

Infrastructure Planning  
and Design

Windows Server® 2008 R2 Remote Desktop Services

Version 2.2

Published: July 2008

Updated: February 2011

For the latest information, please see [www.microsoft.com/ipd](http://www.microsoft.com/ipd)

Copyright © 2011 Microsoft Corporation. This documentation is licensed to you under the Creative Commons Attribution License.  To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/us/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.  When using this documentation, provide the following attribution: Infrastructure Planning and Design is provided with permission from Microsoft Corporation.

This documentation is provided to you for informational purposes only, and is provided to you entirely "AS IS". Your use of the documentation cannot be understood as substituting for customized service and information that might be developed by Microsoft Corporation for a particular user based upon that user’s particular environment. To the extent permitted by law, MICROSOFT MAKES NO WARRANTY OF ANY KIND, DISCLAIMS ALL EXPRESS, IMPLIED AND STATUTORY WARRANTIES, AND ASSUMES NO LIABILITY TO YOU FOR ANY DAMAGES OF ANY TYPE IN CONNECTION WITH THESE MATERIALS OR ANY INTELLECTUAL PROPERTY IN THEM.

Microsoft may have patents, patent applications, trademarks, or other intellectual property rights covering subject matter within this documentation. Except as provided in a separate agreement from Microsoft, your use of this document does not give you any license to these patents, trademarks or other intellectual property.

Information in this document, including URL and other Internet website references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, email addresses, logos, people, places and events depicted herein are fictitious.

Microsoft, Active Directory, Excel, Forefront, Hyper-V, Visio, Windows, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries and regions.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

You have no obligation to give Microsoft any suggestions, comments or other feedback ("Feedback") relating to the documentation. However, if you do provide any Feedback to Microsoft then you provide to Microsoft, without charge, the right to use, share and commercialize your Feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the Feedback. You will not give Feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your Feedback in them.

Contents

[The Planning and Design Series Approach 1](#_Toc283971464)

[Introduction to the Windows Server 2008 R2 Remote Desktop Services Guide 2](#_Toc283971465)

[Step 1: Determine the Scope of the Remote Desktop Services Project 9](#_Toc283971466)

[Step 2: Determine How Workloads Will Be Delivered 10](#_Toc283971467)

[Step 3: Determine the Number of RD Session Host Server Farms 17](#_Toc283971468)

[Step 4: Map Workloads and Users to Farms 19](#_Toc283971469)

[Step 5: Design the Farms 20](#_Toc283971470)

[Step 6: Determine Where to Store User Data 23](#_Toc283971471)

[Step 7: Size and Place the Remote Desktop Services Role Services for the Farms 26](#_Toc283971472)

[Step 8: Secure the Communications 35](#_Toc283971473)

[Step 9: Design VDI and the RD Virtualization Host Role 37](#_Toc283971474)

[Conclusion 39](#_Toc283971475)

[Appendix A: Job Aids 41](#_Toc283971476)

[Appendix B: Server Performance Analyzing and Scaling 46](#_Toc283971477)

[Appendix C: IPD in Microsoft Operations Framework 4.0 57](#_Toc283971478)

[Appendix D: Remote Desktop Services in Microsoft Infrastructure Optimization 58](#_Toc283971479)

[Version History 59](#_Toc283971480)

[Acknowledgments 60](#_Toc283971481)

# 

# The Planning and Design Series Approach

This guide is one in a series of planning and design guides that clarify and streamline the planning and design process for Microsoft® infrastructure technologies.

Each guide in the series addresses a unique infrastructure technology or scenario. These guides include the following topics:

* Defining the technical decision flow (flow chart) through the planning process.
* Describing the decisions to be made and the commonly available options to consider in making the decisions.
* Relating the decisions and options to the business in terms of cost, complexity, and other characteristics.
* Framing the decision in terms of additional questions to the business to ensure a comprehensive understanding of the appropriate business landscape.

The guides in this series are intended to complement and augment the product documentation.

## Benefits of Using This Guide

Using this guide will help an organization to plan the best architecture for the business and to deliver the most cost-effective Remote Desktop Services technology.

Benefits for Business Stakeholders/Decision Makers:

* Most cost-effective design solution for an implementation. Infrastructure Planning and Design (IPD) eliminates over-architecting and overspending by precisely matching the technology solution to the business needs.
* Alignment between the business and IT from the beginning of the design process to the end.

Benefits for Infrastructure Stakeholders/Decision Makers:

* Authoritative guidance. Microsoft is the best source for guidance about the design of Microsoft products.
* Business validation questions to ensure the solution meets the requirements of both business and infrastructure stakeholders.
* High integrity design criteria that includes product limitations.
* Fault-tolerant infrastructure, where necessary.
* Proportionate system and network availability to meet business requirements. Infrastructure that is sized appropriately to meet business requirements.

**Benefits for** Consultants or Partners:

* Rapid readiness for consulting engagements.
* Planning and design template to standardize design and peer reviews.
* A “leave-behind” for pre- and post-sales visits to customer sites.
* General classroom instruction/preparation.

Benefits for the Entire Organization:

Using this guide should result in a design that will be sized, configured, and appropriately placed to deliver a solution for achieving stated business requirements, while considering the performance, capacity, manageability, and fault tolerance of the system.

# Introduction to the Windows Server 2008 R2 Remote Desktop Services Guide

This guide leads the reader step by step through the process of planning a Windows Server® 2008 R2 Remote Desktop Services infrastructure. The guide addresses the following fundamental decisions and tasks:

* Identifying which applications and which desktops will be delivered by Remote Desktop Services.
* Determining whether to deliver desktops by using session-based presentation virtualization or Virtual Desktop Infrastructure (VDI).
* Designing the infrastructure needed to use Remote Desktop Services to serve the selected applications and desktops.

Before starting the technical design, it’s very important to fully understand the business objectives for the project. The following questions should be asked:

* What benefits does the business expect to achieve through the use of Remote Desktop Services?
* What is the value of those benefits and, therefore, the cost case for using Remote Desktop Services to deliver those benefits?
* Is the cost justification jointly entered into by more than one business group and, if so, do they depend on the success of the project for their relationship with each other?

The business objectives should be prioritized right at the start of the project so that they are clearly understood and agreed upon between IT and the business. This is because some workloads will most likely be unsuitable for delivery by Remote Desktop Services. Any changes to those workloads in order to make them suitable will incur cost and should be discussed with the business so that the additional costs can be understood and the best business decision can be reached.

## What’s New in Windows Server 2008 R2 Remote Desktop Services

In Windows Server 2008 R2, Remote Desktop Services (formerly named Terminal Services) has been expanded to provide a brokering service that connects clients to their virtual desktops in a VDI. The server role and all role services have been renamed. The following table lists both the new and former names of each role service.

Table 1. Remote Desktop Services Role Services Name Changes

| Remote Desktop Services role service | Terminal Services role service |
| --- | --- |
| Remote Desktop Session Host (RD Session Host) | Terminal Server |
| Remote Desktop Licensing (RD Licensing) | Terminal Services Licensing (TS Licensing) |
| Remote Desktop Gateway (RD Gateway) | Terminal Services Gateway (TS Gateway) |
| Remote Desktop Connection Broker (RD Connection Broker) | Terminal Services Session Broker (TS Session Broker) |
| Remote Desktop Web Access (RD Web Access) | Terminal Services Web Access (TS Web Access) |

Remote Desktop Virtualization Host (RD Virtualization Host) is a new Remote Desktop Services role service included with Windows Server 2008 R2. RD Virtualization Host requires and integrates with the Hyper-V™ role to provide VDI virtual machines (VMs) that can be used as personal virtual desktops or virtual desktop pools.

The Infrastructure Planning and Design Guide for Windows Server 2008 Terminal Services remains available at <http://go.microsoft.com/fwlink/?LinkID=160987> for use in designing a Terminal Services infrastructure.

## Assumptions

The content in this guide assumes that the reader is familiar with Remote Desktop Services technology and is planning an implementation of servers, farms, or both in Windows Server 2008 R2.

The goal of this Windows Server 2008 R2 Remote Desktop Services Infrastructure Planning and Design Solution Accelerator is to guide the planner through the information gathering, decisions, options, and tasks required to create and design a Remote Desktop Services infrastructure. The objective is an infrastructure that is sized, configured, and appropriately placed in order to deliver the stated business benefits, while considering the end-user experience, security, manageability, performance, capacity, and fault tolerance of the system. The guide addresses the scenarios most likely to be encountered by someone designing a Remote Desktop Services infrastructure. Customers should consider having their architecture reviewed by Microsoft Customer Service and Support prior to implementation as this organization can best comment on the supportability of a particular design.

The primary components of Windows Server 2008 R2 Remote Desktop Services are shown in Figure 1.

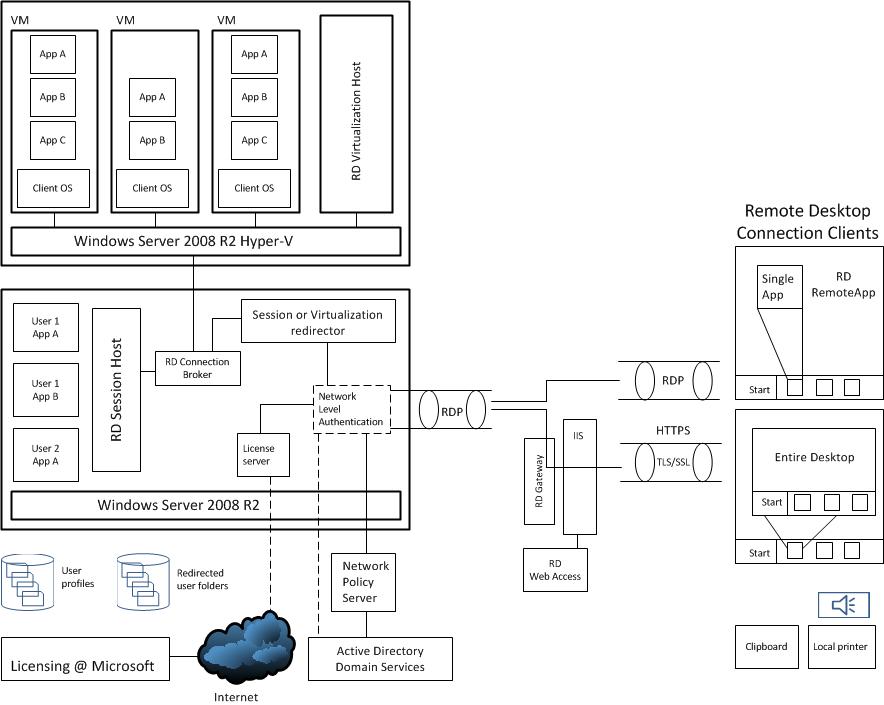


Figure 1. Windows Server 2008 R2 Remote Desktop Services architecture

This diagram illustrates the relationship between the components that can work together to publish applications and desktops through Remote Desktop Services. They are shown together in one possible implementation for illustrative purposes, but other configurations are possible.

Remote Desktop Services provides the following capabilities:

* Centralized systems that host multiple user sessions, all running on the Windows Server 2008 R2 server operating system. All processing is done on the host systems and the user sessions are isolated from each other.
* Delivery of virtual desktops to users from VMs running on Windows Server 2008 R2 Hyper-V. This VDI is delivered by the RD Virtualization Host role service. Connections between clients and VM-based desktops are made by RD Connection Broker.

In both cases, presentation virtualization is used to deliver the desktop or applications to the user. Only the presentation information (such as keyboard and mouse inputs and video updates) is sent between the client and the host systems. The client can be a full Windows®-based workstation, a Windows-based remote desktop device, or a thin client.

Servers running Remote Desktop Services are usually implemented in farms in order to provide high scale and fault tolerance. An RD Session Host server farm is a group of RD Session Host servers that publish an identical set of workloads. In a well-planned solution, all the RD Session Host servers in the farm should be designed to deliver an equivalent experience, hosting the same applications, and in the same configuration, even though there may be differences in the hardware from server to server.

The farm can be load balanced so that users connect to a common name; they are then connected to one of the member servers of the farm.

This guide addresses the following decisions and/or activities that need to occur in preparing for Windows Server 2008 R2 Remote Desktop Services. The nine steps below represent the most critical design elements in a well-planned Windows Server 2008 Remote Desktop Services design:

* Step 1: Determine the Scope of the Remote Desktop Services Project
* Step 2: Determine How Workloads Will Be Delivered
* Step 3: Determine the Number of RD Session Host Server Farms
* Step 4: Map Workloads and Users to Farms
* Step 5: Design the Farms
* Step 6: Determine Where to Store User Data
* Step 7: Size and Place the Remote Desktop Services Role Services for the Farms
* Step 8: Secure the Communications
* Step 9: Design VDI and the RD Virtualization Host Role

Some of these items represent decisions that must be made. Where this is the case, a corresponding list of common response options is presented.

Other items in this list represent tasks that must be carried out. These items are addressed because their presence is significant in order to complete the infrastructure design.

Figure 2 provides a graphical overview of the steps in designing a Windows Server 2008 R2 Remote Desktop Services infrastructure.

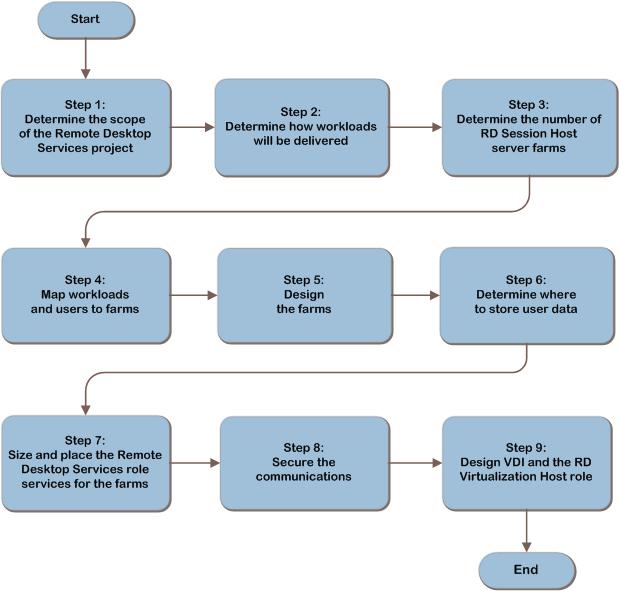


Figure 2. The Windows Server 2008 R2 Remote Desktop Services infrastructure decision flow

## Applicable Scenarios

This guide addresses considerations that are related to planning and designing the necessary components for a successful Windows Server 2008 R2 Remote Desktop Services infrastructure. The scenarios considered in creating this guide include:

* **Organizations planning to centralize their desktop environments.** This enables desktops to be managed centrally in a central data center or a few regional data centers.
* **Organizations that are combining their computing resources and users after a company merger or acquisition.** An alternate rich client desktop or application may be delivered alongside their existing desktops so that they have access to desktops and applications from both parts of the merged enterprise. This can be used to significantly accelerate the integration of the acquired company’s systems. In this case, security and directory issues around application access will of course also need to be addressed, but that is beyond the scope of this document.
* **Delivery of full desktop environments.** Organizations may use Remote Desktop Services to deliver a new desktop environment to clients that are unable or are waiting to upgrade to a new operating system.
* **Rapid deployment of new workloads across an enterprise.** This enables users to be up and running very quickly, without needing to wait for new workloads to be installed on their desktops.
* **Rapid deployment of a new version of an individual workload.** Clients may be unable or may be unwilling to upgrade because of compatibility issues.
* **Providing full corporate desktops to employees who work from home, either occasionally or full time.** This enables the enterprise to provide full-function, secure application delivery to an unmanaged workstation or a thin client without being concerned about the maintenance and security of the user’s device.
* **Providing individual applications and desktops to external third parties in a web browser.** This could include third parties such as vendors and suppliers.
* **Business continuity in the event of a disaster.** Remote Desktop Services can be used to rapidly provision a full working desktop to a newly acquired, or rented, population of user workstations in a new location.
* **Multi-forest environments in which the Remote Desktop Services infrastructure components may span forest boundaries.** This is particularly relevant in the mergers and acquisitions scenario.
* **Provisioning of difficult-to-maintain or infrequently used workloads.** The management overhead of running such workloads on end-user workstations can be significant, so it can make business sense to run them centrally instead, with delivery through Remote Desktop Services.
* Delivering data-intensive client workloads over low-bandwidth links. Remote Desktop Services can be used to deliver an application over bandwidth-constrained links. This is very effective for remotely accessing and manipulating large volumes of data because only a screen view of the data is transmitted over the network to the client, rather than the actual data.

## Out of Scope

This guide concentrates on Remote Desktop Services design and planning exclusively. The following solutions are out of scope for this guide:

* Multi-tenant hosting or remote hosting. Although companies hosting remote application and shared servers quite often use presentation virtualization in some form, this document does not address all of the complexities of a full multi-tenant design.
* Migration from Windows Server 2008 Terminal Services. This guide leads the reader to create the best design to meet the business requirements. Once that design is complete, a separate migration plan should be created, if necessary.
* Citrix MetaFrame. Third-party add-ons to Remote Desktop Services extend the product and offer solutions for various situations, but they are too varied to cover here.
* Planning for Windows Server 2008 Hyper-V. Use the Infrastructure Planning and Design Guide for Windows Server Virtualization, available at <http://go.microsoft.com/fwlink/?LinkID=147617>, as a complement to this guide when using a Remote Desktop Services infrastructure to publish VMs.

## Additional Reading

* Remote Desktop Services (Terminal Services) on the Windows Server 2008 TechCenter site: <http://technet.microsoft.com/en-us/library/dd640164(WS.10).aspx>
* *Infrastructure Planning and Design Guide for* *Selecting the Right* Virtualization:<http://go.microsoft.com/fwlink/?LinkId=160981>
* *Infrastructure Planning and Design Guide for* Windows Server Virtualization:<http://go.microsoft.com/fwlink/?LinkID=147617>
* *Infrastructure Planning and Design Guide for* Microsoft Application Virtualization 4.5:<http://go.microsoft.com/fwlink/?LinkId=160978>
* Anderson, Christa, and Kristin L. Griffin with the Microsoft Presentation Hosted Desktop Virtualization Team. *Windows Server 2008 Terminal Services Resource Kit.* Redmond, WA: Microsoft Press, 2008. ISBN-13:978-0-7356-2585-3
* Remote Desktop Services (Terminal Services) Team Blog: <http://blogs.msdn.com/rds/default.aspx>
* Microsoft Assessment and Planning Toolkit: <http://technet.microsoft.com/en-us/library/bb977556.aspx>

# Step 1: Determine the Scope of the Remote Desktop Services Project

Before designing a Remote Desktop Services infrastructure, an organization needs to determine which parts of its environment to include in the design and the objectives for the project.

The outcomes of this step will in turn drive decisions related to what applications and desktops to make available using Remote Desktop Services and for what user population. Server farm numbers, size, and placement will also be determined by the scope of the project.

## Task 1: Determine Organizational Scope

Before the architecture can be designed, the scope of the project must first be determined so that the planners know the boundaries for which they are building a solution. The scope of the project could be enterprise-wide, one or many locations, or just a single department. If an organization is considering multiple implementations of Remote Desktop Services, it can iterate through this guide for each instantiation.

Decide which locations and workloads will be within the scope of the project. Record this information in Table A-1 “Project Scope” in Appendix A: “Job Aids.”

## Validating with the Business

In order to ensure that the project stays focused on delivering the required services, ask the following questions about the business objectives for the project:

* What benefits does the business expect to achieve through the use of presentation virtualization?
* What is the value of those benefits and, therefore, the cost case for using Remote Desktop Services to deliver those benefits?

The business objectives should be prioritized right from the start so that they are clearly understood and agreed upon between IT and the business.

## Step Summary

In this step, the project scope was defined for the parts of the organization that will participate and the functions that will be provided to them. This will drive decisions in future steps related to capacity requirements. All of the project scope data gathered in this step was recorded in Table A-1 in Appendix A.

In the next step, decisions will be made about how the workloads will be delivered to their users.

# Step 2: Determine How Workloads Will Be Delivered

In the previous step, the scope of the project was determined with respect to locations and the workloads that should be included.

This step’s goal is to look at the end-user populations that fall within that scope and to determine which workloads each person uses and how he or she uses them.

Once this step is complete, project planners will have a list of workloads that are candidates for Remote Desktop Services delivery and an understanding of how they will be used. This information will be used in later steps to design the server infrastructure from which those workloads will be delivered.

## Task 1: Gather Information About Users and Their Working Environment

Using the project scope created in Step 1 as the boundary encompassing the user population, gather the following information about each of the representative users and record it inTable A-2 “User and Application Data” in Appendix A: “Job Aids”:

* User location and number of users. Knowing how many users are at each location assists in determining server sizing and placement.
* Client operating system. When the application is delivered by Remote Desktop Services, the screen updates, keyboard entries, and mouse clicks flow over the network, using Remote Desktop Protocol (RDP), to the Remote Desktop Connection (RDC) client running on the user’s machine. There are significant enhancements in the latest versions of RDC and RDP that provide higher levels of encryption and increase the efficiency of data transmission over the network. The extent to which these can be utilized will depend on which operating system the client has installed because that determines the versions of RDC and RDP that the client system can run. For example, the RemoteApp and Desktop Connection application requires the RDC 7.0 client that is available in Windows 7.

This information will be used in Step 3 to determine the number of RD Session Host server farms since different farms may need to be set up for both high- and low-security clients. It will also be used in Step 7 to design the RD Gateway role service.

* **Encryption requirements**. What is the required level of encryption, if any, for application traffic flowing over the network to the client? This will be used to group workloads together and to determine whether upgrades to the Remote Desktop Services client software may be required.
* Whether single sign-on to applications is provided. If users expect the convenience of a single-sign-on (SSO) experience, that will determine the version of RDC client that must be used to deliver the applications. RDC 6.1 will be required in order to provide this experience in the published applications environment. The version of RDC that is used may affect the number of farms that will be required   
  in Step 3 and the design of the role services in Step 7.
* Applications that are used, their versions, and any special local customizations. If more than one implementation of a given application is in use, then a determination will need to be made about whether those different versions can co-exist in the same Remote Desktop Services environment. If they cannot, each different version, or customization, of the application may have to be delivered by a separate RD Session Host server farm, which will drive up the complexity and cost of the project. This determination will be made in Step 3.

Note   It may be possible to use Microsoft Application Virtualization (App-V) to overcome this by running each application in its own virtual environment on the RD Session Host server. In this case, the application must be supported for delivery by App-V.

* Service level agreement (SLA) levels. What SLAs are in place, what service levels do they commit to, and how many users fall under each one? Understanding what the business and user expectations are for the application defines the requirements for performance and availability. This will drive the sizing of the RD Session Host server farm in Step 7 so that it is able to deliver the expected performance to the user. It will also determine the fault-tolerance approach in that step. It is particularly important to understand the performance and availability expectations that user groups have, whether formalized in an SLA or not, and to use this rather than the actual performance and availability that they experience currently as input to the design.
* Connection type. Are all the users connected through local area networks (LANs), or are some connected over wide area network (WAN) links, over dial-up connections, or through an Internet service provider (ISP)? This information will be used in designing the network, security model, server placement, and sizing. It may not be possible to collect this in user interviews; more likely the network design will need to be examined by IT at the conclusion of these interviews.

## Task 2: Determine Whether the Workload Can Be Delivered Using Presentation Virtualization

Remote Desktop Services provides an environment that delivers workloads to users by publishing them with presentation virtualization, using RDP. This task will determine if any of the workloads that are in scope cannot or should not be delivered using this technology*.*

This information will be recorded in Table A-3 “Application Analysis” in Appendix A. At the end of the step, the list should be reviewed with the business since some applications may not be immediately suited to delivery by publishing with presentation virtualization. These applications may require some changes in order to remain within the scope of the project. Such changes incur cost and, before embarking upon them, this information should be reviewed with the business so that those additional costs and risks can be understood in order to make the best business decision.

The issues can be categorized as follows:

* Issues that affect an application’s business viability.
* Issues that prevent an application from being published.

The data may be gathered from application vendors, Remote Desktop Services community sites, and the application support personnel.

### Issues That Affect Business Viability

A number of business issues may arise when a workload is considered for delivery using presentation virtualization. While the workload and its applications may be technically suitable, these issues could prevent its delivery in this environment. For that reason, the following issues should be considered first:

* Can the applications be licensed for delivery using presentation virtualization without excessive additional cost? Contact the application vendors for this information.
* Are the applications supported for delivery using presentation virtualization? If the vendor does not support the application in a Remote Desktop Services presentation virtualization environment, it may not work correctly, or certain features may not be available. Or the application may still work, but additional testing may be required. A review of the risk of running an unsupported application should also be conducted to see if running the application in Remote Desktop Services is warranted.
* Are there legal reasons for not publishing an application? For example, there could be protected data, such as national security or private patient data, which is not legally allowed to be published by a server that resides outside a country or region’s borders. Additionally, there might data that must not be allowed off-site or data that must remain on certain computers.

### Issues That May Prevent an Application from Being Published

There may be limitations in the way an application is designed or runs that make it unsuitable for delivery using presentation virtualization in Remote Desktop Services:

* Will the user always be connected to the network? Published workloads can only be accessed when the client is connected to the Remote Desktop Services infrastructure. If some users need to be able to run workloads in disconnected mode (for example, when on an airplane), then these workloads cannot be delivered by Remote Desktop Services.
* Does the application require access to client-based resources that may not be available to Windows Server 2008 R2? Client-side devices such as scanners, specialty printers, USB devices, and floppy or hard disk drives used by the application need to be checked for compatibility in a Remote Desktop Services environment.
* Is the required encryption level within the means of most RDP clients? If the RDP clients are not yet at a version that supports the desired encryption level, wait until they are upgraded before deploying the application. Alternatively, the application might be delivered to the clients that are in compliance while leaving the other clients to run the application locally.
* Is the application video-intensive? Video streams from Remote Desktop Services are cached on the client, so the server only needs to send that part of the screen that changes in order to refresh a client. Applications with screen animations may require screen updates to be sent more frequently, placing an increased load on both the server and the network, effectively reducing the scalability of the application.

Record whether the video delivered by the application is Low, Medium, or High intensity, depending upon the level of graphics the application uses. An application such as Notepad with few screen updates is considered a Low-graphics intensity application, whereas an application with several animations and embedded video would be a High-graphics intensity application. High-graphics intensity applications use more server-side processor and memory while consuming more network bandwidth for the frequent screen updates.

* Is the application audio-intensive? Audio is another consumer of network bandwidth. Rate the audio as Low for a few system sounds, Medium for low bandwidth audio delivered occasionally, or High for an audio-intense application such as a media player or computer-based training with voice instruction.

Record the decision for each application in Table A-3 in Appendix A. If the application cannot be delivered using presentation virtualization, remove it from the scope of the project. If the application can be delivered using presentation virtualization, proceed to the next task.

## Task 3: Decide Whether the Workload Will Be Delivered from an RD Session Host or Using VDI

RD Session Host delivers workloads from a shared server environment. In this environment, the operating system, which is Windows Server 2008 R2, is shared by all of the workloads and their users. Applications are also shared by users; a separate instance of each application is created for each user that requires it.

RD Session Host provides a VDI environment in which users have exclusive use of their own desktop operating system and applications that run in their own VM on Windows Server 2008 Hyper-V.

Workloads can be delivered at much greater scale from the RD Session Host environment than through RD Virtualization Host, so review each workload against the questions below to determine whether it can be delivered from an RD Session Host environment, and record the assessments in Table A-3 in Appendix A. This will determine which workloads can be delivered by the shared server environment of RD Session Host. If the workload cannot be delivered from RD Session Host, it will be reviewed for delivery using VDI.

The issues can be categorized as follows:

* Issues with running on a shared Windows Server 2008 R2 instance.
* Issues with the application’s behavior cause problems in a multi-user server.

The data may be gathered from application vendors, Remote Desktop Services community sites, and the application support personnel.

### Issues with Running on a Shared Windows Server 2008 R2 Instance

RD Session Host centralizes client application execution onto a Windows Server 2008 R2 platform. Some client applications may be unable to run effectively on the Windows Server 2008 R2 platform.

Answer the following questions for each of the candidate applications:

* Does the user require administrative rights on the machine? In the shared server environment, administrative rights cannot be granted to individual users.
* Does the user need to be able to restart the operating system on demand? In the RD Session Host shared server environment, a restart would affect all other users.
* Can the application run on Windows Server 2008? An internally developed application may be written to install and run only on earlier operating systems. Verify that the application can run on Windows Server 2008 R2. If not, use the Application Compatibility Toolkit (ACT), available at <http://technet.microsoft.com/en-us/windowsvista/aa905102.aspx>, to see if the application works in compatibility mode.
* Does an application employ code not optimized for either 32-bit or 64-bit architecture? Programs that include 16-bit code must be treated differently by the operating system, which causes the application to use a disproportionate amount of memory and CPU resources compared with programs written in 32-bit or 64-bit code. Use the ACT to detect 16-bit components and move to a newer version, have the software repaired, or decide whether the extra resources required to run the application are acceptable.

### Issues with the Application’s Behavior Cause Problems in a Multi-user Server

At the completion of the project, applications that have been running separately on individual workstations will be run together in a server environment. Multiple instances of the same application can be run on the same server for different users. This requires that the applications co-exist and run in that environment without interfering with each other. The applications must also behave in a way that does not degrade their scalability and performance in this environment to the point that it is no longer cost effective.

The following questions raise the issues that may commonly surface with applications:

* Does the application consume excessive resources or neglect to release them appropriately? An application cannot be allowed to monopolize the CPU, memory, or other shared server resources. For instance, a small memory leak may not present much of a problem on a stand-alone computer, but 50 users concurrently running that same application on a shared server can cause the server to run out of memory very quickly. Monitor memory usage to test for the problem. If discovered, have the software’s creator fix the issue, and evaluate the risk of running the application while it still leaks.
* Does the application follow proper installation and uninstallation standards? Applications need to be functional for all users after installation, and uninstalls must not remove components needed by other applications on the RD Session Host. The ACT tools can help determine setup issues. If any are found, determine the impact to the server and users.
* Does the application alter shared system files and components? An application that changes files such as fonts, dynamic link libraries (DLLs), and drivers affects other applications’ ability to function. Use the ACT to watch for this behavior. If detected, the software must be fixed before the application can be used in the RD Session Host.

Alternatively, it may be possible to use App-V to overcome this by running each application in its own virtual environment on the RD Session Host server. In this case, the application must be supported for delivery by App-V.

* Is user data kept discrete from other data? If an application places user and temporary files in the same location, then other instances of the application can overwrite or alter the files, producing unpredictable results. Test to see if this is occurring. If so, it must be fixed before using the application in RD Session Host.
* Is the registry accessed properly? An application must not make changes to the registry at a level where the change can affect other users or applications. Normally, Remote Desktop Services installation redirects registry entries installed to HKEY\_LOCAL\_MACHINE\Software on a standard desktop to HKEY\_CURRENT\_USER\Software on the RD Session Host server, but software may be written in a way that circumvents these protections. Use the ACT or a registry comparison tool to watch for changes in the wrong areas. If such changes are discovered, evaluate the impact upon the server and users.
* Do changes made by the user to program options affect other instances of the program? Changes made in a user’s environment should not affect other users. For example, when user “A” changes the default view in an email client, every other user’s default view should not change as well. Some ways to fix this issue include preventing changes in programs via permissions, creating mandatory user profiles, or having the vendor fix the issue.

Record the information for each application in Table A-3 in Appendix A. If an application will create a conflict, examine whether that conflict could be overcome by using App-V to run the application in a virtualized box within the RD Session Host server. If so, record that information so that it can be used as input to the App-V infrastructure design, and refer to the IPD Guide for App-V. Information on workloads that are not compatible with the RD Session Host environment or App-V will be used as input to the design of the VDI environment in Step 9. Now, proceed to the next task to categorize the users.

## Task 4: Categorize Users

At this point, the list of users as well as the workloads that will be published by Remote Desktop Services has been validated. In order to properly size the Remote Desktop Services environment, the load that users will place on it needs to be determined.

Place users into one of three categories: Heavy, Normal, or Light, based on their general usage behavior in their workloads.

This categorization will be used in Step 5 to provide the user load on each workload as input to determining the size of the RD Session Host server farm.

Assign each user, or group of users, to only one category since the categorization is reflective of behaviors that are likely to be the same in all workloads. Record the categories for the users, or group of users, in the “User type” row in Table A-2 in Appendix A.

Place each user into one of the following categories:

* Heavy user. Spends most of the day working in one or more applications, often in two or even three at the same time. Uses advanced features of the software and can quickly find new features and employ them. These users could be developers, engineers, graphic artists, research assistants, or project managers. Many of their applications deliver high-resolution graphics that are frequently updated, perhaps with animation. They may copy large multimedia files from application to application. May log on and off applications several times a day, or work in them for extended periods of time. These users are likely to save their files frequently as they work in an application.
* Normal user. Uses the computer frequently, but also performs other work tasks that do not involve the computer. Sometimes exchanges data between their different applications. Their job titles might be administrative assistant, salesperson, doctor, or producer. When leaving their computers to complete other work, they probably will not log off; rather they will leave the application running.
* Light user. Has the computer switched on, but uses it for only a few minutes an hour. Uses only the applications they need to do their jobs, probably not more than one application. They may leave the application running for many days at a time without logging off. Their job titles might be baker, hospital volunteer, corporate fitness expert, cashier, or electrician.

## Step Summary

The list of candidate workloads has now been recorded in Tables A-2 and A-3 in Appendix A, along with characteristics of their usage. The workloads’ suitability for delivery by Remote Desktop Services has been evaluated, and workloads that are not suitable have been removed from the project.

The next step determines the resource requirements to deliver those workloads through Remote Desktop Services.

## Additional Considerations

The ACT, available at <http://technet.microsoft.com/en-us/desktopdeployment/bb414773.aspx>, is a group of tools that can be helpful for gathering and evaluating information about an application as it runs on a stand-alone computer as well as when it runs on an RD Session Host server. Some of the ACT tools most helpful in evaluating an application for Remote Desktop Services are:

* Setup Analysis Tool (SAT). Watches a set up and detects the installation kernel mode drivers and 16-bit components, among other things.
* Inventory Collector. Identifies installed applications within an organization.
* Compatibility Reporting. Helps with compiling and evaluating collected data.
* Find compatibility information from the community. Common repository of compatibility issues and their remediation.

## Additional Reading

* Infrastructure Planning and Design Guide for Selecting the Right Virtualization Technology: <http://go.microsoft.com/fwlink/?LinkId=160981>
* The Application Compatibility Toolkit 5.0 site: <http://technet.microsoft.com/en-us/windowsvista/aa905102.aspx>
* The Application Compatibility page: <http://technet.microsoft.com/en-us/windowsvista/aa905066.aspx>
* “Device Driver INF Changes for Plug and Play Device Redirection on Terminal Server”: <http://www.microsoft.com/whdc/driver/install/TS_redirect.mspx>
* Infrastructure Planning and Design Guide for Microsoft Application Virtualization 4.6: <http://go.microsoft.com/fwlink/?LinkId=160978>
* Infrastructure Planning and Design Guide for *Windows Server Virtualization*: <http://go.microsoft.com/fwlink/?LinkID=147617>

# Step 3: Determine the Number of RD Session Host Server Farms

An RD Session Host server farm is a group of RD Session Host servers that publishes an identical set of workloads.

When a user connects to a farm, the request is passed to the RD Connection Broker role service, which uses a load-balancing algorithm to redirect that request to the least-loaded server in the farm.

NoteA Windows Server 2008 R2 RD Session Host server farm can include servers delivering sessions from Windows Server 2008 Terminal Services. However, servers delivering sessions from Windows Server 2003 Terminal Services are not supported in the same farm. So if there are applications that must be published from Windows Server 2003, it is recommended that they be in a separate farm.

The goal of this step is to arrive at an optimal number of RD Session Host server farms for the Remote Desktop Services design. At the completion of this step, the number of RD Session Host server farms and their locations will have been recorded in Table A-4 “Farm Design” in Appendix A: “Job Aids” so that the remaining sizing and capacity steps can be completed.

## Task 1: Determine the Number of RD Session Host Server Farms

Best practices suggest that a design should start with a single farm, and then add more farms only when required. Use the list below to decide whether additional RD Session Host server farms may be required:

* Clients separated from the current farm by WAN speeds. If more users are accessing the RD Session Host server farm than the link to the farm can accommodate, an additional RD Session Host server farm may be placed at the remote location. In that case, make sure that the links from that farm to back-end services, such as remote databases, have sufficient bandwidth. Alternatively, examine whether increasing the bandwidth of links between the farm and the remote clients may be more effective than instantiating another farm.
* Traveling users. Determine whether additional farms or upgrades to bandwidth may be required to accommodate traveling users. If a number of users or users from a critical group regularly travel to other locations, steps may need to be taken to ensure that their applications can be delivered within the expected service level at those remote locations. This may require that additional farms be instantiated at the remote locations, that the bandwidth to those locations be upgraded, or that capacity be added to accommodate the traveling user. For example, a location in London, England, may host a farm for users that frequently travel there from the United States.
* Requirements to run different versions of the same software, or to have different versions of the same files, such as DLLs. Note that this applies only to session host farms; it does not apply to virtualization host farms. If users require access to more than one version of software or files and if the different versions cannot coexist, the different versions will have to be run on separate RD Session Host server farms. However, it may be possible to use App-V to overcome this by running each application in its own virtual environment on the RD Session Host server. In this case, the application must be supported for delivery by App-V.
* Security limitations in some clients. Limitations in some of the clients may drive implementation of the workloads on two (or more) separate farms at different security levels. There may be applications that require a security implementation that can only be delivered by restricting RD Session Host server access to clients that can support the latest security features available, such as Network Level Authentication. The security features that are available for server-client communication depend on the level of the RDC client and on the level of the operating system. So applications requiring the highest levels of security may need to be instantiated in a separate RD Session Host server farm. Clients running earlier versions of RDC or lower levels of the operating system will not be able to connect to that farm.

Compare required security levels with the security capabilities of the users who need that application. Can each client conform to the security requirements? If not, upgrade the user’s RDC client, lower security for that case, remove the application from the project, or allow the users to run it locally. Or, set up a separate, lower-security RD Session Host server farm for the legacy clients until they can be upgraded.

* Internal and external user populations. Where the same applications are being published both to internal and external users, the external users may be separated on a different farm. This separation may be required by the business even though the applications and their data are identical.
* **Specific encryption requirements.** There are certain levels of encryption that are legal only in certain countries or regions. RD Session Host servers can be configured to allow the client using the highest encryption it is capable of or to deny connections from clients unable to comply with a certain level of encryption. Using the latter setting may require that RD Session Host server farms be split if a legacy client requires access to some applications.
* Organizational requirements to separate business groups. For security reasons, the Accounting department may not want non-accounting personnel to be able to log on to servers containing accounting information, or Research and Development may require the higher security levels of the most recent client, which means that clients unable to comply with the security policy must connect to a different farm.
* Legal considerations requiring a separate farm. National security, privacy issues, and fiduciary laws could require the separation of certain data or prevent other data from crossing national borders. If necessary, add farms to address this need.

Start with one RD Session Host server farm, and then place additional farms in the design as required by the above reasons. Record each farm and its purpose in a new table like the one in Table A-4 in Appendix A.

## Step Summary

The number of RD Session Host server farms and their locations were determined in this step and were recorded in Table A-4 in Appendix A.

The next step is to assign workloads and users to those farms so that the size of each farm can then be calculated in a later step.

# Step 4: Map Workloads and Users to Farms

Now that the number of RD Session Host server farms has been determined, the goal of this step is to assign the applications, virtual desktops, and users to the appropriate RD Session Host server farms. The name of each RD Session Host server farm will be listed in Table A-4 “Farm Design” in Appendix A: “Job Aids,” along with the users and the applications that are to be assigned to each farm. It may also be useful to represent the farms on a geographic map background if they are spread across different locations. An existing network infrastructure diagram can also provide a useful background for this.

## Task 1: Assign Workloads and Users to Their RD Session Host Server Farms

The number of users, farms, virtual desktops, and applications can all become confusing to maintain. In this step, use Table A-4 in Appendix A to record the following information:

* List the applications, associating them with the farm where they will be hosted. Be careful to place applications that have interdependencies together in the same farm.
* Map which users will have access to each application or virtual desktop on each farm. Include any applications that may be unique for traveling users in each farm.
* Where there are significant populations of traveling users, represent them in the job aid in all the locations to which they frequently travel.

## Step Summary

This step, while somewhat easy to describe, is a time consuming and critical task that is necessary for ensuring that the appropriate capacity and performance requirements are analyzed correctly in subsequent steps. The users and workloads were mapped to the most appropriate farms, and this mapping was recorded in Table A-4 in Appendix A. This information is used in the next step to determine what resources each RD Session Host server farm needs in order to handle the users who will be connecting to those server farms.

# Step 5: Design the Farms

The RD Session Host servers in a farm all provide the same set of workloads. This ensures that users receive the same experience no matter to which RD Session Host server they connect.

Many variables are involved in moving a set of applications that are running on client computers to instead run on RD Session Host servers. The goal of this step is to arrive at a reasonable estimate of capacity without requiring excessive modeling and precision. Implement the first set of users, and record the lessons learned; then repeat for the next set of users, adjusting to include what was learned.

This step determines the hardware configuration of the server in each farm and, therefore, the number of servers required to deliver the applications from the farm. Additional servers may then be specified for fault tolerance, load balancing, and maintenance.

Perform the following for each farm, and record the results in Table A-4 in Appendix A.

## Task 1: Select a Form Factor for the RD Session Host Server

The goal of this task is to determine the most appropriate type of hardware on which to deploy the RD Session Host servers.

To make this selection, start with the prerequisites for Windows Server 2008 R2 as a minimum requirement, and then follow these guidelines, which are presented in priority order, to determine the ideal form factor for the servers:

1. Use large memory. In the RD Session Host server, where many users are sharing the same application, large memory can significantly improve performance because it allows each user’s work to remain in memory rather than being swapped out to a physical disk. Additional memory will also reduce the disk input/output (I/O) demand for this reason.
2. More numerous, smaller disk spindles rather than fewer, large disks. Consider using a redundant array of independent disks (RAID) configuration for disk fault tolerance. The key to performance though is in the total disk IOs per second (IOps). In order to increase the IOps for the disk subsystem, consider using more numerous, smaller disks rather than fewer, but larger disks. This approach distributes the work load across more spindles, thereby increasing the overall performance. The number of spindles used has a significant effect on the response times for file access. The disk activity generated on a typical RD Session Host affects the following three areas:

* System files and application binaries
* Page files
* User profiles and user data

Ideally, these three areas should be supported by distinct storage devices. Use storage adapters with a battery-backed cache that allows write-back optimizations. Controllers with write-back cache support offer improved support for synchronous disk writes. Because all RD Session Host server users have an individual registry hive, synchronous disk writes are significantly more common on an RD Session Host server. Registry hives are periodically saved to disk by using synchronous write operations.

1. Use multi-core and multi-processor CPUs. In the multi-user environment of RD Session Host server, multiple processors and cores can help reduce CPU congestion.

Refer to *Performance Tuning Guidelines for Windows Server 2008*, available at<http://www.microsoft.com/whdc/system/sysperf/Perf_tun_srv.mspx>, for more specific guidance on form factor for RD Session Host servers.

Document the selected form factor in Table A-4 in Appendix A, and then proceed to the scaling assessment in the next task.

## Task 2: Determine the Number of RD Session Host Servers Required in the Farm

The optimal number of RD Session Host servers in an RD Session Host server farm delivers a consistent, responsive user experience while balancing system utilization, growth capacity, and cost. In order to consistently arrive at that optimal number for each farm, start with the implementation of a small farm. Use the guidance in “Remote Desktop Session Host Capacity Planning in Windows Server 2008 R2” at <http://www.microsoft.com/downloads/en/confirmation.aspx?familyId=ca837962-4128-4680-b1c0-ad0985939063&displayLang=en> to select a standard server form factor. Put that server form factor into production and measure the results to determine whether the farm delivers the expected performance. Note that more servers can readily be added to the farm if additional capacity is required. Learn from the experience and use it to adjust the method accordingly when planning the next farm.

Record this number in Table A-4 in Appendix A. Repeat the process for each defined RD Session Host server farm.

## Task 3: Determine the Number of Additional Servers Required for Fault Tolerance

The number of servers per farm, determined in the previous task, does not account for fault tolerance. It’s therefore necessary to determine the number of additional servers needed to provide a fault-tolerant implementation.

RD Session Host servers cannot be placed in a Microsoft failover cluster, so fault tolerance, if desired, must be provided by additional servers with a load-balancing solution. If a server goes down, the users on that server will need to restart their sessions, which may be instantiated on one of the remaining servers. Incoming session requests will be load-balanced onto the remaining servers by RD Connection Broker. The user will experience session interruption in this case.

Total server capacity will be reduced during the downtime because the remaining servers will be more heavily loaded, so plan for one or more servers to account for this so that capacity levels are maintained in the event of unplanned or planned downtime for server maintenance.

Determine the number of additional servers required to meet the fault-tolerance requirements, add it to the number in the “Number of RD Session Host servers” row, and record this in Table A-4 in Appendix A.

## Step Summary

In this step, the design of the farm has been completed. The form factor of the RD Session Host servers in the farm has been decided on, the servers have been sized, and the fault-tolerance approach has been determined. The data gathered in this step was recorded in Table A-4 in Appendix A.

Before proceeding to the next step, all of the tasks in this step must be repeated for any other farms that will be implemented. In the next step, decisions will be made on the storage of user profiles and data.

* Additional ReadingPerformance Tuning Guidelines for Windows Server 2008: <http://www.microsoft.com/whdc/system/sysperf/Perf_tun_srv.mspx>
* Infrastructure Planning and Design Guide for *Windows Server Virtualization*: <http://go.microsoft.com/fwlink/?LinkID=147617>

# Step 6: Determine Where to Store User Data

A Remote Desktop Services user may have both a standard user profile on the Windows Server 2008 R2 RD Session Host role service and a Remote Desktop Services profile. This allows the user to maintain different settings for his or her logon to the server operating system and to the RD Session Host server. For roaming users, these profiles should be stored on a fault-tolerant file server. When the user logs off, any profile data can be removed from the RD Session Host server’s hard drive and registry to ensure that the RD Session Host server stays clean. However, to provide a smooth logon experience, ensure that roaming user profiles are cached locally on the RD Session Host server.

Where users may log on to more than one farm, they may need a different Remote Desktop Services profile for each farm. These can be stored on a file server with a path like [\\fileserver\share\Farm1\username](file://fileserver/share/Farm1/username) where Farm1 is the name of the farm. There would then be a similar path for the profile that is used on Farm2, and one for Farm3, and so on. The path would be set up as [\\fileserver\share\%FarmName%\username](file://fileserver/share/%FarmName%25/username) in the user account, and an environment variable created for %FarmName% on each farm. At logon to a particular RD Session Host server farm, the environmental variable would be automatically substituted with the actual name, like Farm2, in the Remote Desktop Services profile path, and the correct profile would then be available.

While user profiles are generally quite small, user data can require significant storage. Separate the user data (in My Documents and Application Data files) from the profile, and store them on a separate network drive, which can be replicated if the user travels.

User profiles and data should not be stored on the RD Session Host server. If they are stored on a particular RD Session Host server and the server becomes unavailable, then the user may be able to establish a new session on another server in the farm, but his or her profile and user data will not be available.

The objective of this step is to determine where to store the user data and profiles so that they are always available to all the servers in the farm.

## Task 1: Design User Profiles

In this task, determine the following about user profiles and record the information in Table A-4 in Appendix A:

* Will users have a mandatory profile? Individual users, or user groups, can be assigned a mandatory profile, which is very useful on multi-user computers and kiosks. There can be more than one mandatory profile assigned to different user groups—for example, one mandatory profile can be assigned for kiosk users and a different mandatory profile can be assigned for call center personnel. Although mandatory profiles are simpler to manage than roaming user profiles, any changes made to the profile during the session are lost at logoff.
* Determine how much space is required for user profiles. If mandatory profiles were chosen, just measure the size of each different mandatory profile, and add them together. These profiles are likely to use less than 1 megabyte (MB), and there is only one copy of the user profile physically stored on disk per defined mandatory profile. If users have a Remote Desktop Services roaming profile, determine the maximum size that the profile could reach, and multiply that by the number of users.
* Determine where to store the user profiles. Since the RD Session Host server, rather than the client computer, accesses the profile, the storage location needs to be easily accessible. The profiles can be held on a storage area network (SAN) or any other file server.

## Task 2: Design User Data

In this task, determine the following about user data and record the information in Table A-4 in Appendix A:

* Determine how much space is required for user data. If users are allowed to store data in a My Documents folder on the storage server, then the storage space required can be significant. Determine how much data users will be permitted to store. This amount, multiplied by the number of users, is the storage requirement. Record this value in the job aid.
* Determine where to store the user data. Since the RD Session Host server, rather than the client computer, accesses the data, the storage location needs to be easily accessible. The data can be held on a SAN or any other file server.

## Task 3: Design the Storage

Now that the storage requirements for user profiles and user data have been determined, the two storage locations should be designed. As explained above, in an environment where users connect to more than one farm, they may have different profiles and data on each farm, and so the following work must be completed separately for each farm.

For the user profile and the user data stores, address the following areas to ensure that the solution will meet the requirements of the users and business:

* Capacity. For each user account, add together the maximum size of the user profile and the maximum size of the Remote Desktop Services profile. If there is more than one pair of profiles for each user (because the user logs on to more than one farm), multiply by the number of farms. Now multiply that number by the number of user accounts. This will be the space required to hold all the user profiles.

[(max size of user profile + max size of Remote Desktop Services profile) x number of farms] x number of user accounts = total user profile storage space required

Multiply the maximum data storage available for each user account by the number of user accounts. This will be the space required to hold all the user data.

Max size of user data x number of user accounts = total user data storage space required

* Performance. The speed at which the user profile files can be accessed is critical at user logon, so they should be stored on a high-speed storage device attached to a file server that has a high-speed network connection.

The speed at which the user data files can be accessed is critical to the users’ perception of application responsiveness, so the data should be stored on a high-speed storage device attached to a file server that is local to the RD Session Host server farm.

The goal of designing storage to meet disk performance requirements is to support the number of IOps. To determine the IOps requirements for these file servers, set up a test environment and record the number of IOps generated by a user working in a data-intensive application. Then multiply this number by the number of user accounts to derive the total IOps load. Next, follow the installation’s standards to add a buffer in order to derive the IOps needed.

* Fault tolerance. Access to the user profile and the Remote Desktop Services profile is required at logon, and access to the user data will be required in order for most applications to run.

For these reasons, the profile and data files should be placed in a highly available file server configuration that provides redundancy, such as a Distributed File System (DFS) share that is replicated or in a failover cluster.

The actual number of disks required is the greater of the number determined necessary for capacity or the number determined necessary for performance. Usually that will be the number determined by performance.

Document the storage design in Table A-4 in Appendix A; then proceed to the next step.

## Step Summary

Decisions have been made about what type of user profiles will be allowed, where they will be stored, and how much storage space they will require. The users’ data storage requirements have also been determined and addressed in the same way. This will ensure that the users’ profiles and their data are available on a consistent basis. All of the data gathered in this step was recorded in Table A-4 in Appendix A.

## Additional Reading

* Step-by-Step Guide for Configuring a Two-Node File Server Failover Cluster in Windows Server 2008: <http://www.microsoft.com/downloads/details.aspx?FamilyID=518D870C-FA3E-4F6A-97F5-ACAF31DE6DCE&displaylang=en>
* Anderson, Christa, and Kristin L. Griffin with the Microsoft Presentation Hosted Desktop Virtualization Team. Windows Server 2008 Terminal Services Resource Kit. Redmond, WA: Microsoft Press, 2008. ISBN-13:978-0-7356-2585-3
* Infrastructure Planning and Design Guide for Windows Server 2008 R2 File Services: <http://go.microsoft.com/fwlink/?LinkId=160976>

# Step 7: Size and Place the Remote Desktop Services Role Services for the Farms

The Remote Desktop Services role services RD Connection Broker, RD Web Access, RD Licensing, and RD Gateway can be shared among multiple farms or can be dedicated to a specific farm. The goal of this step is to optimally place each of these role services and then to determine the number of servers required and how they will be designed.

NoteAll of these role services can be run in a VM. However, this should be carefully tested to ensure that adequate performance is delivered.

At the time of writing, there is limited guidance on capacity and performance planning for the RD Connection Broker, RD Web Access, and RD Licensing role services. However, guidance on sizing the Terminal Services Gateway role for Windows Server 2008 is available at <http://download.microsoft.com/download/9/9/0/990839D2-ED14-4ACA-BFBF-8A40F2A6D2CE/TSG_Scalability_Whitepaper_Final.docx>.

Start by implementing the role services in a small farm, measure the performance, analyze and learn from the results, and adjust the sizing of the role services accordingly.

It may be helpful to take a pragmatic view of the load that is on each of the role services; this may prove useful in sizing the roles (see Table 2).

**Table 2. Load on the Role Services**

|  |  |  |  |
| --- | --- | --- | --- |
| **Role service or function** | **Frequency of access, per Remote Desktop Services client session** | **Other use** | **Relative weight per session** |
| RD Licensing | 0 | First time each client accesses an RD Session Host server. When client licenses expire. | Light |
| RD Connection Broker | 1 | Continuous monitoring of the number of sessions on RD Session Host servers. | Light |
| Redirector | 1 | Not applicable | Light |
| RD Gateway | Many | Not applicable | Normal |
| RD Web Access | 1 | Not applicable | Light |

If more accuracy is required, load tests can be performed to determine the number of users that a server can support for each of the Remote Desktop Services role services. Use the same hardware configuration for the initial testing for all role services. When the initial testing is complete, determine whether role services will reside on shared servers. If they won’t, the form factor may be adjusted to accommodate differences in the demands of the different role services.

At the end of this step, the number of Remote Desktop Services role services and their form factor is determined.

Begin by examining the Active Directory® Domain Services (AD DS) implementation. This should be done first because it has the same implications for all Remote Desktop Services role services discussed below. The Remote Desktop Services role services can work across domain boundaries, so they may be shared across more than one domain. However, there may be organizational reasons to restrict the scope of the role services to particular domains, and this can be implemented using Group Policy settings. In that case, a separate Remote Desktop Services role service will need to be instantiated in each domain.

The Remote Desktop Services role services cannot work across an AD DS forest boundary unless a cross-forest trust is established for that purpose. So if that trust is not in place, plan to implement each of the required Remote Desktop Services role services in each forest.

## Task 1: Design and Place the RD Connection Broker Role Service

RD Connection Broker provides two services for incoming user connection requests: redirection and load balancing. When a user initiates a connection to an RD Session Host server farm, he or she is first connected to a single RD Session Host server where either one of the following services is running:

* Session Redirector role service for connecting to RD Session Host servers
* Virtualization Redirector role service for connecting to RD Virtualization Host role service

Although both of these services run on an RD Session Host server, they are mutually exclusive and cannot coexist, and the server that runs Virtualization Redirector cannot function as an RD Session Host server. So if RD Connection Broker will provide both session redirection and virtualization redirection, at least two servers will be required.

The RD Connection Broker role server can be shared across RD Session Host server farms, but it will not distribute user session requests across farms; the connecting user can be placed only in the farm that redirected the user to RD Connection Broker. When RD Connection Broker is shared between farms, it will operate independently on behalf of each farm.

Using the following process, decide where RD Connection Broker services are needed and in what format, and determine which farm will connect to each RD Connection Broker server. Record the following information in Table A-4 in Appendix A:

1. Determine whether the RD **Connection** Broker role service is required. In an RD Session Host server farm that contains more than one server, RD Connection Broker is used to load balance session requests across RD Session Host servers and to reconnect sessions that have been disconnected. It will reconnect the session to the server where the user’s application is running, thereby avoiding the possibility of an orphan session for that user. So for a multi-server farm, the RD Connection Broker role service will be required.

The RD Connection Broker role service is also used to publish virtual desktops for VDI and to make the initial client connection to those desktops. So if Remote Desktop Services will be used to publish virtual desktops, the RD Connection Broker role service will also be required.

Note   Remote Desktop Services Connection Broker redirection is supported by RDC 5.2 clients and later.

1. Determine the demand on the RD Connection Broker. At the time of this writing, no product guidance is available on capacity and performance planning for RD Connection Broker. Since the RD Connection Broker role service is only used when the user connects to a farm, it can be considered a lightweight service, and its implementation can be designed as such. Start by implementing the role service in a small farm, measure the performance, analyze and learn from the results, and adjust the sizing accordingly.
2. Determine whether to use dedicated redirectors. To increase performance of session and/or virtualization redirection in a large RD Session Host server farm, RD Session Host servers can be dedicated as redirectors. These servers will process incoming requests but will not accept user sessions. This can deliver a faster logon experience to the user.

Separate the redirector onto another server of the chosen form factor. Record whether the separation of the redirector onto a different server increases the number of user sessions that can be supported. If it does, weigh that benefit against the additional cost and maintenance overhead that will be incurred by the addition of a server to determine whether to separate the redirector from RD Connection Broker.

1. Determine the number of servers required. In order to determine the number of servers required for RD Connection Broker, divide the total number of users that will have sessions with the farm by the number of users that an RD Connection Broker server was able to support when tested. Complete this process for each farm.

If the decision has been made to use dedicated redirectors, divide the total number of users that will connect to the farm by the number of users that a dedicated redirector was able to support when tested. This is the number of dedicated redirectors required. Complete this process for each farm.

1. Decide whether the servers that host **RD Connection** Broker can be shared. Now that the resource requirements for a dedicated RD Connection Broker service are known for each farm, a decision can be made on whether the RD Connection Broker role service can be shared. This can be done in the following ways:
2. The RD Connection Broker role service can be shared between farms. In this case, sum the number of users that has been determined for the RD Connection Broker server in each farm, and divide by the number of farms. This will produce the number of users that can be supported on each shared RD Connection Broker server.

To determine the number of servers required for RD Connection Broker, divide the total number of users that will have sessions with the farms by the number of users that an RD Connection Broker server was able to support when tested. Complete this process for each shared farm, and record the value.

1. The RD Connection Broker server can host other Remote Desktop Services role services. Complete all the tasks in this step to determine the number of users that the server form factor can support for each of the role services that will be combined on a server. Then sum the number of users for each role service that will be combined, and divide by the number of shared role services. This will produce the number of users that can be supported on each role-shared server.

Divide the total number of users that will connect to the farm by the number of users that a role service can support to determine the number of servers required for the roles. Complete this process for each shared server, and record the value.

1. **Determine where to place the RD Connection** **Broker role service in the network.** The RD Connection Broker role service needs to communicate with each RD Session Host server farm whose sessions it manages. RD Session Host server farms on a high-speed LAN can share the RD Connection Broker role service. Conversely, farms separated by WAN connections need an RD Connection Broker role service assigned at each location. List the RD Connection Broker role service instances in Table A-4 in Appendix A, ensuring that every RD Session Host server farm requiring an RD Connection Broker role service can connect to one through a high-speed connection.
2. **Determine the fault-tolerance requirements of the RD Connection** **Broker servers**. The RD Connection Broker server provides load balancing for the RD Session Host servers in the farm. It also directs reconnection requests to the server on which a disconnected session has been running. If it is unavailable, these reconnections will not be completed, and orphan sessions will begin to appear on the RD Session Host servers in the farm.

For this reason, the RD Connection Broker role service should be protected from failure by instantiating more than one copy of the role service in a Microsoft failover cluster. This will provide continuity of service in the event of a failure during a session reconnect. Examine the SLAs with users to determine whether the service that a Microsoft failover cluster can deliver is actually required.

## Task 2: Determine the Number of Servers Required for RD Web Access

The RD Web Access role service is installed along with RD Connection Broker and can be used to present applications on a website that is accessed by the user with a browser. However, when the user selects an application from that website and launches it, he or she is connected directly to the remote desktop server, and RD Web Access is no longer involved.

At the time of writing, no authoritative product guidance is available on the server sizing requirements for the RD Web Access role service. RD Web Access is a light-weight application and can be hosted on any hardware configuration that supports a light-weight Microsoft Internet Information Services (IIS) application. In a remote desktop server farm of a few servers, the RD Web Access role service may be hosted on an IIS web server farm that is performing other work. It is therefore prudent to install the RD Web Access application on an existing server, and monitor the system usage to determine if dedicated hardware is required.

The RD Web Access role service can be made fault tolerant by duplicating the role service on additional servers and load balancing them using Network Load Balancing (NLB) or a hardware load balancing solution. If the RD Web Access role service becomes unavailable, users who are accustomed to accessing their applications from its menu may not be able to access their applications. If these users can access their applications in another manner, balance the cost of providing high availability of the RD Web Access role service against the business benefit that it delivers in terms of user convenience. If RD Web Access is the only way for important external users, such as customers, to access applications, high availability may be imperative.

The RD Web Access role service may be used to make a menu of applications available to both internal and external users. Consider whether the organization’s security policy will require that the internal and external facing RD Web Access users be hosted on separate systems.

Record if, where, and how many RD Web Access role services will be hosted for each farm in Table A-4 in Appendix A.

## Task 3: Design and Place the Remote Desktop License Servers

There are two types of Remote Desktop Services client access licenses (RDS CALs):

* **RDS Per Device CAL.** When Per Device licensing mode is used and a client computer or device connects to a remote desktop server for the first time, the client computer or device is issued a temporary license by default. When a client computer or device connects to a remote desktop server for the second time, if the license server is activated and sufficient RDS Per Device CALs are available, the license server issues the client computer or device a permanent RDS Per Device CAL.
* **RDS Per User CAL.** An RDS Per User CAL gives one user the right to access an RD Session Host server from an unlimited number of client computers or devices. When the client or device connects to the RD Session Host server, AD DS is contacted to determine whether the user has an RDS CAL. If the user does not have an RDS CAL, the RD Licensing role service is called to provide a license to associate with that client in AD DS.

RDS Per User CALs are not enforced by the RD Licensing role service. As a result, client connections can occur regardless of the number of RDS Per User CALs installed on the license server.

NoteThis does not absolve administrators from the Microsoft Software License Terms requirements to have a valid RDS Per User CAL for each user.

The RD Licensing role service can be shared across RD Session Host server farms to provide RDS CALs to clients connecting to any of those farms from its pool of RDS CALs.

Using the following process, plan the capacity of the Remote Desktop license server so that it can deal with the request rate for new RDS CALs. Then decide where RD Licensing services are needed and in what format, and determine which farm connects to each license server. Record the following information in Table A-4 in Appendix A:

1. Determine the demand on the license server. At the time of writing, no guidance is available on capacity and performance planning for the RD Licensing role service. Since RD Licensing is used when the user connects to the farm, it can be considered a lightweight service and its implementation can be designed as such. Start by implementing the role service in a small farm, measure the performance, analyze and learn from the results, and adjust the sizing of the role service in the next farm accordingly.
2. Determine the number of servers required. To determine the number of servers required for RD Licensing, divide the total number of users or devices that will have sessions with the farm by the number of users that a license server was able to support when tested. Complete this process for each farm, and record the value.
3. Decide whether the servers that host RD Licensing can be shared. Now that the resource requirements for a dedicated license server are known for each farm, a decision can be made on whether the RD Licensing role service can be shared. This can be done in the following ways:
4. The **l**icense server can be shared between farms. In this case, sum the number of users that has been determined for the license server in each farm, and divide by the number of farms. This will produce the number of users that can be supported on each shared license server.

To determine the number of servers required for RD Licensing, divide the total number of users that will have sessions with the farms by the number of users that a shared license server was able to support when tested. Complete this process for each shared farm, and record the value.

1. **l**icense server can host other Remote Desktop Services role services. Complete all the tasks in this step to determine the number of users that the server form factor can support for each of the role services that will be combined on a server. Take the lowest of these numbers; that is the number of users that can be supported on each role-shared server.

To determine the number of servers required for the roles, divide the total number of users that will connect to the farm by the number of users that a role-shared server was able to support when tested. Complete this process for each shared server, and record the value.

1. **Determine where to place the RD**Licensing **roles in the network.** RD Licensing needs to be able to communicate with each remote desktop server farm whose licenses it manages. RD Session Host server farms on a high-speed LAN can share RD Licensing services. Conversely, farms separated by WAN distances will likely need to have RD Licensing services assigned at each location.

If there are areas where RDS CALs will be procured and administered by different entities (for example, if Research and Development have a separate support group from Engineering), it may be necessary to place additional RD Licensing roles for each of these organizational groups.

1. **Determine the fault-tolerance requirements of RD**Licensing. If RD Licensing is unavailable, new clients and clients whose licenses have expired can be denied access to the remote desktop servers. To reduce the possibility of this occurring, place two license servers with half of the licenses installed on each server. Then publish both of the servers in AD DS.

## Task 4: Design and Place the RD Gateway Servers

External users can connect from an unsecured location to an RD Session Host server farm in the following ways:

* Through a Virtual Private Network (VPN) that extends the secure zone out to the user and provides access to all corporate applications. In this case, the user is authenticated on the network when the VPN is established.
* Through Microsoft DirectAccess, available in Windows Server 2008 R2 and Windows 7.
* Connecting to RD Gateway over RDP. This provides an encrypted connection but requires that the RDP port (3389) be open in the firewall.
* Connecting to RD Gateway over RDP that is encapsulated in Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS). This provides easy Network Address Translation (NAT) and firewall traversal without requiring the RDP port (3389) to be open to the Internet.
* Connecting to RD Gateway through Microsoft Forefront® Unified Access Gateway (UAG). In this case, Forefront UAG can provide an additional level of authentication in front of the RD Gateway, it can terminate Secure Sockets Layer (SSL), and it can be used to implement Network Access Protection (NAP).

When an external client connects to the Remote Desktop Services environment through RD Gateway, RD Gateway acts as a security broker, performing client authentication by calling back-end services. It accepts the connection and authenticates the client through Remote Desktop connection authorization policies (RD CAPs) and Remote Desktop resource authorization policies (RD RAPs) that are called from AD DS. It may also call NAP to test the client’s health. Figure 3 illustrates these connections.

Once authorization is complete, the RD Gateway role service connects the client to the requested server or the RD Session Host server farm through the firewall. The RD Session Host server then performs a Windows authentication challenge with the user. If the user passes authentication, the Remote Desktop Services session can begin.

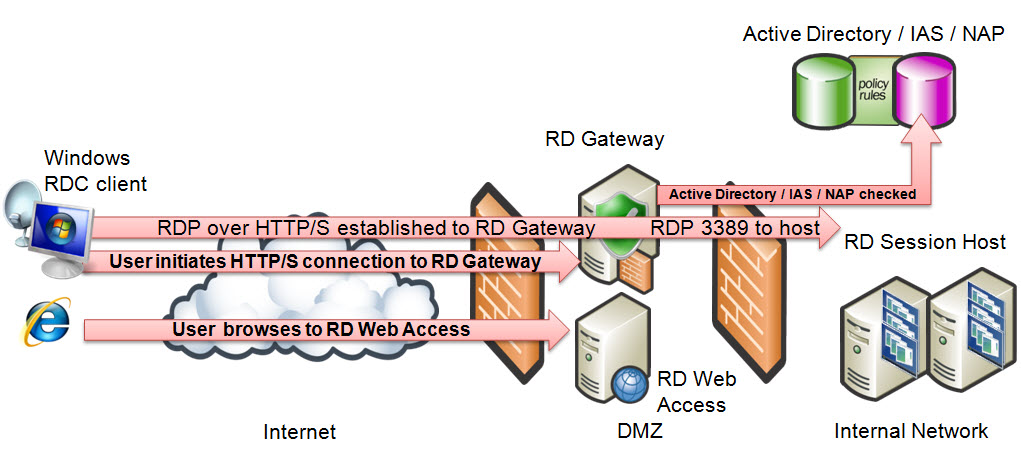


Figure 3. How RD Gateway enables access to the Remote Desktop Services environment for external clients

RD Gateway can also be used to broker authentication within the corporate network if business requirements necessitate.

RD Gateway works with the RDC client to deliver these capabilities and requires RDC client 6.0 or later.

Using the following process, decide where RD Gateway role services are needed and in what format, and determine which farm connects to each RD Gateway:

1. Determine whether the RD Gateway role service is required. First determine whether clients will connect to the RD Session Host server farm through the RD Gateway.

If clients do not require connection to the farm from outside a firewall, RD Gateway is not required.

1. Determine the required security model and the location to place the RD Gateway role. The RD Gateway role can be placed either outside or inside the secure network, and this determines the security model:
2. Place the RD Gateway role service outside the secure network in a screened subnet or perimeter network. In this case, the client connects to the RD Gateway from the Internet over TLS/SSL through port 443 of the RD Gateway server.

This is the simplest possible configuration for RD Gateway and requires the fewest servers. The downside of this configuration is that it requires that RD Gateway be a member of the Active Directory domain, and so the domain must be extended into the perimeter network. Also, port 3389, the default RDP port, will need to be opened behind RD Gateway.

1. Place the RD Gateway role service inside the secure network. In this case, Microsoft Internet Security and Acceleration (ISA) Server or a dedicated SSL terminator is needed to remove SSL encryption.

Port 443 will be open on the external firewall, and either port 443 or port 80 will be on the internal firewall. AD DS is not exposed, nor is port 3389. ISA Server also adds another level of protection as a layer 7 firewall by monitoring the packets bound for RD Gateway and can shut down a suspicious connection.

Decide whether RD Gateway will be placed inside or outside the firewall.

RD Gateway can also be configured to request that clients supply statements of health to the NAP service hosted on the Network Policy Server (NPS). If the client passes the health check policies, it is allowed to complete the connection. If it does not have the proper health levels, it is required to update before connecting.

Determine whether security policy requires NAP to be implemented.

RD Gateway performs a critical part of the client logon process and needs to communicate with the RD Session Host server farm and with the AD DS domain at LAN speeds. Place the RD Gateway role service so that it has high-speed connections to the domain server and the farm.

1. Determine the demand on RD Gateway. At the time of writing, no guidance is available on capacity and performance planning for the RD Gateway role service. However, scalability testing for the TS Gateway role service has been performed by the product group on Windows Server 2008 using sample load test scenarios and is documented at <http://download.microsoft.com/download/9/9/0/990839D2-ED14-4ACA-BFBF-8A40F2A6D2CE/TSG_Scalability_Whitepaper_Final.docx>. This document shows that:

* CPU cycles are the most critical resource on the TS Gateway.
* The number of users that a TS Gateway can support depends on the amount of traffic that flows on the user session.

Compare the results of these tests with the workloads expected of Remote Desktop Services clients and the acceptable CPU threshold for the servers on which the RD Gateway role will run. Use this to estimate the number of users that could be supported on an RD Gateway on the selected server form factor.

1. Determine the number of servers required. To determine the number of servers required for RD Gateway, divide the total number of users that will connect to the RD Session Host server farm by the number of users that an RD Gateway server was able to support when tested. Complete this process for each farm, and record the value in Table A-4 in Appendix A.
2. Decide whether the servers that host RD Gateway can be shared. Now that the resource requirements for a dedicated RD Gateway role service are known for each farm, a decision can be made on whether the RD Gateway role service can be shared. This can be done in the following ways:
3. The RD Gateway server can be shared between farms. In this case, sum the number of users that has been determined for the RD Gateway server in each farm, and divide by the number of farms. This will produce the number of users that can be supported on each shared RD Gateway server.

To determine the number of servers required for RD Gateway, divide the total number of users that will have sessions with the farms by the number of users that a shared RD Gateway server was able to support when tested. Complete this process for each shared farm, and record the value in Table A-4.

1. The RD Gateway server can host other Remote Desktop Services role services. Complete all the tasks in this step to determine the number of users that the server form factor can support for each of the role services that will be combined on a server. Take the lowest of these numbers; that is the number of users that can be supported on each role-shared server.

Divide the total number of users that will connect to the farm by the number of users that a role-shared server can support to determine the number of servers required for the roles. Complete this process for each shared server, and record the value in Table A-4 in Appendix A.

1. **Determine the fault-tolerance requirements of the RD Gateway** role service. If the RD Gateway role service is unavailable, external clients will not be able to connect to the farm. To reduce the possibility of this occurring, plan to implement a fault-tolerance solution using more than one RD Gateway, and load balance the traffic across those RD Gateway role servers.

Note   The RD Gateway role service cannot be implemented in a Microsoft failover cluster.

Record these decisions in Table A-4 in Appendix A.

## Step Summary

The role servers that can service multiple farms have been designed, sized, and placed for the Remote Desktop Services implementation, and that information has been recorded in Table A-4 in Appendix A. Re-examine the design to ensure that every RD Session Host server farm and every user can access the role services they require.

## Additional Reading

* Remote Desktop Services on the Windows Server 2008 TechCenter page: <http://technet.microsoft.com/en-us/windowsserver/ee236407.aspx>
* “What's New in Remote Desktop Services”: <http://technet.microsoft.com/en-us/library/dd560658(WS.10).aspx>
* TS *Connection* Broker Load Balancing Step-by-Step Guide: <http://go.microsoft.com/fwlink/?LinkID=92670>
* Deploying Remote Desktop Licensing Step-by-Step Guide: <http://technet.microsoft.com/en-us/library/dd983943(WS.10).aspx>
* Deploying Remote Desktop Gateway Step-by-Step Guide: <http://technet.microsoft.com/en-us/library/dd983941(WS.10).aspx>
* “Create a Remote Desktop Gateway Server Farm”:<http://technet.microsoft.com/en-us/library/cc732370.aspx>

# Step 8: Secure the Communications

In the previous step, the Remote Desktop Services role services were sized and placed for the farms. Where RD Session Host servers will connect with clients that are outside the corporate network, there are a number of ways to secure the communications. The level of security selected for each farm will depend on the security requirements of the applications that run in the farm and the capabilities and location of the clients. If clients are all inside the corporate network, then there may be no need to secure the communications except where sensitive transactions need additional protection from the possibility of eavesdropping. In this step, the security implementation between the clients and the remote desktop server will be determined.

## Task 1: Determine the Encryption Level Between Clients and the RD Session Host Server

Remote Desktop Services sessions use native 128-bit RDP encryption by default. However, if a client seeking to connect is unable to support 128-bit encryption, the encryption strength may be reduced to 56-bit.

Whenever possible, set the RD Session Host server to negotiate. This will ensure that the highest level of encryption supported by both the remote desktop server and the client is used. Record this information in Table A-4 in Appendix A.

## Task 2: Determine Whether to Seal the Communications

RDP does not provide authentication to verify the identity of an RD Session Host server, which makes it potentially vulnerable to man-in-the-middle attacks. TLS/SSL encryption can be used to enforce mutual authentication between the client and the server before communications are allowed to proceed. This authentication is effected by a certificate exchange.

Perform an assessment of the risk and potential cost of a man-in-the-middle attack. This will be used in the next step to determine the certification authority. Record this information in Table A-4 in Appendix A.

## Task 3: Determine the Certification Authority

Certificates will be required in order to use RDP or HTTPS communications between the clients and the RD Session Host server. There are three ways to source those certificates so that they are available at the RD Session Host server and at all the clients that will need them:

* **Self-signed certificates**. Certificates are generated locally and are then placed in the RD Gateway or in the SSL terminator if the RD Gateway is in the secure zone. The companion certificates must then be distributed and installed on each of the clients that may connect. No infrastructure setup is required and there is no cost, but the installation of the certificates on every client machine can involve significant work.
* **Trusted third party**. A number of trusted certification authorities (CAs), such as VeriSign, are available that provide a certificate service for a fee. The client and the server both trust the CA and exchange certificates from them. No client install is required, and this can be a very convenient implementation if unmanaged clients will connect to the Remote Desktop Services environment.
* Operate a certificate server. In this case, a certificate server is set up outside the secure zone for the clients to access. It requires investment in hardware and software and involves setup work and ongoing maintenance. In addition, the clients may not all be willing to trust this server.

Determine the total cost in hardware, software, and effort of each of the options for the organization, and weigh that against the convenience that they provide for clients. Once that is done, select the option that is most cost-effective overall. Now compare the cost of that selection against the risk of a man-in-the-middle attack, as determined in the previous task. If the benefit outweighs the cost, implement certificates. Record the CA source in Table A-4.

## Task 4: Determine Whether to Encapsulate with HTTPS

If the clients connect using RDP, port 3389 must be open on the external firewall. Many organizations strive to limit the number of ports that are open to the Internet, often limiting it to ports 80 (HTTP) and 443 (HTTPS).

Determine whether policy requires that only ports 80 and 443 can be open in the external firewall, and if that is the case, implement HTTP or HTTPS communications using the RD Gateway role service. Record this information in Table A-4 in Appendix A.

## Step Summary

The security implementation between the clients and the RD Session Host server was determined in this step. This information was recorded in Table A-4 in Appendix A.

## Additional Reading

Windows Server 2008 Security Compliance Management Toolkit: <http://www.microsoft.com/technet/security/prodtech/windowsserver2008/default.mspx>

# 

# Step 9: Design VDI and the RD Virtualization Host Role

In this step, the VDI infrastructure will be designed, including the RD Virtualization Host role. The design decisions made in this step will be used as input to the design of the Windows Server Hyper-V environment in which the client VMs and the RD Virtualization Host role will run.

If the decision in Step 2, Task 3 was to deliver the workloads using RD Session Host and not use VDI, this step can be skipped.

## Task 1: Design the VDI Environment

Start by deciding how VDI desktops will be provided—as personal or pooled desktops. A comparison of these options is provided in Table 3.

Table 3. Options for Providing VDI Desktops

|  |  |  |  |
| --- | --- | --- | --- |
| **Desktop type** | **VHD storage requirements** | **Management overhead** | **Can be customized with applications delivered by App-V** |
| Personal | High: Virtual hard disk (VHD) space required for every user | High: Many unique VMs to service with updates | Yes |
| Pooled | Medium: Space required only for VHD templates | Medium: Template VMs must be serviced | Yes |

Record the desktop type that will be used for each user workload in Table A-5.

If the desktop will be customized by streaming applications with App-V, refer to *Infrastructure Planning and Design Guide for Microsoft Application Virtualization 4.6 at* <http://go.microsoft.com/fwlink/?LinkId=160978>.

## Task 2: Design the RD Virtualization Host Role

The RD Virtualization Host role service performs the following functions:

* Populates VDI VMs into the RD Connection Broker so that they are available for user connection.
* Wakes up a suspended VM when a user initiates a connection to that VM.
* Reports the status of pool VMs to the RD Connection Broker.

Once the client has established a connection to the VM, the RD Virtualization Host and the RD Connection Broker are no longer involved in the connection. So the only load on the RD Virtualization Host occurs when establishing the VDI connection.

The RD Virtualization Host must be run on the same Hyper-V host as the VMs that it serves. The RD Virtualization Host can connect to only one RD Connection Broker at a time; if that RD Connection Broker is in a failover cluster, the RD Virtualization Host will discover it through DNS name resolution and populate it with the VMs that run on the Hyper-V host.

The goal of this step is to determine the most appropriate type of hardware on which to deploy the RD Virtualization Host servers and to decide how many of those machines will be required. The information gathered in this step will be recorded in Table A-4 in Appendix A. Perform this step only if VDI desktops will be delivered from an RD Virtualization Host.

## Step Summary

The VDI and RD Virtualization Host servers were designed in this step and the information was recorded in Tables A-4 and A-5 in Appendix A.

# Conclusion

This guide has summarized the critical decisions, activities, and tasks required to enable a successful design of a Remote Desktop Services infrastructure based on Windows Server 2008 R2.

The guide has addressed the fundamental decisions and tasks involved in:

* Deciding what workloads are to be delivered by Remote Desktop Services and whether Remote Desktop Services is the right approach to use.
* Determining the resources needed to employ Remote Desktop Services to serve the selected workloads.
* Designing the components, layout, security, and connectivity of the Remote Desktop Services infrastructure.

This was done by leading the reader through the nine steps in the decision flow to arrive at a successful design. Where appropriate, the decisions and tasks have been illustrated with typical usage scenarios.

The guide has discussed the technical aspects, service characteristics, and business requirements needed to complete a comprehensive review of the decision-making process.

Once an architecture has been drafted, limited “pilot” tests should be conducted before a major rollout begins so that lessons learned can be incorporated back into the design.

This guide, when used in conjunction with product documentation, allows organizations to confidently plan the implementation of Remote Desktop Services technologies in Windows Server 2008 R2.

## Additional Reading

* Remote Desktop Services Team Blog: <http://blogs.msdn.com/rds/>
* Remote Desktop Services installed Help topics: <http://technet.microsoft.com/en-us/library/cc770412.aspx>
* “Microsoft Application Virtualization 4.5 for Terminal Services”: <http://www.softgridblog.com/?p=142>

# 

# Appendix A: Job Aids

This section provides job aid examples to record data when planning the Remote Desktop Services infrastructure.

Step 1. Use the table below to record information relative to the Remote Desktop Services project scope.

Table A-1. Project Scope

| Location boundary | Workload delivery to users | Workload #1 | Workload #2 |
| --- | --- | --- | --- |
| Enterprise-wide | Session virtualization | Seattle 01 | Seattle 02 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Step 2. Use Table A-2 to collect the required user and application data from interviews with representative users.

Table A-2. User and Application Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User location | Chicago | Denver |  |  |
| Number of users | 200 | 150 |  |  |
| Client operating system | Windows XP SP2 | Windows Vista® |  |  |
| Encryption requirements | 128-bit | 128-bit |  |  |
| SSO | Y | Y |  |  |
| RDC | 6 | 6.1 |  |  |
| Applications used | Microsoft Office Word 2007  No local customization  No SLA  Local support | Microsoft Office Word 2007  No local customization  No SLA  Local support |  |  |
| Purchasing  No local customization  SLA: RT<2 sec  Local support | Purchasing  No local customization  SLA: RT<2 sec  Local support |
| Connection type | LAN-100M | LAN-100M |  |  |
| User type | Normal | Light |  |  |

Steps 2–4. Use Table A-3 to record data about an application’s suitability for delivery by Remote Desktop Services with Windows Server 2008 R2.

Table A-3. Application Analysis

|  |  |  |
| --- | --- | --- |
| Application name | Purchasing V4 | Customer orders |
| Can be licensed | Y | Y |
| License cost accelerates | N | N |
| Supported | Y | Y |
| Legal restrictions | N | N |
| User always connected to network | Y | Y |
| Client hardware | N | N |
| Encryption level correct | Y | Y |
| Video | Low | Medium |
| Audio | None | None |
| Compatible with Windows Server 2008 RD Session Host | Y | Y |
| 16-bit | Y | N |
| User requires administrative rights | N | N |
| Resource consumption | Normal | Normal |
| Install/uninstall | Normal | Normal |
| Shared files | OK | OK |
| Separate user data | OK | OK |
| Registry compliance | OK | OK |
| Isolated user options | OK | OK |
| Application can be run in a VM | Y | Y |
| Business viability | N | Y |
|  | | |
| Good candidate | Y | Y |
| Some issues | Y | N |
| Not suitable | N | Y |

Steps 3–9. Use Table A-4 to record the decisions made in designing each remote desktop server farm.

Table A-4. Farm Design

| Farm name | Reason for design | Applications | **Users** | **Users travel?** |
| --- | --- | --- | --- | --- |
| **Chicago-4** | RDP 7 clients | **Microsoft Office Visio®** | Design engineers - north | N |
| **Office Visio** | Design engineers - south | Y |
| Office Word | Finance Publishing | N |
| Microsoft Office Excel® | Finance | N |
|  |  |  |

Table A-4. Farm Design (continued)

| Number of RD Session Host servers and form factor | User profiles | User profile store | User data store | Number of RD Connection Broker servers and form factor | Fault tolerance |
| --- | --- | --- | --- | --- | --- |
| 4 of 2.66, 64-bit CPU, 4 GB of memory, RAID0+1,  12 terabytes,  3 network adapters | Mandatory | 50 MB, on file server G27-336 | 1 terabyte, on file server G27-338 | 4 of 2.66, 64-bit CPU, 4 GB of memory, RAID0+1,  12 terabytes,  3 network adapters | Active-Passive Cluster |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table A-4. Farm Design (continued)

| Locations of RD Connection Broker servers and farms shared with | Number of RD Web Access servers and form factor | Fault tolerance | Locations of RD Web Access servers and farms shared with | Number of RD Licensing servers and form factor |
| --- | --- | --- | --- | --- |
| Chicago data center  Chicago-7 Farm | On 9 IIS servers in IIS server farm A | Windows NLB | Chicago data center  Chicago-7 Farm | 4x2.66, 64-bit CPU, 4 GB of memory, RAID0+1,  12 terabytes,  3 network adapters |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table A-4. Farm Design (continued)

| Fault tolerance | Locations of RD Licensing servers and farms shared with |
| --- | --- |
| Hardware load balancer | Chicago data center  Chicago-7 Farm |
|  |  |
|  |  |
|  |  |
|  |  |

Table A-4. Farm Design (continued)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RD Licensing | 4x2.66, 64-bit CPU, 4 GB of memory, RAID0+1,  12 terabytes,  3 network adapters | 2 | Chicago-7  Chicago-5 | Chicago-4 Data center | Hardware load balancer |
| RD Gateway | 4x2.66, 64-bit CPU, 4 GB of memory, RAID0+1,  12 terabytes,  3 network adapters | 3 | N | Chicago-4 Data center | Hardware load balancer |
| RD Virtualization Host | 4x2.66, 64-bit CPU, 4 GB of memory, RAID0+1,  12 terabytes,  3 network adapters |  |  |  |  |
|  |  |  |  |  |  |
| Encryption level between clients and remote desktop server | 128-bit |  |  |  |  |
| Use TSL/SSL encryption | Y |  |  |  |  |
| Certification authority | VeriSign |  |  |  |  |
| HTTP/HTTPS | HTTPS |  |  |  |  |

Table A-5. VDI Desktop Types

| Users | Desktop type |
| --- | --- |
| North America sales | <Personal or Pooled> |
|  |  |
|  |  |
|  |  |

# Appendix B: Server Performance Analyzing and Scaling

The following information identifies important monitoring counters used for capacity planning and performance monitoring of a system.

These counters are referenced in Step 5: “Design the Farm,” in which the form factor and size of the remote desktop servers and the RD Web Access server are determined. They are also referenced in Step 7: “Size and Place the Remote Desktop Services Role Services for the Farms*,*” where the form factor and sizing for each of the Remote Desktop Services role services is determined.

Proper design and sizing of the remote desktop server and the Remote Desktop Services role services is critical to the effective operation of the environment, particularly since the servers will host a somewhat unusual workload: multiple copies of applications that were likely originally designed to run on client workstations. The variety of components in this workload means that there is no “one size fits all” answer for remote desktop server sizing, so careful measurements and testing must be performed in order to arrive at a design that is capable of meeting users’ expectations.

## Processor Utilization

Over-committing CPU resources can adversely affect all the workloads on the same server, causing significant performance issues for a larger number of users. Because CPU resource use patterns can vary significantly, no single metric or unit can quantify total resource requirements. At the highest level, measurements can be taken to see how the processor is utilized within the entire system and whether threads are being delayed. Table B-1 lists the performance counters for capturing the overall average processor utilization and the number of threads waiting in the processor Ready Queue over the measurement interval.

Table B-1. Performance Monitor Counters for Processor Utilization

| Object | Counter | Instance |
| --- | --- | --- |
| Processor | % Processor Time | \_Total |
| System | Processor Queue Length | N/A |

### Processor:Percent Processor Time

As a general rule, processors that are running for sustained periods at greater than 90 percent busy are running at their CPU capacity limits. Processors running regularly in the 75–90 percent range are near their capacity constraints and should be closely monitored. Processors regularly reporting 20 percent or less utilization can make good candidates for consolidation.

For response-oriented workloads, sustained periods of utilization above 80 percent should be investigated closely as this can affect the responsiveness of the system. For throughput-oriented workloads, extended periods of high utilization are seldom a concern, except as a capacity constraint.

Unique hardware factors in multiprocessor configurations and the use of Hyper-threaded logical processors raise difficult interpretation issues that are beyond the scope of this document. Additionally, comparing results between 32-bit and 64-bit versions of the processor are not as straightforward as comparing performance characteristics across like hardware and processor families. A discussion of these topics can be found in Chapter 6, “Advanced Performance Topics” in the *Microsoft Windows Server 2003 Performance Guide*.

### System:Processor Queue Length

The Processor Queue Length can be used to identify if processor contention, or high CPU-utilization, is caused by the processor capacity being insufficient to handle the workloads assigned to it. The Processor Queue Length shows the number of threads that are delayed in the processor Ready Queue and are waiting to be scheduled for execution. The value listed is the last observed value at the time the measurement was taken.

On a machine with a single processor, observations where the queue length is greater than 5 is a warning sign that there is frequently more work available than the processor can handle readily. When this number is greater than 10, then it is an extremely strong indicator that the processor is at capacity, particularly when coupled with high CPU utilization.

On systems with multiprocessors, divide the queue length by the number of physical processors. On a multiprocessor system configured using hard processor affinity (that is, processes are assigned to specific CPU cores), which have large values for the queue length, can indicate that the configuration is unbalanced.

Although Processor Queue Length typically is not used for capacity planning, it can be used to identify if systems within the environment are truly capable of running the loads or if additional processors or faster processors should be purchased for future servers.

## Memory Utilization

In order to sufficiently cover memory utilization on a server, both physical and virtual memory usage needs to be monitored. Low memory conditions can lead to performance problems, such as excessive paging when physical memory is low, to catastrophic failures, such as widespread application failures or system crashes when virtual memory becomes exhausted (see Table B-2).

Table B-2. Performance Monitor Counters for Memory Utilization

| Object | Counter | Instance |
| --- | --- | --- |
| Memory | Pages/sec | N/A |
| Memory | Available Mbytes | N/A |
| Memory | Pool Paged Bytes | N/A |
| Memory | Pool Paged Resident Bytes | N/A |
| Memory | Transition Faults/sec | N/A |
| Memory | Committed Bytes | N/A |
| Process | Working Set | <Process Name> |

### Memory:Pages/sec

As physical RAM becomes scarce, the virtual memory manager will free up RAM by transferring the information in a memory page to a cache on the disk. Excessive paging to disk might consume too much of the available disk bandwidth and slow down applications attempting to access their files on the same disk or disks. The Pages/Sec counter tracks the total paging rates, both read and writes, to disk.

For capacity planning, watch for upward trends in this counter. Excessive paging can usually be reduced by adding additional memory. Add memory when paging operations absorbs more than 20–50 percent of the total disk I/O bandwidth. Because disk bandwidth is finite, capacity used for paging operations is unavailable for application-oriented file operations.

The Total Disk I/O Bandwidth is a ratio of the Pages/sec and the Physical Disk:Disk Transfers/sec for all disks in the system:

Memory:Pages/sec ÷ Physical Disk (\_Total):Disk Transfers/sec

### Memory:Available Mbytes

The Available Mbytes displays the amount of physical memory, in megabytes, that is immediately available for allocation to a process or for system use. The percent Available Megabytes can be used to indicate if additional memory is required. Add memory if this value drops consistently below 10 percent. To calculate the percent of Available Megabytes:

(Memory:Available Mbytes ÷ System RAM in Megabytes) \* 100

This is the primary indicator to determine whether the supply of physical memory is ample. When memory is scarce, Pages/sec is a better indicator of memory contention. Downward trends can indicate a need for additional memory. Counters are available for Available Bytes and Available Kbytes.

### Memory:Pool Paged Bytes and Memory:Pool Paged Resident Bytes

The Pool Paged Bytes is the size, in bytes, of the paged pool, an area of system memory used by the operating system for objects that can be written to disk when they are not being used.

The Pool Paged Resident Bytes is the size, in bytes, of the nonpaged pool, an area of system memory used by the operating system for objects that can never be written to disk, but must remain in physical memory as long as they are allocated.

A ratio of Pool Paged Bytes to Pool Paged Resident Bytes can be calculated by:

Memory:Pool Paged Bytes ÷ Memory:Pool Paged Resident Bytes

This ratio can be used as a memory contention index to help in planning capacity. As this approaches zero, additional memory needs to be added to the system to allow both the Nonpaged pool and Page pool to grow.

The size returned by the Pool Paged Resident Bytes can be used for planning additional TCP connections. Status information for each TCP connection is stored in the Nonpaged pool. By adding memory, additional space can be allocated to the Nonpaged pool to handle additional TCP connections.

### Memory:Transition Faults/sec

The Transition Faults counter returns the number of soft or transition faults during the sampling interval. Transition faults occur when a trimmed page on the Standby list is re-referenced. The page is then returned to the working set. It is important to note that the page was never saved to disk.

An upward trend is an indicator that there may be a developing memory shortage. High rates of transition faults on their own do not indicate a performance problem. However, if the Available Megabytes is at or near its minimum threshold value, usually 10 percent, then it indicates that the operating system has to work to maintain a pool of available pages.

### Memory:Committed Bytes

The Committed Bytes measures the amount of committed virtual memory. Committed memory is allocated memory that the system must reserve space for either in physical RAM or on the paging file so that this memory can be addressed by threads running in the associated process.

A memory contention index called the Committed Bytes:RAM can be calculated to aid in capacity planning and performance. When the ratio is greater than 1, virtual memory exceeds the size of RAM and some memory management will be necessary. As the Committed Bytes:RAM ratio grows above 1.5, paging to disk will usually increase up to a limit imposed by the bandwidth of the paging disks. Memory should be added when the ratio exceeds 1.5. The Committed Bytes:RAM is calculated by:

Memory:Committed Bytes ÷ System RAM in Bytes

### Process:Working Set

The Working Set counter shows the amount of memory allocated to a given process that can be addressed without causing a page fault to occur. To see how much RAM is allocated overall across all process address spaces, use the \_Total instance of Working Set. Watch for upward trends for important applications.

Some server applications, such as IIS, Exchange Server, and SQL Server, manage their own process working sets. To measure their working sets, application-specific counters need to be used.

## Disk Storage Requirements

The process of planning for storage requirements is divided into capacity requirements and disk performance. Although a total capacity requirement can be determined, the performance requirements, as well as fault tolerance requirements, of the system will have an impact on the implementation of the storage subsystem. For example, a single drive could provide enough storage space, but the performance of that single disk may not meet the performance needs of the system.

Due to this, both capacity and performance requirements need to be met, which may alter the decision around the size, speed, and configuration of the drives in the storage subsystem.

### Disk Space Capacity

The amount of storage capacity required can be calculated based on OS requirements as well as any application-specific data that needs to be stored on the system.

### Disk Performance

Disk performance is typically expressed as a value of IOps, measured over some period of time during peak usage.

To determine the number of disks needed to meet a system’s IOps requirement, the IOps of a given drive needs to be determined (see Table B-3). To further complicate matters, IOps are very dependent upon the access pattern. For example, a drive will typically have a higher IOps rating for sequential reads than it will for random writes. For this reason, it is normal to calculate a “worst case” IOps measurement based on short random I/O operations.

To calculate the IOps of a drive, information about the drive needs to be collected. The following table lists the information required that normally can be found in the manufacturer’s datasheet about the drive.

Table B-3. Information Required for Calculating IOps

| Required information | Description |
| --- | --- |
| Spindle Rotational Speed (RPM) | The spindle speed expressed as RPM. |
| Average Read Seek Time (ms) | The average seek time for reads. |
| Average Write Seek Time (ms) | The average seek time for writes. |

The first step to calculating the IOps is to determine the Average Seek Time in milliseconds that the drive is capable of doing. There is an assumption that there will be a 50/50 mix of read and write operations. If the ratio of read and write operations is modified, then the Average Seek Time will need to be adjusted. For example, if a drive has an Average Read of 4.7 ms and an Average Write of 5.3 ms, the Average Seek Time for this drive will be 5.0 ms:

5.0ms = (4.7ms + 5.3ms) ÷ 2

Next, the I/O Latency needs to be calculated (see Table B-4). This is calculated by adding the Average Seek Time to the Average Latency. The following table lists the Average Latency of common spindle speeds of drives on the market today.

Table B-4. Average Latency Based on Spindle Rotational Speeds

| Spindle Rotational Speed (rpm) | Average Latency (ms) |
| --- | --- |
| 4,200 | 7.2 |
| 5,400 | 5.6 |
| 7,200 | 4.2 |
| 10,000 | 3.0 |
| 15,000 | 2.0 |

The example drive has a spindle speed of 10,000 RPM. So this drive has an I/O Latency of 8.0 ms:

8.0 ms = 5.0ms + 3.0ms

A drive can only perform one I/O operation at a time. To calculate the number of I/Os that can be performed in a millisecond, 1 is divided by the I/O Latency. Finally, this value is converted to I/O per Second by multiplying by 1000. The IOps calculation for the example drive evaluates to 125 IOps:

125 IOps = (1 I/O ÷ 8.0ms) \* 1000 ms/sec

## Storage Requirements

To determine storage requirements, additional information is needed to be collected around the system being considered. Some of this information is easily identified and self-explanatory, while other information may be more difficult to identify due to lack of quantifiable data. All of the following calculations are for a single server; although if shared storage systems are being considered, then the information can be scaled up based on the number of systems sharing that storage. Table B-5 shows the information that needs to be collected.

Table B-5. Information Required for Calculating Storage Requirements

| Required information | Description | Example |
| --- | --- | --- |
| # Users Per Server | Total number of users hosted by that server. | 700 |
| % Concurrent Users | The percentage of users connected to the server during peak times. | 80% |
| IOps per User Required | The number of IOps generated by a user. | 0.5 |
| Storage Capacity in Gigabytes | The calculated disk storage capacity needed. | 450 |
| % Buffer Factor (for growth) | The percentage of disk storage growth allowed by the system. | 20% |
| Read % of IOps | The percentage of IOps that are Read operations. | 50% |
| Write % of IOps | The percentage of IOps that are Write operations. | 50% |
| Disk Size (GB) | The drive size being considered for the storage system. | 146 |
| Calculated Drive IOps | The calculated IOps of the drive being considered for the storage system. | 125 |

The information in Table B-5 is fairly self-explanatory with the exception of the IOps per user required. This is a measurement of the number of IOps that a single user will generate on the system. Most venders do not have this information for their applications unless the application is extremely I/O intensive. This information may be calculated from observation of a running system, but it is fraught with a number of challenges and is beyond the scope of this guide. For the purpose of this example, this guide will use the value used with Microsoft Exchange Server, which is 0.5 IOps per user.

Based on the information in Table B-5, there are a number of derived values that need to be calculated. Table B-6 lists these values.

Table B-6. Derived Information Required for Calculating Storage Requirements

| Required information | Description | Example |
| --- | --- | --- |
| # of Concurrent Users | The number of concurrent users calculated by calculating the percentage of concurrent users and the number of users per server. | 560 |
| IOps per Server Required | The number of IOps generated by each user multiplied by the number of concurrent users. | 280 |
| Total Storage Requirements | The Storage Capacity increased by the percentage of growth allowed. | 540 |
| Number of Read IOps | The percentage of Reads and the IOps per Server Required. | 140 |
| Number of Write IOps | The percentage of Reads and the IOps per Server Required. | 140 |
| Drive Size Actual (GB) | After formatting, drives capacity is roughly 10 percent less than the advertised total capacity of the disk. This value adjusts for this loss of space. This value is calculated by taking 90 percent of the Disk Size (GB). | 132 |

RAID 0+1 and RAID 5 are the most common drive configurations for storing data in a redundant manner on a system. However, these two RAID systems have different IOps calculations due to how they operate.

### RAID 0+1 Calculations

To calculate the number of drives necessary to meet the storage requirements with RAID 0+1, divide the Total Storage Requirements by the Drive Size Actual. Round the result up and multiply by 2. For this example, 10 drives are needed for RAID 0+1 to meet the storage requirements:

10 = ROUNDUP(540÷132)\*2

To calculate the number of drives required to meet the performance requirements with RAID 0+1, multiply the Number of Write IOps by 2 and add the Number of Read IOps. Divide this total by the Calculated Drive IOps and round the result up. For this example, 4 drives are needed for RAID 0+1 to meet the performance requirements:

4 = ROUNDUP(((140\*2)+140)÷125)

Although RAID 0+1 can meet the performance requirements with just 4 disks, 10 disks are required to meet the capacity requirements.

### RAID 5 Calculations

To calculate the number of drives necessary to meet the storage requirements with RAID 5, the Total Storage Requirements needs to be multiplied by 1.2 to adjust for parity storage requirements. This value is then divided by the Drive Actual Size and then rounded up. For this example, five drives are needed for RAID 5 to meet the storage requirements:

5 = ROUNDUP((540\*1.2)÷132)

To calculate the number of drives required to meet the performance requirements with RAID 5, multiply the Number of Write IOps by 4 and add the Number of Read IOps. Divide this total by the Calculated Drive IOps and round the result up. For this example, six drives are needed for RAID 5 to meet the performance requirements:

6 = ROUNDUP(((140\*4)+140)÷125)

Although RAID 5 can meet the storage requirements with just five disks, six disks are required to meet the capacity requirements.

### RAID 0+1 versus RAID 5 Calculations

As can be seen in this example, RAID 5 looks to be the better choice when using 10K 146-GB drives. However, it is important to look at different types of drives when doing these calculations. For example, if a drive that has 300 GB is substituted for the 146-GB drives and if all other characteristics remain the same, then the choices drastically change.

Using 300-GB drives, RAID 0+1 requires just four drives to meet both capacity and performance characteristics. RAID 5 will require three drives to meet capacity, but six drives to meet performance requirements. By changing the size of the drive, the best choice changed, as well.

## Storage Monitoring

IOps are used to help characterize the performance requirements of the system. However, once the system is up, additional performance monitoring can be utilized to determine whether the disk subsystem is slowing the system down (see Table B-7).

Table B-7. Performance Monitor Counters for Disk Performance

| Object | Counter | Instance |
| --- | --- | --- |
| Physical Disk | % Idle Time | <All Instances> |
| Physical Disk | Disk Transfers/sec | <All Instances> |
| Physical Disk | Avg. Disk secs/Transfers | <All Instances> |
| Physical Disk | Split IO/sec | <All Instances> |

### Physical Disk:Percent Idle Time

The Percent Idle Time counter is the percent of time that a disk was idle during the sampling interval. An idle time of less than 20 percent indicates that the disk may be overloaded.

Physical Disk(n):Disk utilization can be derived by subtracting the Physical Disk(n):Percent Idle Time from 100 percent.

### Physical Disk:Disk Transfers/sec

The Disk Transfers/sec is the number I/O request Packets (IRP) that has been completed during the sampling interval. A disk can only handle one I/O operation at a time. So the number of physical disks attached to the computer serves as an upper bound on the sustainable disk I/O rate. Where disk arrays are used, divide the Disk Transfers/sec by the number of disks in the array to estimate individual disk I/O rates.

The Physical Disk(n):Average Disk Service Time/Transfer can be calculated by taking the Physical Disk(n):Disk Utilization and dividing it by the Physical Disk(n):Disk Transfers/sec for each physical disk. This indicates how fast the drive responds to a request. If this climbs above what is specified for the disk, it can indicate that the subsystem is overloaded.

### Physical Disk:Avg. Disk secs/transfers

The Avg. Disk secs/transfer is the overall average of the response time for physical disk requests during the sampling interval. This includes both the time the request was serviced by the device and the time it spent waiting in the queue. If this climbs to over 15–25 disk IOps per disk, then the poor disk response time should be investigated.

The Physical Disk(n):Average Disk Queue Time/Transfer can be calculated by taking the Physical Disk(n):Avg. Disk secs/Transfer and subtracting the Physical Disk(n):Avg. Disk Service Time/Transfer. The Average Disk Queue Time/Transfer indicates the amount of time a request spent waiting in the disk queue for servicing. A high queue time can indicate a poorly responding disk subsystem or specifically a poorly responding physical disk.

### Physical Disk:Split Input Output per Second

Split IOps is the rate physical disk requests were split into multiple disk requests during the sampling interval. A large number of split IOps indicates that the disk is fragmented and performance is being affected. The percentage of Split I/Os can be calculated by the following formula where “n” is a specific disk:

(Physical Disk(n):Split IO/sec ÷ Physical Disk(n):Disk Transfers/sec) \* 100

If this percentage is greater than 10 to 20 percent, check to see whether the disk needs to be defragmented.

## Network Performance

Most workloads require access to production networks to ensure communication with other applications and services and to communicate with users. Network requirements include elements such as throughput—that is, the total amount of traffic that passes a given point on a network connection per unit of time (see Table B-8). Other network requirements include the presence of multiple network connections.

Table B-8. Performance Monitor Counters for Network Performance

| Object | Counter | Instance |
| --- | --- | --- |
| Network Interface | Bytes Total/sec | (Specific network adapters) |
| Network Interface | Current Bandwidth | (Specific network adapters) |
| IPv4 & IPv6 | Datagrams/sec | N/A |
| TCPv4 & TCPv6 | Connections Established | N/A |
| TCPv4 & TCPv6 | Segments Received/sec | N/A |

### Network Interface:Bytes Total/sec and Network Interface:Current Bandwidth

This Bytes Total/sec is the number of bytes transmitted and received over the specified interface per second The Current Bandwidth counter reflects the actual performance level of the network adaptor, not its rated capacity. If a gigabit network adapter card on a segment is forced to revert to a lower speed, the Current Bandwidth counter will reflect the shift from 1 Gps to 100 Mpbs.

Using these two values, the Network interface utilization can be calculated for each interface, designated as “n” with the following equation:

(Network Interface(n):Bytes Total/sec ÷ Network Interface(n):Current Bandwidth) \*100

If Percent Busy for a given adapter exceeds 90 percent, then additional network resources will be necessary. Generally, the maximum achievable bandwidth on a switched link should be close to 90–95 percent of the Current Bandwidth counter.

### IPv4 & IPv6:Datagrams/sec

These counters show the total number of IP datagrams transmitted and received per second during the sampling interval. By generating a baseline around this counter, a trending and forecasting analysis of the network usage can be performed.

### TCPv4 & TCPv6:Connections Established

The Connections Established counter is the total number of TCP connections in the ESTABLISHED state at the end of the measurement interval. The number of TCP connections that can be established is constrained by the size of the Nonpaged pool. When the Nonpaged pool is depleted, no new connections can be established.

Trending and forecasting analysis on this counter can be done to ensure that the system is properly scaled to handle future growth. The server can be tuned using the TCP registry entries like *MaxHashTableSize* and *NumTcTablePartitions* based on the number of network users seen on average.

### TCPv4 & TCPv6:Segments Received/sec

The Segments Received/sec is the number of TCP segments received across established connections, averaged over the sampling interval. The average number of segments received per connection can be calculated. This can be used to forecast future load as the number of users grows. The following formula can be used to calculate the average number of segments received per connection:

TCPvn:Segments Received/sec ÷ TCPvn:Connections Established/sec

## Windows Server–based File Servers

For Windows Server–based file servers, additional performance counters are available in addition to the ones mentioned previously, which can be used to monitor the performance of the system. Table B-9 lists the most important counters.

Table B-9. Performance Monitor Counters for File Server Performance

| Object | Counter | Instance |
| --- | --- | --- |
| Server | Work Item Shortages | N/A |
| Server Work Queues | Available Threads | <All Instances> |
| Server Work Queues | Queue Length | <All Instances> |

### Server:Work Item Shortages

The Work Item Shortages counter indicates the number of times a shortage of work items caused the file server to reject a client request. This error usually results in session termination. This is the primary indicator that the File Server service is short on resources.

Server Message Block (SMB) requests are stored in a work item and assigned to an available worker thread. If there are not enough available threads, the work item is placed in the Available Work Items queue. If this queue becomes depleted, then the server can no longer process SMB requests.

### Server Work Queues: Available Threads

The Available Threads reports the number of threads from the per-processor Server Work Queue that are available to process incoming SMB requests. When the number of available threads reaches 0, incoming SMB requests must be queued. This is the primary indicator for identifying if the number of work threads defined for the per-processor Server Work queues is a potential bottleneck.

The MaxThreadsPerQueue registry DWORD value located at HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters controls the number of threads that are created in the per-processor thread pools. By default, the system will create 10 threads per-processor thread pool. This value should be increased when the following conditions hold true:

* Available Threads is at or near zero for a sustained period.
* Queue Length of waiting requests is greater than 5.
* Percent Processor Time for the associated processor instance is less than 80.

By increasing the available threads, additional work items can be handled; however, care should be taken not to overstress the system with the addition of the extra threads.

### Server Work Queues: Queue Length

The Queue Length counter reports the number of incoming SMB requests that are queued for processing, waiting for a worker thread to become available. There are separate per-processor Server Work queues to minimize inter-processor communication delays.

This is a primary indicator to determine whether client SMB requests are delayed for processing at the file server. It can also indicate that the per-processor Work Item queue is backed up because of a shortage of threads or processing resources.

If any of these queues grows beyond 5, then the underlying reason for this growth should be investigated. File server clients whose sessions are terminated because of a work item shortage must restore their sessions manually.

# Appendix C: IPD in Microsoft Operations Framework 4.0

Microsoft Operations Framework (MOF) offers integrated best practices, principles, and activities to assist an organization in achieving reliable solutions and services. MOF provides guidance to help individuals and organizations create, operate, and support technology services, while helping to ensure the investment in technology delivers expected business value at an acceptable level of risk. MOF’s question-based guidance helps to determine what is needed for an organization now as well as providing activities that will keep the organization running efficiently and effectively in the future.

Use MOF with the IPD guides to ensure that people and process considerations are addressed when changes to an organization’s technology services are being planned (see Figure C-1):

* Use the Plan Phase to maintain focus on meeting business needs, consider business requirements and constraints, and align business strategy with the technology strategy. IPD helps to define an architecture that delivers the right solution as determined in the Plan Phase.
* Use the Deliver Phase to build solutions and deploy updated technology. In this phase, IPD helps IT pros design their technology infrastructures.
* Use the Operate Phase to plan for operations, service monitoring and control, and troubleshooting. The appropriate infrastructure, built with the help of IPD guides, can increase the efficiency and effectiveness of operating activities.
* Use the Manage Layer to work effectively and efficiently to make decisions that are in compliance with management objectives. The full value of sound architectural practices embodied in IPD will help deliver value to the top levels of a business.



Figure 4. The architecture of MOF 4.0

# Appendix D: Remote Desktop Services in Microsoft Infrastructure Optimization

The Infrastructure Optimization (IO) Model at Microsoft groups IT processes and technologies across a continuum of organizational maturity (for more information, see <http://www.microsoft.com/infrastructure>). The model was developed by industry analysts, the Massachusetts Institute of Technology (MIT) Center for Information Systems Research (CISR), and Microsoft based on its experience with its own enterprise customers. A key goal for Microsoft in creating the IO Model was to develop a simple way to use a maturity framework that is flexible and that can easily be used as the benchmark for technical capability and business value.

IO is structured around three information technology models: Core IO, Application Platform Optimization, and Business Productivity IO (see Figure 5). According to the Core IO Model, organizations that are implementing presentation virtualization for production workloads with Windows Server 2008 R2 are meeting one of the requirements to move to the Rationalized maturity level.

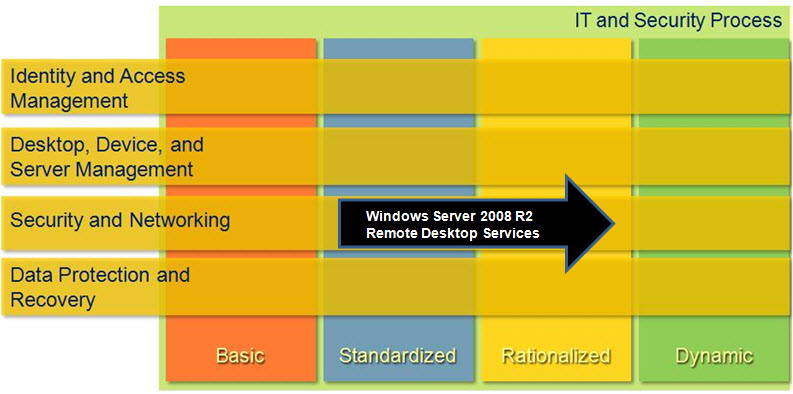


Figure 5. Mapping of Windows Server 2008 R2 Remote Desktop Services technology into Core IO

# Version History

|  |  |  |
| --- | --- | --- |
| **Version** | **Description** | **Date** |
| 2.2 | Updated guide to reflect VDI language and minor format changes. | February 2011 |
| 2.1 | Minor updates to this guide to reflect current IPD template. | July 2010 |
| 2.0 | The server role and all role services have been renamed.  In Windows Server 2008 R2, Remote Desktop Services (formerly named Terminal Services) has been expanded to provide a brokering service that connects clients to their virtual desktops in a Virtual Desktop Infrastructure (VDI).  Added the “What’s New for Windows Server 2008 R2 for Remote Desktop Services”: Remote Desktop Virtualization Host (RD Virtualization Host) is a new Remote Desktop Services role service included with Windows Server 2008 R2. RD Virtualization Host requires and integrates with the Hyper-V role to provide Virtual Desktop Infrastructure (VDI) virtual machines that can be used as personal virtual desktops or virtual desktop pools. | November 2009 |
| 1.0 | First release. | July 2008 |

# Acknowledgments

The Solution Accelerators – Management and Infrastructure (SA-MI) team acknowledges and thanks the people who produced the Infrastructure Planning and Design Guide for Windows Server 2008 R2 Remote Desktop Services. The following people were either directly responsible for or made a substantial contribution to the writing, development, and testing of this guide.

Contributors:

* Jude Chosnyk – GrandMasters
* Joe Coulombe – Microsoft
* Michael Kaczmarek – Microsoft
* Robin Maher – Microsoft
* Reed G. Porter – Studio B
* Fergus Stewart – Microsoft
* Melissa Stowe – Microsoft

Reviewers:

* Mahadev Alladi – Microsoft
* Christa Anderson – Microsoft
* Alex Balcanquall – Microsoft
* Daniel H. Brown – Microsoft
* Charles Denny – Microsoft
* Max Herrmann – Microsoft
* Olga Ivanova – Microsoft
* Michael Kleef – Microsoft
* Klaas Langhout – Microsoft
* John Morello – Microsoft
* Aurora Santiago – Microsoft
* Tony Soper – Microsoft
* Tessa Wooley – Microsoft

Editors:

* Laurie Dunham – *Xtreme Consulting Group, Inc.*
* Pat Rytkonen – *Volt Technical Services*
* Michelle Anderson – *Xtreme Consulting Group, Inc.*

## Feedback

Please direct questions and comments about this guide to [ipdfdbk@microsoft.com](mailto:ipdfdbk@microsoft.com?subject=IPD%20-%20Remote%20Desktop%20Services).

We value your feedback on the usefulness of this guide. Please complete the following **Solution Accelerators Satisfaction Survey**, available at: <http://go.microsoft.com/fwlink/?LinkID=132579>, and help us build better guidance and tools.