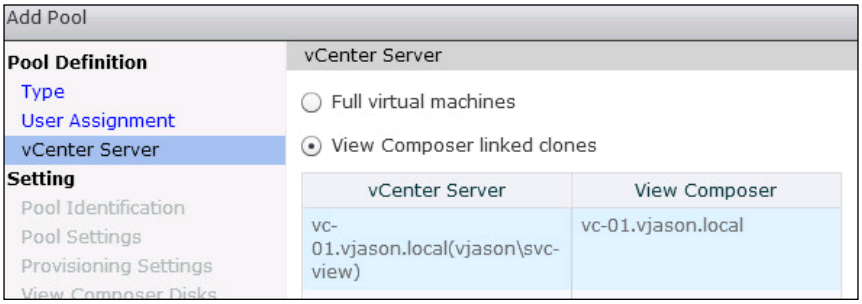
1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Inventory** | **Pools** page within the console.
3. Click on the **Add** button in the **Pools** window to open the **Add Pool** window.
4. In the **Type** section under **Pool Definition**, select the **Automated Pool** radio button as shown in the following screenshot and then click on **Next**:



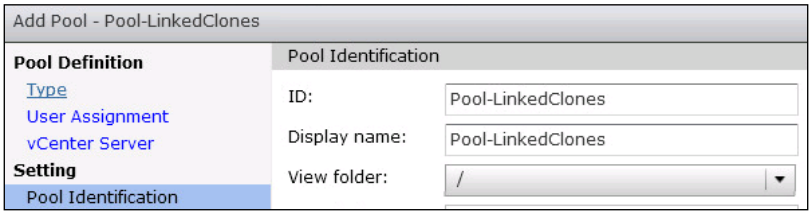
5. In the **User Assignment** section under **Pool Definition**, select the **Dedicated** radio button and check the **Enable automatic assignment** checkbox as shown in the following screenshot. Then click on **Next**:



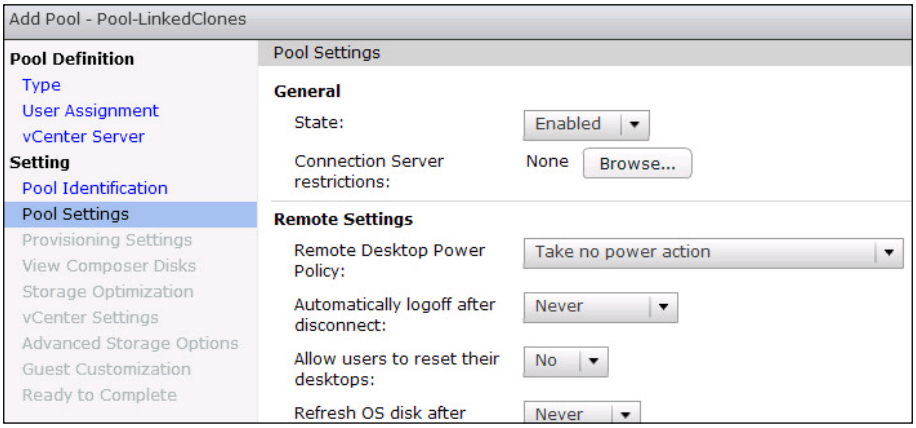
6. In the **vCenter Server** section under **Pool Definition**, select the **View Composer linked clones** radio button, highlight the vCenter server as shown in the following screenshot, and then click on **Next**:



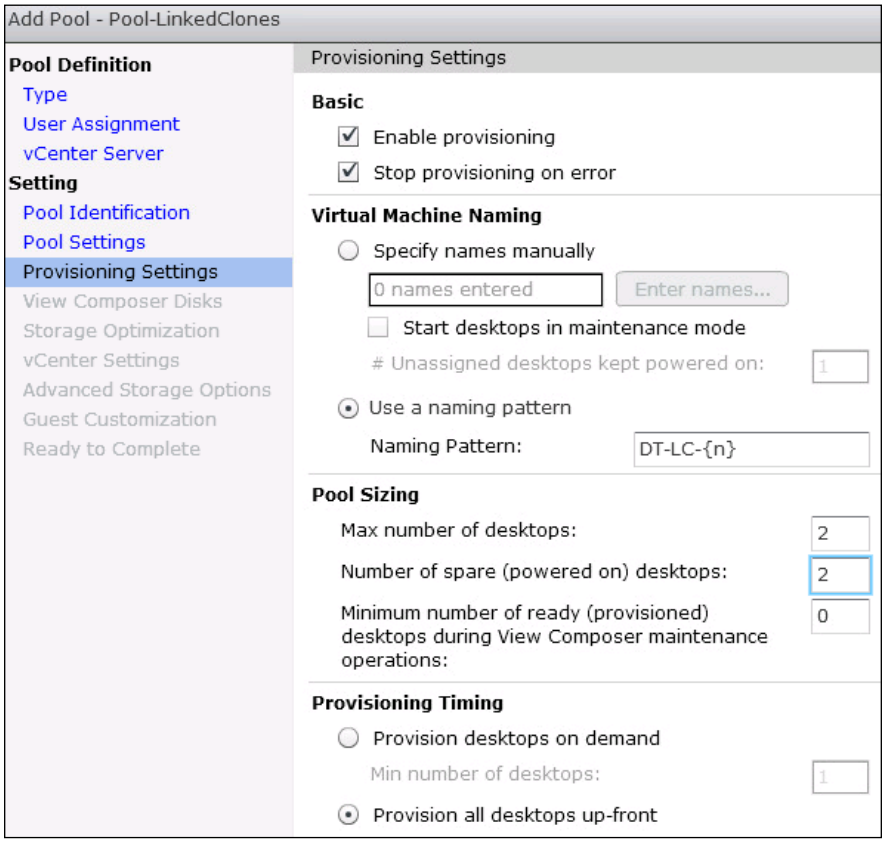
7. In the **Pool Identification** section under **Setting**, populate the pool **ID:** and **Display Name:** fields as shown in the following screenshot, and then click on **Next**:



8. In the **Pool Settings** section under **Setting**, configure the various settings for the desktop pool. Many of these options are self-explanatory; those that are not are described in the *Desktop pool common terms* section of this chapter. These settings can also be adjusted later if desired. Once finished, click on **Next**:

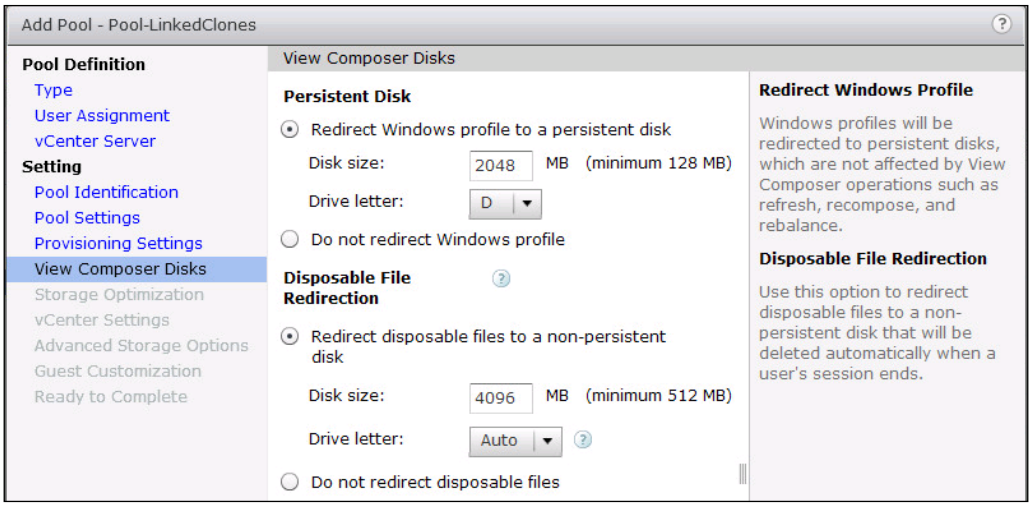


9. In the **Provisioning Settings** section under **Setting**, configure the various provisioning options for the desktop pool that include the desktop naming format, the number of desktops, and the number of desktops that should remain available during View Composer maintenance operations. Once finished, click on **Next**:

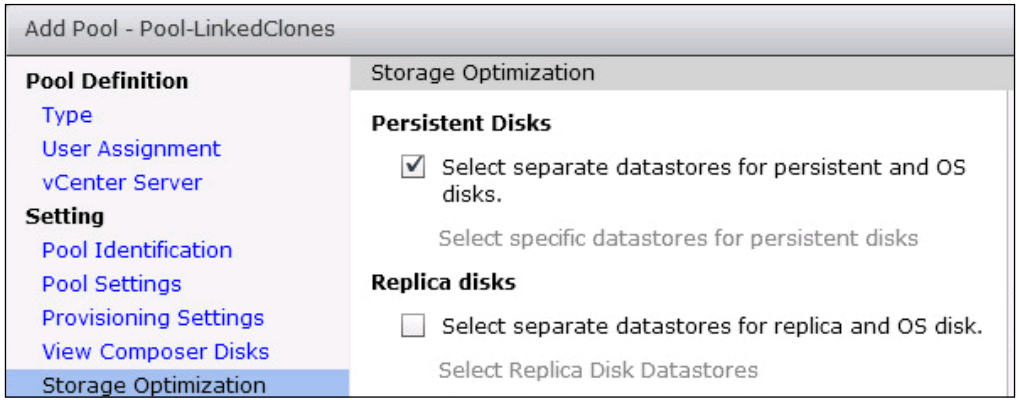


When creating a desktop naming pattern, use a `{n}` to instruct View to insert a unique number in the desktop name. For example, using `DT-LC-{n}` as shown in the preceding screenshot will name the first desktop `DT-LC-1`, the next `DT-LC-2`, and so on. Refer to the *Appendix*, *Advanced Details about Key Horizon View Features*, for a more detailed explanation of how to customize the naming patterns.

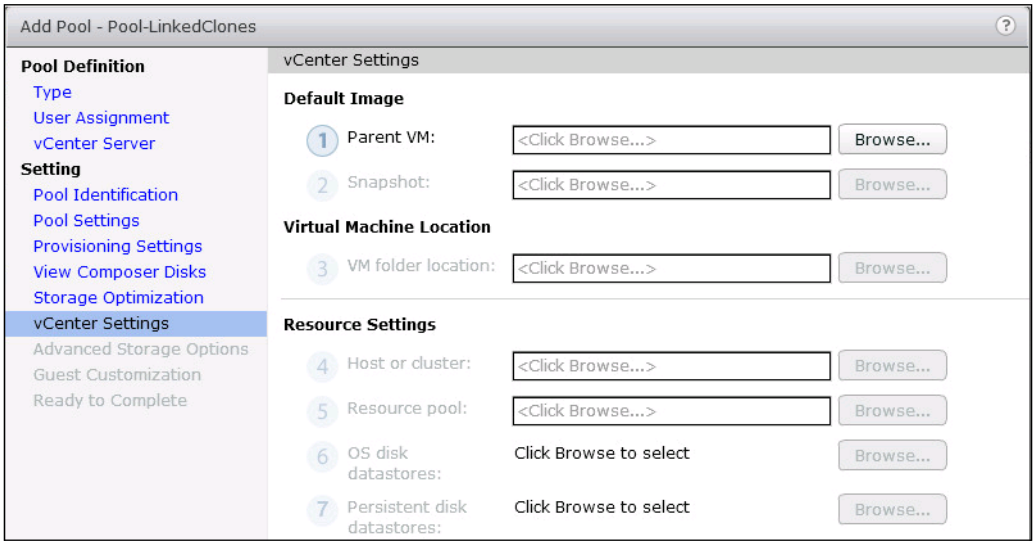
10. In the **View Composer Disks** section under **Setting**, configure the settings for your optional linked-clone disks. By default, both a **Persistent Disk** for user data and a non-persistent disk for **Disposable File Redirection** are created. Once finished, click on **Next**:



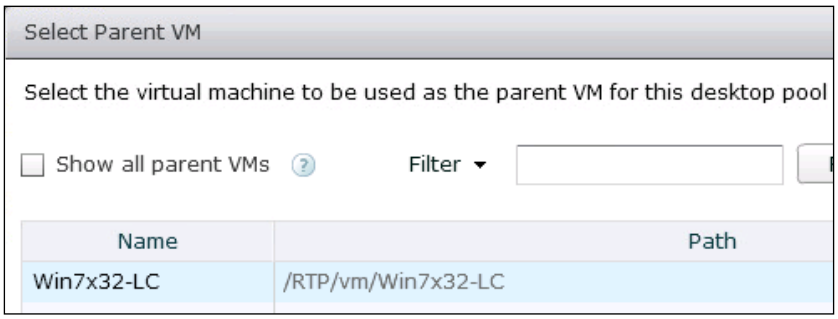
11. In the **Storage Optimization** section under **Setting**, configure the optional settings to separate some of your linked-clone disks. In our example, we will separate the persistent disks from the linked-clone OS disks. Once finished, click on **Next**:



12. In the **vCenter Settings** section under **Setting**, we will need to configure seven different options that include selecting the parent virtual machine, which snapshot of that virtual machine to use, what vCenter folder to place the desktops in, what vSphere cluster and resource pool to deploy the desktops to, and what datastores to use. Click on the **Browse** button next to the **Parent VM** setting to begin the process:

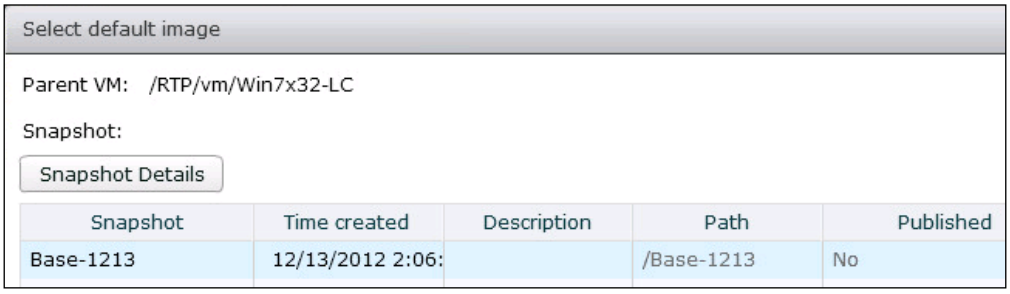


13. In the **Select Parent VM** window, highlight the Virtual Desktop master image that you wish to deploy desktops from, as shown in the following screenshot. Click on **OK** when the image is selected to return to the **vCenter Settings** window:

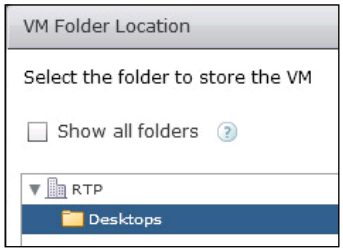


The virtual machine will only appear if a snapshot has been created.

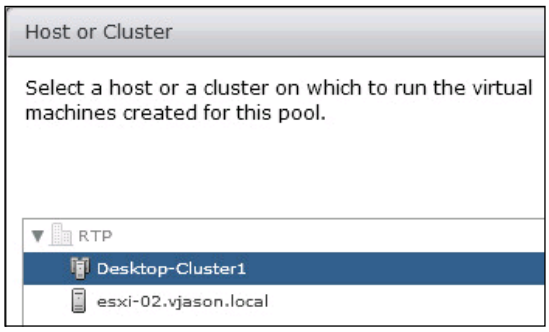
14. Click on the **Browse** button next to the **Snapshot:** setting to open the **Select default image** window. Select the desired snapshot, as shown in the following screenshot, and click on **OK** to return to the **vCenter Settings** window:



15. Click on the **Browse** button next to the **VM folder Location** setting to open the **VM Folder Location** window, as shown in the following screenshot. Select the folder within vCenter where you want the desktop virtual machines to be placed, and click on **OK** to return to the **vCenter Settings** window:

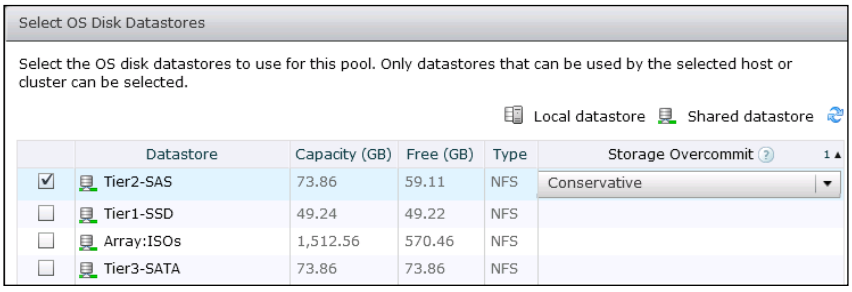


16. Click on the **Browse** button next to the **Host or cluster** setting to open the **Host or Cluster** window, as shown in the following screenshot. Select the cluster or individual vSphere host within vCenter where you want the desktop virtual machines to be created, and click on **OK** to return to the **vCenter Settings** window:



17. Click on the **Browse** button next to the **Resource pool** setting to open the Resource Pool window, as shown in the following screenshot. If you intend to place the desktops within a resource pool you would select that here; if not select the same cluster or vSphere host you chose in the previous step. Once finished, click on **OK** to return to the **vCenter Settings** window:

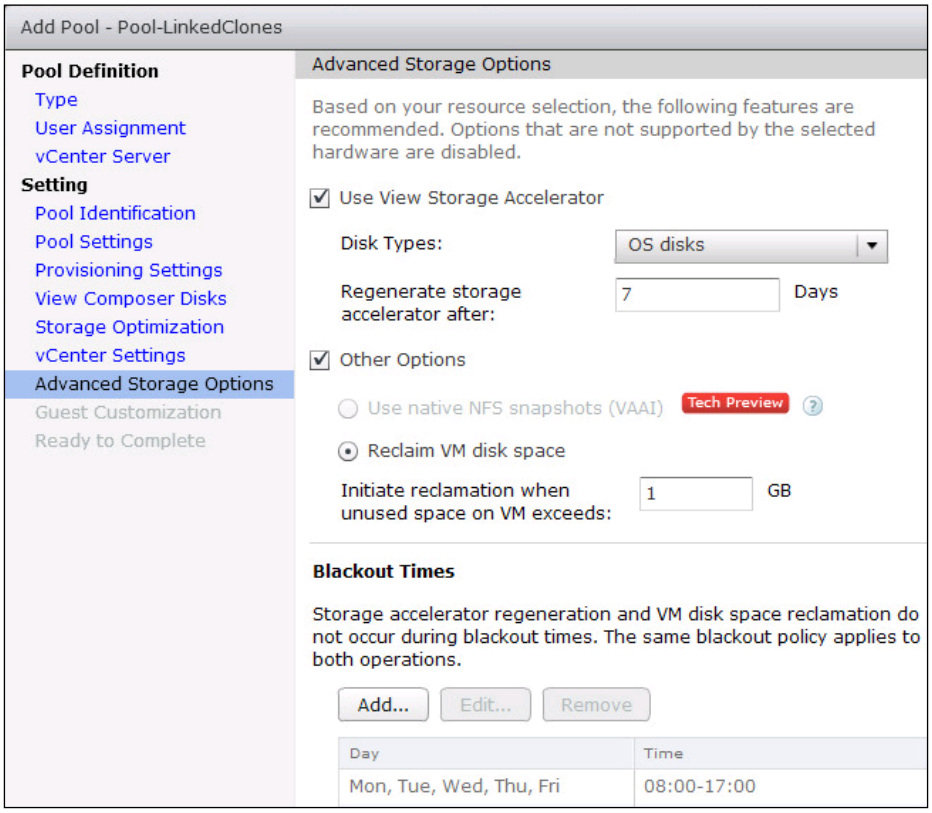
18. Click on the **Browse** button next to the OS disk datastores setting to open the **Select OS Disk Datastores** window, as shown in the following screenshot. Select the datastore or datastores where you want the disks to be created, configure the **Storage Overcommit** value if desired, and click on **OK** to return to the **vCenter Settings** window:



19. Click on the **Browse** button next to the **Persistent disk datastores** setting to open the **Select Persistent Disk Datastores** window, as shown in the following screenshot. Select the datastore or datastores where you want the disks to be created, configure the **Storage Overcommit** value if desired, and click on **OK** to return to the **vCenter Settings** window.

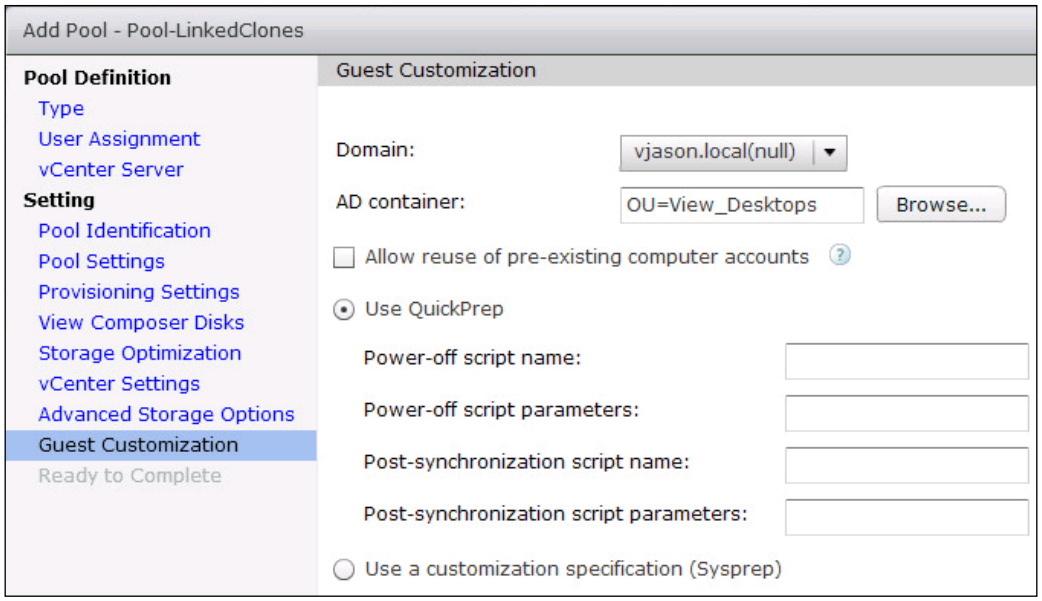
If you had not opted to use a different datastore for your persistent disks in step 11, this option would not have been presented.

20. The **vCenter Settings** window should now have all options selected, enabling the **Next** button. Click on the **Next** button to move on to the **Advanced Storage Options** window.
21. In the **Advanced Storage Options** section under **Setting**, if desired select and configure the **Use View Storage Accelerator** and **Other Options** radio buttons to enable those features. In our example, we have enabled both the **Use View Storage Accelerator** and **Reclaim VM disk space** options, and configured **Blackout Times** to ensure that these operations do not occur between 8 A.M. and 5 P.M. on weekdays. When finished, click on **Next**:



The **Use native NFS snapshots (VAAI)** feature is a technical preview provided by VMware but not yet suggested for production use. In addition, the feature requires the support of the storage array.

22. In the **Guest Customization** section under **Setting**, select the **Domain** where the desktops will be created, the **AD container** where the computer accounts will be placed, whether to use QuickPrep or Sysprep, and any other options as required. When finished, click on **Next**:



23. In the **Ready to Complete** section under **Setting**, verify that the settings we selected were correct, using the **Back** button if needed to go back and make changes. If all the settings are correct, click on **Finish** to initiate the creation of the desktop pool.

The desktop pool and Virtual Desktops will now be created. To monitor the creation of the desktops, review the *Monitoring the desktop creation process* section of this chapter. Also located in this chapter is the *Entitling access to desktop pools* section, which outlines how to grant users access to the desktop pools that we have created.

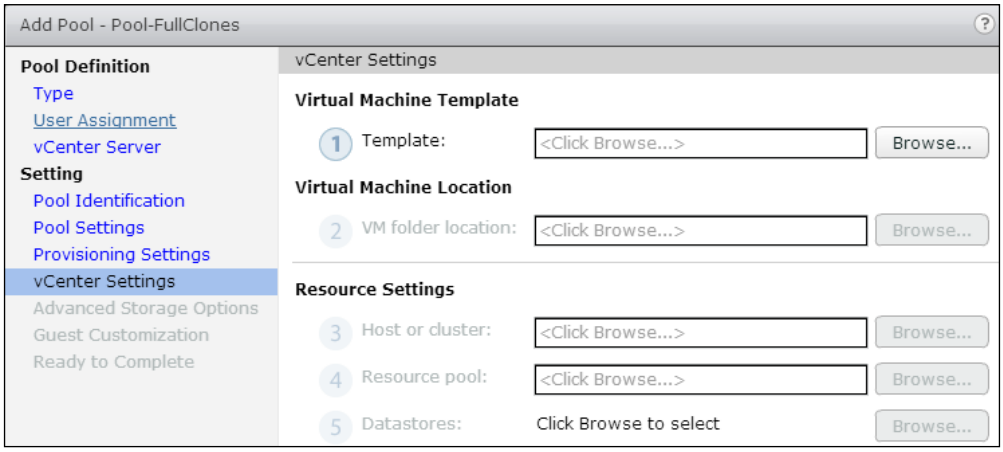
Creating a pool using full clones

The process used to create a pool used full-clone Virtual Desktops, which is similar to that used to create a linked-clone pool. As discussed previously, it is assumed that you already have a Virtual Desktop master image that you have converted to a vSphere template. In addition, if you wish View to perform the virtual machine customization, you will need to create a **Customization Specification** using the **vCenter Customization Specifications Manager**. The Customization Specification is used by the Windows Sysprep utility to complete the guest customization process. Visit the VMware vSphere virtual machine administration guide (`http://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-pubs.html`) for instructions on how to create a Customization Specification.

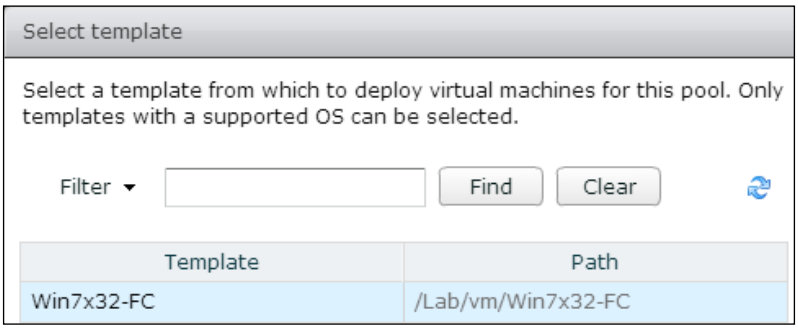
The following steps outline the process used to create the desktop pool. Screenshots are included only when the step differs significantly from the same step in the *Creating a pool using View Composer linked clones* section.

1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Inventory** | **Pools** page within the console.
3. Click on the **Add** button in the **Pools** window to open the **Add Pool** window.
4. In the **Type** section under **Pool Definition**, select the **Automated Pool** radio button and then click on **Next**.
5. In the **User Assignment** section under **Pool Definition**, select the **Dedicated** radio button, check the **Enable automatic assignment** checkbox, and then click on **Next**.
6. In the **vCenter Server** section under **Pool Definition**, select the **Full virtual machines** radio button, highlight the desired vCenter server, and then click on **Next**.
7. In the **Pool Identification** section under **Setting**, populate the pool **ID:** and **Display Name** fields and then click on **Next**.
8. In the **Pool Settings** section under **Setting**, configure the various settings for the desktop pool. These settings can also be adjusted later if desired. Once finished, click on **Next**.

9. In the **Provisioning Settings** section under **Setting**, configure the various provisioning options for the desktop pool that include the desktop naming format and number of desktops. Once finished, click on **Next**.
10. In the **vCenter Settings** section under **Setting**, we will need to configure settings that set the virtual machine template, what vSphere folder to place the desktops in, which vSphere host or cluster to deploy the desktops to, and which datastores to use. Other than the **Template** setting described in the next step, each of these settings is identical to those seen when creating a View Composer linked clone pool in the previous section. Click on the **Browse** button next to each of the settings in turn and select the appropriate options:

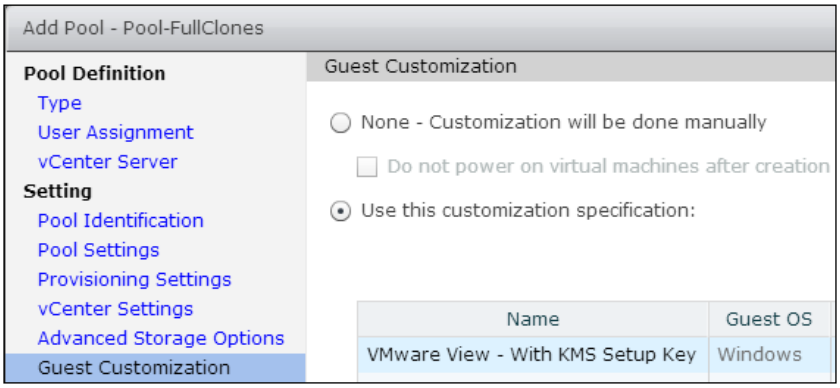


11. To configure the **Template** setting, select the vSphere template that you created from your Virtual Desktop master image as shown in the following screenshot:



A template will only appear if one is present within vCenter.

12. Once all the settings in the **vCenter Settings** window have been configured, click on **Next** to move to the **Add Pool - Advanced Storage Options** window.
13. In the **Advanced Storage Options** window, if desired select and configure the **Use View Storage** radio buttons and configure **Blackout Times**. When finished, click **Next**.
14. In the **Guest Customization** section under **Setting**, select either the **None – Customization will be done manually** or **Use this customization specification** radio button. In our example, we have selected the **Customization Specification** that we created within vCenter. QuickPrep cannot be used when creating full-clone Virtual Desktops.



Manual customization is typically used when the template has been configured to run Sysprep automatically upon startup, without requiring any interaction from either View or VMware vSphere.

15. In the **Ready to Complete** section under **Setting**, verify that the settings we selected were correct, using the **Back** button if needed to go back and make changes. If all the settings are correct, click on **Finish** to initiate the creation of the desktop pool.

The desktop pool and Virtual Desktops will now be created. To monitor the creation of the desktops, review the *Monitoring the desktop creation process* section of this chapter. Also located in this chapter is the *Entitling access to desktop pools* section that outlines how to grant users access to the desktop pools that we have created.

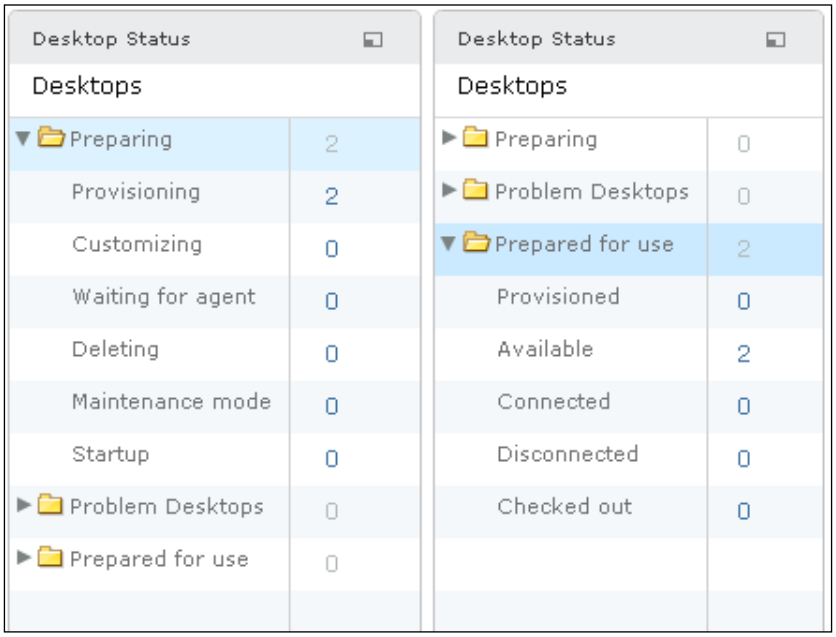
Monitoring the desktop creation process

The amount of time it takes to create the desktop pool varies based on a number of factors. There are multiple locations within the View Manager Admin console and the vSphere Client where you can monitor the progress of the desktop deployment.

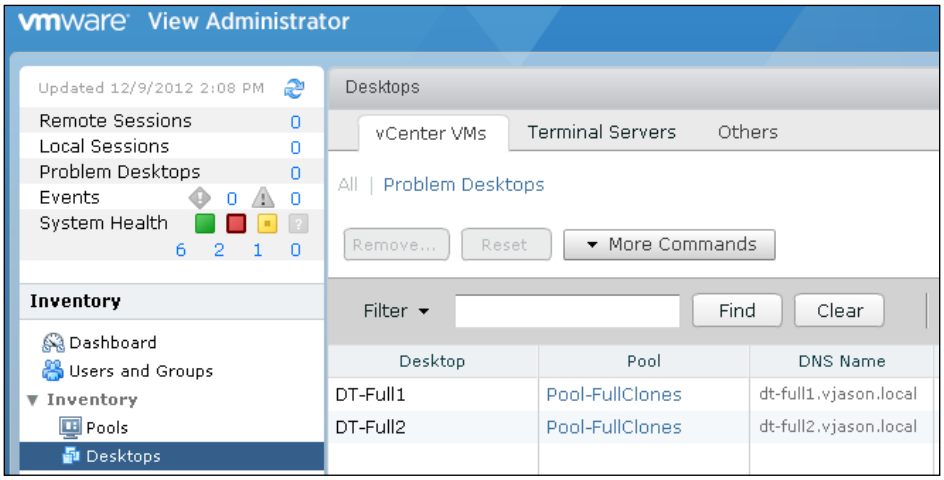
View Manager Admin console

View desktops are displayed throughout the View Manager Admin console as soon as they are initially created within vCenter. This section details two areas of the console where you can view the current status of the desktops.

- **Dashboard**: As desktops move through the deployment and configuration process, their status on the **Desktop Status** dashboard will change. The following screenshot shows how the **Desktop Status** dashboard window changes as desktops move from **Preparing** stage (left) to the **Prepared for use** stage (right).



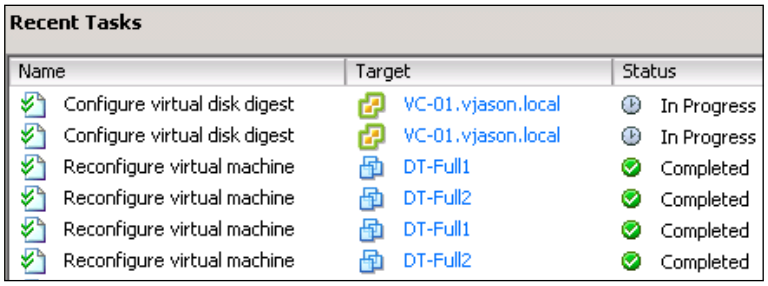
- **Desktops**: This option appears under **Inventory**. This window will display each of the View desktops, along with their current status.



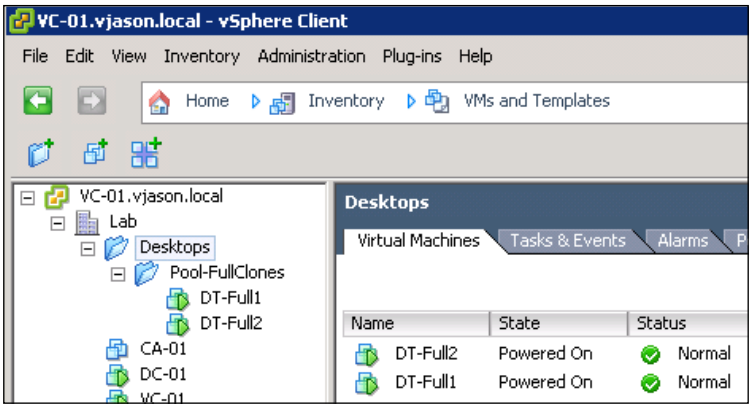
The vSphere client

Creating Virtual Desktops will generate a number of vCenter tasks, during which the desktops will begin to appear within the vCenter Console. Monitor the following areas of the vSphere client to verify that the desktop pool is being created.

- **Recent Tasks**: This window is shown in the following screenshot and will display the tasks associated with the creation and configuration of the Virtual Desktops:



- **VMs and Templates** or **Hosts and Clusters**: View desktops will appear in these views of the vSphere client just as with any other virtual machine. The following screenshot shows View desktops as seen in the VMs and Templates view, within the folder we specified during the creation of the desktop pool:



Common provisioning problems

There are a number of different issues that can arise during the deployment of desktop pools. While it is impossible to try and list them all, the following represent some of the more common issues that can occur:

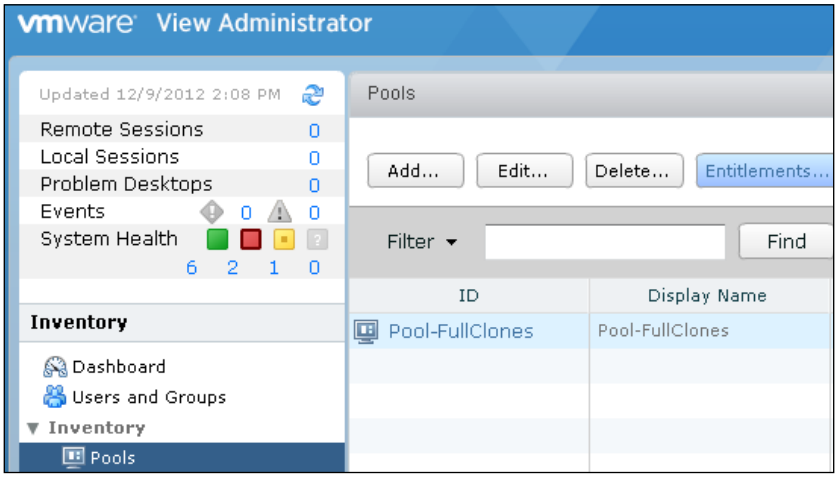
- **Undersized or misconfigured DHCP address pool**: This is more common with linked clones, which change MAC addresses when redeployed; this can exhaust a DHCP pool. Linked-clone environments typically work best when the DHCP lease time is very short. Desktops cannot complete the provisioning process without access to the network.
- **Issues with Windows operating system activation**: If KMS services are not functioning within the domain, the provisioning process will fail.
- **Insufficient permissions within vCenter**: If the accounts used by View and View Composer do not have the required permissions within vCenter, the provisioning process will fail.
- **DNS not functioning properly**: DNS is integral to desktop provisioning. If the desktops cannot resolve the IP addresses or infrastructure services including Active Directory and View components, the provisioning process will fail.

The View event log, located in the **Monitoring | Events** menu of the View Manager Admin console, contains detailed information that can be used to troubleshoot the provisioning process.

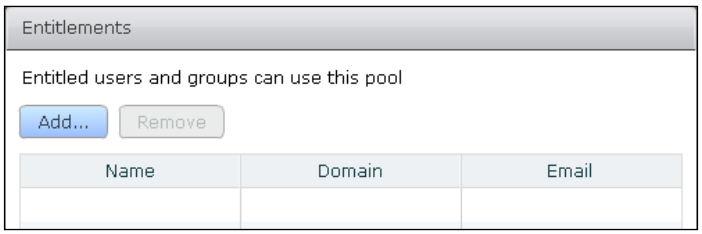
Entitling access to desktop pools

The following steps outline how to grant users or AD security groups access to a View desktop pool, a necessary task since no access is granted by default. This can be done while the pool is still being provisioned.

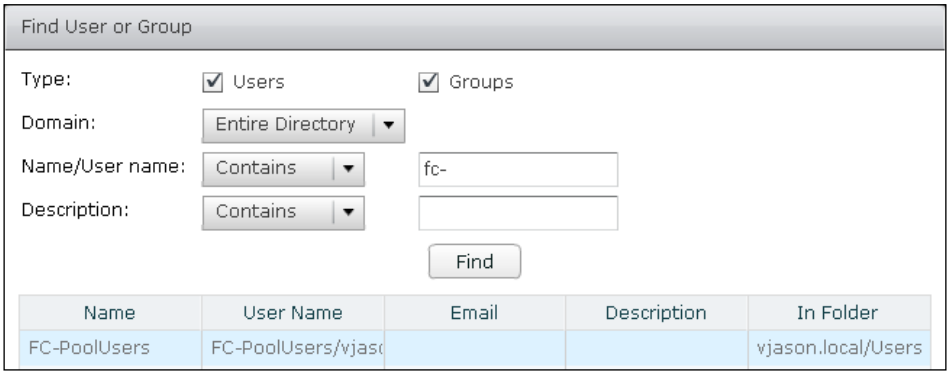
1. Log on to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Inventory** | **Pools** page within the console.
3. Highlight the pool you wish to entitle, as shown in the following screenshot, and click on the **Entitlements** button to open the **Entitlements** window:



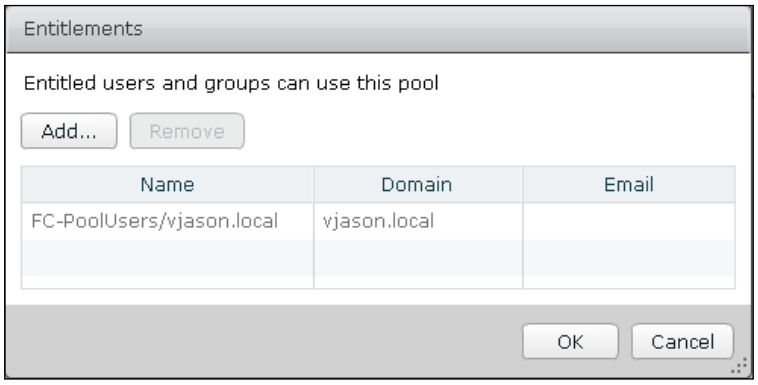
4. In the **Entitlements** window shown in the following screenshot, click on the **Add** button to open the **Find User or Group** window:



5. In the **Find User or Group** window shown in the following screenshot, use the **Name/User name:** or **Description:** fields to search for the user or group to which you wish to grant access. In the following example, we searched for a security group that was created specifically for a View named `FC-PoolUsers`. Highlight the desired user or group, and click on **OK** to return to the **Entitlements** window:



6. Repeat step 4 as needed to entitle additional users or groups.
7. If all the required users and groups have been added, click on the **OK** button in the **Entitlements** window shown in the following screenshot to complete the action:



The selected users and groups now have access to the available desktops within the desktop pool. *Chapter 9, VMware Horizon View Client Options*, contains instructions on how to use the View Client to connect to these newly provisioned desktops.

Summary

In this chapter, we have learned about View desktop pools. In addition to learning how to create two different types of desktop pools, we were introduced to a number of key concepts that are part of the pool creation process.

We discussed the difference between a full-clone and linked-clone Virtual Desktop, how Sysprep differs from QuickPrep, how to monitor the provisioning of a View desktop pool, the types of issues that can prevent a pool from provisioning successfully, and how to grant users or security groups access to desktop pools through a process called entitlement.

In *Chapter 9*, *VMware Horizon View Client Options*, we will discuss the various options of the VMware Horizon View Client, including both the software-based client as well as dedicated thin clients that are used solely to access View.

9
VMware Horizon View Client Options

VMware Horizon View supports a number of different client hardware and software platforms, providing organizations with maximum flexibility when deciding what endpoint they wish to use. This chapter will examine a number of different View client options and discuss their characteristics, requirements, advantages, and disadvantages.

In this chapter we will learn:

- What are the supported operating systems for the standalone View Client?
- What is the difference between thin and zero View Clients?
- What are the requirements for the View Client with Local Mode?
- Why would you choose one client type over another?
- How to install the View Client in Windows
- View client command-line options for installation and usage

View Client platforms

There are four different types of clients that can be used to connect to a View environment. These client types include:

- Traditional View software clients installed within an existing OS
- Thin clients, which are devices that use a limited customized or embedded OS that is optimized to be used as a View Client

- Zero clients, which are devices that use a **system on a chip** (**SoC**) or similar technology that is built to be used as a View Client
- Access using a HTML5-compliant browser, either through the VMware Horizon Suite or a View Connection Server that has Feature Pack 1 installed

The traditional software client is the preferred method for organizations who wish to re-use their existing desktops or laptops as View Clients. Assuming these machines meet the required specifications, all that is required is to install the software client within the existing operating system.

Thin clients and zero clients are used in place of traditional desktops or laptops, greatly reducing endpoint management and maintenance while still providing a high-quality user experience.

VMware Horizon View Feature Pack 1 is a free add-on that extends the capabilities of the View Connection Server to support access to desktops over a supported HTML5-compliant browser. Information about Feature Pack 1 can be found in *Chapter 15, VMware Horizon View Feature Pack 1*.

The VMware Horizon Suite is a separate VMware product that enables browser-based access to View desktops, applications packaged with ThinApp, and user data. No client software is required when accessing View desktops through the Horizon portal; only a HTML5-compliant browser is required.

Additional information about each of the View Client types will be discussed in this section.

Software clients

VMware provides software-based clients for a number of different operating system platforms. The following is a list of all the operating systems currently supported by the latest View software-based clients:

- Microsoft Windows
    - Windows 8 RT, 32-bit or 64-bit
    - Windows 7 32-bit or 64-bit (None or SP1)
    - Windows Vista 32-bit (SP1 or SP2)
    - Windows XP 32-bit (SP3)

- Apple OS X
  - Mountain Lion (10.8)
  - Lion (10.7)
  - Snow Leopard (10.6.8)
- Apple iOS
  - iOS 4.2 and later
  - Devices including iPhone 4 or later, and all iPad models
- Linux
  - 32-bit Ubuntu Linux (10.04 or 12.04)
- Android
  - Android 4.1 or 4.2 (Jelly Bean)
  - Android 4 (Ice Cream Sandwich)
  - Android 3 (Honeycomb)

Each of the View Clients has specific hardware requirements. For an up-to-date listing of supported operating system versions and their hardware requirements, consult the View Clients Documentation page (`http://www.vmware.com/support/viewclients/doc/viewclients_pubs.html`) as well as the *VMware Compatibility Guide* (`http://www.vmware.com/resources/compatibility/search.php%3FdeviceCategory=vdm`).

The VMware View with Local Mode client is supported only on the Windows OS. Consult the VMware document *Using VMware View Client for Windows* (`http://www.vmware.com/support/viewclients/doc/viewclients_pubs.html`) for additional host computer requirements.

Thin clients

A thin client is similar to a traditional desktop or laptop, in that it runs a traditional OS and has a limited amount of storage. Thin clients typically use a minimal or customized build of the selected OS, which can result in some amount of maintenance needs but less than that of the full version of the same OS. Linux and Windows Embedded are two common OSs that can be found in thin clients. Those thin clients that use Linux or Windows Embedded typically support some of the core features of the OS, which for Windows Embedded includes the ability to join an AD domain.

Thin clients typically contain minimal storage, which is usually flash-based. This helps reduce the power and cooling needs of a thin client compared to a traditional full desktop or laptop. If you intend to use the View group policy templates to enable client caching, described in the *View Client Configuration ADM Template* section of *Chapter 13*, *Implementing VMware Horizon View Group Policies*, you must verify that the thin client has sufficient available storage based on the settings that you choose.

Thin clients that are available are serial, parallel, audio, microphone, and USB ports, which are useful if an organization needs to support locally attached devices. This is in addition to other common features such as wireless networking, support for **Power over Ethernet** (**PoE**), and support for multiple displays. A number of thin client vendors offer tools that can be used to manage their devices, which are useful when an organization needs to perform maintenance on a large number of thin clients. Before choosing a thin client that is appropriate for your environment, it is important to verify that it supports all the features that you need such as the PCoIP protocol and smart card readers.

Thin clients retain the ability to use some locally installed applications, such as a web browser. While View can be configured to deliver a superior user experience under almost any type of workload, there may be scenarios where an organization prefers that users access a web service using the locally installed browser. One example of this is streamed media; a properly sized View environment will have no problem delivering streamed media but an organization may prefer that a user employ the local browser rather than viewing the streamed media through the View Client.

If your environment uses a Unified Communications platform, only certain thin client hardware or software (OS) platforms may be supported. Please consult your Unified Communications vendor for compatibility guidelines.

Existing desktops or laptops can also be converted into thin clients using a software package such as VDI Blaster by Devon IT. These software packages install a highly optimized version of an OS as well as the View Client software, creating a thin client that often requires only a few GB of local storage. These software packages are ideal for organizations that wish to repurpose their existing hardware while also eliminating the need to maintain a full host OS.

Thin clients are offered from a number of different companies, including those that also offer traditional desktops and laptops. Some of the larger thin client suppliers include:

- Chip PC
- Dell Wyse
- Fujitsu
- HP
- Lenovo

Zero clients

A zero client shares many of the same features as a thin client but is built to be used only as a client for VMware View or competing virtual desktop solutions. While a thin client uses a customized or embedded OS that is installed on local storage, a zero client uses SoC or **field-programmable gate array** (**FPGA**) technology as the underlying operating environment. The software for a zero client is typically stored as a firmware image, which is updated solely through the updates provided by the thin client manufacturer.

The manufacturers that offer thin clients also offer zero clients as well as similar management utilities. These management utilities are even more important for zero clients, as there is no underlying OS to manage as there is with the thin client. As with the thin clients, these management platforms are useful for organizations that need to maintain large numbers of devices. While zero clients do not require traditional OS updates, it is likely that the View Client will need updates from time to time to support protocol enhancements and bug fixes.

Like thin clients, zero clients that are available are serial, parallel, audio, microphone, USB ports, wireless networking, support for Power over Ethernet (PoE), and support for multiple displays.

The list of zero client manufacturers includes all of those listed in the thin clients section as well as the following additional ones:

- LG
- Samsung
- Sun Microsystems
- Teradici

HTML5 clients

VMware Horizon View Feature Pack 1 enables access to View desktops over a supported HTML5-compliant browser. *Chapter 15*, *VMware Horizon View Feature Pack 1* details what is required to install and enable HTML client access, which includes installing additional software on the View Connection Servers and the virtual desktop master image.

VMware Horizon Suite enables access to VMware View desktops using only a HTML5-compliant browser; no View Client software is required. Horizon is a fully separate product that requires its own infrastructure, software licenses, and design considerations. Go to `vmware.com` for more information about the VMware Horizon Suite.

Choosing your View Client

The decision on which client to use is likely to differ based on a number of different factors. Each client type has distinct advantages but those advantages are secondary compared to the needs of the end user. This section will consider some common scenarios that are likely to impact which solution you use for your View Clients.

Why software clients?

One of the main reasons why organizations choose to use the View Clients for existing OS's is that they likely already own hardware and an OS capable of running the client software. Based on the current supported hardware and OS requirements, it is possible to run the View Client on a Windows XP computer with an 800 MHZ processor, both of which are over 10 years old. By comparison, the average thin or zero clients cost several hundred dollars or more, which can dramatically increase the capital costs of a View implementation project.

Traditional desktop and laptop clients offer the maximum flexibility compared to thin or zero clients as they can always be used as a standalone device. Zero clients can only be used as a client and, while thin clients may offer locally installed applications, they are still used primarily as a client as well.

One additional consideration is that the View Software Client supports client-side caching, which preserves the appearance of the View Client window even when packet loss occurs. While some zero clients do offer this feature, not all do.

Why thin or zero clients?

Thin and zero clients offer a number of basic advantages over using existing desktops or laptops as View Clients. Their cost may be prohibitive for organizations who already own hardware that is capable of acting as a View Client, but it is important to be familiar with their advantages when you decide to implement View. The following are the common advantages that thin or zero clients have over using traditional desktop or laptop computers that are running a full version of an OS:

- **Less power usage**: Thin and zero clients use a fraction of the power of traditional desktop hardware and generate very little heat, which results in less power usage over time; often, less hardware maintenance is required as the devices usually contain no mechanical hard drives or cooling fans.
- **Ability to use PoE**: PoE-capable thin or zero clients can be installed without having to worry about existing power outlet availability.
- **Zero clients have no resident software**: This virtually eliminates the likelihood that they would be vulnerable to traditional security threats.
- **Thin clients run a more limited OS**: This still has some security requirements but in many cases they are more locked down than a traditional desktop or laptop computer.
- **Zero clients have no local storage**: This eliminates common data security concerns. While thin clients have a small amount of local storage, it is a fraction of that of a traditional desktop or laptop computer.
- Compared to traditional desktops and laptops, thin and zero clients are much simpler to manage throughout their lifecycle.
- When purchasing new clients, thin and zero clients will generally be less expensive than traditional desktops and laptops.
- A zero client provides a simplified interface that streamlines the View Client login process, and ensures that users must log in to View to perform any tasks since the client hosts no applications.

Thin and zero clients are very compelling choices for organizations who intend to fully embrace the benefits of virtual desktops. Rather than attempting to continue to manage the legacy computer that sits on every desktop, organizations can deploy a purpose-built device that requires virtually no maintenance and uses a fraction of the power, while still offering a superior user experience.

Installing the View Client for Windows

The View Client can be installed using multiple methods; for this section we will perform the installation using the interactive installer. In the *View Client Command-line options* section, we will discuss how to perform the installation silently using various command-line options:

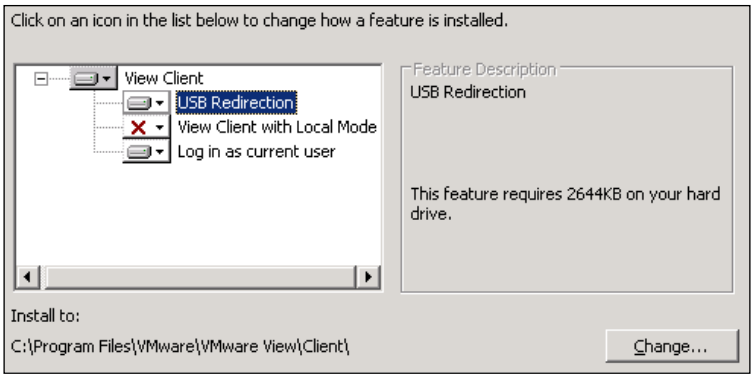
The View Client for Windows is provided in four different packages:

- 32-bit installer
- 64-bit installer
- 32-bit with Local Mode installer
- 64-bit with Local Mode installer

The installers that include the Local Mode option include the Type 2 Hypervisor software that is used to host the virtual desktop running in Local Mode. The Local Mode feature can be deselected during the installation of the client, which effectively leaves you with just the traditional View Client software installed.

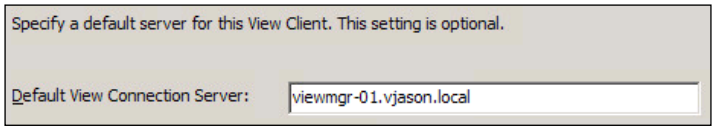
This section will show the installation of the View Client using the installer that includes the Local Mode option, although we will deselect the Local Mode option during installation. The following steps outline the installation process, and explain the options that are presented:

1. Double-click on the View Client installer file to launch the installation process. In our example, the file is named similar to `VMware-viewclientwithlocalmode-x.x.x-yyyyyy`.
2. In the **Welcome to the Installation Wizard for VMware View Client** screen, click on **Next**.
3. Review the license agreement, select the **I accept the terms in the license agreement** radio button, and click on **Next**.
4. In the **Custom Setup** screen (shown in the following screenshot), deselect any options that are not required, change the installation directory if needed, and click on **Next**.

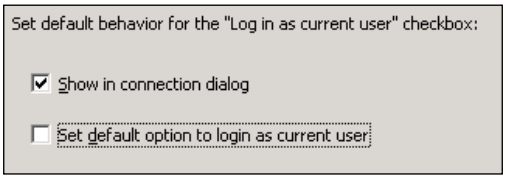


USB redirection is an optional feature that separates View Client USB traffic from the RDP or PCoIP connection to TCP port 32111, which can provide better mouse, keyboard, and screen performance. **Log in as current user** enables single sign on to View using the credentials of the currently logged in user. Both of these features can be disabled using group policies, command-line options, or at runtime within the View Client GUI.

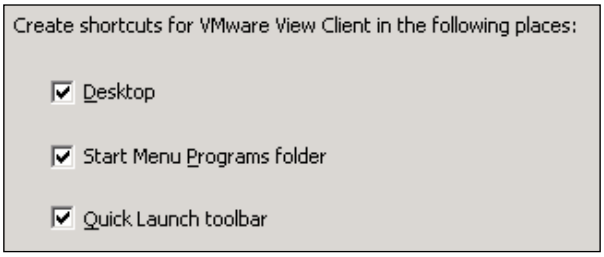
5. In the **Default Server** screen (shown in the following screenshot), specify the **Default View Connection Server** that the client will use and click on **Next**. This setting is optional and can be changed later.



6. In the **Enhanced Single Sign On** screen (shown in the following screenshot), specify the default behavior for the single sign on option and click on **Next**.



7. In the **Configure Shortcuts** screen (shown in the following screenshot), deselect the checkbox next to any shortcuts that are not needed and click on **Next**.



8. In the **Ready to Install the Program** window, verify the installation folder and click on **Install** to begin the client installation process.

In most cases, the View Client installer will prompt for a system reboot when the installation process completes. It is recommended you perform this reboot before trying to use the View Client.

The View Client is now available for use, and can be launched using the shortcuts created during the installation process. In the next section, we will use the client to connect to one of our desktops.

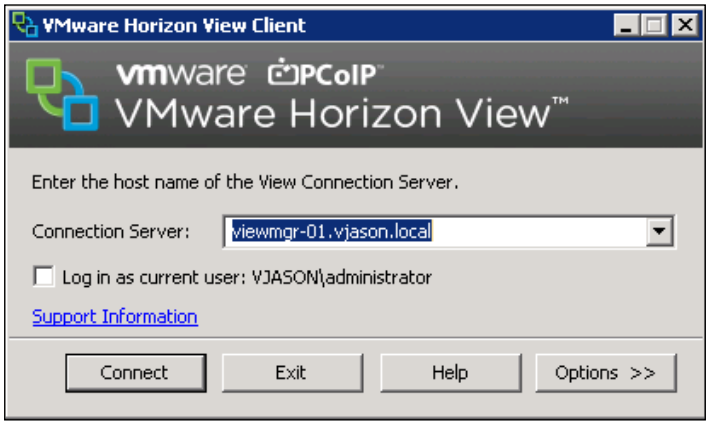
Many of the View Client options can be configured using the included AD Group Policy templates. These templates and their usage will be discussed in detail in *Chapter 13*, *Implementing VMware Horizon View Group Policies*.

Using the View Client

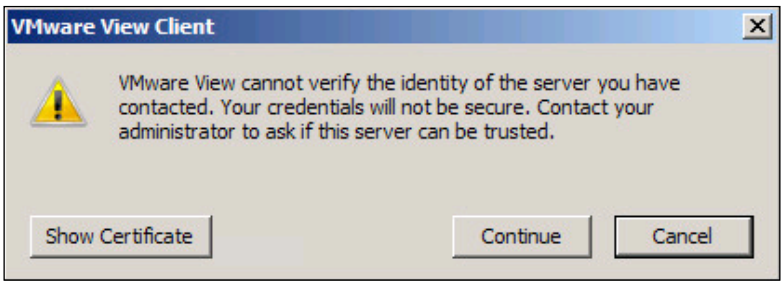
This section will show how to use the View Client to connect to a virtual desktop. In our example, we will be connecting to the View Connection Server named `VIEWMGR-01.vjason.local`, and selecting a desktop from a pool named `LC-Pool`. The following steps detail the login process from start to finish:

1. Launch the View Client using any of the shortcuts that were created during the installation process.

2. In the **VMware Horizon View Client** window (shown in the following screenshot), supply the **Connection Server** name if it is not present, and click on **Connect**.

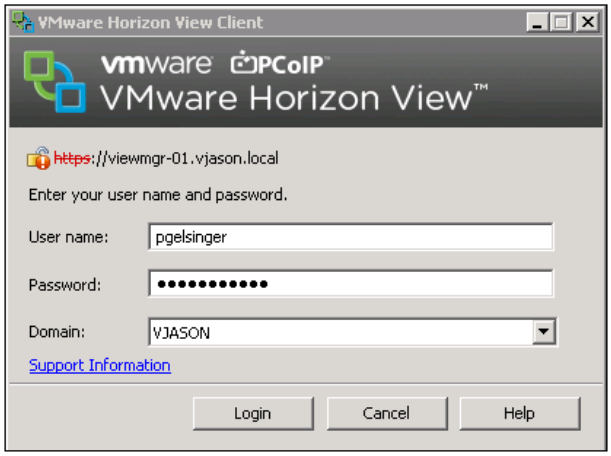


3. If the View Connection server is still using the default self-signed certificate, or the View Client has not been configured to bypass certificate verification, you will be presented with a warning that the client cannot verify the identity of the server, as shown in the following screenshot. For now, click on **Continue**.

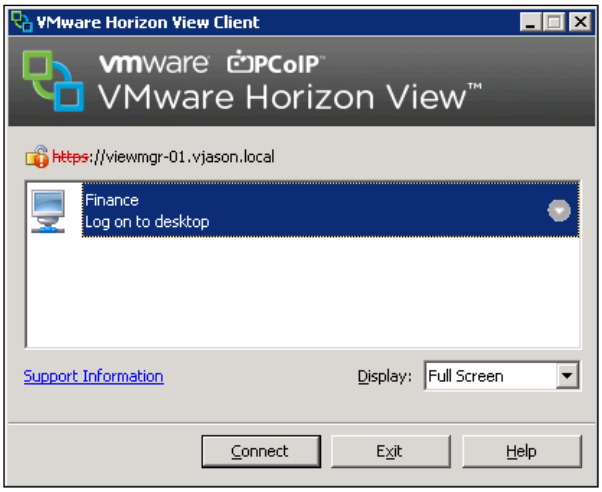


The ideal method to address this issue is to replace the default SSL certificates with ones that will be trusted by your View Clients. This process is discussed in *Chapter 12*, *Managing View SSL Certificates*. You may also configure the View Client to bypass certificate verification using the client options or AD group policies; however, this method is not recommended as it makes the View Clients less secure.

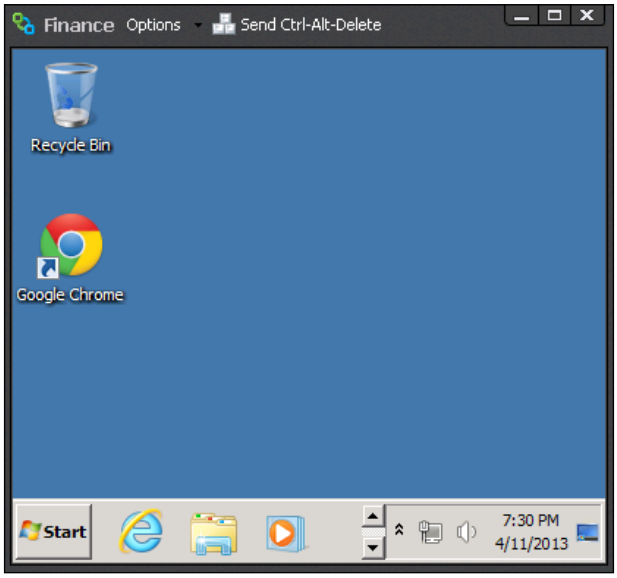
4. Provide the **User name** and **Password** for a user that has been entitled for access to the desktop pool, select the **Domain** if required, and click on **Login** as shown in the following screenshot:



5. Select a desktop pool from the list as shown in the following screenshot, and click on **Connect**. As we are using automatic assignment with our desktop pool, the user will be assigned a desktop from the pool upon the first login.



6. The View Client will briefly display a message informing the user that the desktop is being prepared for use, after which the login will proceed and the user will be logged in to the desktop. The following screenshot shows a View Client window after the View Client has completed the login process. The client window has been reduced in size so that it can be displayed more easily within the book.



The user has completed the login process and may now begin to use the desktop. Once the user is finished with the session, he/she can simply log off as they would with a traditional desktop.

View Client command-line options

VMware has made available a number of command-line options that make it possible for organizations to automate both the installation and operation of the View Client software. This section will provide information about these options and show some common examples of how they are used.

Command-line options available during installation

The View Client for Windows is provided in a **Microsoft Installer** (**MSI**) format that supports a silent installation. The installer also supports custom configuration by allowing multiple configuration options to be specified at the time of installation. The following table provides a list of the different configuration options that can be configured using the command line at the time of installation.

Installation option	Description	Default value
`INSTALLDIR`	Software installation directory	`%ProgramFiles%\VMware\VMware View\Client`
`VDM_SERVER`	FQDN or IP address of the preferred View Connection Server	None
`DESKTOP_SHORTCUT`	Installs a shortcut for the client on the desktop	`1` (install the shortcut). Specify `0` if no shortcut is desired.
`QUICKLAUNCH_SHORTCUT`	Installs a shortcut for the client in the quick-launch tray	`1` (installs the shortcut). Specify `0` if no shortcut is desired.
`STARTMENU_SHORTCUT`	Installs a shortcut for the client in the Start menu	`1` (installs the shortcut). Specify `0` if no shortcut is desired.
`REBOOT="Really Suppress"`	Suppresses a reboot of the View Client computer after the installation completes	The system will reboot if required after the installation completes.
`ADDLOCAL=ALL`	Installs all features of the View Client with Local Mode	During an interactive installation, all features are installed by default.
`ADDLOCAL=CORE`	Installs the core features of the View Client; required when selecting the MVDI, ThinPrint, TSSO, and USB options	During an interactive installation, all features are installed by default.
`ADDLOCAL=MVDI`	Installs the Local Mode option when using the View Client with Local Mode installer	During an interactive installation of the View Client with Local Mode, this feature is installed by default.

Installation option	Description	Default value
`ADDLOCAL=ThinPrint`	Installs the ThinPrint option of the View Client, enabling printers to be presented to the virtual desktop without requiring the associated drivers	During an interactive installation, this feature is installed by default.
`ADDLOCAL=TSSO`	Installs the **Single Sign On** (**SSO**) feature of the View Client	During an interactive installation, this feature is installed by default.
`ADDLOCAL=USB`	Installs the USB redirection feature of the View Client	During an interactive installation, this feature is installed by default.

The following command-line options are also available and are used to further customize how the installation proceeds:

- `/a`: Performs an administrative installation
- `/a` <full path to an existing administrative install>: Patches an existing administrative installation
- `/s`: Hides the initialization dialog (For silent mode use `/s /v /qn`)
- `/v`: Indicates parameters to pass to the installer
- `/c`: Cleans out installation registration information
- `/l`: Performs detailed logging
- `/l` <full path to a log file>: Performs detailed logging

When installing the View Client with Local Mode, you will need the View Client installer that includes that feature. The installer is easily identified as it contains the text `withlocalmode` in the filename. A sample filename for the View Client with Local Mode is `VMware-viewclientwithlocalmode-x.x.x-yyyyyy.exe`. The feature is installed by default when using the Local Mode installation package but it can be excluded if desired.

Some sample command-line installations are shown in this section. Elevated privileges are required if you are installing the client within the context of the currently logged in Windows user account.

```
VMware-viewclient-x86_64-x.x.x-yyyyyy.exe /S /V /qn DESKTOP_SHORTCUT=0
VDM_SERVER=VIEWMGR-01.vjason.local INSTALLDIR="C:\Program Files\VMware\
View\Client" /l C:\Logs\VMware_Client.log
```

The previous command will silently install the View Client to the specified directory, avoid creating a desktop shortcut for the client, default to the specified View Connection Server, and save a detailed log file of the installation process to the `C:\Logs` folder.

```
VMware-viewclient-x86_64-x.x.x-yyyyyy.exe /S /V /qn VDM_
SERVER=VIEWMGR-01.vjason.local ADDLOCAL=All
```

The previous command will silently install the View Client with Local Mode and default to the specified View Connection Server.

Using the information provided in this chapter, View administrators will be able to automate as much of the View Client installation process as they want, which is helpful when you need to deploy a large number of View Clients.

View administrators can also use ThinApp to package the View Client, which may be useful in some client deployment scenarios where end users have very limited privileges. VMware does not provide official guidelines on how to use ThinApp to package the View Client, but VMware employees have provided their own instructions on the official VMware blog site. The `blogs.vmware.com` article titled *Virtualizing the VMware View Client using ThinApp* (`http://blogs.vmware.com/thinapp/2012/09/virtualizing-the-vmware-view-client-using-thinapp.html`) provides detailed instructions on how to use ThinApp to capture and package the View Client using ThinApp.

Command-line options for launching the View Client

The View Client software allows all client options to be specified at the command line, enabling organizations to maintain full control over the client login process. The following table provides a list of the parameters that can be specified when launching the View Client either from a command line or when using a preconfigured script. The View Client executable is named `wswc.exe`, and is located in the directory specified during the client installation process.

Option	Description
`-checkin`	This is applicable to Local Mode View desktops only. This is used to check in the specified desktop and subsequently unlock the version hosted by vSphere in the datacenter. You must also specify the `-desktopName` option.

Option	Description
`-checkout`	This is applicable to Local Mode View desktops only. This is used to check out the specified desktop and then lock the version hosted by vSphere in the datacenter.
`-confirmRollback`	This is applicable to Local Mode View desktops only. This is used to suppress the confirmation dialog box that appears when using the `-rollback` option. To perform a rollback in non-interactive mode, you must also specify `-nonInteractive`.
`-connectUSBOn Startup`	When this is set to `true`, redirects all USB devices to the View desktop that is currently connected to the host. This option is `false` by default, and is set automatically if you specify `-unattended`.
`-connectUSB OnInsert`	When this is set to `true`, connects a USB device to the View desktop when you plug in the device. This option is `false` by default, and is set automatically if you specify `–unattended`.
`-desktopLayout window_size`	This is used to specify the View desktop client window size. The available options include `fullscreen`, `multimonitor`, `windowLarge`, and `windowSmall`.
`-desktopName desktop_name`	This is used to specify the name of the desktop. Use the same value that would appear in the View Client's **Select Desktop** dialog box. In many desktop pool configurations, this will be the desktop pool display name.
`-desktopProtocol protocol`	This is used to specify the desktop protocol to be used. The protocol can be PCoIP or RDP but may be limited based on the desktop pool configuration.
`-domainName domain_name`	This is used to specify the AD domain that the user will use to log in to View Client.
`-file file_path`	This is used to specify the path to and filename of a configuration file that contains the View Client command-line options. This text file should contain one command-line option per line.
`-languageId Locale-ID`	This is used to provide localization support for different languages in View Client. If the required language resource library is available, specify the **Locale ID** (**LCID**) to be used. For US English, the LCID is `0x409`.
`-localDirectory directory_path`	This is applicable to Local Mode View desktops only. This is used to specify which directory on the local system to use for storing the Local Mode desktop. By contrast, if the local directory is selected in View Client, a subfolder with the desktop name is created under the selected directory and local files are stored in that subfolder. This option requires that you also specify `–desktopName`.

Option	Description
-logInAsCurrent User	When this is set to true, the View Client AD user credential information is used to log in to the View Connection Server and the View desktop. The default value is false. This feature requires the associated option be installed with the View Client.
-nonInteractive	This is used to suppress error message boxes when starting View Client from a script. This option is set by default if you specify –unattended.
-password password	This is used to specify the password that will be used to log in to the View Client.
-printEnvironment Info	This is used to display the IP address, MAC address, and machine name of the View Client device.
-rollback	Applicable to Local Mode View desktops only. This is used to unlock the vSphere hosted version of a checked-out View desktop and then discard the local session. This option requires that you also specify –desktopName.
-serverURL connection_server	This is used to specify the URL, IP address, or FQDN of the View Connection Server used to log in.
-smartCardPIN PIN	This is used to specify the pin code when using a smart card to log in to View.
-unattended	This is used to start View Client software in a non-interactive mode that is typically used for View clients configured in kiosk mode. The kiosk mode clients are discussed further in the *VMware Horizon View Administration Guide* (http://www.vmware.com/support/pubs/view_pubs.html).
-userName user_ name	This is used to specify the AD account name that the end user will use when logging in to View Client.

A sample login using command-line options is done as follows:

```
wswc.exe -serverURL VIEWMGR-01.vjason.local –username jventresco
-password Password! -domainname vjason.local -desktopname LC-Pool
-logInAsCurrentUser False -connectUSBOnStartup False
```

This command will log my AD user account into the VIEWMGR-01.vjason.local View Connection Server, select a desktop from the LC-Pool desktop pool, and use the password provided. The command shown also disables the **Log in as Current User** and **Connect USB on Startup** features.

Summary

In this chapter, we learned about different VMware Horizon View Client options. We discussed the different methods that organizations can use to connect to View desktops, and discussed why an organization may choose one option over another.

We went through a basic installation of the View Client software for Windows and provided examples of how to install that client silently using command-line options. We ended by discussing the different command-line options that can be used when launching the View Client, enabling full control over the client login session.

In the next chapter, we will discuss how to perform maintenance on virtual desktops, which often requires different techniques from those you would use with traditional desktops.

10
Performing View Desktop Maintenance

Maintaining desktops deployed using VMware Horizon View requires a different approach depending on what desktop type you have selected. Full clone desktops are typically managed using the same techniques as traditional physical desktops, as each is a fully independent virtual machine with dedicated underlying virtual hard disks. Many organizations choose full clone desktops for this reason as they can continue to manage them using tools and procedures that are already in place.

Linked-clone desktops are an entirely different matter, especially if you wish to minimize the amount of per-desktop storage that is required. If an organization were to manage its linked-clone desktops using the same traditional techniques used with physical or full clone desktops, they would find that over time those desktops used more and more storage space, negating the benefits of using linked-clone desktops.

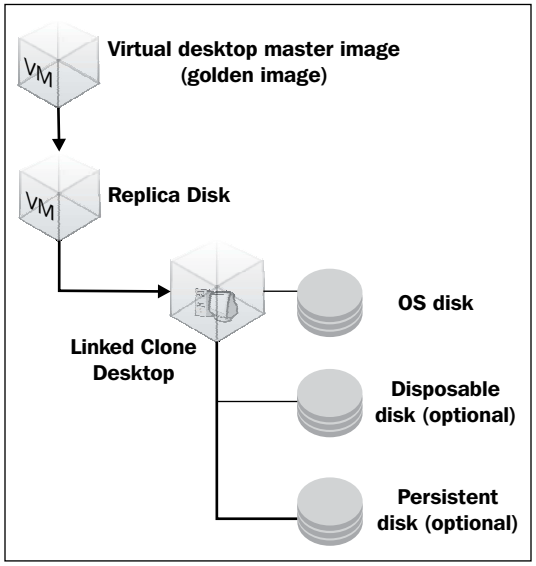
This chapter will focus primarily on managing linked-clone desktops using the various features of View itself.

In this chapter, we will learn:

- An overview of linked-clone maintenance
- How to recompose a linked-clone desktop pool
- How to refresh a linked-clone desktop pool
- How to rebalance a linked-clone desktop pool
- How to manage View Composer persistent disks

An overview of linked-clone maintenance

To understand why a linked-clone desktop requires different maintenance techniques than a physical or full clone desktop, we must again understand what makes it different. The following diagram shows the relationship between the linked-clone disk and the shared replica disk.



The replica disk is a read-only copy of the virtual desktop master image virtual hard disk; it is shared among as many as 2,000 desktops within a given View desktop pool. The linked-clone disk is used by the virtual desktop when it needs to write data; one virtual desktop is created for each linked-clone desktop.

One of the primary advantages of linked-clone desktops is that they require far less storage space than full-clone desktops; this is made possible by the shared replica disk. This reduced storage utilization is certainly useful at the time the desktops are deployed, but to maintain this advantage over time, you must use the View native recompose or refresh features.

Each of the maintenance operations described in this section requires the desktop to be powered off. Due to this, as well as the storage IO associated with each operation, it is recommended you perform these tasks during off-peak hours. Each of these tasks can be scheduled, making it easier for View administrators to accomplish this.

Desktop refresh

A desktop refresh returns the desktop's linked-clone disk, also known as the OS disk, to the original state and size as configured in the desktop pool options. If the desktop is configured with an optional persistent disk for storing user profile data, that data will be retained during the refresh operation. A refresh can be performed on either a desktop pool or an individual desktop.

A desktop refresh provides several benefits:

- A quick way of reducing linked-clone storage utilization
- If a desktop develops software problems, a refresh can be used to restore it to the original state

A refresh operation is also performed during linked-clone recompose and rebalance operations. A refresh operation typically requires fewer than 10 minutes of downtime per desktop, although the time required may vary depending on the performance capabilities of the View infrastructure and the specified number of concurrent refresh operations allowed. View Composer performs up to 12 concurrent refresh operations at once by default. Instructions on how to perform a desktop refresh operation are provided later in this chapter.

Any maintenance task that includes a refresh operation will force the Windows desktop to reactivate the OS and Office software if installed. Due to this, it is recommended that any organization that wishes to use linked-clone desktops deploy a Windows KMS server to handle Windows and Office license activation. Windows MAK keys would be quickly exhausted in a linked-clone desktop environment.

Desktop recompose

A desktop recompose is used to replace the existing linked-clone replica disk, usually in response to a configuration change, software installation, or software update. A desktop recompose is the preferred method of updating the linked-clone desktop as the changes only affect the replica disk. Were the same updates or changes to be applied directly to the linked-clone desktops themselves, each of the linked-clone OS disks would increase in size by the amount needed to process the change. The following example shows the difference between updating a virtual desktop master image and using a recompose to deploy an updated replica disk, versus installing the updates directly on the linked-clone desktops.

In this example, it is determined that installing the updates on a single desktop requires 215 MB of additional space:

- If the virtual desktop master image is updated, and a recompose operation is performed, only the 215 MB of additional space will be required to update all 2,000 desktops in the pool
- If the linked-clone desktops are patched individually, a pool of 2,000 linked-clone desktops would require an additional 430 GB of storage, or 215 MB for each linked-clone desktop

In addition to the additional storage required to install the patches directly on the linked-clone desktops, the patches or software installed would not persist if any maintenance that requires a refresh operation were performed.

A desktop recompose operation consists of the following steps:

1. The View administrator (or other responsible party) updates the virtual desktop master image with the required changes.
2. The View administrator takes a new snapshot of the updated virtual desktop master image.
3. The View administrator uses the View Manager Admin console to initiate a recompose, selecting the updated snapshot. A new master image with snapshot can also be selected, provided it is running on the same operating system.
4. View Composer clones the selected virtual desktop master image and snapshot to a new replica disk.
5. View Composer powers down the virtual desktop that will be recomposed.
6. View Composer returns the existing linked-clone OS disk to the original size and state, and associates it with the new replica disk. In addition, if the desktop has a persistent disk configured, it will be attached to the recomposed desktop at this point.
7. View Composer powers on the recomposed linked clone and configures it using the View Agent.
8. The View Agent informs the View Manager server that it is available for use.

A recompose operation typically requires fewer than 10 minutes of downtime per desktop, and View Composer performs up to 12 concurrent recompose operations at once by default. Instructions on how to perform a desktop recompose operation are provided later in this chapter.

A Local Mode desktop can also be recomposed, but the recompose operation will not occur until the desktop has been checked back in to View.

Desktop rebalance

A desktop rebalance is used to rebalance linked-clone desktop storage across existing datastores, including any new datastores that were added to the desktop pool configuration. As mentioned previously, a rebalance operation will also refresh the desktop as part of the process, so a rebalance cannot be used as a way of balancing the linked-clone OS disks.

A rebalance is typically most useful to balance persistent disk storage, as the persistent disk will remain in place until the desktop is deleted or the persistent disk is detached from the desktop and later deleted. Organizations that choose not to deploy a persistent disk may find that regular refresh or recompose operations are all that is required to maintain consistent desktop storage utilization throughout the lifecycle of their linked-clone desktops.

Managing View maintenance tasks

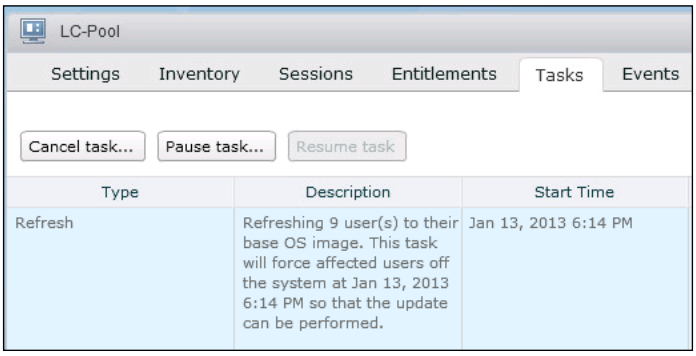
On-going or scheduled refresh, recompose, or rebalance tasks can be paused, resumed from a pause, or canceled at any time using the View Manager Admin console. When a maintenance task is canceled or paused, any operations currently underway will finish but no new operations will start. When a paused task is resumed, the maintenance operation will continue.

The resources required to perform View desktop maintenance tasks may impact the performance of the View infrastructure. If the View desktop maintenance is causing performance problems for other View desktops, or for other resources that share the View infrastructure, simply pause the maintenance task. Resume the maintenance task during a period of low infrastructure utilization.

The following steps outline how to manage a task assigned to a desktop pool or an individual desktop:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Go to **Inventory** | **Pools** within the console.

3. In the **Pools** window, click on the pool that has the task that you wish to cancel. In our example, we will click on the pool titled **LC-Pool** to open the **LC-Pool** window.
4. Under the **Tasks** tab of the **LC-Pool** window (shown in the following screenshot), highlight the task that you wish to update and click on **Cancel task...**, **Pause task...**, or **Resume task** as required. Since the sample task is not currently paused, the **Resume task** button is grayed out.



The **Tasks** tab will show tasks assigned to individual desktops as well as those assigned to the pool as a whole. The tasks are managed using the same process regardless of their assignment.

Global settings for View maintenance

There are multiple configuration options within View that affect View refresh, recompose, and rebalance operations. This section will explain where those options can be found, and their purposes.

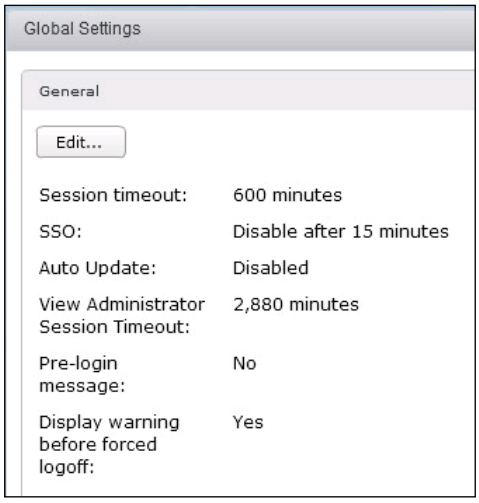
Logoff warning and timeout

When a View administrator chooses to forcibly log off the user to perform a linked-clone maintenance operation, the user is notified and the log off proceeds after five minutes. The notification message and the timeout value can both be configured in the **View Global Settings** window.

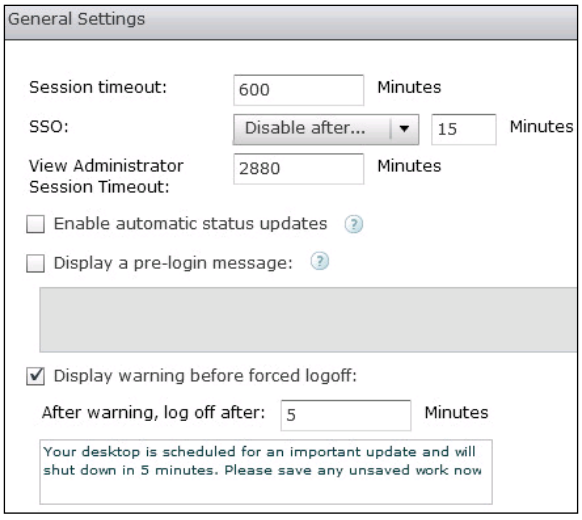
If you choose to automatically log off users to perform desktop maintenance, your warning message should instruct them not to log in again until maintenance is complete. This will help prevent users from immediately trying to reconnect to their desktops after they have been logged off, which can interfere with the maintenance process.

The following steps outline how to update these global settings:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Global Settings** window under **View Configuration** within the console. Click on the **Edit** button in the **General** section as shown in the following screenshot:



3. In the **General Settings** window, update the **Display warning before forced logoff** and **After warning, log off after** settings as needed. Click on **OK** to update the settings.

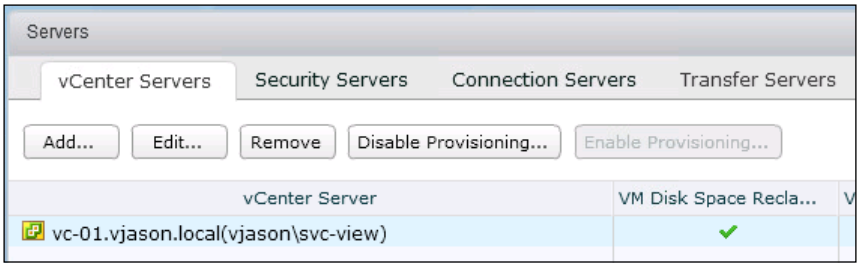


Concurrent maintenance operations

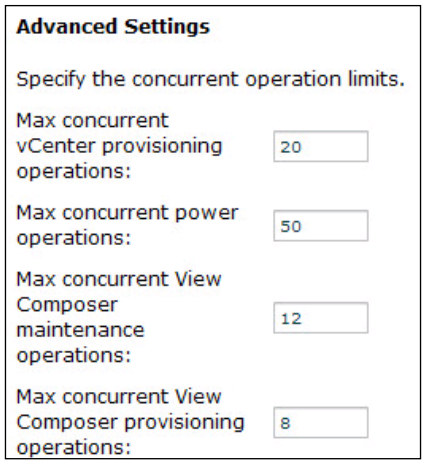
By default, View Composer will perform no more than 12 maintenance operations at one time. While this is considered the optimal setting for this option, it is possible to increase or decrease the number if required. This number is set on a per-vCenter Server basis, so if multiple vCenter Servers are being used, each one will need to be changed individually.

The following steps outline how to update the number of concurrent maintenance operations that View Composer will perform:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Servers** window under **View Configuration** within the console.
3. In the **vCenter Servers** tab of the **Servers** window (shown in the following screenshot), highlight the vCenter Server you wish to update and click on the **Edit** button.



4. In the **Edit vCenter Server** window, click on the **Edit** button underneath the **vCenter Server Settings** section.
5. In the second **Edit vCenter Server** window, under the **Advanced Settings** section (shown in the following screenshot), update the **Max concurrent View Composer maintenance operations** value as needed. Click on **OK** twice to close both **Edit vCenter Server** windows and update the settings.



The **Edit vCenter Server** window also allows you to change other settings that affect the speed at which View desktops are provisioned, deployed, and powered on. For each of these settings, the default value is considered optional and changes are not recommended.

Storage overcommit

Storage overcommit levels are configured on a per-datastore basis and affect how many linked clones View Composer will provision on each datastore. Storage overcommit is typically configured when the desktop pool is created but the settings can be updated at any time.

The following are the four different storage overcommit levels supported by View. Each is calculated based on the size of the parent virtual machine.

- **None**: Storage is not overcommitted
- **Conservative**: The default; storage will be overcommitted up to four times the size of the datastore
- **Moderate**: Storage will be overcommitted up to seven times the size of the datastore
- **Aggressive**: Storage will be overcommitted up to 15 times the size of the datastore

Consider an example where the overcommit level is set to Conservative, the parent virtual machine uses a disk that is 12 GB in size, and linked clones will be configured on datastores that are 240 GB in size.

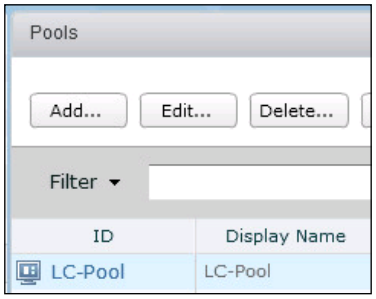
- 120 GB (datastore) X 4 (overcommit level) = 480 GB
- 480 GB / 12 GB (parent virtual machine size) = 40 linked clones

Based on these figures, when using the default storage overcommit level, View Composer will place up to 40 linked clones on each datastore at the time of the linked-clone deployment or rebalance operation.

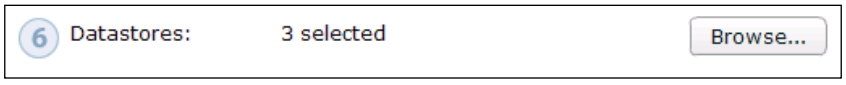
Updating datastore storage overcommit settings

The following steps outline how to update the storage overcommit levels of an existing desktop pool.

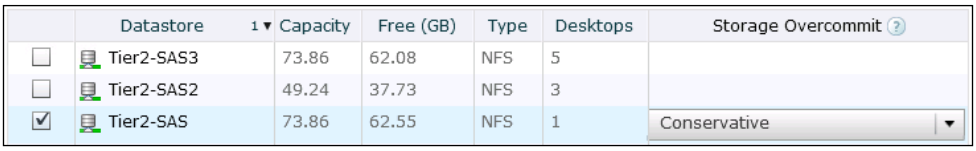
1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Pools** window under **Inventory** within the console.
3. In the **Pools** window, highlight the pool you wish to refresh. In our example we will highlight the pool titled LC-Pool. Click on the **Edit** button shown below to open the LC-Pool window.



4. In the **LC-Pool** window, click on the **vCenter Settings** tab.
5. In the **vCenter Settings** tab, click on the **Browse...** button next to the **Datastores** setting shown in the following screenshot. This will open the **Select Linked Clone Datastores** window.



6. In the **Select Linked Clone Datastores** window (shown in the following screenshot), open the **Storage Overcommit** drop-down menu next to each datastore to set the storage overcommit level. The level can only be changed for the datastores that are in use by the pool.



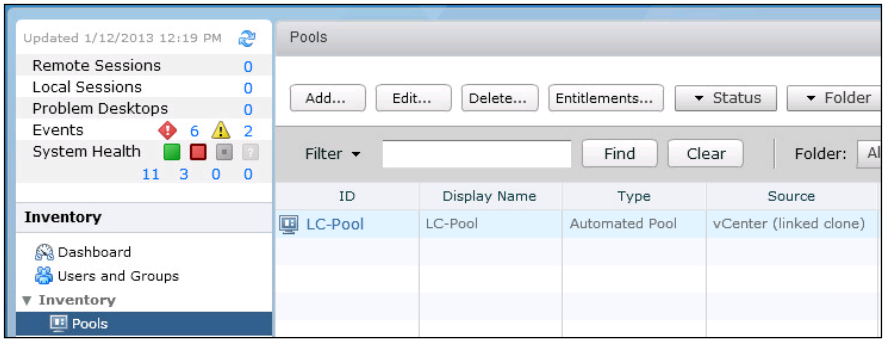
7. Click **OK** twice to close the **Select Linked Clone Datastores** and **Edit LC-Pool** windows and implement the changes.

Changing the storage overcommit settings does not, by itself, initiate any desktop maintenance activities. To enforce the updated storage overcommit policies on an existing desktop pool, simply perform a desktop rebalance using the procedure described later in this chapter.

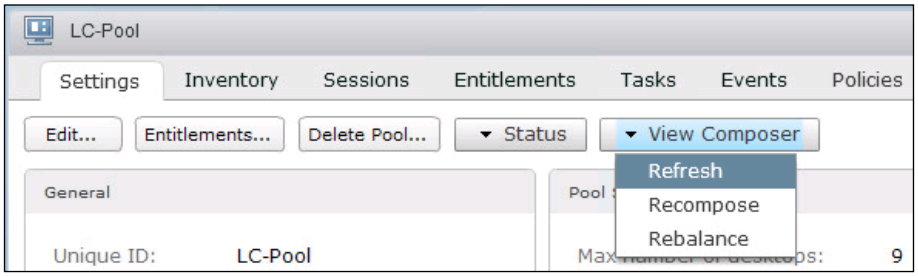
Refreshing linked-clone desktops

The following steps outline how to refresh a desktop pool using the View Manager Admin console.

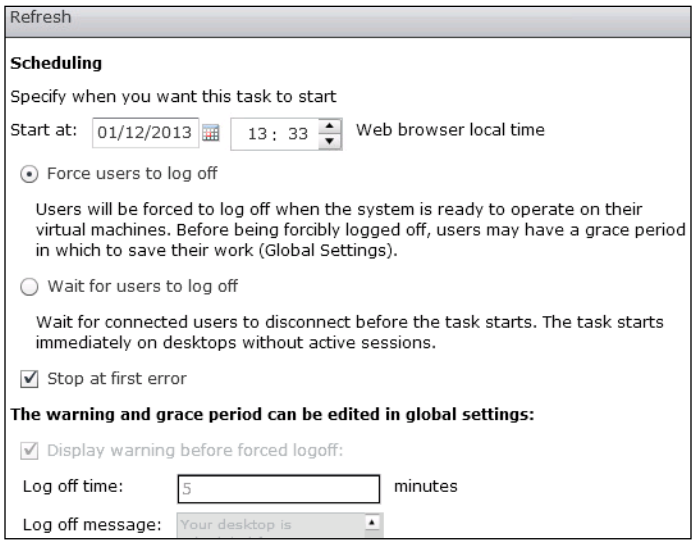
1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Pools** window under **Inventory** within the console.
3. In the **Pools** window (shown in the following screenshot), click on the pool you wish to refresh. In our example, we will click on the pool titled **LC-Pool** to open the **LC-Pool** window.



4. On the right side of the **LC-Pool** window (shown in the following screenshot), open the **View Composer** drop-down menu and click on **Refresh** to open the **Refresh** window.



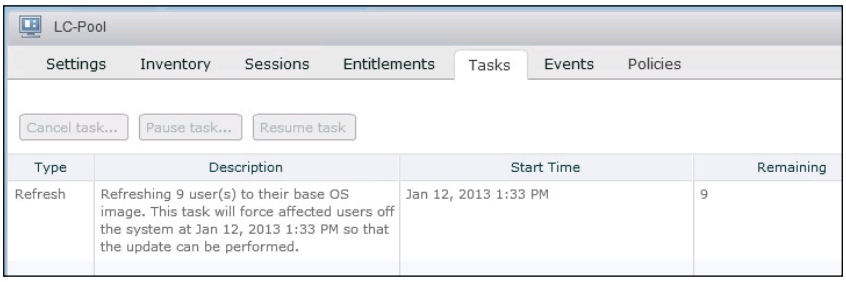
5. In the **Refresh - Scheduling** window (shown in the following screenshot), accept the default settings and click on **Next** to continue. If no changes are made, the refresh operation will begin immediately and users will be logged off from their desktops automatically after 5 minutes. The following are the optional settings:
    - The date and time the refresh should start
    - Whether or not to force the users to log off, or wait for them to log off
    - Whether or not to stop the refresh if an error occurs
    - Updates to the warning and grace period settings must be made in the View's global settings



6. Review the options in the **Refresh – Scheduling** window. If changes are required, click on the **Back** button to return to the previous screen. Click on **Finish** to begin or schedule the refresh operation, depending on what was configured in the previous step.

The time required to complete a desktop refresh operation varies based on a number of different factors beyond that of the View configuration itself. Generally speaking, under average circumstances it will take no more than 10 minutes per desktop starting from the time that View Composer performs the initial power down of the desktop.

The status of the refresh operation can be viewed in the **Tasks** tab of the desktop pool window as shown in the following screenshot:

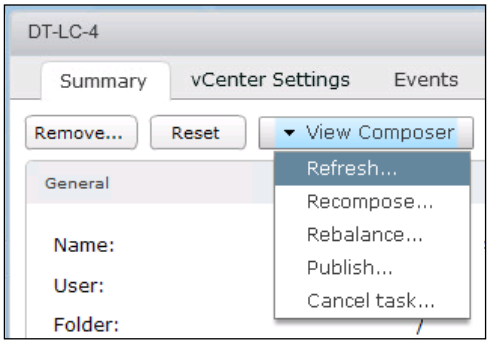


Refreshing individual desktops

A refresh can also be performed on an individual desktop. This is often done in response to an event, such as problems with the guest OS, that affects only the desktop that is to be refreshed. The following steps outline how to refresh a single desktop using the View Manager Admin console.

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Desktops** window under **Inventory** within the console.
3. In the **Desktops** window, click on the pool you wish to refresh. In our example, we will click on the desktop named **DT-LC-4** to open the **DT-LC-4** window.

4. In the **DT-LC-4** window, open the **View Composer** drop-down menu as shown in the following screenshot. Click on **Refresh** to open the **Refresh** window.

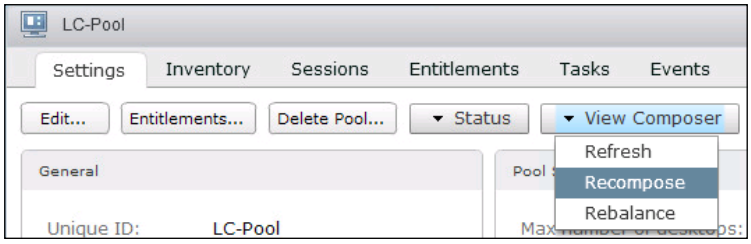


5. In the **Refresh** window, select the desired options and click on **Next**. These are the same options that appear when refreshing an entire desktop pool.
6. Click on **Finish** to begin or schedule the refresh operation, depending on what was configured in the previous step.

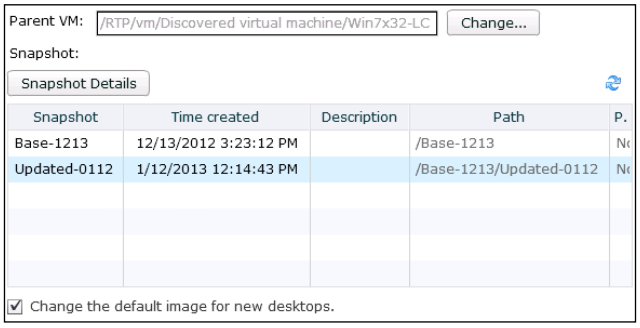
Recomposing linked-clone desktops

The following steps outline how to recompose a desktop pool using the View Manager Admin console:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Pools** window under **Inventory** within the console.
3. In the **Pools** window, click on the pool you wish to recompose. In our example, we will click on the pool titled **LC-Pool** to open the **LC-Pool** window.
4. On the right side of the **LC-Pool** window (shown in the following screenshot), open the **View Composer** drop-down menu and click on **Recompose** to open the **Recompose** window.



5. On the **Image** page of the **Recompose** window (shown in the following screenshot), highlight the updated snapshot that you wish to use with your desktops. You may also select a different parent VM and accompanying snapshot by clicking on the **Change...** button, as long as they use the same OS as the existing desktops. In our example, we have chosen a new snapshot of the existing parent VM. Click on **Next** to move to the next step.



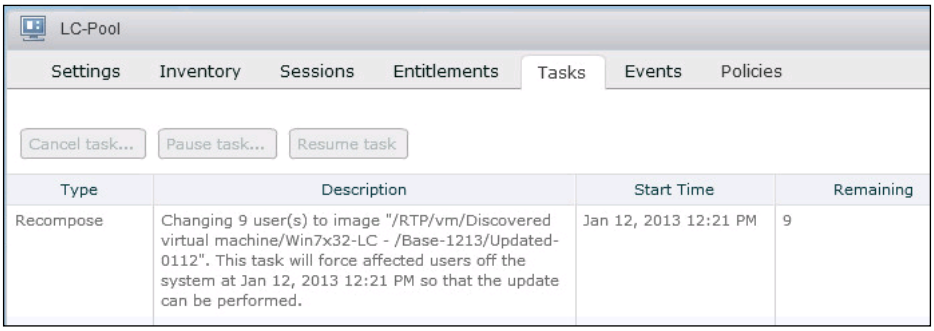
By default, View will use the selected snapshot when deploying new desktops within the desktop pool. Uncheck the **Change the default image for new desktops** checkbox to change this behavior and force new desktops to use the preceding snapshot.

6. On the **Scheduling** page of the **Recompose** window, select the desired scheduling options and click on **Next**. These are the same scheduling options that appear when performing a desktop refresh.
7. Review the options in the **Ready to Complete** page of the **Recompose** window. If changes are required, click on the **Back** button to return to the previous screen. Click on **Finish** to begin or schedule the recompose operation, depending on what was configured in the previous step.

The time required to complete a desktop recompose operation varies based on a number of different factors beyond that of the View configuration itself. Generally speaking, under average circumstances it will take no more than 15 minutes per desktop, starting from the time that View Composer performs the initial power down of the desktop.

If a desktop pool is configured to use Windows Sysprep for machine customization, a new Windows Machine **System Identifier** (**SID**) will be generated during a recompose operation. Consider any potential issues this may cause within your environment. The only alternative is to redeploy the desktops using VMware QuickPrep instead of Windows Sysprep.

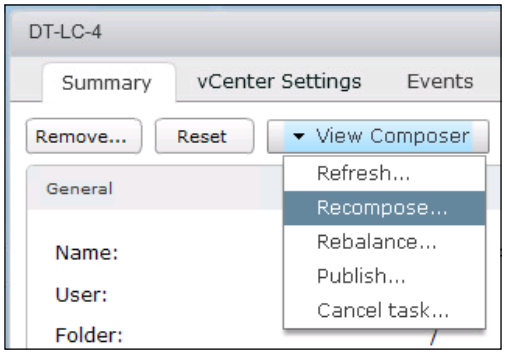
The status of the recompose operation can be viewed in the **Tasks** tab of the desktop pool window as shown in the following screenshot:



Recomposing individual desktops

A recompose can also be performed on an individual desktop. Reasons to do this might include the need to test out an updated desktop configuration on a small number of users prior to recomposing all the desktops. The following steps outline how to recompose a single desktop using the View Manager Admin console:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Desktops** window under **Inventory** within the console.
3. In the **Desktops** window, click on the desktop you wish to recompose. In our example, we will click on the desktop named **DT-LC-4** to open the **DT-LC-4** window.
4. In the **DT-LC-4** window, open the **View Composer** drop-down menu as shown in the following screenshot. Click on **Recompose** to open the **Recompose** window.

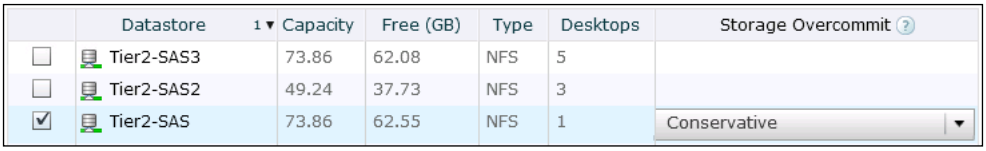


5. In the **Image** page of the **Recompose** window, select the desired options and click on **Next**. These are the same options that appear when performing a desktop pool recompose.
6. On the **Scheduling** page of the **Recompose** window, select the desired scheduling options and click on **Next**. These are the same scheduling options that appear when performing a desktop pool recompose.
7. Click **Finish** to begin or schedule the recompose operation, depending on what was configured in the previous step.

Rebalancing linked-clone desktops

The following steps outline how to rebalance a desktop pool using the View Manager Admin console. In this example, we will be adding additional datastores to our desktop pool prior to the rebalance operation. These datastores will then be used by the rebalance.

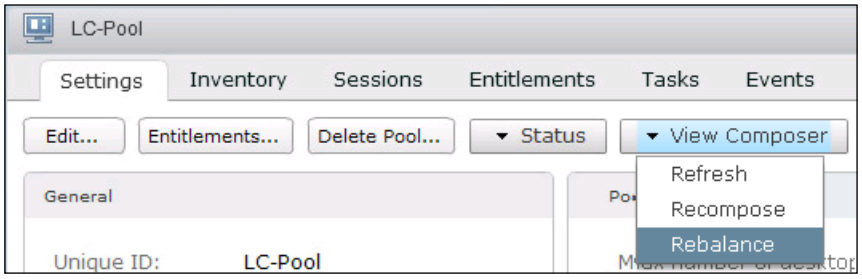
1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Pools** window under **Inventory** within the console.
3. In the **Pools** window, highlight the pool to which you wish to add the datastores and then rebalance. In our example, we will highlight the pool titled **LC-Pool**. Click on the **Edit** button to open the **Edit LC-Pool** window.
4. In the **Edit LC-Pool** window, click on the **vCenter Settings** tab.
5. In the **vCenter Settings** tab, click on the **Browse** button next to the **Datastores** setting. This will open the **Select Linked Clone Datastores** window.
6. In the **Select Linked Clone Datastores** window (shown in the following screenshot), click on the checkboxes next to the datastores you wish to add to the desktop pool. In our example, we will check the box next to both the **Tier2-SAS2** and **Tier2-SAS3** datastores.



	Datastore	Capacity	Free (GB)	Type	Desktops	Storage Overcommit
☐	Tier2-SAS3	73.86	62.08	NFS	5	
☐	Tier2-SAS2	49.24	37.73	NFS	3	
☑	Tier2-SAS	73.86	62.55	NFS	1	Conservative

7. Click on **OK** twice to close the **Select Linked Clone Datastores** and **Edit LC-Pool** windows, implement the changes, and return to the **LC-Pool** window.

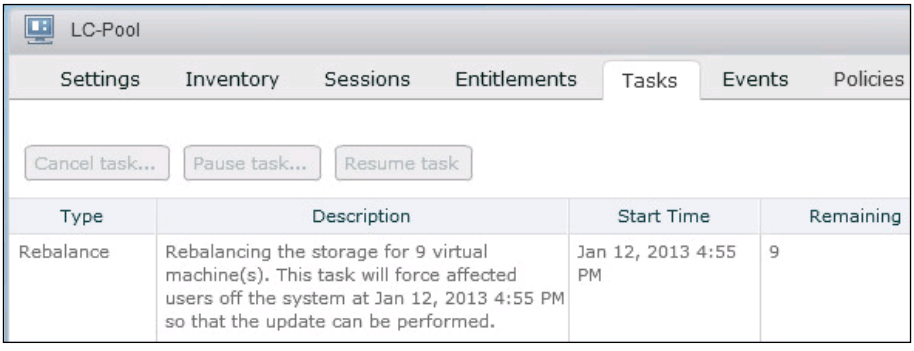
8. On the right side of the **LC-Pool** window (shown in the following screenshot), open the **View Composer** drop-down menu and click on **Rebalance** to open the **Rebalance** window.



9. On the **Rebalance** page of the **Rebalance** window, review the message and click on **Next**.
10. On the **Scheduling** page of the **Rebalance** window, select the desired scheduling options and click on **Next**. These are the same scheduling options that appear when performing a desktop refresh or recompose.
11. Review the options in the **Scheduling** page of the **Rebalance** window. If changes are required, click on the **Back** button to return to the previous screen. Click on **Finish** to begin or schedule the rebalance operation, depending on what was configured in the previous step.

As with other View maintenance operations, the time required to complete a desktop rebalance operation varies based on a number of different factors beyond that of the View configuration itself. Generally speaking, under average circumstances it will take no more than 15 minutes per desktop, starting from the time that View Composer performs the initial power down of the desktop.

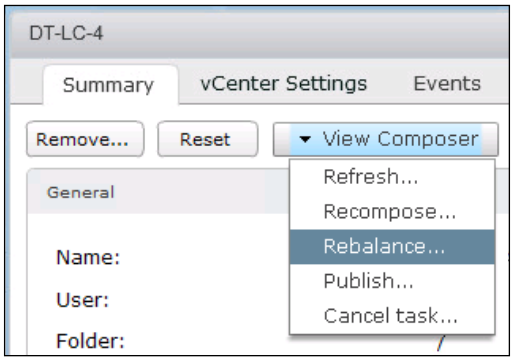
The status of the rebalance operation can be viewed in the **Tasks** tab of the desktop pool window as shown in the following screenshot:



Rebalancing individual desktops

A rebalance can also be performed on an individual desktop. This can be helpful in scenarios where only a small number of desktops need to be rebalanced, and not the entire desktop pool. The following steps outline how to rebalance a single desktop using the View Manager Admin console.

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Desktops** window under **Inventory** within the console.
3. In the **Desktops** window, click on the desktop you wish to rebalance. In our example, we will click on the desktop named **DT-LC-4** to open the **DT-LC-4** window.
4. In the **DT-LC-4** window, open the **View Composer** drop-down menu as shown in the following screenshot. Click on **Rebalance** to open the **Rebalance** window.



5. In the **Rebalance** window, review the message and click on **Next**.
6. In the **Scheduling** page of the **Rebalance** window, select the desired scheduling options and click on **Next**. These are the same scheduling options that appear when performing a desktop refresh or recompose.
7. Review the options in the **Scheduling** page of the **Rebalance** window. If changes are required, click on the **Back** button to return to the previous screen. Click on **Finish** to begin or schedule the rebalance operation, depending on what was configured in the previous step.

Managing View Composer persistent disks

View Composer persistent disks are used to store user profile data, and enable it to persist during the View Composer maintenance tasks described in this chapter. A linked clone is not required to have a persistent disk; features such as user profile folder redirection and View Persona management enable a linked-clone desktop to appear to be persistent, even if it lacks a persistent disk.

Organizations that rely on View Composer persistent disks to store critical user data should be familiar with how to manage them using native features of View. This section will provide examples of the different View maintenance operations that involve View Composer persistent disks.

Persistent disks will work only with the operating system version with which they were deployed. In the event that the original operating system is unavailable, and the data on the disk must be accessed, the persistent disks will need to be manually attached to a new virtual desktop and an assigned Windows drive letter. When attached this way, the persistent disks will simply appear as another hard drive.

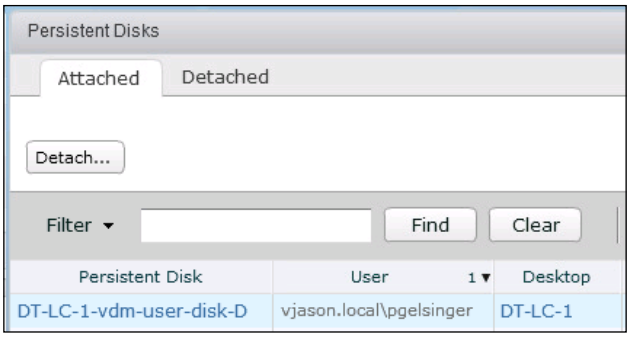
Detaching persistent disks

Detaching the persistent disk from a desktop allows it to remain managed by View while discarding the linked-clone files that are no longer required. If the persistent disk is needed again at a later date, a desktop can quickly be deployed and the persistent disk is associated with it.

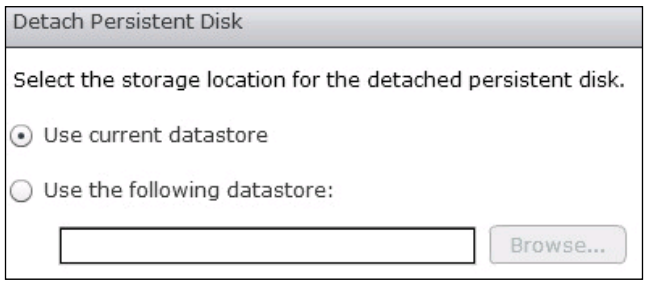
The following steps outline how to detach a persistent disk using the View Manager Admin console.

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Persistent Disks** window under **Inventory** within the console.

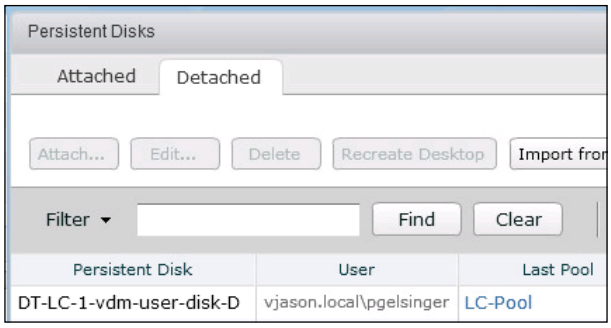
3. Highlight the persistent disk that you wish to detach. In the following example, we have highlighted the persistent disk associated with the desktop **DT-LC-1**, belonging to the user **vjason.local\pgelsinger**. Click on the **Detach** button to open the **Detach Persistent Disk** window.



4. In the **Detach Persistent Disk** window (shown in the following screenshot), select where to store the persistent disk. In this example, we will leave it on the current datastore, although organizations may choose to move the disk elsewhere for organizational or archival purposes. Click on **OK** to detach the disk.



The persistent disk will be detached from the linked clone it was associated with, the linked clone will be deleted, and a new unassigned one will be deployed in its place. The detached persistent disk can be found under the **Detached** tab in the **Persistent Disks** window under **Inventory**, as shown in the following screenshot:

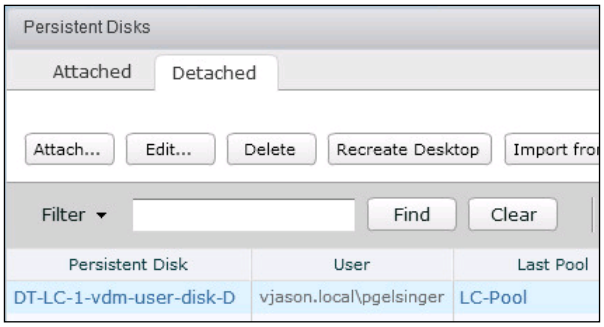


View maintains the information required to quickly recreate the linked-clone desktop, including the desktop pool and user it was assigned to.

Recreating a desktop using a persistent disk

The following steps outline how to recreate a linked clone desktop using a previously detached persistent disk:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Persistent Disks** window under **Inventory** within the console, and click on the **Detached** tab.
3. Highlight the persistent disk you wish to use, and click on the **Recreate Desktop** button as shown in the following screenshot:



4. In the **Recreate Desktop** window, review the information and click on **OK**.

Since View retained information about the desktop pool to which the persistent disk was previously assigned, no further information is required in order to recreate the desktop.

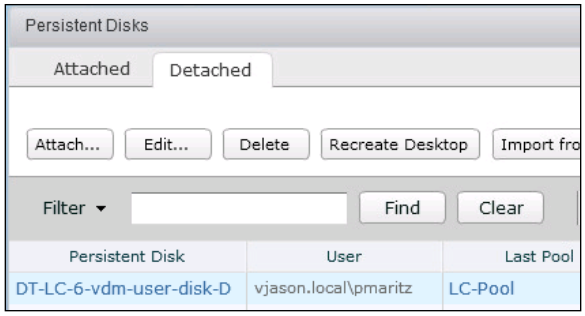
Attaching a detached persistent disk to an existing desktop

View provides the ability to attach a detached persistent disk to an existing desktop, enabling the user of that desktop to have access to that persistent disk as well as his/her own disk. This can be useful in scenarios where someone needs quick access to the data of a departed user, and you want to accomplish the task using only the View console.

Remember that linked-clone desktops should never have their storage configuration changed from within vCenter, as this can render the desktop unmanageable by View. Always use the View Manager Admin console to make changes that affect the linked-clone storage configuration.

The following steps outline how to attach the detached persistent disk to an existing desktop:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Persistent Disks** window under **Inventory** within the console, and click on the **Detached** tab.
3. Highlight the persistent disk that you wish to use and click on the **Attach** button, as shown in the following screenshot:



4. In the **Attach Persistent Disk** window, select the desktop that you wish to assign the disk to and click on **OK**. Only desktops with assigned users will appear in the list of choices.

The persistent disk will be attached to the existing desktop and the contents will be accessible to the user whose desktop it was assigned to.

Importing a persistent disk

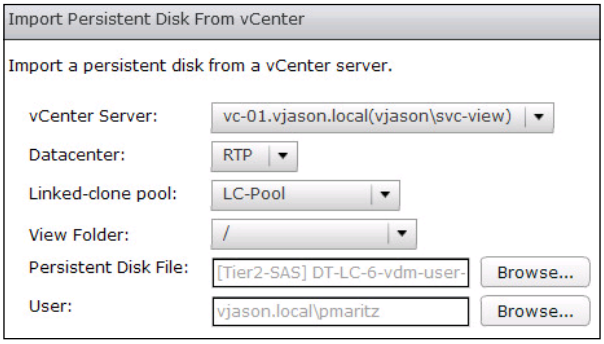
To support a wider range of recovery scenarios, View supports importing persistent disks and using them to create a new desktop. An imported persistent disk will not have any owner or desktop pool information, so the View Administrator will have to choose a new owner and desktop pool. This can be useful in scenarios where an organization wishes to retain persistent disks offline or on secondary storage that is not attached to the View infrastructure.

The following steps outline how to import persistent disks and use them to create a new desktop:

1. Log in to the View Manager Admin console using an AD account that has administrative permissions within View.
2. Open the **Persistent Disks** window under **Inventory** within the console, and click on the **Detached** tab.
3. Click on the **Import from vCenter...** button as shown in the following screenshot:

4. In the **Import Persistent Disk From vCenter** window (shown in the following screenshot), select the appropriate resources including:
    - **vCenter Server**: Target vCenter server
    - **Datacenter**: Target vCenter datacenter object
    - **Linked-clone pool**: Pool where the linked clone will be created
    - **View Folder**: Optional; destination folder for the virtual machine object

- **Persistent Disk File**: The file you will be importing
- **User**: The user who will be assigned the desktop that will use the persistent disk



5. Click on **OK** to import the persistent disk and associate it with the linked-clone desktop.

To access the data on the imported persistent disk, the user it was assigned to will need to log in to the newly configured desktop.

Summary

In this chapter, we have learned about different View Composer linked-clone maintenance operations. We discussed each of these maintenance tasks and went through the examples of how they are used.

We learned about linked-clone refresh, recompose, and rebalance operations. We learned about what they are for and what to be aware of concerning their use; we then went through the operation of each. We also learned about persistent disk maintenance, including how to detach them from existing desktops, how to reattach them, and how to use them to recreate a new linked-clone desktop.

In the next chapter, we will discuss how to create a virtual desktop master image, an important task that requires careful consideration and planning.

11
Creating a Master Virtual Desktop Image

When designing your VMware Horizon View infrastructure, creating a Virtual Desktop master image is second only to infrastructure design in terms of importance. The reason for this is simple; as ubiquitous as Microsoft Windows is, it was never designed to be a hosted Virtual Desktop. The good news is that with a careful bit of planning, and a thorough understanding of what your end users need, you can build a Windows desktop that serves all your needs, while requiring the bare minimum of infrastructure resources.

A default installation of Windows contains many optional components and configuration settings that are either unsuitable for, or not needed in, a Virtual Desktop environment, and understanding their impact is critical to maintaining Virtual Desktop performance over time and during peak levels of use.

Uninstalling unneeded components and disabling services or scheduled tasks that are not required will help reduce the amount of resources the Virtual Desktop requires, and ensure that the View infrastructure can properly support the planned number of desktops even as resources are oversubscribed.

Oversubscription is defined as having assigned more resources than what is physically available. This is most commonly done with processor resources in Virtual Desktop environments, where a single server processor core may be shared between multiple desktops. As the average desktop does not require 100 percent of its assigned resources at all times, we can share those resources between multiple desktops without affecting the performance.

This chapter will focus on a number of different topics related to the planning and creation of a Virtual Desktop master image.

In this chapter, we will learn:

- Why it is important to optimize a Virtual Desktop master image
- What kinds of results we will see if we optimize the Windows OS
- How to optimize the virtual machine hardware
- How to optimize the underlying Windows file system
- How to optimize the Windows OS
- Why you should customize the default Windows user local profile

Why is desktop optimization important?

To date, Microsoft has only ever released a version of Windows designed to be installed on physical hardware. This isn't to say that Microsoft is unique is this regard, as neither Linux and Mac OS X offers an installation routine that is optimized for a virtualized hardware platform.

While nothing stops you from using a default installation of any OS or software package in a virtualized environment, you may find it difficult to maintain consistent levels of performance in Virtual Desktop environments where many of the resources are shared, and in almost every case oversubscribed in some manner. In this section, we will examine a sample of the CPU and disk IO resources that can be recovered were you to optimize the Virtual Desktop master image.

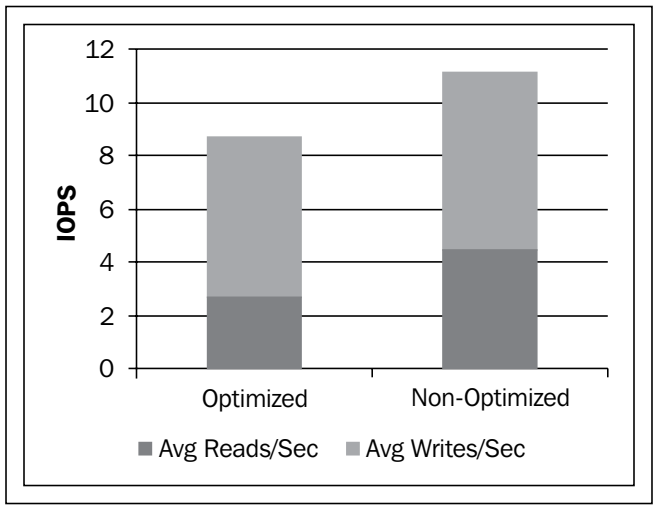
Due to the technological diversity that exists from one organization to the next, optimizing your Virtual Desktop master image is not an exact science. The optimization techniques used and their end results will likely vary from one organization to the next due to factors unrelated to View or vSphere. The information contained within this chapter will serve as a foundation for optimizing a Virtual Desktop master image, focusing primarily on the operating system.

Optimization results – desktop IOPS

Desktop optimization benefits one infrastructure component more than any other: storage. Until all flash storage arrays achieve price parity with the traditional spinning disk arrays many of us use today, reducing the per-desktop IOPS required will continue to be an important part of any View deployment.

On a per-disk basis, a flash drive can accommodate more than 15 times the IOPS of an enterprise SAS or SCSI disk, or 30 times the IOPS of a traditional desktop SATA disk. Organizations that choose an all-flash array may find that they have more than sufficient IOPS capacity for their Virtual Desktops, even without doing any optimization.

The following graph shows the reduction in IOPS that occurred after performing the optimization techniques described later in this chapter.

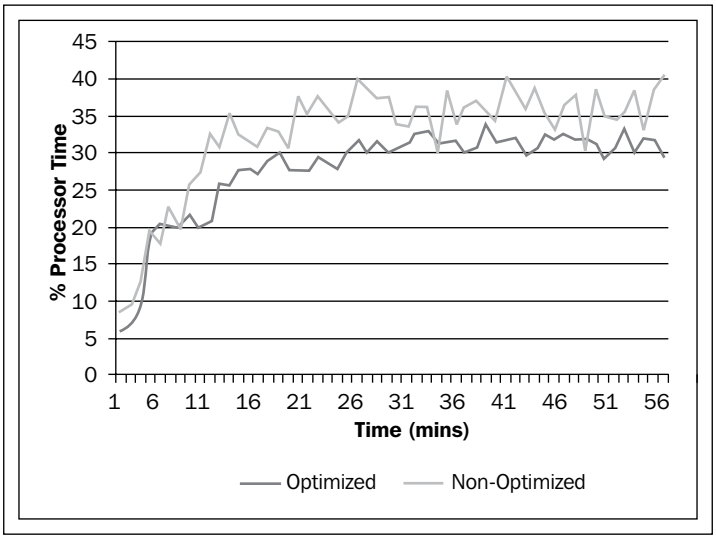


The optimized desktop generated 15 percent fewer IOPS during the user workload simulation. By itself that may not seem like a significant reduction, but when multiplied by hundreds or thousands of desktops the savings become more significant.

Optimization results – CPU utilization

View supports a maximum of 16 Virtual Desktops per physical CPU core. There is no guarantee that your View implementation will be able to attain this high consolidation ratio, though, as desktop workloads will vary from one type of user to another. The optimization techniques described in this chapter will help maximize the number of desktops you can run per each server core.

The following graph shows the reduction in vSphere host % **Processor Time** that occurred after performing the optimization techniques described later in this chapter:



% Processor Time is one of the metrics that can be used to measure server processor utilization within vSphere. The statistics in the preceding graph were captured using the vSphere ESXTOP command line utility, which provides a number of performance statistics that the vCenter performance tabs do not offer, in a raw format that is more suited for independent analysis.

The optimized desktop required between 5 to 10 percent less processor time during the user workload simulation. As was the case with the IOPS reduction, the savings are significant when multiplied by large numbers of desktops.

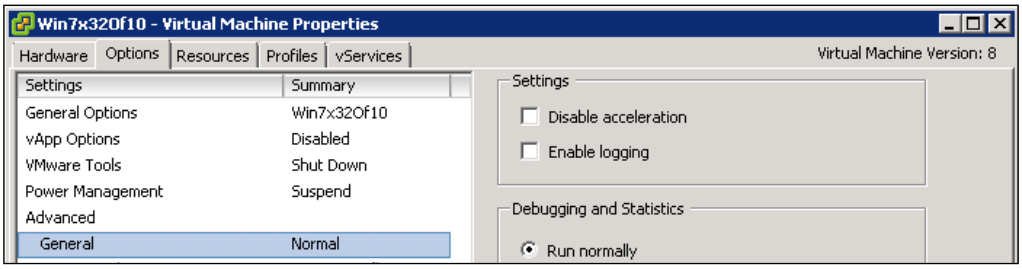
Virtual Desktop hardware configuration

The Virtual Desktop hardware configuration should provide only what is required based on the desktop needs and the performance analysis that was done in *Chapter 1*, *Designing a VMware Horizon View Infrastructure*. This section will examine the different virtual machine configuration settings that you may wish to customize, and explain their purpose.

Disabling virtual machine logging

Every time a virtual machine is powered on, and while it is running, it logs diagnostic information within the datastore that hosts its VMDK file. For environments that have a large number of Virtual Desktops, this can generate a noticeable amount of storage I/O. The following steps outline how to disable virtual machine logging:

1. In the vCenter client, right-click on the desktop master image virtual machine and click on **Edit Settings** to open the **Virtual Machine Properties** window.
2. In the **Virtual Machine Properties** window, select the **Options** tab.
3. Under **Settings**, highlight **General**.
4. Clear **Enable logging** as shown in the following screenshot, which sets the **logging = "FALSE"** option in the virtual machine VMX file:



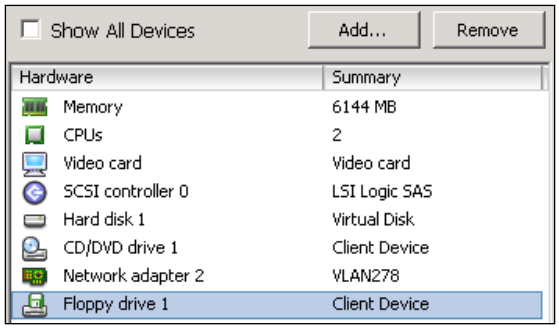
While disabling logging does reduce disk IO, it also removes log files that may be used for advanced troubleshooting or auditing purposes. The implications of this change should be considered before placing the desktop into production.

Removing unneeded devices

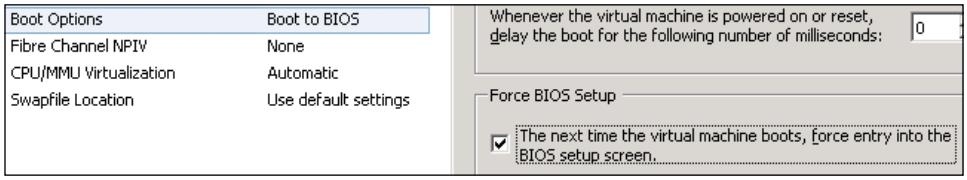
By default, a virtual machine contains several devices that may not be required in a Virtual Desktop environment. In the event that these devices are not required, they should be removed to free up server resources. The following steps outline how to remove the unneeded devices:

1. In the vCenter client, right-click on the desktop master image virtual machine and click on **Edit Settings** to open the **Virtual Machine Properties** window.

2. In the **Virtual Machine Properties** window, under **Hardware**, highlight **Floppy drive 1** as shown in the following screenshot and click on **Remove**:

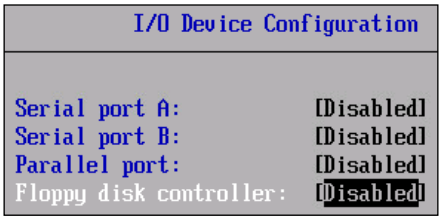


3. In the **Virtual Machine Properties** window, select the **Options** tab.
4. Under **Settings**, highlight **Boot Options**.
5. Check the checkbox under the **Force BIOS Setup** section as shown in the following screenshot:



6. Click on **OK** to close the **Virtual Machine Properties** window.
7. Power on the virtual machine; it will boot into the **PhoenixBIOS Setup Utility**.
8. The **PhoenixBIOS Setup Utility** menu defaults to the **Main** tab. Use the down arrow key to move down to the **Legacy Diskette A**, and then press the Space bar key until the option changes to **Disabled**.
9. Use the right arrow key to move to the **Advanced** tab.
10. Use the arrow down key to select **I/O Device Configuration** and press *Enter* to open the **I/O Device Configuration** window.

11. Disable the serial ports, parallel port, and floppy disk controller as shown in the following screenshot. Use the up and down arrow keys to move between devices, and the Space bar to disable or enable each as required:



12. Press the *F10* key to save the configuration and exit the **PhoenixBIOS Setup Utility**.

Do not remove the virtual CD-ROM device, as it is used by vSphere when performing an automated installation or upgrade of the VMware Tools software.

Customizing the Windows desktop OS cluster size

Microsoft Windows uses a default cluster size, also known as allocation unit size, of 4 KB when creating the boot volume during a new installation of Windows. The cluster size is the smallest amount of disk space that will be used to hold a file, which affects how many disk writes must be made to commit a file to disk. For example, when a file is 12 KB in size, and the cluster size is 4 KB, it will take three write operations to write the file to disk.

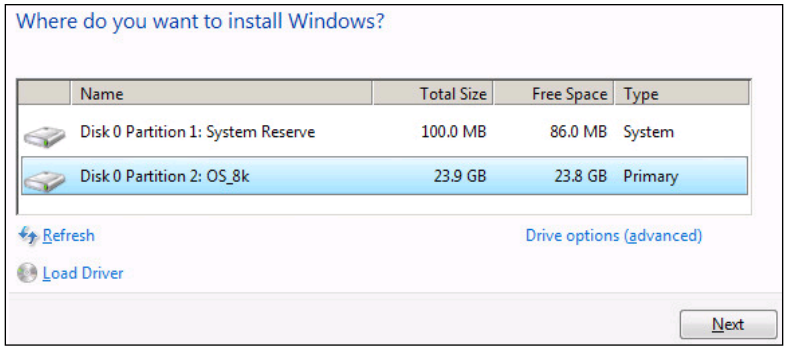
The default 4 KB cluster size will work with any storage option that you choose to use with your environment, but that does not mean it is the best option. Storage vendors frequently do performance testing to determine which cluster size is optimal for their platforms, and it is possible that some of them will recommend that the Windows cluster size should be changed to ensure optimal performance. The following steps outline how to change the Windows cluster size during the installation process; the process is the same for both Windows 7 and Windows 8. In this example, we will be using an 8 KB cluster size, although any size can be used based on the recommendation from your storage vendor.

- The cluster size can only be changed during the Windows installation, not after.
- If your storage vendor recommends the 4 KB Windows cluster size, the default Windows settings are acceptable.

1. Boot from the Windows OS installer ISO image or physical CD and proceed through the install steps until the **Where do you want to install Windows?** dialog box appears.
2. Press *Shift* + *F10* to bring up a command window.
3. In the command window, enter the following commands:

```
diskpart
select disk 0
create partition primary size=100
active
format fs=ntfs label="System Reserve" quick
create partition primary
format fs=ntfs label=OS_8k unit=8192 quick
assign
exit
```

4. Click on **Refresh** to refresh the **Where do you want to install Windows?** window.
5. Select **Drive 0 Partition 2: OS_8k**, as shown in the following screenshot, and click on **Next** to begin the installation:



The **System Reserve** partition is used by Windows to store files critical to the boot process and will not be visible to the end user. These files must reside on a volume that uses a 4 KB cluster size, so we created a small partition solely for that purpose. Windows will automatically detect this partition and use it when performing the Windows installation.

In the event that your storage vendor recommends a different cluster size than shown in the previous example, replace the `8192` in the sample command in step 3 with whatever value the vendor recommends, in bytes, without any punctuation.

Windows OS pre-deployment tasks

The following tasks are unrelated to the other optimization tasks that are described in this chapter but they should be completed prior to placing the desktop into production.

Installing VMware Tools

VMware Tools should be installed prior to the installation of the View Agent software. To ensure that the master image has the latest version of the VMware Tools software, apply the latest updates to the host vSphere Server prior to installing the tools package on the desktop.

The same applies if you are updating your VMware Tools software. The View Agent software should be reinstalled after the VMware Tools software is updated to ensure that the appropriate View drivers are installed in place of the versions included with VMware Tools.

Cleaning up and defragmenting the desktop hard disk

To minimize the space required by the Virtual Desktop master image and ensure optimal performance, the Virtual Desktop hard disks should be cleaned of nonessential files and optimized prior to deployment into production. The following actions should be taken once the Virtual Desktop master image is ready for deployment:

- Use the Windows **Disk Cleanup** utility to remove any unnecessary files.

- Use the Windows Defragment utility to defragment the virtual hard disk.

  If the desktop virtual hard disks are thinly provisioned, you may wish to shrink them after the defragmentation completes. This can be performed with utilities from your storage vendor if available, by using the vSphere **vmkfstools** utility, or by using the vSphere storage vMotion feature to move the virtual machine to a different datastore. Visit your storage vendor or the *VMware vSphere Documentation* (`http://www.vmware.com/support/pubs/vsphere-esxi-vcenter-server-pubs.html`) for instructions on how to shrink virtual hard disks or perform a storage vMotion.

Windows OS optimizations

A default installation of Microsoft Windows contains a number of configuration settings, components, and scheduled tasks that may not be required or are not desirable in a Virtual Desktop environment. This section will detail these settings, and provide instructions on how to make the recommended changes.

Many of these optimizations are implemented using Windows Group Policies, which can be applied to the Virtual Desktop master image prior to deployment, or by using domain-enforced Active Directory Group Policies. It is recommended to apply the majority of the policies directly to the master image when using linked-clone desktops. Doing this allows the refresh or recompose operation to proceed more quickly, as the majority of the settings the desktop requires will already have been applied. Were the necessary policies to be applied using only domain-based Group Policy templates, a refresh or recompose is likely to take more time and resources to complete as each desktop must process the policy updates and make the necessary configuration changes. In addition, the desktops may require a reboot to fully implement the policy changes.

Most of the information in this section applies to both Windows 7 and Windows 8. If a specific recommendation applies only to Windows 8, it will be identified.

Disabling Windows Error Reporting

Windows Error Reporting compiles error reports that occur when an application crashes and, if configured to, forwards the information on to Microsoft. Linked-clone desktops, both persistent and non-persistent, are less likely to require this feature, as the underlying OS is likely to be refreshed or recomposed on a regular basis. This feature may be needed when using full-clone desktops, though, as those desktops generally have a much longer life cycle that may require occasional application troubleshooting. The following steps outline how to disable Windows Error Reporting:

1. Using the Group Policy console, edit the local desktop or domain-based Group Policy.
2. Select the **Computer Configuration** | **Administrative Templates** | **Windows Components** | **Windows Error Reporting** policy object.
3. Set **Disable Windows Error Reporting** to **Enabled**.

Disabling automatic updates

Linked-clone desktops are typically updated using a recompose operation, which negates the need for the Windows update service. To prevent linked-clone desktops from installing updates, which would significantly grow the linked-clone OS disk size, this Windows feature should be disabled.

If your environment uses full-clone View desktops, which are deployed using a vSphere template, you should regularly update the template with the latest Windows patches or other required configuration changes. This ensures that as new desktops are deployed, they will require little or no additional configuration as the template will already have been updated. By default, any new linked-clone desktops in a given desktop pool will automatically be created using either the virtual machine snapshot selected during the most recent recompose operation, or the original virtual machine snapshot if no recompose operations have been performed.

The following steps outline how to disable Windows Updates:

1. Using the Group Policy console, edit the local desktop or domain-based Group Policy.
2. Select the **Computer Configuration** | **Administrative Templates** | **Windows Components** | **Windows Update** policy object.
3. Set the **Configure Automatic Updates** to **Disabled**.

Delete the `C:\Windows\SoftwareDistribution\Download` folder to remove any update packages that may already been downloaded to the desktop.

Do not disable Automatic Updates in environments that use **System Center Configuration Manager** (**SCCM**), as it will prevent updates deployed using SCCM from being installed.

Removing unnecessary application updaters

A number of applications install their own updater utility, such as Adobe AIR, Adobe Acrobat Reader, and the Java Runtime Environment. If linked-clone desktops are being used, and regular recomposes are being performed, these updater utilities should be disabled to reduce the growth of the linked-clone OS disk. Full-clone desktops may wish to leave these updaters enabled, unless they will be installed using alternative methods, such as Microsoft SCCM and other similar desktop management platforms.

The procedure used to disable or remove these components will vary based on the architecture of the individual application. The following are examples of how to disable the application updaters that were mentioned in this section, which include Adobe AIR, Adobe Acrobat Reader, and the Java Runtime Environment.

Disabling the Adobe AIR updater

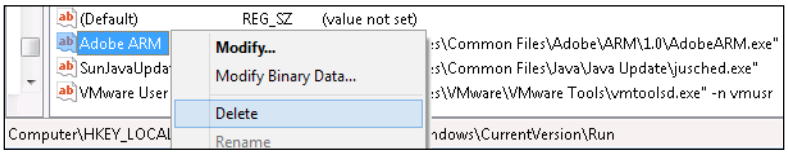
The following steps outline how to disable the Adobe AIR updater:

1. Download and install the Adobe AIR Settings Manager application, which is available from the `www.adobe.com` website.
2. Launch the AIR Settings Manager application, click on the **Disable Updates** button, and then close the application.

Disabling the Adobe Acrobat automatic updater

The following steps outline how to disable the Adobe Acrobat automatic updater:

1. From a Windows command prompt, load the `regedit.exe` application.
2. Navigate to **HKLM** | **SOFTWARE** | **Microsoft** | **Windows** | **Current Version** | **Run**.
3. Right-click on the **Adobe ARM** registry key shown in the following screenshot, and click on **Delete**:



The **HKLM** | **SOFTWARE** | **Microsoft** | **Windows** | **Current Version** | **Run** registry key will contain other programs that are executed upon system startup. Review the entries in this registry key, and if the program is not required, delete the key to prevent the application from loading upon system startup. Prior to making any changes, research the key being removed to ensure that the parent program will continue to operate properly were this component of the program not executed at startup.

If you wish to prevent users from performing manual updates within the Acrobat Reader application itself, you can use the Adobe Acrobat Group Policy templates referenced in the *Adobe Acrobat Enterprise Administration Guide* located at `http://www.adobe.com/devnet-docs/acrobatetk/tools/AdminGuide/index.html`. These Group Policy objects enable you to completely disable the ability to update Acrobat Reader using the application menus.

Disabling the Java updater utility

The following steps outline how to disable the Java updater utility:

1. From a Windows command prompt, load the `regedit.exe` application.
2. Navigate to **HKLM** | **SOFTWARE** | **Microsoft** | **Windows** | **Current Version** | **Run**.
3. Right-click on the **SunJavaUpdateSched** registry key and click on **Delete**. This registry key is shown in the screenshot in the previous section beneath the Adobe ARM registry key.

Removing unneeded Windows components

There are a number of Windows components that are installed by default that may not be needed in a Virtual Desktop environment. To further reduce the resources required by the desktop, remove any Windows components that are not required. Some components that may not be required include Indexing Service, Internet Printing Client, Media Features, Tablet PC Components, and Windows Search.

The following steps outline how to remove unneeded Windows components:

1. Open the Windows Control Panel.
2. Navigate to **Programs** | **Turn Windows features on or off**.
3. Remove any unneeded components.

Changing NTFS filesystem settings

There are several NTFS options than can be optimized using the `fsutil` command to minimize filesystem overhead. These options include the ability to disable the creation of DOS style 8.3 filenames and disabling the last accessed timestamp. The "last accessed" option reduces write workload for users who run applications that access many files.

- Applications or shortcuts that access files using file or folder short names may not function properly if 8.3 names are disabled. Do not change this setting without researching if your applications require that functionality.
- Do not disable last accessed timestamps if think that your organization might need to audit the usage of any files that reside on the Virtual Desktop filesystem.

These features can be disabled from a Windows command prompt using the following commands:

```
fsutil behavior set disablelastaccess 1
fsutil 8dot3name set 1
```

Additional information about the `fsutil` command is available from Microsoft TechNet (`http://technet.microsoft.com/en-us/library/cc753059(v=ws.10).aspx`).

Pre-compiling .NET Framework assemblies

Microsoft .NET compiles framework assemblies on an as-needed basis when .NET-dependent programs are launched for the first time. This process can be both CPU- and disk-intensive, so you should pre-compile all .NET Framework assemblies on the Virtual Desktop master image prior to deployment. The following steps outline how to pre-compile all versions of the .NET Framework assemblies:

1. Open an elevated Windows command prompt.
2. Navigate to the `C:\Windows\Microsoft.NET\Framework\v4.0.30319` directory.
3. Type `ngen.exe executequeueditems` and hit *Enter*. This process may require several minutes to complete.

Microsoft .NET 3.5 is not installed by default on Windows 7 or Windows 8. If Microsoft .NET 3.5 is required on the desktop image, it should be installed prior to running the `ngen.exe executequeueditems` command. Microsoft .NET 3.5 can be installed using the Control Panel entry **Control Panel** | **Programs** | **Turn Windows features on or off**.

Disabling Windows hibernation

When the Windows OS goes into hibernation mode, a significant amount of storage write I/O is required to write the contents of the systems RAM to the `hiberfil.sys` file. An equivalent amount of read storage I/O is needed to resume the desktop from hibernation. During periods of heavy use, this additional I/O may affect the performance of other desktops that share the same storage. In addition to that, the `hiberfile.sys` file requires disk space equivalent to that of the desktops configured RAM, which further increases the amount of per-desktop space required. To reduce desktop storage utilization, hibernation should be disabled.

To disable hibernation, open an elevated Windows command prompt and execute the following command:

```
powercfg /hibernate off
```

If required, View can manage workstation power states using native vSphere features. Desktops can be powered down or suspended as required based on the configuration of the desktop pool.

Disabling Windows System Restore

Windows System Restore is used to restore a Windows desktop to a previous state, a useful feature when using a traditional physical desktop. This feature is generally not required when using linked-clone desktops, though, as those desktops can be refreshed as needed to restore them to their original state.

vSphere snapshots can be used in place of System Restore if you need the ability to quickly undo changes made to your Virtual Desktop master images or full-clone Virtual Desktops. vSphere snapshots should not be used with linked-clone desktops, as the snapshot would prevent View Composer maintenance operations from completing successfully.

Generating Windows System Restore snapshots generates intermittent spikes in storage I/O, and also requires additional disk space. To minimize per-desktop storage utilization, this feature should be disabled on all linked-clone desktops and full-clone desktops as well unless the feature is explicitly required. The following steps outline how to disable system restore:

1. Right-click **My Computer** and select **Properties**.
2. Select **Advanced system settings** | **System Protection**.
3. Click on the **Configure** button to open the **System Protection** window.
4. Under the **Restore Settings** section, click on the **Turn off system protection** radio button.

Sizing virtual machine RAM properly

The amount of RAM used for the desktop affects both the amount of storage space required and the likelihood that it will need to swap memory into the Windows page file.

Windows initially sizes the `C:\pagefile.sys` system file based on the amount of RAM the virtual machine is granted, which is then expanded on demand to meet the virtual memory requirements of the Windows OS. The page file will also increase in size when the desktop RAM is increased.

The amount of RAM assigned to the virtual machine affects whether or not it is likely to need to utilize the Windows page file. Using the page file generates additional storage I/O, which we prefer to avoid in a Virtual Desktop environment where the storage is shared among multiple desktops. Using the techniques described in *Chapter 1*, *Designing a VMware Horizon View Infrastructure*, the desktop should be assigned sufficient RAM so that under normal circumstances it will not need to use the page file.

The Microsoft TechNet article *Pushing the Limits of Windows: Virtual Memory* (`http://blogs.technet.com/b/markrussinovich/archive/2008/11/17/3155406.aspx`) contains additional guidance about how to properly size Windows system RAM.

Setting the Windows page file to a fixed size

By default, Windows dynamically expands and shrinks the Windows page file as required. This leads to fragmentation of the page file, and additional storage I/O. To minimize the storage I/O associated with page file operations, set the page file to a fixed size.

Not every Virtual Desktop configuration will require a page file. If you determine that a desktop pool has low per-desktop memory requirements, common when minimal applications are being used, you can disable the page file entirely. vSphere transparent page sharing, described in the VMware technical paper *Understanding Memory Resource Management in VMware vSphere 5.0* (`http://www.vmware.com/files/pdf/mem_mgmt_perf_vsphere5.pdf`) is enabled by default and will further reduce per-desktop vSphere memory utilization by removing redundant virtual machine memory pages from the vSphere server.

The following steps outline how to configure a fixed size page file:

1. Right-click **My Computer** and select **Properties**.
2. Select **Advanced system settings** | **Advanced**.
3. Click on the **Settings** button under **Performance** to open the **Performance Options** window.
4. In the **Performance Options** window, click on the **Advanced** tab.
5. In the **Advanced** tab, click on the **Change** button.
6. Click on the **Custom size** radio button, and populate the **Initial size** and **Maximum size** fields with the same value in MB.
7. Click on the **Set** button to implement the changes, and then click on **OK** three times to complete the action. Reboot the desktop if prompted.

Refer to the Microsoft TechNet article *Pushing the Limits of Windows: Virtual Memory* for additional guidance about how to determine the fixed page file size.

Disabling paging the executive

By default, Windows writes kernel-mode drivers and system code to the Windows page file when not in use, which leaves more RAM available for the system. This action generates additional storage I/O, which we prefer to limit in a Virtual Desktop environment.

If the virtual machine is assigned sufficient memory, this feature is unneeded and therefore should be disabled to reduce the per-desktop storage I/O. The following steps outline how to disable paging of the executive:

1. From a Windows command prompt, load the `regedit.exe` application.
2. Navigate to **HKLM** | **System** | **CurrentControlSet** | **Control** | **Session Manager** | **Memory Management**.
3. Double-click on the **DisablePagingExecutive** registry key to edit the value.
4. Change the value from `0` (default) to `1` to disable the feature.

Disabling Content Indexing of the C drive

Content Indexing creates storage I/O overhead for a desktop, as it builds the desktop content index cache. If Content Indexing is not required, or if the desktop is a linked-clone, this feature should be disabled to reduce desktop storage I/O.

Linked-clone desktops would require a content index each time they are refreshed or recomposed, significantly increasing the storage I/O required to complete these operations. Content Indexing should be disabled when using linked-clone desktops.

The following steps outline how to disable indexing of the local disk:

1. Open the **My Computer** window, right-click on `C:`, and click on **Properties**.
2. On the **General** tab, clear the **Allow files on this drive to have contents indexed in addition to file properties** checkbox.
3. Click on **OK** to initiate the change, and click on **OK** again to update the indexing settings using the default option (**Apply changes to Drive C:\, subfolders and files**).

During the application of the new indexing settings, an error message stating that a file is in use may occur. If this happens, select **Ignore All**. If the desktop has additional hard disks, repeat this process for each of those disks.

Disabling the content indexing of the remaining file locations

Windows indexes a number of system and user-specific folders by default. To reduce the storage I/O overhead associated with these indexing operations, remove any unneeded folder locations from the index list.

When using linked-clone desktops, it is suggested to uncheck all file locations from the **Indexed Locations** window.

The following steps outline how to disable indexing of the remaining default locations:

1. Open the Windows **Control Panel**.
2. Navigate to **Indexing Options**.
3. Click on the **Modify** button to open the **Indexed Locations** window.
4. Deselect any locations or folders in the list that you do not want indexed, and click on **OK**.

Disabling unnecessary services

There are multiple Windows services that are not useful in Virtual Desktop environments; to reduce desktop resource requirements, they should be disabled. Services are disabled by launching the Services MMC plugin, which can be launched from a Windows command prompt by executing the command `services.msc`.

The following is a list of services that are generally not required in a Virtual Desktop environment. A description for each of these services is provided in the Services MMC plugin:

- **Diagnostic Policy Service**
- **IP Helper**
- **Network Location Awareness**
- **Security Center**
- **Shell Hardware Detection**
- **SSDP Discovery**
- **SuperFetch** (disable only when using non persistent desktops; additional information about SuperFetch follows this section)

- **Telephony**
- **Themes**
- **Touch Keyboard and Handwriting Panel Service**
- **Windows Defender Service** (disable only when using alternative antivirus or anti-malware platforms)
- **Remote Registry**
- **Windows Audio**
- **Windows Connect Now** (Config Registrar; Windows 8 only)
- **Windows Update**
- **WLAN AutoConfig**
- **WWAN AutoConfig**

For a comprehensive list of Windows services and configuration options, review the *Windows 7 Service Configurations* and *Windows 8 Service Configurations* guides at `www.blackviper.com`.

SuperFetch

SuperFetch analyzes desktop usage patterns and pre-populates system RAM with programs the user is likely to launch. When using non-persistent desktops, this results in unnecessary storage I/O as the optimizations will not persist and will be repeated each time the desktop is used. SuperFetch also allocates more of the system RAM for its use, increasing the amount of per-desktop RAM required.

For persistent desktops with larger memory configurations, it is recommended to leave SuperFetch enabled so that Windows can optimize the disk layout of the prefetch data and proactively load user binaries into memory, which will make the desktop more responsive.

Removing unnecessary scheduled tasks

Windows has a number of scheduled tasks that are either undesirable or not required in a Virtual Desktop environment. These tasks can be removed or disabled using the Windows Control Panel **Schedule tasks** utility or an elevated command prompt. The following is a list of tasks that should be reviewed to determine if they are required within the Virtual Desktop environment; if not, they should be disabled or removed. Details about each task are available in the **Schedule tasks** Windows Control Panel utility.

- \Microsoft\Windows\Application Experience\AitAgent
- \Microsoft\Windows\Application Experience\ProgramDataUpdater
- \Microsoft\Windows\Application Experience\StartupAppTask: Windows 8 only
- \Microsoft\Windows\Autochk\Proxy
- \Microsoft\Windows\Bluetooth\UninstallDeviceTask
- \Microsoft\Windows\Customer Experience Improvement Program\BthSQM: Windows 8 only
- \Microsoft\Windows\Customer Experience Improvement Program\Consolidator
- \Microsoft\Windows\Customer Experience Improvement Program\KernelCeipTask
- \Microsoft\Windows\Customer Experience Improvement Program\UsbCeip
- \Microsoft\Windows\Defrag\ScheduledDefrag
- \Microsoft\Windows\DiskDiagnostic\Microsoft-Windows-DiskDiagnosticDataCollector
- \Microsoft\Windows\DiskDiagnostic\Microsoft-Windows-DiskDiagnosticResolver
- \Microsoft\Windows\FileHistory\File History (maintenance mode): Windows 8 only
- \Microsoft\Windows\Live\Roaming\MaintenanceTask: Windows 8 only
- \Microsoft\Windows\Live\Roaming\SynchronizeWithStorage: Windows 8 only
- \Microsoft\Windows\Maintenance\WinSAT: Windows 8 only
- \Microsoft\Windows\Mobile Broadband Accounts\MNO Metadata Parser: Windows 8 only
- \Microsoft\Windows\MobilePC\HotStart: Windows 8 only
- \Microsoft\Windows\Power Efficiency Diagnostics\AnalyzeSystem
- \Microsoft\Windows\Ras\MobilityManager
- \Microsoft\Windows\SideShow\AutoWake
- \Microsoft\Windows\SideShow\GadgetManager
- \Microsoft\Windows\SideShow\SessionAgent

- `\Microsoft\Windows\SideShow\SystemDataProviders`
- `\Microsoft\Windows\SpacePort\SpaceAgentTask`: Windows 8 only
- `\Microsoft\Windows\SystemRestore\SR`
- `\Microsoft\Windows\UPnP\UPnPHostConfig`
- `\Microsoft\Windows\Windows Defender\Windows Defender Cache Maintenance`: Windows 8 only
- `\Microsoft\Windows\Windows Defender\Windows Defender Cleanup`: Windows 8 only
- `\Microsoft\Windows\Windows Defender\Windows Defender Scheduled Scan`: Windows 8 only
- `\Microsoft\Windows\Windows Defender\Windows Defender Verification`: Windows 8 only
- `\Microsoft\Windows\Windows Error Reporting\QueueReporting`
- `\Microsoft\Windows\Windows Media Sharing\UpdateLibrary`
- `\Microsoft\Windows\WindowsBackup\ConfigNotification`

To remove a task using an elevated Windows command prompt, use a command similar to the following example:

```
SCHTASKS /Delete /TN "\Microsoft\Windows\Application Experience\
AitAgent" /F
```

Removing unnecessary Windows 8 Metro applications

Windows 8 includes a number of Metro applications that may not be required in a Virtual Desktop environment. In addition, some of the Metro applications are active even if they are not being used by the end user, placing unnecessary load on the View infrastructure. The full list of installed applications can be reviewed using the PowerShell command `Get-AppxPackage`.

The following PowerShell commands will uninstall Windows Metro applications related to Microsoft Bing, Xbox, Zune, and SkyDrive. Additional Metro applications are removed by using the same command syntax and replacing the `-name` portion of the command with the name of the application you wish to remove:

```
get-appxpackage -name Microsoft.Bing* | Remove-AppxPackage
get-appxpackage -name Microsoft.XBox* | Remove-AppxPackage
get-appxpackage -name Microsoft.Zune* | Remove-AppxPackage
get-appxpackage -name microsoft.microsoftsky* | Remove-AppxPackage
```

Disabling login success logging

By default, Windows records both successful and failed login attempts. For some environments, the overhead required to log successful logons is not required and the setting can be disabled. Non-persistent desktops generally do not require this setting, as they are destroyed after each use.

Disabling account login event auditing is considered a potential security issue, as this information may be required in order to fully audit access to the desktop.

The following steps outline how to disable successful login events so that they are not written to the security log:

1. Using the Group Policy console, edit the local desktop or domain-based group policy.
2. Select the **Computer Configuration** | **Windows Settings** | **Security Settings** | **Local Policies** | **Audit Policy** policy object.
3. Double-click the **Audit account logon events** policy to open the **Policy Properties** window.
4. Click on the **Failure** checkbox, and uncheck the **Success** checkbox if it is checked. Click on **OK** to close the window.

Changing the Group Policy refresh interval

By default, all computers in an Active Directory domain attempt to refresh their Group Policy settings every 90 minutes with a 30 minute offset. This is extended to limit the peak amount of network bandwidth that is consumed when refreshing the Group Policies. By default, Group Policy is also updated at every boot of the OS. The following steps outline how to change the Group Policy refresh interval:

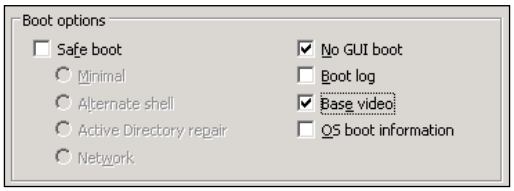
1. Using the Group Policy console, edit the local desktop or domain-based Group Policy.
2. Select the **Computer Configuration** | **Administrative Templates** | **System** | **Group Policy** policy object.
3. Double click the **Group Policy refresh internal for computers** policy to open the policy properties window.
4. Click on the **Enabled** radio button to enable the policy.

5. In the **Options** area, set how often the Group Policy will be applied to computers by typing or selecting the number of **Minutes**.
6. In the same area, set the amount of random time to be added to the Group Policy refresh interval by again typing or selecting the number of **Minutes**. This randomizes the policy refresh interval to prevent the desktops from refreshing the policies at the same time.

Disabling the Windows boot animation

Windows displays a start-up animation during the boot process. As this animation cannot be seen by users who are not connected to the virtual machine console, and additional vSphere Server resources are required to render it, it should be disabled. The following steps outline how to disable the Windows boot animation:

1. From a Windows command prompt, load the `msconfig.exe` application.
2. Select the **Boot** tab.
3. Under **Boot options**, check the **No GUI boot** and **Base video** checkboxes as shown in the following screenshot. Click on **OK** to finalize the changes.



Optimizing the Windows profile

There are various Windows settings that cannot be changed using normal Group Policies, or other post-installation customizations such as those we have already described in this chapter. To implement these additional settings, we can customize the default Windows local user profile, implementing changes that will be applied to all users who log on to the desktop for the first time.

In most cases, it is possible to create customized scripts or Group Policies that make these changes after a user has already logged into the account. This generally requires changes to the desktop system registry and an in-depth understanding of how the settings are recorded and updated within Windows.

As we discussed earlier, and this is particularly the case with linked-clone desktops, it is preferable to apply as much of your customization as you can to the Virtual Desktop master image. This ensures that the desktops are prepared using the minimum system resources required and are fully configured prior to their use. Policies that apply after the desktops are deployed may require an additional reboot to fully implement, which is not ideal for Virtual Desktop environments.

The process used to customize the default local user profile is outlined in the Microsoft KB article *Customize the default local user profile when preparing an image of Windows* (`http://support.microsoft.com/kb/973289`). The following settings, each of which helps reduce desktop resource utilization, are recommended to be made to the default local user profile.

The changes in this section will only apply to other desktop users if they are applied using the default user profile. The changes will not affect any profiles that are already present on the desktop image.

Adjusting for best performance

Some of the more advanced UI features, such as menu fading and animations, require additional desktop CPU and memory resources. The following steps outline how to disable these effects:

1. Right-click on **My Computer** and click on **Properties**.
2. Click on **Advanced system settings** to open the **System Properties** window.
3. In the **System Properties** window, click on the **Advanced** tab.
4. Under **Performance**, click on the **Settings** button.
5. Click on the **Adjust for best performance** radio button and then click on **OK** twice to close the window and update the settings.

Turning off system sounds

System sounds require additional server and network resources and may not be required in every Virtual Desktop environment. If sounds are required, the View administrator may want to configure a custom sound scheme based on the specific needs of the organization. The following steps outline how to turn off the system sounds:

1. Navigate to **Control Panel** | **Sound** | **Sounds**.
2. Set **Sound Scheme:** to **No Sounds**, or create and then select a custom sound scheme.

Disabling the Windows background and screen saver

Displaying a custom Windows wallpaper or screen saver requires additional server and network resources. The Windows wallpaper should be changed to either none, or a solid color, choosing the appropriate option based on the version of Windows being used. The screen saver should be disabled, or set to a blank screen, by choosing the appropriate option based on the version of Windows being used.

Summary

In this chapter, we have learned about how to configure the Virtual Desktop master image and about what makes it different from configuring a traditional desktop.

We discussed recommended Virtual Desktop hardware settings, how to customize the Windows file system during the installation process, optimizations that can reduce the desktop resource requirements, and when to use a custom Windows default user profile.

In *Chapter 12*, *Managing View SSL Certificates*, we will discuss how to manage and replace the default View self-signed SSL certificates.

12
Managing View SSL Certificates

VMware View, similar to many other applications that require SSL-based encryption, installs self-signed SSL certificates by default. A **self-signed certificate** is one that is signed by the creator, in this case the VMware View component that is being installed. While self-signed certificates do enable secure communications, by default they will not be trusted by any client or server who connects to them. An untrusted certificate leads to the familiar "There is a problem with this website's security certificate" message in Microsoft Internet Explorer, or the "VMware View cannot verify the identity of the server you have contacted" message in the VMware View client.

In addition, the default self-signed SSL certificates may have a smaller key length than is required within your organization; this is something that can only be addressed by replacing them after the installation has completed.

While it is possible to add exceptions that will make the default View SSL certificates trusted by the different components of the View infrastructure and the View clients themselves, it is preferable to replace the certificates with those signed by a commercial or private certificate authority. A number of commercial certificate authorities are supported by many OS's by default and organizations can distribute their own root certificate from a private certificate authority, which will enable trusted connections to any resource using certificates issued by that authority.

This chapter will show how to replace the default SSL certificates installed by each of the components of View. We will use Microsoft Active Directory Certificate Services to issue the certificates, although the process is similar if you were to use a commercial certificate authority.

By the end of this chapter we will learn:

- How to install the Microsoft **Internet Information Services** (**IIS**) console
- How to create a Local Computer Certificates console
- How to create a certificate request
- How to request a certificate using Microsoft Active Directory Certificate Services
- How to request a certificate with Subject Alternative Names using Microsoft Active Directory Certificate Services
- How to replace the certificate in a View Connection Server
- How to replace the certificate in a View Security Server
- How to replace the View Composer certificate
- How to replace the certificate in a View Transfer Server

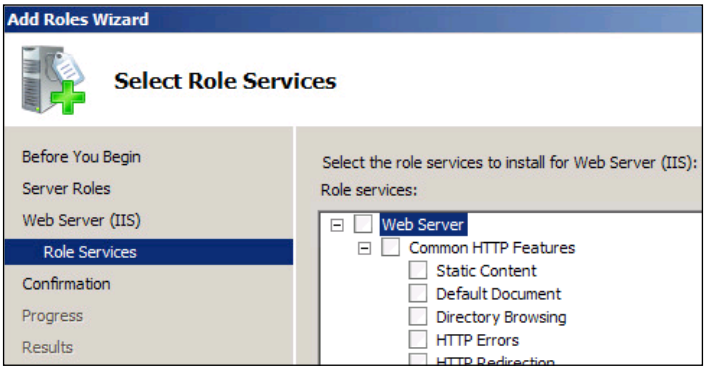
Installing the Microsoft Internet Information Services (IIS) console

The Microsoft IIS console is used to create the certificate request for the View servers. You can install the console on each View server or on another server entirely; alternatively, you can use an existing console that is already installed elsewhere. The certificate requests can be generated from any server with the console installed.

If you are installing the IIS console on a server running one of the View components, make sure that you follow the instructions explicitly and deselect the **Web Server** checkbox referenced in step 6 in the following procedure. Each of the View components runs its own web server; if additional web server software is installed, it will likely conflict with the one used by View.

1. Open the Windows Server Manager.
2. In the **Server Manager** window, click on **Add Roles** to open the **Add Roles Wizard** page.
3. In the **Before You Begin** tab under **Add Roles Wizard**, click on **Next**.
4. In the **Server Roles** tab, select **Web Server (IIS)** and click on **Next**.
5. In the **Web Server (IIS)** tab, review the introduction and click on **Next**.

6. In the **Role Services** tab, check the **Web Server** checkbox as shown in the following screenshot and click on **Next**. This will deselect all role services except for the IIS Manager console.



7. In the **Confirmation** tab, review the summary, verify that only the IIS Management console is being installed, and click on **Install**.
8. Reboot the server if prompted.

The IIS Manager administrative console should now appear in the **Administrative Tools** folder of the Windows Start menu. In the next section, we will create the console required to import certificates into the local computer certificate store.

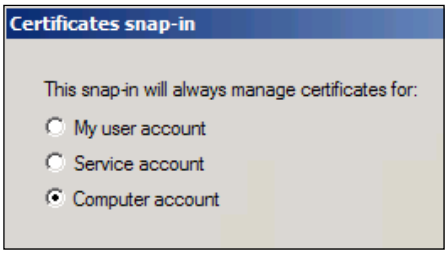
Creating a Local Computer Certificates console

The Local Computer Certificates MMC console will be used to replace the certificate on the View Connection, Security, and Composer Servers. The console will also be used to convert the certificate for the View Transfer Server to a Personal Information Exchange (PKCS#12 or PFX) format.

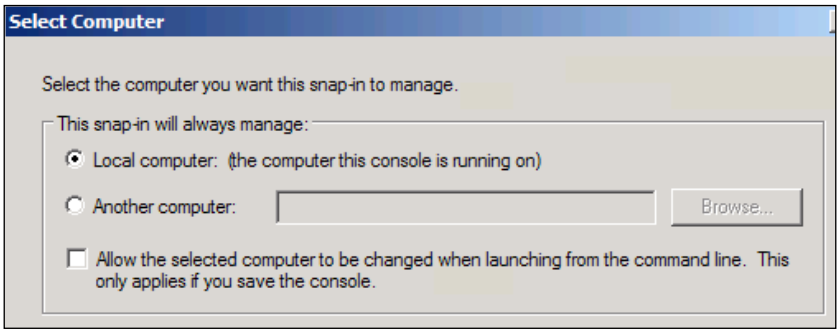
As with the IIS console, it is not necessary to create this console on each of the View servers. The same console can be created on a remote server and, in step 5 in the following procedure, a remote connection is made rather than a local one.

1. From the Windows Start menu, open the Microsoft Management Console by searching for and opening the `mmc.exe` application.

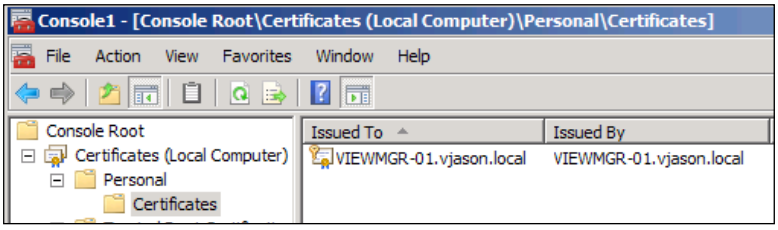
2. In the MMC console window, open the **File** menu and select **Add/Remove Snap-in** to open the **Add or Remove Snap-ins** window.
3. In the **Available snap-ins** section of the **Add or Remove Snap-ins** window, select **Certificates** and click on **Add** to open the **Certificates snap-in** window.
4. In the **Certificates snap-in** window, select the **Computer account** radio button as shown in the following screenshot. Click on **Next** to move to the **Select Computer** window.



5. In the **Select Computer** window shown in the following screenshot, if the computer us not already selected, click on the **Local computer** radio button and then click on **Finish** to return to the **Add or Remove Snap-ins** window.



6. Click on **OK** to close the **Add or Remove Snap-ins** window and return to the MMC console window. The console will now include the **Certificates** (**Local Computer**) snap-in, as shown in the following screenshot:



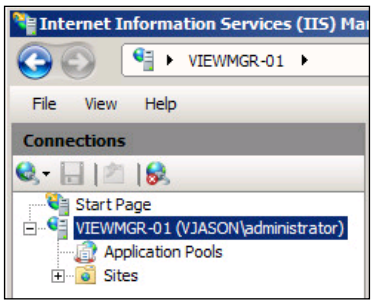
To create a shortcut for the console, open the **File** menu and click on **Save As** to open the **Save As** window. Provide a name and location for the shortcut and click on **Save**. This shortcut can be used to access the console without having to reconfigure it again.

Creating a certificate request

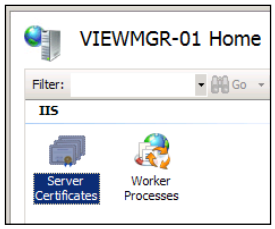
A **certificate request** is an encrypted string of text contained within a standard text file that contains most of the information needed for the certificate authority to generate a SSL certificate. You can generate requests using a number of different methods but we will focus on using the Microsoft IIS Manager administrative console.

The instructions in this section will show the request being made from the View Connection Server itself. As mentioned previously, the process is the same even if the request is made using the IIS Manager console on another server.

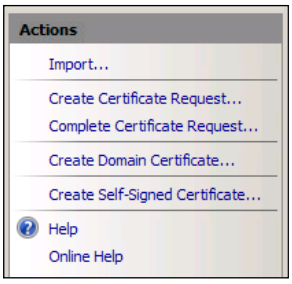
1. Open the **Internet Information Services (IIS) Manager** administrative console.
2. Under the **Connections** section of the **Internet Information Services (IIS) Manager** administrative console, click on the object that corresponds with the local server name, as shown in the following screenshot:



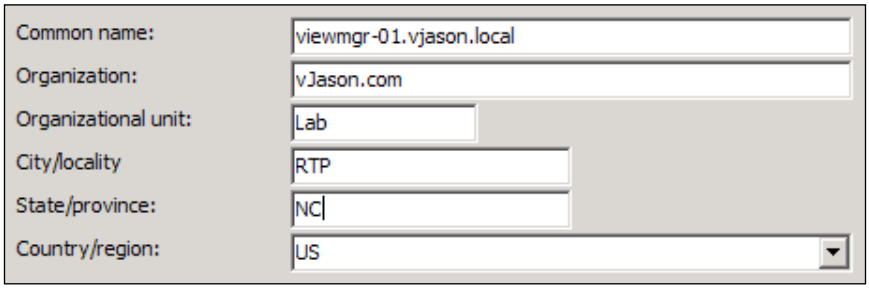
3. Under the **Home** section of the **Internet Information Services (IIS) Manager** administrative console (shown in the following screenshot), double-click on the **Server Certificates** object to open the **Server Certificates** window.



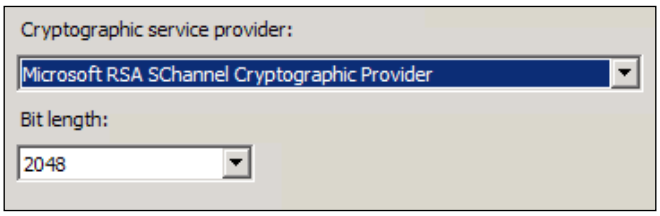
4. In the **Actions** section under **Server Certificates** of the **Internet Information Services (IIS) Manager** administrative console (shown in the following screenshot), click on **Create Certificate Request** to open the **Request Certificate** window.



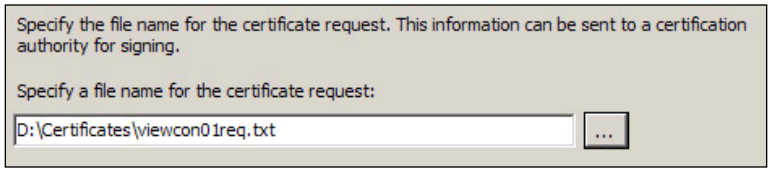
5. In the **Request Certificate – Distinguished Name Properties** window, populate the fields required, as shown in the following screenshot, and click on **Next**. The **Common name** field should be populated with the FQDN of the server you are requesting the certificate for.



6. In the **Request Certificate - Cryptographic Service Provider Properties** window, the **Cryptographic service provider** field should be set to **Microsoft RSA SChannel Cryptographic Provider** as shown in the following screenshot. Update the **Bit length** field, if required by your organization, and click on **Next**.



7. In the **Request Certificate - File Name** window (shown in the following screenshot), provide a file name and location for the certificate request and click on **Finish**.



8. Submit the certificate request to a certificate authority. If asked, the certificate should be provided in the PFX or **Canonical Encoding Rules** (**CER**) format.

If the certificate and its private key are in the **Privacy Enhanced Mail** (**PEM**) or other format, which may occur if you are using a commercial certificate authority, you must convert it to PKCS#12 (PFX) or CER format before you can import the certificate into a Windows certificate store on a View server host. Your certificate vendor should be able to provide you with a certificate in the required format. Certificates obtained from Active Directory Certificate Services will already be in the required CER format.

In the next section, we will submit the certificate request to a private Microsoft Active Directory Certificate Services certification authority.

Requesting a certificate using Microsoft Active Directory Certificate Services

Microsoft Active Directory Services is an optional component of the Windows Server operating system that enables organizations to create their own private certificate authority. Similar functionality is available with alternative operating systems but many organizations that have a large Windows infrastructure often already rely on AD Certificate Services to provide a number of different client and server certificates.

We will use AD Certificate Services to request certificates for each of the View components. The process is the same for each, although View Security Server may require additional information to be provided in step 5 in the following procedure. Consult the next section of this chapter for instructions on how to add additional DNS names to a certificate request.

If you wish to replace the View Security Server with a certificate suitable for both private and public FQDNs, and intend to use Microsoft Active Directory Certificate Services to issue the certificate, you will need to enable support for **Subject Alternative Names** (**SAN**) on an AD Certificate Services server.

Review Microsoft KB article 931351 (`http://support.microsoft.com/kb/931351`) for instructions on how to enable SAN support on a AD Certificate Services server.

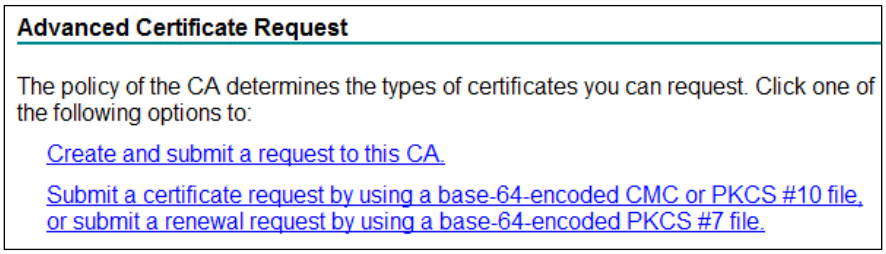
1. From a web browser, open the Microsoft Active Directory Certificate Services website. The URL for the site is typically in the format `https://FQDN of the server/certsrv`.

To access the Microsoft Active Directory Certificate Services website remotely, it must be configured with a SSL certificate. If your certificate services server has not been configured with a SSL certificate, you can only access the certificate services website from a web browser on the certificate services server itself, using HTTP instead of HTTPS. This section assumes that you can access the certificate services server remotely.

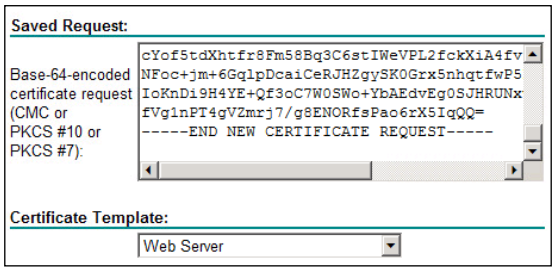
2. In the **Welcome** page shown in the following screenshot, click on **Request a certificate**:

3. In the **Request a Certificate** page shown in the following screenshot, click on **advanced certificate request**.

4. In the **Advanced Certificate Request** page (shown in the following screenshot), click on **Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file**.



5. In the **Submit a Certificate Request or Renewal Request** window (shown in the following screenshot), paste the contents of the certificate request file (obtained in step 7 of the **Creating a Certificate Request** section) in the **Saved Request** window. In the **Certificate Template** drop-down menu, select **Web Server**. Click on **Submit** to submit the certificate request and open the **Web Access Confirmation** window.

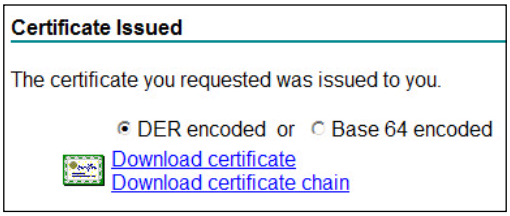


6. In the **Web Access Confirmation window**, review the message and click on **Yes** to complete the certificate request.

If Microsoft Active Directory Certificate Services is configured to approve web server requests automatically, the certificate will be available immediately. If not, the certificate administrator must approve the request to create the certificate.

This section assumes that the certificate request will automatically be approved and available for immediate download. If you need to wait for the certificate administrator to approve the certificate request, simply return to the certificate services website after the request has been approved and select **View the status of a pending certificate request** to download the certificate.

7. In the **Certificate Issued** page (shown in the following screenshot), click on the **DER encoded** radio button and then click on **Download certificate**. When prompted, save the certificate to the local computer.



The certificate is now available to be imported into the Windows certificate store.

New certificates should be copied and archived to a secure location prior to being imported into the Windows certificate store. This enables the certificates to be reused if any of the View servers ever need to be rebuilt.

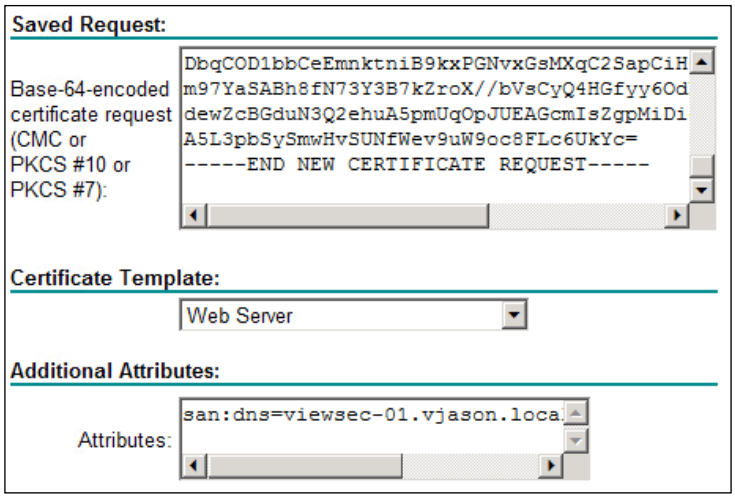
Requesting a certificate's Subject Alternative Names

A certificate's **Subject Alternative Name** (**SAN**) is often used on servers that have only one web service running but are accessed from multiple DNS names. A View Security Server is an example of this, as it may be accessed using a publicly known name such as `view.vjason.com`, as well as a private (internal) name such as `viewsec-01.vjason.local`.

While a certificate will work regardless of what DNS name was used to connect to it, if the DNS name used is not present as a SAN in the certificate, the client will receive an error that the certificate name does not match. To prevent these errors from occurring, when requesting the certificate we simply need to provide a list of the different DNS names that will be used to connect to the server.

In this section, we will request a certificate with SANs from our AD Certificate Services server. As mentioned previously, you will need to enable support for SANs on your AD Certificate Services server or the SANs will not be added to the certificate.

To request a certificate with SANs, all that is needed is to provide additional information in step 5 of the previous section, *Requesting a certificate using Microsoft Active Directory Certificate Services*. As shown in the following screenshot, we will use the **Additional Attributes** section of the **Submit a Certificate Request or Renewal Request** window to request the SANs:



Using the examples provided in this chapter, the following text should be inserted to add SANs to the certificate request.

```
san:dns=viewsec-01.vjason.local&dns=view.vjason.com
```

If additional SANs are required, simply append the text with an ampersand (`&`), and then the additional DNS name in the format `dns=my.domain.com`. Do not insert spaces in any part of the string of text.

The rest of the certificate request process is the same; once the request has been approved, save the certificate to the local computer to later import into the Windows certificate store.

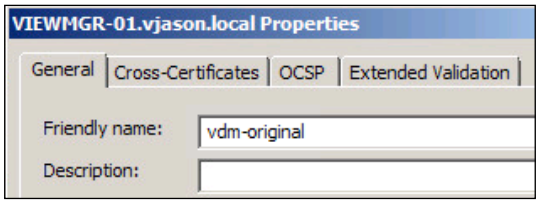
Replacing the certificate in a View Connection Server

The following steps outline how to replace the certificate on the View Connection Server, and assume that you have already obtained the replacement certificate. The View Connection Server will be unavailable while the certificate is being replaced, so plan for downtime accordingly.

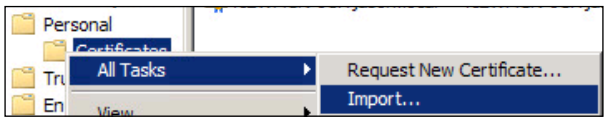
1. Using the Services MMC console, stop the VMware View Connection Server service. This will also stop other View-related services.
2. Open the Local Computer Certificates MMC console you created in the *Creating a Local Computer Certificates console* section.

3. Right-click on the existing View Manager certificate (shown in the following screenshot) and click on **Properties** to open the **Properties** window. This certificate is easily identified as it has a **Friendly Name** of **vdm**.

4. In the certificate's **Properties** window, append the friendly name of the certificate with `-original`, as shown in the following screenshot, and click on **OK** to return to the Local Computer Certificates MMC console:

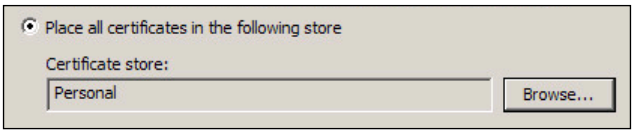


5. In the Local Computer Certificates MMC console, go to **Certificates (Local Computer)** | **Personal**. Then, right-click on the **Certificates** folder and click on **All Tasks** | **Import...** as shown in the following screenshot. This will open the **Welcome to the Certificate Import Wizard** window.

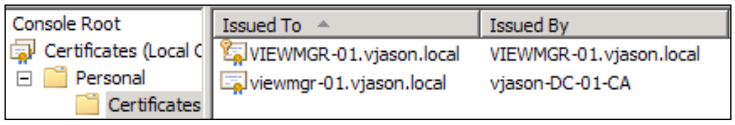


6. In the **Certificate Import Wizard** window, click on **Next**.
7. In the **Certificate Import Wizard - File to Import** window, click on the **Browse...** button, select the certificate file that you obtained from your certificate authority, and click on **Next**. In the following example, we have selected the certificate file **D:\Certificates\certnew.cer**:

8. In the **Certificate Import Wizard - Certificate Store** window (shown in the following screenshot), the **Place all certificates in the following store** radio button should already be checked. Select the **Personal** store and click on **Next**.



9. In the **Certificate Import Wizard - Completing the Certificate Import Wizard** window, review the settings and click on **Finish** to close the **Certificate Import Wizard** window. If changes are required, click on the **Back** button and make the changes where necessary.
10. The Certificate Import Wizard will open an additional window to confirm the successful import of the certificate. Click on **OK** to close this window.
11. The new certificate will appear alongside the existing certificate as shown in the following screenshot:

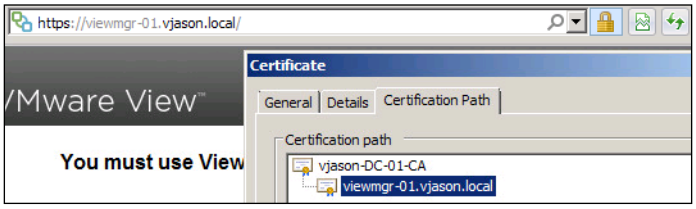


12. Right-click on the new certificate and click on **Properties** to open the **Properties** window. This certificate is easily identified as it has no friendly name.
13. In the certificate's **Properties** window, set the friendly name of the certificate to **vdm** and click on **OK** to return to the Local Computer Certificates MMC console.

The View Connection and Security Server software identifies which certificate to use by the value of the friendly name, which should be **vdm**.

14. Using the Services MMC console, start the **VMware View Connection Server** service.

15. From a web browser, access the View Connection Server using HTTPS and the FQDN. As shown in the following screenshot, verify that no SSL errors are shown, the new certificate is being used, and the certificate is trusted:

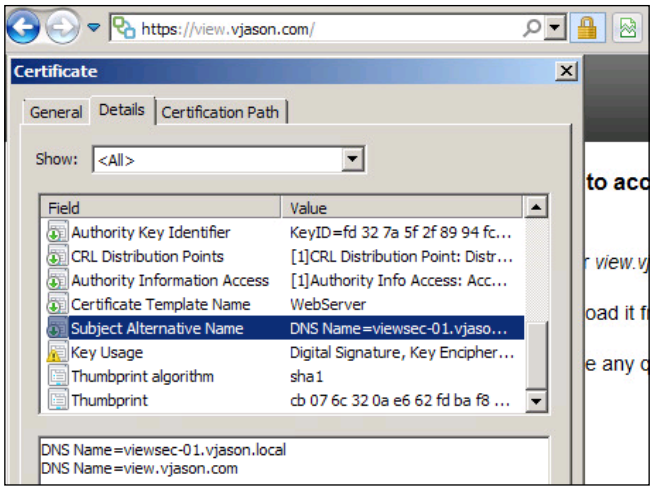


The same process should be repeated for any additional View Connection Servers, using unique certificates for each one.

Replacing the certificate on a View Security Server

The process used to replace the certificate on a View Security Server is nearly identical to that of the View Connection Server. This section will detail which steps from the previous section differ when replacing the certificate on a View Security Server. The View Security server will be unavailable while the certificate is being replaced, so plan for downtime accordingly. These steps assume that you have already obtained the replacement certificate. Follow the steps outlined in the *Replacing the certificate in a View Connection Server* section, replacing steps 1, 14, and 15 with the updated ones as follows:

- **Step 1**: Using the Services MMC console, stop the VMware View Security Server service. This will also stop other View-related services.
- **Step 14**: Using the Services MMC console, start the VMware View Security Server service.
- **Step 15**: From a web browser, access the View Security Server using HTTPS and all of the FQDNs that are defined in the certificate. As shown in the following screenshot, verify that no SSL errors are shown, the new certificate is being used, the certificate is trusted, and that the expected SANs are present.



The same process should be repeated on any additional View Security Servers, using unique certificates for each one.

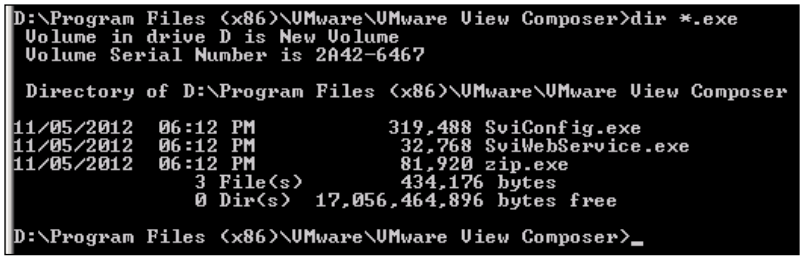
Replacing the View Composer certificate

View Composer uses a default certificate that is not trusted by the View Connection Server. While an exception can be made to trust this certificate when Composer is enabled, replacing the certificate with one that is trusted is straightforward and enables Composer to be trusted without an exception being made. The following steps outline how to replace the View Composer SSL certificate and assume that you have already obtained the replacement certificate. View Composer will be unavailable while the certificate is being replaced, so plan for downtime accordingly.

This process should be done before any linked clone desktops are deployed. This isn't a requirement; however, if any problems occur with View Composer, it is much easier to rebuild if no linked clone desktops are deployed.

1. Open the Local Computer Certificates MMC console.
2. In the Local Computer Certificates MMC console, go to **Certificates (Local Computer)** | **Personal**, right-click on the **Certificates** folder, and click on **All Tasks** | **Import...**. Complete the steps to import the certificate.

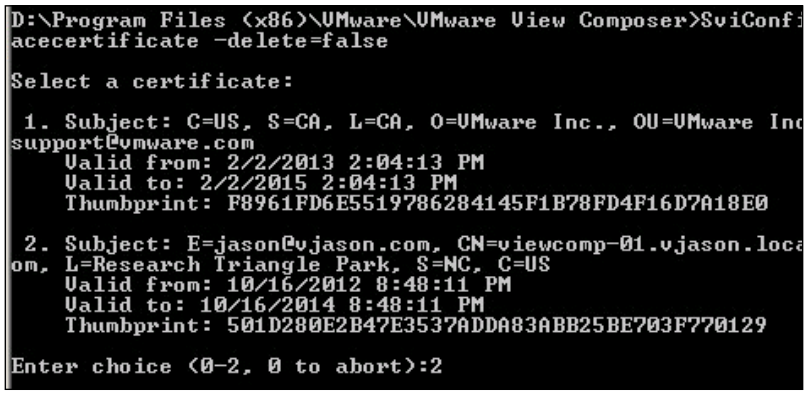
3. The new certificate will appear alongside the existing certificate. Unlike the certificates for the View Connection and Security Servers, you do not need to change the certificate's **Friendly Name**.
4. Open the Services MMC console and stop the VMware View Composer service.
5. Open a Windows command prompt and change to the `\Program Files (x86)\VMware\VMware View Composer` directory, as shown in the following screenshot:



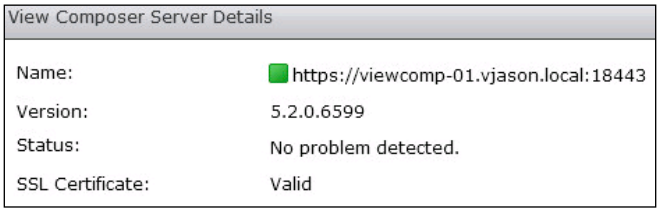
6. Enter the following command and press *Enter*. The `delete=false` option leaves the existing certificate in place, allowing us to use it again if ever required.

```
SviConfig -operation=replacecertificate -delete=false
```

7. Type in the number for the new certificate from the list provided and press *Enter*. As shown in the following screenshot, we will select certificate 2 as that is the newly issued certificate that we wish to enable. The output of the `SviConfig` command should verify that the operation completed successfully.



8. Open the Services MMC console and start the VMware View Composer service.
9. To verify that the certificate is trusted by the View Connection Servers, open the View Manager Admin console.
10. In the View Manager Admin console's dashboard, under **System Health**, expand the **View Composer Servers** object.
11. Click on the View Composer Server you wish to check, to open the **View Composer Server Details** window as shown in the following screenshot. Verify that the SSL Certificate field is shown as **Valid**.



The same process should be repeated on any additional servers that host View Composer, using unique certificates for each.

Replacing the certificate on a View Transfer Server

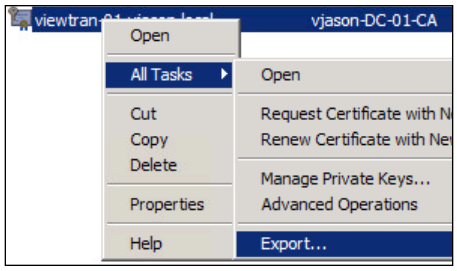
Replacing the certificates on a View Transfer Server requires a certificate in the PFX format, which necessitates a different procedure when using AD Certificate Services certificate authority. The following items are needed to prepare a certificate in the CER format to be used with a View Transfer Server.

- A computer with the open source OpenSSL software installed, which is available at `www.openssl.org`. OpenSSL is used to convert the certificate into the format needed by the Apache Web Server used with the View Transfer Server. In our example, OpenSSL was installed on a Windows computer.
- A certificate with an exportable private key, which is not allowed with the default AD Certificate Services web server template. To use certificates issued by AD Certificate Services, review Microsoft Technet article cc725621 (`http://technet.microsoft.com/en-us/library/cc725621(v=ws.10).aspx`) for instructions on how to create a new certificate template that allows exportable private keys.

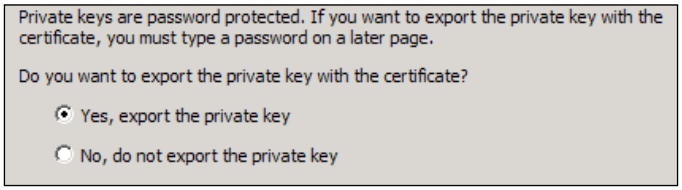
In this section, we will use the Local Computer Certificates console to convert the certificate obtained from AD Certificate Services from the CER format to the PFX format.

The following steps outline how to replace the certificate on a View Transfer Server and assume that you have already obtained the replacement certificate. Steps 1 through 10 can be completed on any server that has OpenSSL installed. The View Transfer Server will be unavailable while the certificate is being replaced, so plan for downtime accordingly.

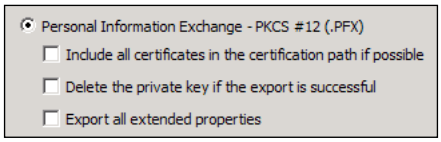
1. Open the Local Computer Certificates MMC console.
2. In the Local Computer Certificates MMC console, go to **Certificates (Local Computer) | Personal** and right-click on the **Certificates** folder. Then click on **All Tasks | Import...**. Complete the steps to import the certificate.
3. In the Local Computer Certificates MMC console, right-click on the newly imported certificate and click on **All Tasks | Export...** as shown in the following screenshot. This will open the **Welcome to the Certificate Export Wizard** window.



4. In the **Certificate Export Wizard** window. click on **Next**.
5. In the **Certificate Export Wizard - Export Private Key** window (shown in the following screenshot), click on the **Yes, export the private key** radio button and click on **Next**.



6. In the **Certificate Export Wizard - Export File Format** window (shown in the following screenshot), use the default settings and click on **Next**.



7. In the **Certificate Export Wizard - Password** window, provide a password for the private key and click on **Next**. In our example, the password will be `1234`.

Use a secure password when exporting the private key, and again in step 9 when performing the second conversion.

8. In the **Certificate Export Wizard - File to Export** window, provide a filename and location for the PFX file and click on **Next**. In our example, the file is named `viewtran.pfx`.
9. Execute each of the following commands in turn, providing the password from step 7 when prompted. These commands are executed on a computer with OpenSSL installed, from within the `OpenSSL-Win32\bin` directory.

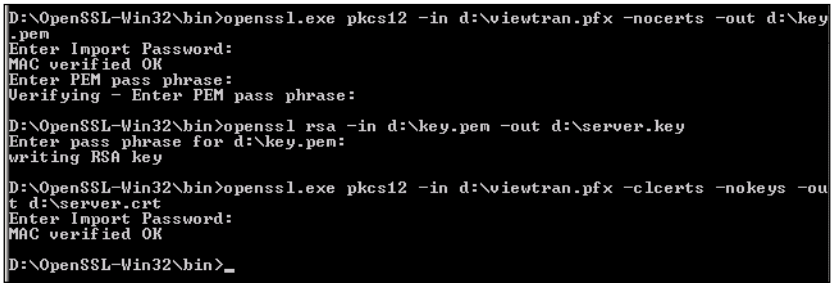
```
openssl.exe pkcs12 -in d:\viewtran.pfx -nocerts -out d:\key.pem

openssl rsa -in d:\key.pem -out d:\server.key

openssl.exe pkcs12 -in d:\viewtran.pfx -clcerts -nokeys -out d:\
server.crt
```

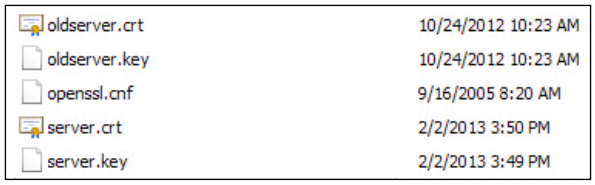
Replace the file and path name in the `-in` and `-out` switches in the preceding commands with the name and location of your files.

10. The following screenshot shows the execution of these commands and their expected output. Three files are generated during the conversion process: `key.pem`, `server.key`, and `server.crt`.

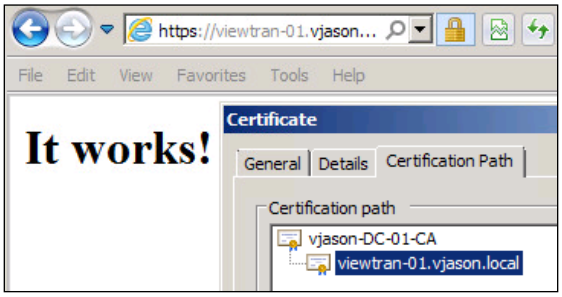


Each of the three files should be copied and archived to a secure location. This enables them to be reused if any of the View servers ever need to be rebuilt.

11. Browse to the `Program Files\VMware\VMware View\Server\httpd\conf` directory on the View Transfer Server.
12. Rename or archive the existing `server.key` and `server.crt` files.
13. Copy the new `server.key` and `server.crt` files into the folder. In the following screenshot, the existing files were appended with `old` in front of the filename, and the new files were copied into the same directory.



14. Reboot the View Transfer Server to implement the new certificate.
15. From a web browser, access the View Transfer Server using HTTPS and the FQDN. As shown in the following screenshot, verify that no SSL errors are shown, the new certificate is being used, and the certificate is trusted:



The same process should be repeated on any additional View Transfer Servers, using unique certificates for each one.

Summary

In this chapter, we have learned about how to replace the SSL certificates in each of the View servers.

We discussed how to use the standard Windows tools and utilities to create certificate requests and generate certificates, how the new certificates are imported and enabled, and how to use OpenSSL to convert the certificate into the format needed by the View Transfer Server.

In the next chapter, we will discuss the different Group Policy templates included with VMware View and how to use those to manage and customize your View environment.

13
Implementing VMware Horizon View Group Policies

VMware Horizon View includes six Microsoft Active Directory Group Policy templates to enable the customization of various aspects of the View environment. The templates focus on the View Agent, Client, Server, PCoIP settings, and Persona Management settings within View. Of these configuration items, only Persona Management requires the use of Group Policies in order to enable the feature. The remaining configuration items will all function without any changes of any kind, although most organizations will use one or more of the templates in order to fine-tune various aspects of their View environment.

These templates are provided in the Administrative Template (ADM) format, which allows them to be imported into an Active Directory Group Policy template. In this section, we will learn about each of the templates, their general purpose, and what settings are contained within each.

In this chapter, we will learn:

- What are the names and the description of each View ADM template file
- Where the View ADM template files are located
- What Group Policy Loopback Processing is
- Where are the View Group Policies applied
- What settings are contained within the six View ADM template files

View Group Policy overview

View includes six different Group Policy templates:

- `vdm_agent.adm`: This consists of settings related to the configuration of the View Agent
- `vdm_client.adm`: This consists of settings related to configuration of the View Client
- `vdm_server.adm`: This consists of settings related to the configuration of the View Connection Server
- `vdm_common.adm`: This consists of settings that are common to all View servers
- `ViewPM.adm`: This consists of settings used to enable and configure View Persona Management
- `pcoip.adm`: This consists of settings related to the configuration of the PCoIP display protocol

The Group Policy template files are installed along with the View Connection Server software in the `Program Files\VMware\VMware View\Server\extras\GroupPolicyFiles` folder.

Packages such as the View Client install its ADM template alongside the client software, in the `Program Files\VMware\VMware View\Client\extras` folder. This enables customization of the PCoIP settings on View Clients that are not members of the domain.

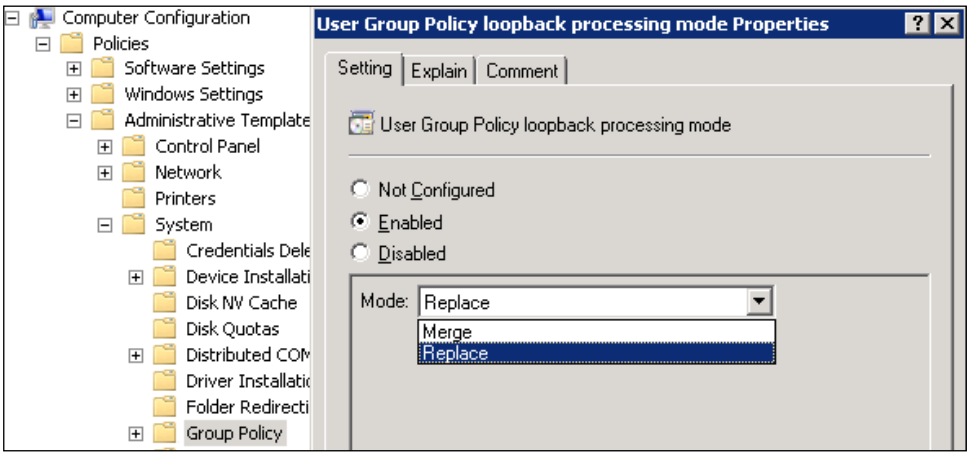
Loopback processing for Group Policies

View includes a number of user-based Group Policies that are only required when using a View desktop, such as settings for View Persona Management or the View Agent. In addition, an organization may be using some user-based Group Policies that they do not want, or should not want, applied to users when they are using View desktops, such as policies that install software applications.

Group Policy loopback processing applies policies based solely on the location of the Active Directory computer object, discarding or lowering the priority of other user policies that may have been applied by other GPOs.

Loopback processing is an advanced Group Policy feature that should not be used without understanding how it will affect the Group Policies of View desktop users. Since loopback processing will discard or reorder the application of other GPOs that apply to the user, before enabling it research what policies are affected and plan your View GPOs accordingly.

You can set the loopback policy in the Group Policy Object Editor snap-in by using the **User Group Policy loopback processing mode** policy setting under **Computer Configuration** | **Administrative Template** | **System** | **Group Policy** as shown in the following screenshot:

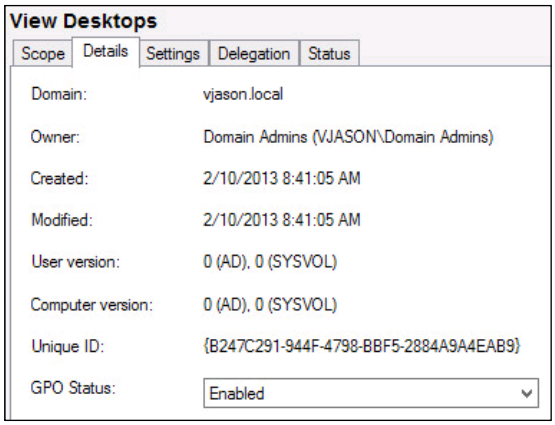


Two loopback processing modes are available:

- **Merge**: The list of user GPOs are gathered during the login process, followed by the computer GPOs. When the policies are applied, the computer GPOs are applied last, and as a result have higher precedence than the user GPOs.
- **Replace**: The other user GPOs are not gathered, only the computer GPOs are. Only the user GPOs within the current policy are applied to the user. All Computer GPOs will be applied.

When you choose **Replace** mode, all other user policies are discarded. As a result, it is important to ensure that any domain-critical user policies are included in the GPO that has loopback processing enabled in **Replace** mode.

When enabling loopback processing on a GPO, ensure that the GPO has user policy settings enabled in the **Details** tab's **GPO Status** drop-down menu shown in the following screenshot. A setting of **Enabled** means that both user and computer policies in the GPO will be applied:



For additional information about loopback processing, consult Microsoft TechNet.

View Agent Configuration ADM template

The View Agent Configuration ADM template file (`vdm_agent.adm`) contains policy settings related to the authentication and environmental components of View Agent. Where applicable, the default values for the settings are provided.

Agent Configuration base settings

The following settings apply to the base level of the View Agent Configuration ADM template:

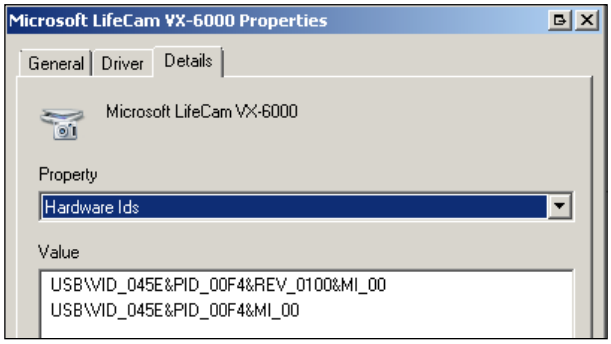
Policy	Applied to	Default Value	Description
Recursive Enumeration of Trusted Domains	Computers	**Enabled**	This setting ensures that View Clients will have access to all trusted AD domains when they attempt to log in. This setting should be enabled if the View environment supports users in other domains or other trusted AD forests.

The View Agent ADM template also includes USB Configuration policy settings. The same policies can also be applied to the View Client using the View Client ADM template. If the policies are applied using the View Agent ADM template, the View Client will download the USB policy settings from View Agent and use them in conjunction with the View Client USB policy settings to decide which devices it will allow to be available for redirection from the client computer. The USB policies are all computer configuration objects.

View USB Configuration settings

When configuring any setting that involves USB device families or specific devices, you have the option of overriding or merging the View Agent setting with those of the View Client. To override the View Client setting, enter an **O:** before the identifier; to merge the setting, enter an **M:** before the identifier. All the following examples use the override option.

USB vendor and product IDs can be obtained from the device properties page in the Windows Device Manager. The following screenshot shows the USB hardware IDs obtained from the **Details** tab of the device properties:



The portion of the vendor and hardware ID that will be used in the USB settings is **VID_045E&PID_00F4**. This value must be converted to a specific format for the USB policy setting. The following guidelines can be used to convert the ID into the value needed:

- The underscores should be replaced by dashes
- The ampersand should be replaced by an underscore

Using these guidelines, the new vendor and hardware ID will be **VID-045E_PID-00F4**. **045E** represents the vendor ID, while **00F4** represents the hardware ID. To specify all hardware devices made by that vendor, you can simply replace the hardware ID with ****, which acts as a wildcard. The resultant ID would be **VID-045E_PID-****.

USB device families are standardized names that are used by all vendors who sell USB devices. A full list of USB device families can be found in the VMware Horizon View Administration guide (`http://www.vmware.com/support/pubs/view_pubs.html`).

The following settings apply to the View USB Configuration section of the View Agent Configuration ADM template. Each of these settings are computer configuration objects and undefined by default unless otherwise noted:

Setting	Description
Exclude All Devices	This prevents all USB devices attached to the View Client from being forwarded to the View desktop. Most View environments allow some devices, such as a USB keyboard, mouse, or printer to be forwarded, while disabling the forwarding of devices not needed in the View desktops, such as Bluetooth or Smart Card readers. When set to either **true** or **false**, you can still use other policy settings to allow or prevent specific USB devices or families of USB devices to be forwarded. The default setting is **false**, which allows all devices. If set to **true** and the settings are passed to View Client, this setting will always override the settings on View Client.
Exclude Device Family	This prevents an entire family of devices from being forwarded to the View desktop. This setting is used by organizations that want to allow USB devices in general to be forwarded, but prevent specific device families from being forwarded. To prevent View Client attached storage devices and imaging devices from being forwarded, you would enter the following text in the policy setting field: `O:storage;imaging` **Storage** and **imaging** are the US device family names for mass storage devices and imaging devices. When using multiple device names, separate them with a semicolon.

Exclude VID/PID Device	This prevents devices with specific vendor and product IDs from being forwarded to the View desktop. This setting can be used to prevent a specific device from being forwarded, while still allowing other devices within that family. To prevent the sample device referenced earlier in the chapter from being forwarded, you would enter the following text in the policy field:
	
	`O: VID-045E_PID-00F4`
	
	To prevent all devices by that vendor from being forwarded, you would use:
	
	`O: VID-045E_PID-****`

Setting	Description
Include Device Family	This specifies which device families can be forwarded to the View desktop. This setting is often used in tandem with the **Exclude All Devices** setting, to enable the forwarding of specific device families. This setting is configured using the same format as the **Exclude Device Family** setting.
Include VID/PID Device	This allows devices with specific vendor and product IDs to be forwarded to the View desktop. This setting is often used in tandem with the **Exclude All Devices** setting, to enable the forwarding of specific devices or devices from a specific vendor. This setting is configured using the same format as the **Exclude VID/PID Device** policy setting.

Client Downloadable only Settings

The following settings apply to the Client Downloadable only Settings section of the View Agent Configuration ADM template. While these settings exist within the View Agent Group Policy template, they are only applied to the View Clients when they connect. Each of these settings are computer configuration objects and undefined by default unless otherwise noted:

Setting	Description
Allow Audio Input Devices	This allows audio input devices such as microphones to be forwarded to the View desktop. The default setting within View is to allow forwarding, which may not be desirable in environments where client bandwidth is tightly controlled.
Allow Audio Output Devices	This allows audio output devices such as speakers to be forwarded to the View desktop. The default setting within View is to block forwarding, which is recommended, as the local View Client audio output typically does not need to be forwarded to the View desktop.
Allow Auto Device Splitting	This allows the automatic splitting of composite USB devices, such as a headset that has both headphones and a microphone. Splitting these devices allows the components to be managed individually using View Group Policies, rather than as just one USB device. The setting should be enabled if an organization uses the View USB policies to tightly control which USB devices can be forwarded to the desktop.

Setting	Description
Allow HID Bootable	This allows input devices, other than keyboards or mice that are available at boot time, to be forwarded to the View desktop. One common example is a keyboard or mouse that connects using a USB Bluetooth receiver. The default setting within View is to allow forwarding, which ensures that these devices will be available to the desktop at all times.
Allow Other Input Devices	This allows input devices, other than HID-bootable devices or keyboards with integrated pointing devices, to be forwarded to the View desktop. The default setting within View is to allow forwarding, which ensures that these devices will be available to the desktop at all times.
Allow Keyboard and Mouse Devices	This allows keyboards with integrated pointing devices to be forwarded to the View desktop. The default setting within View is to block forwarding, although organizations that use these devices may wish to enable this option.
Allow Smart Cards	This allows Smart Card devices to be forwarded to the View desktop. The default setting within View is to block forwarding, which is recommended unless the Smart Card readers are required for logging in to the View desktop.
Allow Video Devices	This allows video devices to be forwarded to the View desktop. The default setting is to allow forwarding, which is the recommended setting.
Exclude VID/PID Device From Split	This excludes a specific composite USB device from splitting. This setting is used when Auto Device Splitting is allowed, but when you need to prevent specific devices from splitting so that you can manage it as a single USB device. This is also used to enable the policy to use the same format as **Exclude VID/PID Device** policy, which uses the USB vendor and product IDs to identify the device.
Split VID/PID Device	This is used to allow a specific composite USB device to split even when Allow Auto Device Splitting is disabled. This is also used to enable the policy to use the same format as **Exclude VID/PID Device** policy, which uses the USB vendor and product IDs to identify the device.

Agent Configuration settings

The following settings apply to the Agent Configuration section of the View Agent Configuration ADM template:

Policy	Applied to	Default value	Description
AllowDirectRDP	Computers	Enabled	This allows anyone to directly connect to a View desktop using the Microsoft RDP. This setting should be disabled if you wish to prevent direct access to View desktops for security purposes. Do not disable this setting if your environment has View Clients running Mac OSX, as this will prevent their normal View sessions from connecting successfully.
AllowSingle Signon	Computers	Enabled	This enables the pass through of your View Client logon credentials to the View desktop, allowing you to log in automatically. When the setting is disabled, users will be required to log in again at the View desktop Windows login screen.
CommandsTo RunOnConnect	Computers	Undefined	This specifies a list of commands or scripts to be run when a View Client session is first connected. This setting is similar to a traditional login script, and can be used to run commands that are only required when logging in to View desktops.

Policy	Applied to	Default value	Description
CommandsTo RunOnReconnect	Computers	Undefined	This specifies a list of commands or scripts to be run when a View Client session is reconnected. These commands will only run if the user session is still active within the desktop when the user attempts to reconnect.
Connect using DNS Name	Computers	Disabled	This determines whether View Connection Server uses the DNS name instead of the IP address of the View desktop when connecting a View Client. This setting is often enabled when the View Client or View Connection Server cannot connect directly to the View desktop using the IP address.
Connection TicketTimeout	Computers	900 seconds (View default)	This specifies the amount of time in seconds that the View connection ticket is valid. View Clients use a connection ticket for verification and single sign-on when connecting to View Agent on the View desktop. For security reasons, the ticket is valid for a limited amount of time. If authentication of the View Client does not complete within the connection ticket timeout period, the logon session times out.

Policy	Applied to	Default value	Description
CredentialFilter Exceptions	Computers	Undefined	This specifies the executable files that are not allowed to load the agent Credential Filter, preventing those executables from accessing the View Client SSO login information. Filenames must not include a path or suffix, and use a semicolon to separate multiple filenames. This setting is often left at the default to ensure that all components on the desktop function properly.
Disable Time Zone Synchronization	Either	Disabled	This determines whether the time zone of the View desktop is synchronized with that of the View Client. When set to **Enabled**, the setting only applies if the Disable time zone forwarding setting of the View Client Configuration policy is not set to **Disabled**. This setting should be left at the default to ensure that the desktop time matches that of the View Client.
Enable multi-media acceleration	Computers	Enabled	MMR is a Microsoft DirectShow filter that forwards multimedia data from specific codecs on the View desktops directly through a TCP socket to the client, where it is decoded and played. MMR should be disabled if the View Client has insufficient resources or lacks the support for local multimedia decoding.

Policy	Applied to	Default value	Description
Force MMR to use software overlay	Computers	Disabled	Enabling this setting forces MMR to use a software overlay on all monitors, required in multi-monitor systems where only the primary monitor would normally support it. This setting should only be enabled if multi-monitor MMR support is required.
ShowDiskActivityIcon	Computers	Enabled	This displays the disk activity icon in the Windows system tray. This setting should be disabled if the disk activity icon is not required.
Toggle Display Settings Control	Computers	Enabled	This determines whether to disable the **Settings** tab in the Display Windows control panel when a View Client session uses the PCoIP display protocol. This setting should remain enabled to ensure that the View desktops use the display settings as configured in the desktop pool.

Agent Security settings

The following settings apply to the Agent Security section of the View Agent Configuration ADM template:

Policy	Applied to	Default value	Description
Accept SSL encrypted framework channel	Computers	Undefined	This determines whether the View Agent will accept a SSL-encrypted framework channel from the View Client. If all your View Clients have full support for View 5.2 or later, SSL encryption should be enforced to ensure that only SSL connections will be used. If legacy View Clients need to connect to the View infrastructure, the setting may need to remain at the default, which allows those legacy clients to connect without SSL while still allowing SSL for newer clients.

View Client Configuration ADM template

The View Client Configuration ADM template file (`vdm_client.adm`) contains policy settings related to the View Client configuration. Where applicable, the default values for the settings are provided.

Agent Configuration base settings

The following settings apply to the base level of the View Agent Configuration ADM template:

Policy	Applied to	Default value	Description
Delay the start of replications when starting the View Client with Local Mode	Computers	900 seconds	This specifies the number of seconds to delay the start of replication of Local Mode desktop data after the View Client with Local Mode starts. Replication will begin after the delay period, based on the replication schedule as configured in local mode policies in the View Manager Admin console.

Policy	Applied to	Default value	Description
Determines if the View Client should use proxy.pac file	Computers	Disabled	This determines whether View Client uses a **Proxy Auto Config** (**PAC**) file, which is used to control which proxy server is used for a specified URL. This setting is often used when the View Client would otherwise be prevented from accessing View with the default PAC settings.
Disable time zone forwarding	Computers	Undefined	This determines whether time zone synchronization between the View desktop and the View Client is disabled. This setting should be left at the default to ensure that the desktop time matches with that of the View Client.
Disable toast notifications	Users	Undefined	This determines whether to disable toast notifications from View Client, such as the warning when a View Session is nearing timeout. This setting should be left at the default to ensure that the View Client receives important status messages.
Redirect smart card readers in Local Mode	Computers	Enabled	This determines whether Smart Card readers are redirected to the View local mode Desktops. When enabled, the readers will be shared with the client system. This setting should be left at the default if Smart Cards are required for authentication to the local mode desktop.

Policy	Applied to	Default value	Description
Tunnel proxy bypass address list	Computers	Undefined	This specifies a list of tunnel IP addresses, which are IP addresses that should be accessed directly and not through a proxy server. Use a semicolon (;) to separate multiple entries. This setting should be enabled if specific IP addresses should be accessed directly.
URL for View Client online help	Computers	Undefined	This specifies an alternate URL from which View Client can retrieve help pages, such as an internal help desk page. This setting should be used when an organization has created a custom internal web page for View Client support.
Always on top	Users	Enabled	This ensures that the View Client window is always the topmost window, preventing the Windows taskbar from obscuring a full-screen View Client window. The setting should be left at the default value, but can be enabled if the View Client needs ongoing access to locally installed applications.
Default Exit Behavior For Local Mode Desktops	Users	Shutdown	This controls the default exit behavior of View local mode Desktops. The default setting is optimal in most environments, as it ensures that the desktop will be powered down when not in use.

Policy	Applied to	Default value	Description
Don't check monitor alignment on spanning	Users	Disabled	By default, the View Client desktop will not span multiple monitors if the screens do not have the same height (when beside one another) or width (when positioned top and bottom). This setting should be left at the default to ensure that spanned displays display properly.
Enable multi-media acceleration	Users	Undefined	This determines whether MMR is enabled on the View Client. If MMR is disabled in the View Agent, this setting has no effect. This setting is often used when the desktops have MMR enabled, but a subset of the View Clients should not use it due to performance or support concerns.
Enable the shade	Users	Enabled	This determines whether the View Client shade menu bar is visible. This setting should be left at the default to ensure that the View Clients have access to the menu bar commands.
Pin the shade	Users	Enabled	This determines whether the pin on the View Client shade menu bar is enabled, which prevents the menu bar from auto-hiding. This setting should be left at the default to ensure that the share menu is displayed by default.

Scripting Definitions settings

The following settings apply to the Scripting Definitions section of the View Client Configuration ADM template. Each of these settings can be applied to either a user or computer and is undefined by default:

Setting	Description
Connect all USB devices to the desktop on launch	This determines whether all of the available View Client USB devices are connected to the View desktop upon login. This setting should be left at the default, as the View Agent USB policies allow for more granular control over what devices are forwarded to the View desktop.
Connect all USB devices to the desktop when they are plugged in	This determines whether View Client USB devices are connected to the View desktop when they are plugged in. This setting should be left at the default, as the View Agent USB policies allow for more granular control over what devices are forwarded to the View desktop.
DesktopLayout	This specifies the layout of the View Client window when logging into a View desktop. This setting should be configured based on the needs of your environment. In environments where a user will do all of their work within the View Client, the preferred setting is **Full Screen**.
DesktopName to select	This specifies the desktop pool that View Client will use during login. This setting eliminates the need for the View Client to select the desktop pool, and can be useful when an environment has a large number of pools.
Disable third-party Terminal Services plugins	This determines whether the View Client disables third-party Terminal Services plugins. This setting does not affect View-specific plugins, such as USB redirection. Third-party plugins are enabled by default, which ensures that they will be available if required. This setting can be disabled if none of the View Clients use the RDP protocol.
Logon DomainName	This specifies the NetBIOS name of the AD domain that View Client uses during login. This setting can be used to preselect the AD domain name to further simplify the View Client login process.
Logon Password	This specifies the password that View Client will use to log in. The password is stored in plain text by Active Directory, so for security reasons this setting should not be used under most circumstances.

Setting	Description
Logon UserName	This specifies the username that View Client will use to log in. This setting can be used to populate the AD user name to further simplify the View Client login process.
Server URL	This specifies the URL that the View Client will use to log in. This setting can be used to populate the View Connection Server URL name in the View Client, eliminating the need for users to memorize it.
Suppress error messages (when fully scripted only)	This determines whether View Client error messages are hidden during the login process. This setting applies only when the login process is fully scripted, meaning that all information needed to complete the login was provided by a script or Group Policy settings. This setting is only required when your environment includes clients that use fully automated logins.

Security settings

The following settings apply to the Security Settings section of the View Client Configuration ADM template:

Policy	Applied to	Default value	Description
Allow command line credentials	Computers	Enabled	This determines whether the View Client allows user credentials to be provided using command line options. If this setting is enabled, when users execute the View Client using the command line the **SmartCardPIN** and **password** options will not be accepted. The default setting is enabled, which is preferred as it prevents users from creating scripts or batch files to log in that include their password or Smart Card PIN.

Policy	Applied to	Default value	Description
Certificate verification mode	Computer	Undefined	This configures SSL certificate checking settings on the View Client, which affect how the client handles connections to untrusted certificates. The default setting is **Warn But Allow**, which is acceptable for new View deployments, but is not the ideal configuration. This setting allows clients to connect to the untrusted certificate after accepting a warning. Wherever possible, the View Connection and Security Servers should be given trusted certificates, and the policy should be set to **Full Security**. This allows View Clients to connect only to servers with trusted certificates.
Default value of the Log in as current user checkbox	Either	Disabled	This specifies the default value of the View Client **Log in as current user** checkbox, and will override the default value specified during View Client installation. Enabling this setting ensures that SSO will be used to log in to the View Client, unless the user manually unchecks the checkbox.
Display option to Log in as current user	Either	Enabled	This determines whether the View Client **Log in as current user** checkbox is visible. When the checkbox is visible, users can select or deselect it, overriding its default value. When the checkbox is hidden by using the policy, users cannot override its default value. This policy, when combined with the default value of the **Log in as current user** checkbox policy, can be used to force the View Client to use SSO to log in.

Policy	Applied to	Default value	Description
Enable jump list integration	Computer	Enabled	This determines whether a Windows jump list appears in the View Client icon in the taskbar of Windows 7 and later systems. A jump list lets users quickly connect to recent View Connection Server instances and View desktops. The default setting is **Enabled**, which is recommended to enable quick logins to frequently used View Connection Servers.
Enable Single Sign-On for smart card authentication	Computer	Undefined	This determines whether SSO is enabled for Smart Card authentication when logging into the View Client. When single sign-on is disabled, the View Client does not display a custom Smart Card PIN dialog. This setting is often left at the default in order to support SSO when using Smart Cards.
Enable SSL encrypted framework channel	Either	Undefined	This determines whether the View Client will accept an SSL-encrypted framework channel from the View Agent. If all your View Agents have full support for View 5.2 or later, SSL encryption should be enforced. Assuming your View Clients and View desktops are all running the current versions of their respective View software components, this setting should be set to **Enforce** to ensure that only SSL connections are used.
Ignore bad SSL certificate date received from the server	This setting applies to View 4.6 and earlier releases only.		

Policy	Applied to	Default value	Description
Ignore certificate revocation problems	Either	Disabled	This determines whether View Client errors associated with a revoked SSL certificate will be ignored. This setting should be left at the default, which ensures that any errors will be communicated to the View Client.
Ignore incorrect SSL certificate common name (host name field)	These settings apply to View 4.6 and earlier releases only.		
Ignore incorrect usage problems			
Ignore unknown certificate authority problems			
Servers Trusted For Delegation	Computer	Undefined	This specifies which View Connection Servers will accept the user credentials that are passed when a user selects the **Log in as current user** checkbox. If you do not specify any View Connection Server instances, all View Connection Servers will accept the login information. All servers are trusted by default, which is the typical setting for most environments. If you do not want to use SSO in your environment, or for specific View Connection Servers, you can use this setting to restrict it.

RDP settings

The following settings apply to the RDP Settings section of the View Client Configuration ADM template. These settings apply only to View Clients using the RDP protocol. Each of these settings are computer configuration objects, and undefined by default unless otherwise noted:

Setting	Description
Audio redirection	This determines whether the audio played within the View desktop is redirected. By default, the audio is redirected to the View Client, which is the most common setting.
Bitmap cache file size in	This specifies the size of the bitmap cache, in kilobytes or megabytes, to use for specific **bits per pixel** (**bpp**) bitmap color settings. Increasing the bitmap cache provides more storage on the client for image data, potentially reducing the need to retransmit that data when it needs to be displayed again, which may reduce the bandwidth required for the connection. Separate versions of this setting are provided in different unit and bpp combinations. This setting should not be altered without measuring the impact at both the View desktop and the View Client to ensure that performance is not affected.
Bitmap caching/cache persistence active	This determines whether persistent bitmap caching is enabled on the View Client. Persistent bitmap caching can improve performance by reducing the amount of image data that is transmitted between the View desktop and the View Client, but it requires additional disk space. This setting should not be altered without measuring the impact at both the View desktop and the View Client to ensure that performance is not affected.
Color depth	This specifies the display color depth. This setting is often used in limited-use cases when an organization wishes to reduce the display color depth in order to help minimize the bandwidth required by the View Clients.
Cursor shadow	This determines whether a mouse cursor shadow appears. This setting can be disabled as part of a larger effort to optimize View desktop performance. By itself, this setting does not have a significant impact on desktop performance.

Setting	Description
Desktop background	This determines whether the Windows desktop background will appear. This setting helps minimize the bandwidth required to display the Windows desktop background. This setting can be disabled as part of a larger effort to optimize View desktop performance.
Desktop composition	This determines whether Windows desktop composition, also known as **Aero**, is enabled. Unless explicitly required, desktop composition should be disabled to optimize the performance of the View desktop.
Enable audio capture redirection	This specifies whether the default Windows audio input device is redirected. This default setting is **Disabled** , which is the preferred setting unless audio capture is explicitly required.
Enable compression	This determines whether RDP data is compressed, which reduces the amount of bandwidth required for the RDP session. The default setting is **Enabled**, which is the optimal setting for View environments that use the RDP protocol.
Enable Credential Security Service Provider	This specifies whether the View desktop RDP connection uses **Windows Network Level Authentication** (**NLA**), a more secure authentication method for the RDP protocol. This setting must be enabled if Windows is configured to require NLA authentication.
Enable RDP Auto-Reconnect	This determines whether the RDP client component of the View Client will attempt to reconnect to a View desktop after an RDP protocol connection failure. This setting has no effect if the **Use secure tunnel connection to desktop** option is enabled in the View Manager Admin console. The default setting is **Disabled**. If RDP auto-reconnect is desired, this setting should be enabled.
Font smoothing	This determines whether anti-aliasing is used when displaying fonts. This setting causes a slight increase in resources used by the View desktop, but provides for a superior user experience.

Setting	Description
In-memory bitmap cache in KB for 8bpp bitmaps	This specifies the size, in KB, of the RAM bitmap cache for the 8 bpp color setting. This setting should not be altered without measuring the impact at both the View desktop and the View Client, to ensure that performance is not affected.
Menu and window animation	This determines whether animation for menus and windows is enabled. This setting helps eliminate the resources and bandwidth required to render and transmit display updates that involve menu and window animations. This setting can be disabled as part of a larger effort to optimize View desktop performance.
Redirect clipboard	This determines whether the View Client clipboard information is redirected. This setting can be used to control whether or not clipboard data on the View Client can be pasted within the View desktop. Most organizations allow this, but depending on security concerns, you may wish to disable this option.
Redirect drives	This determines whether local disk drives on the View Client are redirected. By default, local drives are redirected from the View Client to the View desktop. If this is not desired, this feature should be disabled.
Redirect printers	This determines whether local printers on the View Client are redirected. By default, local printers are redirected from the View Client to the View desktop. If this is not desired, this feature should be disabled.
Redirect serial ports	This determines whether COM ports on the View Client are redirected. COM port redirection is only required if the View Clients use serial port devices whose input must be forwarded to the View desktop, as found in some **point of sale** (**POS**) platforms.
Redirect smart cards	This determines whether Smart Cards on the View Client are redirected. If Smart Card redirection is not required in your environment, this feature should be disabled.

Setting	Description
Redirect supported plug-and-play devices	This determines whether local **plug-and-play** (**PnP**) and POS devices on the View Client are redirected. This setting is often enabled to allow these devices to connect. If specific restrictions are required, they can be applied using the View Agent USB policies.
Shadow bitmaps	This determines whether Windows bitmaps are shadowed when using the View Client in anything less than full screen mode. Shadow bitmaps may be disabled to reduce View desktop resource utilization as part of a larger effort to optimize View desktop performance. By itself, this setting does not have a significant impact on desktop performance.
Show contents of window while dragging	This determines whether the folder contents appear when users drag a folder to a new location. This setting can be disabled to help reduce View desktop resource utilization as part of a larger effort to optimize View desktop performance. By itself, this setting does not have a significant impact on desktop performance.
Themes	This determines whether Windows themes appear. Windows themes should be disabled unless explicitly required. This ensures that the View desktops have a similar base resource requirement, while still providing optimal performance.
Windows key combination redirection	This determines whether Windows key combinations are applied at the View Client or View desktop. This setting should be configured based on the needs of your organization.

View USB Configuration settings

The View USB Configuration section of the View Client Configuration ADM template contains the same settings as the *View USB Configuration* and *Client Downloadable only Settings* sub sections of the *View Agent Configuration ADM template* section. Refer to that section for information about those settings.

Settings not configurable by Agent

The following settings apply to the Settings not configurable by Agent section of the View Client Configuration ADM template. Each of these settings are computer configuration objects and undefined by default unless otherwise noted:

Policy	Description
Allow Device Descriptor Failsafe behavior	This allows devices on the View Client to be redirected even if config/device descriptors could not be read. This setting is often enabled when troubleshooting issues with devices not redirecting when expected.
Disable Remote Configuration	This prevents the download of View Client configuration settings from the View desktop. View Client configuration settings are downloaded by default. Enabling this setting prevents those configuration settings from being applied, which may be useful for troubleshooting purposes.
Exclude Path	This is used to prevent specific USB hub or port paths from being redirected. To configure this setting, the path must be specified in the following format, where **0** is the USB Bus, and **01** is the USB port number: **bus-0_port-01** All values should be in hexadecimal format. The device numbers can be found in the Windows device properties window under **Details** | **Bus number**, and the port numbers under **Details** | **Location information** for a device plugged into the given port. This setting has very limited use since the USB Bus and port values will vary on a platform-by-platform basis. Wildcards are not supported when using this setting.
Include Path	This is used to identify specific USB hub or port paths that can be redirected. This setting is configured using the same format as that of the **Exclude Path** setting.

View Common Configuration ADM template

The View Common Configuration ADM template file (`vdm_common.adm`) contains policy settings common to all View components. Each of settings in this section are computer configuration settings and undefined by default. Where applicable, the default values for the settings are provided.

Common Configuration Base Settings

The following settings apply to the base level of the View Common Configuration ADM template. These settings apply to all View components.

Policy	Description
Disk threshold for log and events in Megabytes	Specifies the remaining disk space threshold in MB for View server logs and events. When the specified value is met, event logging stops. The default is 200 MB, which should prevent the View server from filling up the disk with event and log data.
Enable extended logging	Determines whether the View log files include trace and debug events. This setting should be enabled when performing advanced troubleshooting of the View environment.
Override the default View Windows event generation	Specifies which View events will be recorded in the Windows event log, overriding the default settings.

Log Configuration Settings

The following settings apply to the Log Configuration Settings section of the View Common Configuration ADM template.

Policy	Description
Number of days to keep production logs	This specifies how long the View log files will be retained. If no value is set, the log files will be kept for seven days. This setting is often increased for auditing and troubleshooting purposes.
Maximum number of debug logs	This specifies the maximum number of debug log files to retain. When a log file reaches its maximum size, a new log file is created. When the number of log files reaches the value of this setting, the oldest log file is deleted. This setting is often increased for auditing and troubleshooting purposes.
Maximum debug log size in Megabytes	This specifies the maximum size in of a debug log in MB. Once this limit is reached, the log file is closed and a new one created. This setting should be left at the default to make the log files easier to review.
Log Directory	This specifies the full path to the log file directory. If the specified location is not writeable, the default location will be used. An extra directory with the client name will be created for client log files. For View servers, this setting is often updated to redirect the log files to a location where they cannot impact the operation of the server itself, such as an alternate hard disk.

Performance Alarm Settings

The following settings apply to the Performance Alarm Settings section of the View Common Configuration ADM template. These settings apply only to View Connection Servers and desktops running the View Agent software:

Policy	Description
CPU and Memory Sampling Interval in Seconds	This sets the CPU and memory polling interval, which determines how often their usage is measured. If the value is set too low, a large amount of log data will be generated. This value should be kept to the minimum required to properly monitor resource utilization.
Overall CPU usage percentage to issue log info	This sets the threshold at which the overall CPU utilization of the system will be logged. When multiple processors are available, this percentage represents the combined usage of those processors. This value should be used to reduce the amount of log data that is recorded by only monitoring CPU utilization above a certain level.
Overall memory usage percentage to issue log info	This sets the threshold at which the overall committed system memory use will be logged. This value should be used to reduce the amount of log data that is recorded by only monitoring memory usage above a certain level.
Process CPU usage percentage to issue log info	This sets the threshold at which the CPU usage of specific processes will be logged. This setting is often useful for troubleshooting purposes.
Process memory usage percentage to issue log info	This sets the threshold at which the memory usage of specific processes will be logged. This setting is often useful for troubleshooting purposes.
Process to check, comma separated name list allowing wild cards and exclusion	A comma-separated list of queries that correspond to the name of one or more processes to be examined by the Process CPU and memory monitoring policies. The following example shows how to specify two different processes: `McTray.exe,winlogon.exe` You can filter the list by using wildcards within each query.

Security Configuration Settings

The following settings apply to the Security Configuration Settings section of the View Common Configuration ADM template:

Policy	Description
Type of certificate revocation check	This sets the type of certificate revocation check that the View servers will perform. The default setting is **WholeChainButRoot**, which checks all the certificates in the certification chain except the root certificate. Any setting is acceptable for this policy, as all will yield the same revocation check results.
Only use cached revocation URLs	This specifies whether certificate revocation checking will use only cached URLs. The default setting is **false**, which allows the service to check both cached and non-cached URLs. This is the preferred setting.
Revocation URL check timeout milliseconds	This specifies the cumulative timeout across all certificate revocation URL wire retrievals in ms. The default is **0**, which instructs the server to use the default value in Windows. The setting is typically not altered.

View Server Configuration ADM template

The View Server Configuration ADM template file (`vdm_server.adm`) contains policy settings related to the View Connection Servers. Each of the settings in this section are computer configuration settings and undefined by default. Where applicable, the default values for the settings are provided.

Server Configuration base template

The following settings apply to the base level of the View Server Configuration ADM template:

Policy	Default value	Description
Recursive Enumeration of Trusted Domains	Enabled	This setting ensures that View Clients will have access to all trusted AD domains when they attempt to log in. This setting should be enabled if the View environment supports users in other domains or other trusted AD forests.

View Persona Management ADM Template

The View Persona Management ADM Template file (`ViewPM.adm`) contains group policy settings used to enable and configure View Persona Management, which is disabled by default. This section expands upon the topics covered in the *Using the Persona Management Group Policies* and *Advanced Persona Management options* sections of *Chapter 7*, *Implementing View Persona Management*. Where applicable, the default values for the settings are provided.

Roaming and Synchronization settings

The **Roaming & Synchronization** Group Policy settings are used to enable and disable View Persona Management, set the location of the remote profile repository, determine which folders and files belong to the user profile, and control how to synchronize folders and files.

The following settings apply to the Roaming and Synchronization Group Policy settings section of the View Persona Management ADM template. Each of these settings are computer configuration objects and undefined by default unless otherwise noted.

Policy	Description
Manage user persona	This setting is used to enable or disable View Persona Management. If this setting is disabled or not configured, and Windows roaming profiles are configured for the user, roaming profiles will be used. When enabled, specify a profile upload interval in minutes, which determines how often changes in the user profile are copied to the remote repository. The default profile upload interval is 10 minutes.
Persona repository location	This setting is used to specify the location of the user profile repository. If you do not specify a location, Persona Management will use the roaming profile folder if it exists. If roaming profile is enabled for the user, but you wish to use a dedicated Persona Management repository, you must check the **Override Active Directory user profile path if it is configured** checkbox to prevent Persona Management from using the existing Windows roaming profile folder. When this setting is enabled, configure the **Share path** with the location of the user profile repository, specified as a UNC path.

Policy	Description
Remove local persona at log off	When enabled, removes the profile data from the View desktop when the user logs off. In addition, you can select the checkbox to **Delete each user's local settings folders when the user profile is removed**. Checking this box removes the `AppData\Local` folder in Windows Vista and newer OSs. For Windows XP, checking the box removes the `Local Settings` folder.
Roam local settings folders	Roams the local settings folder along with the rest of the user profile. By default, the local settings folders are not roamed. This setting is typically enabled when the Virtual Desktop uses applications that require configuration before they can be used, such as Microsoft Outlook.
Files and folders to preload	This specifies a list of files and folders that will be downloaded to the local user profile when the user logs in. Changes in the files are copied to the remote repository as they occur. This setting is used to preload files that will make specific applications run faster. If an application loads poorly due to an on-demand download of user profile data, you can use this policy to preload the data the application is requesting. Do not attempt to preload large amounts of information, or this may lead to unacceptably long login times. This setting should only be used if the data is needed immediately. The setting is configured using folder paths relative to the user profile folder. For example, the following policy setting would preload the `Credentials` folder within the user profile: `AppData\Local\Microsoft\Credentials`
Files and folders to preload (exceptions)	This prevents the specified files and folders from being preloaded by Persona Management. This setting has no effect if the parent folder is not configured to preload. The selected folder paths must reside within the folders that you specify in the **Files and folders to preload** policy or this setting will have no effect. Using the example from the previous setting, to exclude a folder within the `Credentials` folder you would use the following setting: `AppData\Local\Microsoft\Credentials\DontCopyMe`

Policy	Description
Windows roaming profiles synchronization	This specifies a list of files and folders that are managed using Windows roaming profiles and not by Persona Management. Files and folders that are affected by this setting will only be copied from or to the Windows roaming profile share when the user logs on or off. This setting is configured using folder paths relative to the user profile folder. This setting is often used to troubleshoot issues with application behavior when using Persona Management.
Windows roaming profiles synchronization (exceptions)	The selected files and folders are exceptions to the paths that are specified in the **Windows roaming profiles synchronization** setting. This setting has no effect if the parent folder is not configured to use roaming profile synchronization. The selected folder paths must reside within the folders that you specify in the **Windows roaming profile synchronization** policy or this setting will have no effect. This setting is configured using folder paths relative to the user profile folder.
Files and folders excluded from roaming	This specifies a list of files and folders that will not be roamed with the rest of the user profile. The specified files and folders will reside on the View desktop only. This setting is often used when there is no need to retain specific profile data in the Persona Management repository. The setting is configured using folder paths relative to the user profile folder.
Files and folders excluded from roaming (exceptions)	The selected files and folders are exceptions to the paths that are specified in the **Files and folders excluded from roaming** policy. This setting has no effect if the parent folder is not excluded from roaming. The selected folder paths must reside within the folders that you specify in the **Files and folders excluded from roaming** policy or this setting will have no effect. This setting is configured using folder paths relative to the user profile folder.
Enable background download for laptops	This setting allows to download all files in the user profile when a user logs in to a View-managed laptop that has the View Persona Management software installed. Files are downloaded in the background, which ensures that if the laptop is taken offline all of the profile data will be available.

Policy	Description
Folders to background download	The selected folders are downloaded in the background after a user logs in to the View desktop, which allows for a fast login while the Persona Management client continues to work to make the specified folders available locally. This policy is used when data should be available on the View desktop, but it does not need to be available immediately. This setting has a minimal impact on the desktop compared to the **Files and folders to preload** policy. This setting is configured using folder paths relative to the user profile folder.
Folders to background download (exceptions)	The selected files and folders are exceptions to the paths that are specified in the **Folders to background download** policy. This setting has no effect if the parent folder is not configured to background download. The selected folder paths must reside within the folders that you specify in the **Folders to background download** policy or this setting will have no effect. This setting is configured using folder paths relative to the user profile folder.
Excluded processes	The specified processes will be ignored by View Persona Management, which prevents those processes from initiating a download of profile data. This setting is often used to prevent an anti-virus program from initiating the download of profile data while it performs a scan of the profile. Specify the full process name similar to the following: `McTray.exe` Wildcards are not supported in the process names.

Folder redirection settings

Folder Redirection Group Policy settings can be used to redirect user profile folders to a network share. When a folder is redirected, all the data is stored directly on the network share during the user session. Refer to the *Folder redirection* section of *Chapter 7*, *Implementing View Persona Management*, for a list of folders that can be redirected. The following settings apply to the Folder Redirection Group Policy settings section of the View Persona Management ADM template. Each of these settings are computer configuration objects and undefined by default unless otherwise noted.

The following settings allow further customization of the Folder Redirection policies by allowing specific subfolders to be excluded and included from folder redirection:

Policy	Description
Folders excluded from Folder Redirection	The selected file and folder paths will not be redirected to a network share. The setting is configured using folder paths relative to the user profile folder, such as `Desktop\ExcludeMe`. This setting is often enabled when the redirection of a specific folder leads to issues with View desktop performance or application functionality.
Folders excluded from Folder Redirection (exceptions)	The selected file and folder paths within those folders that are excluded will be redirected to a network share. This setting has no effect if the parent folder is not excluded from redirection. This setting is configured using folder paths relative to the user profile folder. Using the previous example, to redirect a folder that lies within a folder that is excluded from redirection, you would use the following setting: `Desktop\ExcludeMe\ButIncludeMe`

Desktop UI settings

The Desktop UI Group Policy settings control which View Persona Management UI objects that users see on their desktops. The following settings apply to the Desktop UI Group Policy Settings section of the View Persona Management ADM template. Each of these settings are computer configuration objects and all UI objects are hidden by default:

Policy	Description
Hide local offline file icon	This hides the offline icon when a user views locally stored files that are part of their profile. This setting should be left disabled unless explicitly required.
Show progress when downloading large files	This displays a progress window on a user's desktop when the client retrieves large files from the Persona Management repository. This setting should be left disabled to prevent the display of unnecessary status windows. When the setting is enabled, it is typically to assess Persona Management throughput.

Policy	Description
Show critical errors to users via tray icon alerts	This displays critical error icon alerts in the Windows desktop tray when Persona Management replication or network connectivity failures occur. This setting should be left disabled unless performing troubleshooting with Persona Management.

Logging Group Policy settings

The Logging Group Policy settings determine the name, location, and behavior of the View Persona Management log files. The following settings apply to the Logging Group Policy settings section of the View Persona Management ADM template. Each of these settings are computer configuration objects:

Policy	Default value	Description
Logging filename	`VMWVvp.txt`	The full pathname and filename of the local View Persona Management log file. By default the logs are written to: `Application Data\VMware\VDM\logs\ VMWVvp.txt` When changing the default value, you must provide a filename and a path to a folder on the local system. UNC paths are not supported.
Logging destination	Log file only	This instructs Persona Management to log messages to the log file, the debug port, or both destinations. This setting should be left at the default unless performing advanced troubleshooting of Persona Management.
Logging flags	Error and information only	This determines what types of Persona Management messages to log. This setting should be left at the default unless performing advanced troubleshooting of Persona Management. If that is the case, debug messages should be logged as well.
Debug flags	None	This determines the types of Persona Management debug messages to log. This setting should be left at the default unless performing advanced troubleshooting of Persona Management. If that is the case, you may wish to enable all options for debug logging.

View PCoIP Session Variables ADM template

The View PCoIP Session Variables ADM template file (`pcoip.adm`) contains policy settings related to the PCoIP display protocol. Each of the settings in this section are computer configuration settings and undefined by default. Wherever applicable, the default values for the settings are provided.

Refer to *Advanced Details about Key Horizon View Features* for examples of how this policy can be used to customize the View PCoIP settings.

The settings can be configured as not overridable or overridable by an administrator. If a policy is overridable, users with local administrator permissions can make changes to the settings. To configure the policies, simply choose them from the following subfolders based on which option you prefer:

- `Overridable Administrator Defaults`
- `Not Overridable Administrator Settings`

Each subfolder contains the same policy items; the only difference is whether or not they can be overridden.

PCoIP Session Variables base settings

The General Session Variable Group Policy settings are used to configure general session characteristics, such as PCoIP image quality, USB devices, and network ports.

The following settings apply to the base level of the View Common Configuration ADM template. Each of these settings are computer configuration objects and undefined by default unless otherwise noted.

Policy	Description
Configure clipboard redirection	This determines whether the View Client clipboard information is redirected. This setting can be used to control whether or not clipboard data on the View Client can be pasted within the View desktop. The default setting is **Enabled client to server only**, which you may wish to disable depending on security concerns.

Policy	Description
Configure PCoIP client image cache size policy	This controls the size of the View Client PCoIP image cache. The View Client uses image caching to store portions of the display that were previously transmitted, which reduces the amount of data that is retransmitted between the View desktop and the client. The default setting is 250 MB, but can be increased assuming the View Client has sufficient available resources.
Configure PCoIP event log verbosity	This specifies the PCoIP event log verbosity, which affects how much data is recorded in the logs. The default setting is 2, but may be increased to 3 when performing advanced PCoIP troubleshooting.
Configure PCoIP image quality levels	This controls how PCoIP renders images during periods of network congestion. The Minimum Image Quality, Maximum Initial Image Quality, and Maximum Frame Rate values work in tandem to provide granular control in environments where network bandwidth is limited. Refer to *Advanced Details about Key Horizon View Features* for examples of how this setting is configured.
Configure PCoIP session encryption algorithms	Controls the encryption algorithms that will be advertised by the View PCoIP Client during session negotiation. The default settings enable the `Salsa20-256round12` and `AES-128-GCM` algorithms. This policy is typically updated to enable or disable a particular algorithm for security purposes.
Configure PCoIP USB allowed and unallowed device rules	This specifies which USB devices are allowed or not allowed for PCoIP sessions that are initiated from a zero client running the Teradici firmware. By default all devices are allowed, which ensures that all devices will be supported. Devices are specified using class or device IDs as described in the View Agent USB policy settings.

Policy	Description
Configure PCoIP virtual channels	This specifies which virtual channels can or cannot operate over View PCoIP Client sessions. Virtual channels are used for operations such as USB or clipboard redirection. All channels are enabled by default. When specifying channels you must separate them with a |, and if the channel name contains a | or \ you must place a \ before them. The following are examples of possible policy settings based on these rules: • `sample1|sample2` • `sam|ple3\\sample4` This setting is often left at the default unless you have specific security policies that you must enforce.
Configure SSL connections to satisfy security tools.	This specifies how the View Client SSL session negotiation connections are established. This setting is often enabled when testing the View desktops with port scanner utilities. This setting should otherwise be left at the default.
Configure the Client PCoIP UDP port	This specifies the UDP client port that is used by software-based View PCoIP Clients. The UDP port value specifies the base UDP port to use. The UDP port range value determines how many additional ports to try if the base port is not available. The base port is *50002* and the port range is *64* by default. These values should not be changed unless the client ports are being used for other applications.
Configure the maximum PCoIP session bandwidth	Specifies the PCoIP session maximum bandwidth, in kb per second. The bandwidth includes the imaging, audio, virtual channel, USB, and control PCoIP traffic. The default value is 90000 kilobits per second, which can be adjusted as needed based on bandwidth availability. Refer to *Advanced Details about Key Horizon View Features* for examples of how this setting is configured.
Configure the PCoIP session audio bandwidth limit	This specifies the maximum PCoIP session bandwidth that can be used for audio (sound) playback. The default value is **500** kilobits per second, which provides acceptable quality in most cases.

Policy	Description
Configure the PCoIP session bandwidth floor	This specifies a lower limit, in kb per second, for the bandwidth reserved by the PCoIP session. The default value is **0** (no minimum). The default setting is often changed when you wish to provide a guaranteed minimum bandwidth to View PCoIP Clients.
Configure the PCoIP session MTU	This specifies the **Maximum Transmission Unit** (**MTU**) size in bytes (B) for UDP packets within PCoIP session. The default value is **1300** bytes, which is not typically changed.
Configure PCoIP transport header	This configures the View PCoIP transport header to support QoS, which is used by supported network devices to prioritize traffic when experiencing network congestion. The default setting is enabled, which ensures that QoS enables networking equipment to prioritize View Client traffic (when configured to do so).
Configure the TCP port to which the PCoIP host binds and listens	This specifies which TCP server port will be bound to by software-based PCoIP hosts. The default TCP port is **4172** and the port range is **1**, neither of which is typically changed.
Configure the UDP port to which the PCoIP host binds and listens	This specifies which UDP server port will be bound to by software PCoIP hosts. The default UDP port is **4172** and the port range is **10**, neither of which are typically changed.
Disable sending CAD when users press Ctrl + Alt + Del	When this policy is enabled, users must press *Ctrl* + *Alt* + *Insert* rather than *Ctrl* + *Alt* + *Del* (CAD) to send a **Secure Attention Sequence** (**SAS**) to the desktop during a View PCoIP session. *Ctrl* + *Alt* + *Del* is enabled by default. Enable this setting if you want CAD sequences to be sent to View desktop only.
Enable access to a PCoIP session from a vSphere console	Allows a vSphere Client console to display an active PCoIP session, and allows input to be sent to the View desktop. The default setting is disabled, which is recommended to prevent vSphere Clients from interrupting with active View Client sessions.
Enable Right SHIFT behavior when a PCoIP client is connected	Substitute the Right *Shift* key with a Left *Shift* key, which allows the Right *Shift* key to function properly when using RDP within a PCoIP session. The default setting is disabled, which is the ideal setting unless you expect RDP sessions to be launched from within a PCoIP session.

Policy	Description
Enable the FIPS 140-2 approved mode of operation	This enables the use of **Federal Information Processing Standard** (**FIPS**) 140-2 approved cryptographic algorithms and protocols when establishing a remote PCoIP connection. Enabling this setting will override the disabling of `AES128-GCM` encryption. The default setting is disabled. This policy should be enabled for environments that require this level of encryption.
Enable/disable audio in the PCoIP session	Enables or disables audio within PCoIP sessions. Both endpoints must have audio enabled or the policy will have no effect. Audio is enabled by default, but can be disabled for those environments that do not require audio within their PCoIP sessions.
Enable/disable microphone noise and DC offset filter in PCoIP session	Enables or disables the microphone noise and DC offset filter for microphone input during PCoIP sessions. The default setting is enabled, which is recommended to ensure optimal quality of microphone input.
Turn off Build-to-Lossless feature	This disables the build-to-lossless feature of the PCoIP protocol. Build-to-lossless uses PCoIP bandwidth to build the display to a lossless state over time, meaning that the View Client display represents the exact state of the remote View desktop display. This default setting is **Enabled**, which is the optimal setting for environments with bandwidth constraints. Build-to-lossless is often used for desktops that require maximum image quality, such as those in the medical or design fields.
Turn on PCoIP user default input language synchronization	This synchronizes the default input language for the user in the PCoIP session with the default input language of the PCoIP client. The default setting is **Disabled**, which ensures that no errors will be encountered if a client were to attempt to connect to a View desktop that does not have the appropriate input language installed. This setting should only be enabled if the View desktops support all input languages that could possibly be required.

Policy	Description
Use alternate key for sending Secure Attention Sequence	This specifies an alternate key to be used when sending a Secure Attention Sequence (SAS). The default setting uses the *Ctrl* + *Alt* + *Insert* key sequence. This setting is typically updated when *Ctrl* + *Alt* + *Insert* SAS is already in use for some other operation.
Use enhanced keyboard on Windows client if available	This restricts direct keyboard sequences to the guest operating system in PCoIP desktop sessions. Direct keyboard sequences are multiple key sequences such as *Ctrl* + *C*. The default setting is disabled, but may be enabled if you wish for the direct keyboard sequences to be sent to both the View Client host and the View desktop. This setting has no effect if the VMware keyboard filter driver (`vmkbs.sys`) is not installed on the View Client, or if the user running the client does not have local administrator permissions. The driver is installed automatically when installing the View Client with Local Mode software.

Summary

In this chapter we learned about the different Group Policy templates that are included in VMware Horizon View.

We discussed what Group Policy loopback processing is, and why an organizations might need to use it as part of their View Deployment. We also went into detail about each of the View Group Policy templates, including what they are used for and the meaning of each policy setting.

In *Chapter 14*, *Managing View with PowerCLI* (available for free download from the Packt Publishing website), we will discuss how to perform View administrative tasks using VMware PowerCLI, which extends the capabilities of Windows PowerShell to provide support for View.

Advanced Details about Key Horizon View Features

This appendix contains additional information about a number of key topics discussed throughout the book.

In this appendix you will learn:

- How to configure the advanced View event-logging options
- How to configure the advanced vCenter provisioning options
- How to configure the View Local Mode policy settings
- How to customize the View desktop names
- How to configure the most common PCoIP optimization settings

The information provided in this chapter will help ensure that you are familiar with some of the most common advanced topics concerning the implementation of VMware Horizon View.

View Event configuration options

Chapter 2, *Implementing VMware Horizon View Connection Server*, discussed the setup of the Connection Server, including the configuration of the View events database. This section will discuss advanced options that exist for displaying and logging View event data. To access these advanced options, log in to the View Manager Admin console and open the **Event Configuration** page from **View Configuration**.

The following settings can be configured to further customize how View event data is captured and displayed. The settings are arranged into two sections on the **Event Configuration** page:

- **Event Settings**:
    - **Show events in View Administrator for**: This setting controls how long events are displayed in the console. Options are 1, 2, or 3 weeks, or 1, 2, 3, or 6 months.
    - **Classify events as new for**: This setting controls how long events are classified as new in the console. Options are 1, 2, or 3 days.
- **Syslog**:
    - **Send to syslog servers**: This setting enables you to send View event data to the specified syslog servers. This data will be sent unencrypted, so consult with your organization's security policy before enabling this feature.
    - **Log to file**: This checkbox enables you to write syslog data to a file rather than sending it to syslog servers directly.
    - **Copy to location**: This setting is used when the **Log to file** option is checked. Syslog data will be saved to the specified location. Required information includes the UNC path, username, password, and domain. This data is sent and recorded unencrypted, so consult with your organization's security policy before enabling this feature.

vCenter provisioning options

Chapter 2, *Implementing VMware Horizon View Connection Server*, discussed the setup of the View Connection Server, including the linking of a vCenter Server to View. This section will discuss advanced options that exist for vCenter Server provisioning settings. To access these advanced options, log in to the View Manager Admin console and open the vCenter Servers page by navigating to **View Configuration | Servers**.

Highlight the vCenter Server you wish to edit and click on the **Edit** button to open the **Edit vCenter Server** window. Click on the **Edit** button under vCenter Server settings to access the **Advanced Settings** page.

The following settings can be updated to increase or decrease the rate at which View provisions, operates, and maintains desktops when using vCenter:

- **Max concurrent vCenter provisioning operations**: The maximum number of concurrent requests that View will request to vCenter for full-clone desktop virtual machine provision or delete operations.
- **Max concurrent power operations**: The maximum number of concurrent suspend, resume, power on, or power off operations that View will request to vCenter for both full-clone and linked-clone desktops.
- **Max concurrent View Composer maintenance operations**: The maximum number of concurrent refresh, recompose, and rebalance operations that View will request to vCenter and View Composer for linked-clone desktops.
- **Max concurrent View Composer provisioning operations**: The maximum number of concurrent requests that View will request to vCenter and View Composer for linked-clone desktop virtual machine provision or delete operations.

Any increments to the default values of these settings should be carefully monitored to ensure that the vCenter Servers and View Composer Servers can handle the additional load and that the performance of the desktops is not affected during the operation.

View Local Mode desktop policy settings

Chapter 4, *Implementing VMware View Transfer Server*, discussed how to implement a View Transfer Server, which enables desktops to be downloaded and run directly on the View Client machine. The Local Mode policies are used to control a number of key Local Mode options. To access these settings, log in to the View Manager Admin console and open the **Policies - Global Policies - Edit Local Mode Policies** page.

The following is a detailed explanation of each setting:

- **Local Mode**: If this is set to **Allow**, desktops can be checked out by clients using the View Client with Local Mode. If this value is changed while desktops are checked out, those desktops will not be able to be used.
- **User-initiated rollback**: If this is set to **Allow**, a user can discard the current Local Mode desktop and revert to the current remote version.
- **Max time without server contact**: If a Local Mode client does not contact the View environment within the specified time, a warning will be displayed and the desktop will be suspended. This can be set to **Unlimited**, ensuring that desktops will never suspend.

- **Target replication frequency**: This setting specifies how often the Local Mode desktop will attempt to replicate its current state to the remote version, measured from the time the previous replication began. If this is set to **No replication**, and user-initiated replication is allowed, users may initiate replication themselves using the View Client. This can be specified in days, hours, or minutes, and is set to **No replication** by default.
- **User deferred replication**: If this is enabled, allows users to pause an ongoing replication for 2 hours.
- **Disks replicated**: This setting configures which disks are replicated. Available options are **Persistent disks** (default), **OS disks**, or **OS and persistent disks**.
- **User-initiated check-in**: If this is set to **Allow**, users may check in their Local Mode desktops.
- **User-initiated replication**: If this is set to **Allow**, users may initiate replication of their desktop manually.

Changes made to the Global Policies will apply to all desktop pools by default. If you wish to configure policies on a specific desktop pool, click on the target pool in the **Inventory – Pools** page to open the **Settings** page for that pool. Click on the **Policies** tab, and make the changes as required.

Naming View desktops

Chapter 8, *Creating VMware Horizon View Desktop Pools*, discussed how to create View desktop pools. This section contains additional information about how to customize the name of the desktops that View creates.

Naming patterns

Naming patterns are used when you wish View to automatically name desktops based on rules that you provide. For example:

- `Finance{n}`: Every desktop name will start with `Finance` and be appended with a unique number such as `Finance1`, `Finance2`, and so on.
- `Finance{n:fixed=4}`: Every desktop name will start with `Finance`, and be appended with a number that must have four digits, even if leading zeroes are needed. For example, `Finance0001`, `Finance0002`, and so on.

Specifying names manually

When creating a desktop pool, you have the option of providing a list of your own desktop names by selecting the **Specify names manually** radio button in the **Provisioning Settings** section of the **Add Pool** wizard. Once the **Specify names manually** radio button has been selected, click on the **Enter names** button. If you are creating a dedicated pool, you can also specify the owner of the desktop.

You must type or paste in the list of desktop names in the **Enter Desktop Names** window in the format of one name per line. If you wish to specify the desktop owner, append it to the desktop name with the username in the format `FullDomainName\UserLoginID`. For example, `FinanceJV,vjason.local\jventresco`. If the pool is configured for floating assignment, the username assignment will be ignored.

Configuring View PCoIP settings

The amount of time you spend customizing PCoIP settings will vary from one environment to the next. Factors such as the applications that users employ, the number of WAN-connected clients, the number of remote clients, and even LAN configuration can each impact the default settings. This section will focus on a small subset of settings that you are most likely to update as you adjust PCoIP settings. PCoIP settings are set using the `pcoip.adm` group policy template.

PCoIP image quality levels

Configuring PCoIP bandwidth quality levels can help you deliver consistent image quality even while there is network congestion. The following is an example of settings that I have used in the past that should deliver acceptable image quality for task and knowledge workers.

As described in *Chapter 13*, *Implementing VMware Horizon View Group Policies*, you should use the **Not Overridable Administrator Defaults** section of the PCoIP group policy template to configure the PCoIP settings. This ensures that the settings cannot be overridden by the View Client.

The recommended changes include the following settings. The definitions for each policy option can be found in *Chapter 13*, *Implementing VMware Horizon View Group Policies*. Additional information is provided where applicable:

- **Configure PCoIP image quality levels**: **Enabled**
- **Set the Minimum Image Quality value**: **50** (the default)
- **Set the Maximum Initial Image Quality value**: **70**
- **Set the maximum Frame Rate value**: **24**
- **Use image settings from client if available**: **Disabled**
- **Turn off Build-to-Lossless**: **Enabled**

It is not recommended you employ these settings for users who use graphics-intensive applications, as they may reduce the quality of the display to a level that is unacceptable to the user.

Maximum PCoIP session bandwidth

The maximum PCoIP session bandwidth settings should be used when there are endpoints through which your View clients connect that have specific bandwidth limitations. Examples of this include remote clients connecting over WAN links or Internet connections. This setting is typically not used when all the View clients are on the local LAN, unless endpoints of that LAN require it.

The following is an example of settings for an organization that has 20 users in a remote office and 1 GB (1,048,576 KB) of bandwidth available for View Client connections.

The default maximum PCoIP session bandwidth is 90,000 KB, which when multiplied by 20 client connections would almost double the capacity of our 1 GB link (20 connections * 90,000 KB = 1,800,000 KB). If we reduce the maximum to 47,000 KB, we will ensure that our View client connections use no more than 90 percent of the bandwidth available. For example, 47,000 KB (maximum per-client bandwidth) * 20 (number of clients) = 940,000 KB = 0.896 GB

Summary

In this chapter, we learned additional information about some of the topics discussed throughout the book including View event log configuration options, vCenter provisioning options, View Local Mode desktop policy settings, customizing virtual desktop naming parameters, and PCoIP protocol settings.

Index

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z

Thank you for buying

Implementing VMware Horizon View 5.2

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, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike.
For more information, please visit our website: `www.packtpub.com`.

About Packt Enterprise

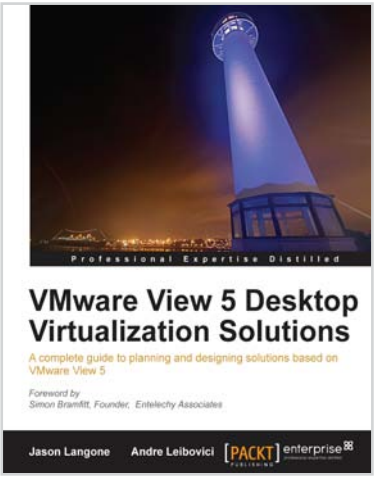
In 2010, Packt launched two new brands, Packt Enterprise and Packt Open Source, in order to continue its focus on specialization. This book is part of the Packt Enterprise brand, home to books published on enterprise software – software created by major vendors, including (but not limited to) IBM, Microsoft 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, 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



VMware View 5 Desktop Virtualization Solutions

ISBN: 978-1-84968-112-4 Paperback: 288 pages

A complete guide to planning and designing solutions based on VMware View 5

1. Written by VMware experts Jason Langone and Andre Leibovici, this book is a complete guide to planning and designing a solution based on VMware View 5
2. Secure your Visual Desktop Infrastructure (VDI) by having firewalls, antivirus, virtual enclaves, USB redirection and filtering and smart card authentication
3. Analyze the strategies and techniques used to migrate a user population from a physical desktop environment to a virtual desktop solution

---

VMware ThinApp 4.7 Essentials

ISBN: 978-1-84968-628-0 Paperback: 256 pages

Learn how to quickly and efficiently virtualize your applications with ThinApp 4.7

1. Practical book which provides the essentials of application virtualization with ThinApp 4.7
2. Learn the various methods and best practices of application packaging and deployment
3. Save money and time on your projects with this book by learning how to create portable applications

[PACKT] enterprise
professional expertise distilled
PUBLISHING

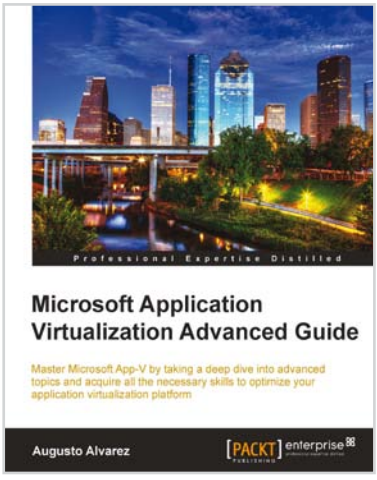
Getting Started with Microsoft Application Virtualization 4.6

ISBN: 978-1-84968-126-1 Paperback: 308 pages

Virtualize your application infrastructure effi ciently using Microsoft App-V Augusto Alvarez P U

1. Publish, deploy, and manage your virtual applications with App-V
2. Understand how Microsoft App-V can fit into your company.
3. Guidelines for planning and designing an App-V environment.
4. Step-by-step explanations to plan and implement the virtualization of your application infrastructure

---



Microsoft Application Virtualization Advanced Guide

ISBN: 978-1-84968-448-4 Paperback: 474 pages

Master Microsoft App-V by taking a deep dive intoadvanced topics and acquire all the necessary skills tooptimize your application virtualization platform

1. Understand advanced topics in App-V; identify some rarely known components and options available in the platform
2. Acquire advanced guidelines on how to troubleshoot App-V installations, sequencing, and application deployments
3. Learn how to handle particular applications, adapting companys' policies to the implementation, enforcing application licenses, securing the environment, and so on