Cisco UCS B200 M4 Blade Server with VMware Horizon 6 and Red Hat Enterprise Linux 6.6

Reference Architecture

January 2016
Executive Summary

VMware recently extended the power of mobility to Linux desktop users. For the first time, customers can provision both Microsoft Windows and Linux desktops using a single management console based on their users’ needs.

With the increased processing power of today’s Cisco UCS® B-Series Blade Servers and C-Series Rack Servers, applications with demanding 3D graphics such as computer-aided design and computer-aided manufacturing (CAD/CAM) can now be deployed on a Linux virtual desktop.

This document is written for customers who want to take advantage of the cost savings, security, and customization options that are available with Linux-based virtual desktops. The objective of this document is to provide the reader with specific configuration guidelines for integrating Cisco UCS B200 M4 Blade Servers with Red Hat Enterprise Linux (RHEL) 6.6 virtual desktops on VMware Horizon View 6.1.

Cisco Unified Computing System

Cisco UCS Manager provides unified, embedded management of all software and hardware components of the Cisco Unified Computing System™ (Cisco UCS) through an intuitive GUI, a command-line interface (CLI), and an XML API. The manager provides a unified management domain with centralized management capabilities and can control multiple chassis and thousands of virtual machines.

Cisco UCS is a next-generation data center platform that unites computing, networking, and storage access. The platform, optimized for virtual environments, is designed using open industry-standard technologies and aims to reduce total cost of ownership (TCO) and increase business agility. The system integrates a low-latency; lossless 10 Gigabit Ethernet unified network fabric with enterprise-class, x86-architecture servers. It is an integrated, scalable, multichassis platform in which all resources participate in a unified management domain.

The main components of Cisco UCS (Figure 1) are:

- **Computing**: The system is based on an entirely new class of computing system that incorporates blade servers based on Intel® Xeon® processor E5-2600/4600 v3 and E7-2800 v3 family CPUs.
- **Network**: The system is integrated on a low-latency, lossless, 10-Gbps unified network fabric. This network foundation consolidates LANs, SANs, and high-performance computing (HPC) networks, which are separate networks today. The unified fabric lowers costs by reducing the number of network adapters, switches, and cables needed, and by decreasing the power and cooling requirements.
- **Virtualization**: The system unleashes the full potential of virtualization by enhancing the scalability, performance, and operational control of virtual environments. Cisco security, policy enforcement, and diagnostic features are now extended into virtualized environments to better support changing business and IT requirements.
- **Storage access**: The system provides consolidated access to local storage, SAN storage, and network-attached storage (NAS) over the unified fabric. With storage access unified, Cisco UCS can access storage over Ethernet, Fibre Channel, Fibre Channel over Ethernet (FCoE), and Small Computer System Interface over IP (iSCSI) protocols. This capability provides customers with choice for storage access and investment protection. In addition, server administrators can preassign storage-access policies for system connectivity to storage resources, simplifying storage connectivity and management and helping increase productivity.
- **Management**: Cisco UCS uniquely integrates all system components, enabling the entire solution to be managed as a single entity by Cisco UCS Manager. The manager has an intuitive GUI, a CLI, and a robust API for managing all system configuration processes and operations.
Cisco UCS is designed to deliver:

- Reduced TCO and increased business agility
- Increased IT staff productivity through just-in-time provisioning and mobility support
- A cohesive, integrated system that unifies the technology in the data center; the system is managed, serviced, and tested as a whole
- Scalability through a design for hundreds of discrete servers and thousands of virtual machines and the capability to scale I/O bandwidth to match demand
- Industry standards supported by a partner ecosystem of industry leaders

Cisco UCS Fabric Interconnect

The Cisco UCS 6200 Series Fabric Interconnects are a core part of Cisco UCS, providing both network connectivity and management capabilities for the system. The Cisco UCS 6200 Series offers line-rate, low-latency, lossless 10 Gigabit Ethernet, FCoE, and Fibre Channel functions.
The fabric interconnects provide the management and communication backbone for the Cisco UCS B-Series Blade Servers and Cisco UCS 5100 Series Blade Server Chassis. All chassis, and therefore all blades, attached to the fabric interconnects become part of a single, highly available management domain. In addition, by supporting unified fabric, the Cisco UCS 6200 Series provides both LAN and SAN connectivity for all blades in the domain.

For networking, the Cisco UCS 6200 Series uses a cut-through architecture, supporting deterministic, low-latency, line-rate 10 Gigabit Ethernet on all ports, 1-terabit (Tb) switching capacity, and 160 Gbps of bandwidth per chassis, independent of packet size and enabled services. The product series supports Cisco low-latency, lossless, 10 Gigabit Ethernet unified network fabric capabilities, increasing the reliability, efficiency, and scalability of Ethernet networks. The fabric interconnects support multiple traffic classes over a lossless Ethernet fabric, from the blade server through the interconnect. Significant TCO savings come from an FCoE-optimized server design in which network interface cards (NICs), host bus adapters (HBAs), cables, and switches can be consolidated.

Cisco UCS B200 M4 Blade Server
The Cisco UCS B200 M4 Blade Server (Figures 2 and 3) is a density-optimized, half-width blade server that supports two CPU sockets for Intel Xeon processor E5-2600 v3 series CPUs and up to 24 DDR4 DIMMs. It supports one modular LAN-on-motherboard (LOM) dedicated slot for a Cisco virtual interface card (VIC) and one mezzanine adapter. In additions, the Cisco UCS B200 M4 supports an optional storage module that accommodates up to two SAS or SATA hard disk drives (HDDs) or solid-state disk (SSD) drives. You can install up to eight Cisco UCS B200 M4 servers in a chassis, mixing them with other models of Cisco UCS blade servers in the chassis if desired.

Figure 2. Cisco UCS B200 M4 Front Panel
Cisco UCS combines Cisco UCS B-Series Blade Servers and C-Series Rack Servers with networking and storage access into a single converged system with simplified management, greater cost efficiency and agility, and increased visibility and control. The Cisco UCS B200 M4 Blade Server is one of the newest servers in the Cisco UCS portfolio.

The Cisco UCS B200 M4 delivers performance, flexibility, and optimization for data centers and remote sites. This enterprise-class server offers market-leading performance, versatility, and density without compromise for workloads ranging from web infrastructure to distributed databases. The Cisco UCS B200 M4 can quickly deploy stateless physical and virtual workloads with the programmable ease of use of the Cisco UCS Manager software and simplified server access with Cisco® SingleConnect technology. Based on the Intel Xeon processor E5-2600 v3 product family, it offers up to 768 GB of memory using 32-GB DIMMs, up to two disk drives, and up to 80 Gbps of I/O throughput. The Cisco UCS B200 M4 offers exceptional levels of performance, flexibility, and I/O throughput to run your most demanding applications.

In addition, Cisco UCS has the architectural advantage of not having to power and cool excess switches, NICs, and HBAs in each blade server chassis. With a larger power budget per blade server, it provides uncompromised expandability and capabilities, as in the new Cisco UCS B200 M4 server with its leading memory-slot capacity and drive capacity.

Cisco UCS B200 M4 Features
The Cisco UCS B200 M4 provides:

- Up to two multicore Intel Xeon processor E5-2600 v3 series CPUs for up to 36 processing cores
- 24 DIMM slots for industry-standard DDR4 memory at speeds up to 2133 MHz, and up to 768 GB of total memory when using 32-GB DIMMs
Two optional, hot-pluggable SAS and SATA HDDs or SSDs

Cisco UCS VIC 1340, a 2-port, 40 Gigabit Ethernet and FCoE–capable modular (mLOM) mezzanine adapter
  - Provides two 40-Gbps unified I/O ports or two sets of four 10-Gbps unified I/O ports
  - Delivers 80 Gbps to the server
  - Adapts to either 10- or 40-Gbps fabric connections

Cisco FlexStorage local drive storage subsystem, with flexible boot and local storage capabilities that allow you to:
  - Configure the Cisco UCS B200 M4 to meet your local storage requirements without having to buy, power, and cool components that you do not need
  - Choose an enterprise-class RAID controller, or go without any controller or drive bays if you are not using local drives
  - Easily add, change, and remove Cisco FlexStorage modules

The Cisco UCS B200 M4 server is a half-width blade. Up to eight can reside in the 6-rack-unit (6RU) Cisco UCS 5108 Blade Server Chassis, offering one of the highest densities of servers per rack unit of blade chassis in the industry.

Cisco UCS B200 M4 Benefits
The Cisco UCS B200 M4 server is well suited for a broad spectrum of IT workloads, including:

- IT and web infrastructure
- Virtualized workloads
- Consolidating applications
- Virtual desktops
- Middleware
- Enterprise resource planning (ERP) and customer-relationship management (CRM) applications
- Single-instance and distributed databases

The Cisco UCS B200 M4 is one member of the Cisco UCS B-Series Blade Servers platform. As part of Cisco UCS, Cisco UCS B-Series servers incorporate many innovative Cisco technologies to help customers handle their most challenging workloads. Cisco UCS B-Series servers within a Cisco UCS management framework incorporate a standards-based unified network fabric, Cisco Data Center Virtual Machine Fabric Extender (VM-FEX) virtualization support, Cisco UCS Manager, Cisco UCS Central Software, Cisco UCS Director software, and Cisco fabric extender architecture.

The Cisco UCS B200 M4 Blade Server delivers:

- Suitability for a wide range of applications and workload requirements
- Highest-performing CPU and memory options without constraints in configuration, power, or cooling
- Half-width form factor that offers industry-leading benefits
- Latest features of Cisco UCS VICs

**Linux Distributions**

A Linux distribution (often called a distro for short) is an operating system composed of a software collection based on a Linux kernel and, often, a package management system.

A typical Linux distribution consists of a Linux kernel, GNU tools and libraries, additional software, documentation, a window system (the most common is the X Window System), a window manager, and a desktop environment. Most of the included software is free, open-source software made available both in compiled binary files and in source-code form, allowing modifications to the original software. Usually, Linux distributions optionally include some proprietary software that may not be available in source-code form, such as binary blobs required for some device drivers. Almost all Linux distributions are UNIX like; the most notable exception is Android, which does not include a CLI or programs made for typical Linux distributions.

Almost 600 Linux distributions exist, with nearly 500 of those in active development, constantly being revised and improved. Because of the wide availability of software, distributions have taken a wide variety of forms, including distributions suitable for use on desktops, servers, laptops, netbooks, mobile phones, and tablets as well as in minimalist environments: typically for use in embedded systems. There are commercially backed distributions, such as Fedora (Red Hat), openSUSE (SUSE), and Ubuntu (Canonical), and entirely community-based distributions, such as Debian, Slackware, Gentoo, Elementary OS, Fedora, Steam OS, and Arch Linux. Most distributions are ready to use and precompiled for a specific instruction set, but some distributions (such as Gentoo) are distributed mostly in source-code form and compiled locally during installation.

**Linux Distribution Package Management**

Distributions are normally segmented into packages. Each package contains a specific application or service. For example, you can get a library package for handling the Portable Network Graphics (PNG) image format: a collection of fonts for a web browser.

A package typically is provided as compiled code, with installation and removal of packages handled by a package management system (PMS) rather than by a simple file archiver. Each package intended for such a PMS contains metadata such as a package description, version number, and dependencies. The package management system can evaluate this metadata to perform package searches, automatically upgrade the package to a newer version, and check that all dependencies of a package are present and, if necessary, add them automatically.

Although Linux distributions typically contain much more software than proprietary operating systems, it is normal for local administrators to also install software not included in the distribution. For example, the administrator might install a newer version of a software application than that supplied with a distribution, or an alternative to that chosen by the distribution (for example, KDE Plasma Workspaces rather than GNOME, or the reverse, for the user interface layer). If the additional software is distributed in source-only form, this approach requires local compilation. However, if additional software is locally added, the state of the local system may fall out of synchronization with the state of the package manager's database. If so, the local administrator will be required to take additional measures to help ensure that the entire system is kept up-to-date. The package manager may no longer be able to do so automatically.

Most distributions install packages, including the kernel and other core operating system components, in a predetermined configuration. Few now require or even permit configuration adjustments during the initial installation. This feature makes installation less daunting, particularly for new users, but is not always acceptable.
For specific requirements, much software must be carefully configured to be useful, to work correctly with other software, or to be secure, and local administrators often need to spend time reviewing and reconfiguring assorted software.

Some distributions go to considerable lengths to specifically adjust and customize most or all of the software included in the distribution. Not all do so, however. Some distributions provide configuration tools to assist in this process.

By replacing everything provided in a distribution, an administrator may reach a "distribution-less" state: everything is retrieved, compiled, configured, and installed locally. Such a system also can be built from the foundation, avoiding a distribution altogether. You need a way to generate the first binary files until the system is self-hosting. You can do this through compilation on another system capable of building binary files for the intended target (possibly through cross-compilation).

Figure 4 provides an overview of Linux package management.

**Figure 4.** A Linux Distribution Is Usually Built Around a Package Management System, Which Includes the Linux Kernel, Free and Open-Source Software, and Sometimes Some Proprietary Software

Linux Distribution Package Use Cases

A wide variety of types of Linux distribution packages are available, including:

- Commercial or noncommercial packages
- Packages designed for enterprise users, power users, or home users
- Packages supported on multiple types of hardware, or platform-specific packages that may even be certified by the platform vendor
- Packages designed for servers, desktops, or embedded devices
- General-purpose packages or packages with highly specialized specific machine functions (for example, firewalls, network routers, and computer clusters)
● Packages for specific user groups: for example, packages with language internationalization and localization, or packages for music production or scientific computing

● Packages built primarily for security, usability, portability, or comprehensiveness

The diversity of Linux distributions is the result of the technical, organizational, and philosophical variation among vendors and users. The permissive licensing of free software means that any user with sufficient knowledge and interest can customize an existing distribution or design one to suit the user’s specific needs.

VMware vSphere 6.0

VMware provides virtualization software. VMware’s enterprise software hypervisors for servers—VMware vSphere ESX, vSphere ESXi, and vSphere—are bare-metal hypervisors that run directly on server hardware without requiring an additional underlying operating system. VMware vCenter Server for vSphere provides central management and complete control and visibility into clusters, hosts, virtual machines, storage, networking, and other critical elements of your virtual infrastructure.

VMware vSphere 6.0 introduces many enhancements to vSphere Hypervisor, VMware virtual machines, vCenter Server, virtual storage, and virtual networking, further extending the core capabilities of the vSphere platform.

VMware ESXi 6.0 Hypervisor

vSphere 6.0 introduces a number of new features in the hypervisor.

Scalability Improvements

ESXi 6.0 dramatically increases the scalability of the platform. With vSphere Hypervisor 6.0, clusters can scale to as many as 64 hosts, up from 32 in previous releases. With 64 hosts in a cluster, vSphere 6.0 can support 8000 virtual machines in a single cluster. This capability enables greater consolidation ratios, more efficient use of VMware vSphere Distributed Resource Scheduler (DRS), and fewer clusters that must be separately managed. Each vSphere Hypervisor 6.0 instance can support up to 480 logical CPUs, 12 terabytes (TB) of RAM, and 1024 virtual machines. By using the newest hardware advances, ESXi 6.0 enables the virtualization of applications that previously had been thought to be nonvirtualizable.

Security Enhancements

ESXi 6.0 offers these security enhancements:

● **Account management**: ESXi 6.0 enables management of local accounts on the ESXi server using new ESXi CLI commands. The capability to add, list, remove, and modify accounts across all hosts in a cluster can be centrally managed using a vCenter Server system. Previously, the account and permission management functions for ESXi hosts were available only for direct host connections. The setup, removal, and listing of local permissions on ESXi servers can also be centrally managed.

● **Account lockout**: ESXi Host Advanced System Settings have two new options for the management of failed local account login attempts and account lockout duration. These parameters affect Secure Shell (SSH) and vSphere Web Services connections, but not ESXi direct console user interface (DCUI) or console shell access.

● **Password complexity rules**: In previous versions of ESXi, password complexity changes had to be made by manually editing the /etc/pam.d/passwd file on each ESXi host. In vSphere 6.0, an entry in Host Advanced System Settings enables changes to be centrally managed for all hosts in a cluster.
● **Improved auditability of ESXi administrator actions:** Prior to vSphere 6.0, actions at the vCenter Server level by a named user appeared in ESXi logs with the vpxuser username: for example, [user=vpxuser]. In vSphere 6.0, all actions at the vCenter Server level for an ESXi server appear in the ESXi logs with the vCenter Server username: for example, [user=vpxuser: DOMAIN\User]. This approach provides a better audit trail for actions run on a vCenter Server instance that conducted corresponding tasks on the ESXi hosts.

● **Flexible lockdown modes:** Prior to vSphere 6.0, only one lockdown mode was available. Feedback from customers indicated that this lockdown mode was inflexible in some use cases. With vSphere 6.0, two lockdown modes are available:
  ◦ In normal lockdown mode, DCUI access is not stopped, and users on the DCUI access list can access the DCUI.
  ◦ In strict lockdown mode, the DCUI is stopped.

● **Exception users:** vSphere 6.0 offers a new function called exception users. Exception users are local accounts or Microsoft Active Directory accounts with permissions defined locally on the host to which these users have host access. These exception users are not recommended for general user accounts, but they are recommended for use by third-party applications—for service accounts, for example—that need host access when either normal or strict lockdown mode is enabled. Permissions on these accounts should be set to the bare minimum required for the application to perform its task and with an account that needs only read-only permissions on the ESXi host.

● **Smart card authentication to DCUI:** This function is for U.S. federal customers only. It enables DCUI login access using a Common Access Card (CAC) and Personal Identity Verification (PIV). The ESXi host must be part of an Active Directory domain.

VMware Horizon 6 with View
Horizon Version 6.2 includes the following new features and enhancements:

● **Windows 10**
  ◦ Windows 10 is supported as a desktop guest operating system
  ◦ Horizon Client runs on Windows 10
  ◦ Smart card is supported on Windows 10.
  ◦ The View User Profile Migration tool migrates Windows 7, 8/8.1, Server 2008 R2, or Server 2012 R2 user profiles to Windows 10 user profiles.

● **RDS Desktops and Hosted Apps**
  ◦ View Composer. View Composer and linked clones provide automated and efficient management of RDS server farms.
  ◦ Graphics Support. Existing 3D vDGA and GRID vGPU graphics solutions on VDI desktops have been extended to RDS hosts, enabling graphics-intensive applications to run on RDS desktops and Hosted Apps.
  ◦ Enhanced Load Balancing. A new capability provides load balancing of server farm applications based on memory and CPU resources.

● **One-Way AD Trusts**

● **One-way AD trust domains are now supported.** This feature enables environments with limited trust relationships between domains without requiring View Connection Server to be in an external domain.
- **Cloud Pod Architecture (CPA) Enhancements**
  - Hosted App Support. Support for application remoting allows applications to be launched using global entitlements across a pod federation.
  - HTML Access (Blast) Support. Users can use HTML Access to connect to remote desktops and applications in a Cloud Pod Architecture deployment.

- **Virtual SAN 6.1**
  - All-Flash support (Requires vSphere 6.0)
  - Stretched cluster support (Requires vSphere 6.0 U1)

- **Access Point Integration**
  - Access Point is a hardened Linux-based virtual appliance that protects virtual desktop and application resources to allow secure remote access from the Internet. Access Point provides a new authenticating DMZ gateway to View Connection Server. Smart card support on Access Point is available as a Tech Preview. Security server will continue to be available as an alternative configuration. For more information, see Deploying and Configuring Access Point.

- **FIPS**
  - Install-time FIPS mode allows customers with high security requirements to deploy Horizon 6.

- **Graphics Enhancements**
  - AMD vDGA enables vDGA pass-through graphics for AMD graphics hardware.
  - 4K resolution monitors (3840x2160) are supported.

- **View Administrator Enhancements**
  - View Administrator shows additional licensing information, including license key, named user and concurrent connection user count.
  - Pool creation is streamlined by letting View administrators clone existing pools.

- **Horizon 6 Interoperability with vSphere 6 Update 1**

- **Horizon 6 for Linux Desktop Enhancements**

  Several new features are supported on Horizon 6 for Linux desktops, including NVIDIA GRID vGPU, vSGA, RHEL 7.1 and Ubuntu 14.04 guest operating systems, and View Agent installation of JRE 8 with no user steps required.

  For more information, please visit [here](#).

**VMware Horizon with View Client**

With VMware Horizon with View Client for Linux, you can easily access your Linux virtual desktop from a Ubuntu, Red Hat, or any other Linux distribution supported by VMware with the best possible user experience on the LAN or WAN.

- **Support for Linux distribution**: The View Agent currently supports specific Ubuntu, Red Hat, SUSE, and Cento OS Linux distributions.

**Note:** The latest version of Horizon View Client 3.5 is just being released. Check for the supported Linux distributions.
- **Exceptional performance:** The adaptive capabilities of the PC over IP (PCoIP) display protocol are optimized to deliver the best user experience, even over low-bandwidth and high-latency connections. Your desktop is fast and responsive regardless of where you are.

- **Simple connectivity:** Horizon View Client for Linux is tightly integrated with Horizon View for simple setup and connectivity.

- **Security from any location:** Whether you are at your desk or away from the office, your data is delivered securely to you. Enhanced certificate checking is performed on the client. Horizon View Client for Linux also supports optional RADIUS and RSA SecurID authentication. (RADIUS support was added with VMware View 5.1 and View Client for Linux 1.5.)

Horizon Version 6.2 supports several new features for Linux desktops, including:

- NVidia GRID vGPU and vSGA deployment
- RHEL 7.1 and Ubuntu 14.04 guest OS
- View Agent installation of Java Runtime Environment (JRE) 8 (this Java version is required for scripts) with no user steps required

Table 1 lists the Linux operating systems that Horizon Client 3.4 supports.

**Table 1.** Horizon Client 3.4 Linux Operating System Support

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ubuntu</td>
<td>Version 12.04 and 14.04</td>
</tr>
<tr>
<td>Ubuntu x64</td>
<td>Version 12.04</td>
</tr>
<tr>
<td>Red Hat Enterprise Linux (RHEL)</td>
<td>Version 6.6</td>
</tr>
<tr>
<td>SUSE Linux Enterprise Desktop (SLED)</td>
<td>Version 11 SP3</td>
</tr>
<tr>
<td>CentOS</td>
<td>Version 6.6</td>
</tr>
</tbody>
</table>

VMware has recently released View Client 3.5. For more information about View Agent for Linux and its supported versions, see the following links:


Table 2 lists the Linux operating systems that Horizon Client 3.5 supports.

**Table 2.** Horizon Client 3.5 Linux Operating System Support

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ubuntu</td>
<td>Version 12.04 and 14.04</td>
</tr>
<tr>
<td>Ubuntu x64</td>
<td>Version 12.04</td>
</tr>
<tr>
<td>Red Hat Enterprise Linux (RHEL)</td>
<td>Version 6.6 and 6.7</td>
</tr>
<tr>
<td>SUSE Linux Enterprise Desktop (SLED)</td>
<td>Version 11 SP3</td>
</tr>
<tr>
<td>CentOS</td>
<td>Version 6.6</td>
</tr>
</tbody>
</table>
Deploying Red Hat Enterprise Linux 6.6 Virtual Desktops on VMware Horizon with View 6

Before you can deploy Linux virtual desktops in your environment, the following components must already be installed:

- Microsoft Active Directory
- User data VMware Virtual Machine File System (VMFS) data stores
- VMware vCenter
- VMware ESXi 6 hosts joined to the VMware vCenter
- VMware Horizon Connection Servers

This document does not detail the deployment of these prerequisite resources.

The RHEL 6 Installation Guide provides complete details about the Installation of RHEL 6 for various hardware-based architectures. RHEL 6 systems can be deployed as either physical or virtual machines.

Regardless of whether physical or virtual machines are used, a RHEL 6 installation involves the following general steps:

1. Install RHEL 6.
2. Perform the first bootup (FirstBoot).
3. Apply updates.
4. Configure features.
Install RHEL 6 and Deploy the First Linux Virtual Machine

1. Create a virtual machine on the ESXi host (Figure 5).

**Figure 5.** Create a New Virtual Machine

![Image of Create New Virtual Machine window]

2. Provide a name for the virtual machine (Figure 6).

**Figure 6.** Provide a Name for the Virtual Machine

![Image of Name and Location window]
3. Select the guest operating system. For example, select Linux and choose the RHEL distribution version (Figure 7).

**Figure 7.** Select the Guest OS

![Select Guest OS](image)

**Note:** Allocate the required CPU and memory and choose the network to which the virtual machine will be connected along with all other required parameters.

4. Select the required number of monitors and video RAM for the virtual machine to be created (Figure 8).

**Note:** Consult the VMware documentation for the recommended number of monitors, vRAM guidelines, and resolution best practices.

**Figure 8.** Select the Number of Monitors and vRAM for the Virtual Machine

![Select Monitors and vRAM](image)
5. Attach the Linux ISO image (Figure 9).

**Note:** RHEL 6.6 64-bit is deployed in this study.

**Figure 9.** Attach the Linux Distribution ISO Image

![Image of Virtual Machine Properties]

6. Be sure that the ISO image is verified and installable (Figure 10).

**Figure 10.** The RHEL Distribution ISO Image Is Verified and Acceptable for Installation

![Image of ISO Verification]

**Note:** You will need select certain settings for your virtual machine to be installed: for example, the desktop mode, remote desktop clients, desktop deduplication setting, server tools, system administration tools, and troubleshooting options.
7. View all the additional virtual machine features and optional settings before you begin the installation (Figure 11).

**Figure 11.** Examples of Additional Options for the Virtual Machine

8. Click Next. The installer deploys the RHEL virtual machine and all the selected packages (Figure 12).

**Figure 12.** Linux ISO Image Installation Is in Progress

9. After the installation is complete, reboot the virtual machine (Figure 13).

**Note:** A system reboot is required.
Perform FirstBoot
After the operating system has been installed, the system reboots and enters what is referred to as FirstBoot. During FirstBoot, administrators are guided through the process of setting the date and time, configuring software updates, registering with Red Hat Network (RHN), creating initial user accounts, and setting options for kernel failure dumps (Kdumps). The system then reboots to activate the changes.

Apply Updates
After login has been completed under the newly created user account, you need to apply updates to the system to implement the latest versions of all software in the RHEL 6 system. You can perform updates to apply the most recent patches and security updates for RHEL 6 at any time by running Yum Update.

You need to update all the Linux virtual machines either by registering the virtual machines with the Linux distribution or by configuring a local repository.

The command for updating the installation package for the Linux distribution is shown here:

```
# yum update <chose the package being installed>
```

Configure Security-Enhanced Linux Security Parameters
By default, Security-Enhanced Linux (SELinux) is enabled during the RHEL 6 installation process. For maximum security, Red Hat recommends running RHEL 6 with SELinux enabled. Verify that SELinux is enabled and configured at system bootup.

1. Use the getenforce utility to verify that SELinux is enabled:
   
   ```
   # getenforce
   Enforcing
   ```

2. If getenforce returns Permissive, then set it to Enforcing and verify again:
Enforcing

3. Edit the file /etc/selinux/config and make SELinux persistent across reboots:

SELINUX=enforcing

The RHEL 6 Installation Guide provides complete instructions. Consult the guide for additional installation details.

Install and Configure Samba

Install the Samba packages:

```
# yum -y install samba samba-client samba-common samba-winbind \samba-winbind-clients
```

```
[root@LINUX-VDII ~]# service smb status
smbd (pid 2233) is running...
```

```
[root@LINUX-VDII ~]# service smb start
```

```
[root@LINUX-VDII ~]# service smb stop
```

```
[root@LINUX-VDII ~]# chkconfig smb on
[root@LINUX-VDII ~]# chkconfig --list smb
smb 0:off 1:off 2:on 3:off 4:off 5:off 6:off
[root@LINUX-VDII ~]#
```

Synchronize Time Services

Edit the file /etc/ntp.conf so that the RHEL 6 system time is synchronized from a known, reliable time service:

```
[root@LINUX-VDII ~]# vi /etc/ntp.conf
[root@LINUX-VDII ~]# service ntpd status
[root@LINUX-VDII ~]# service ntpd start
Starting ntpd:
```

```
[root@LINUX-VDII ~]# chkconfig ntpd on
[root@LINUX-VDII ~]# chkconfig --list ntpd
ntpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off
[root@LINUX-VDII ~]#]
```
**Configure DNS**

Edit the file `/etc/resolv.conf` so that the fully qualified domain name (FQDN) of the Domain Name Server (DNS) is specified:

```
[root@LINUX-VDI1 /]# vi /etc/resolv.conf
[root@LINUX-VDI1 /]# hostname
LINUX-VDI1.VDILAB.V.LOCAL
[root@LINUX-VDI1 /]# hostname -I
122.0.0.15
```

Similarly, set the hostname of the RHEL 6 system to the FQDN. Edit the file `/etc/sysconfig/network` and set the hostname to use the FQDN. Figure 14 shows the `sysconfig/network` file settings.

![ Sysconfig/network File Settings](image)

**Install and Configure the Kerberos Client**

As a best practice, install and configure the Kerberos client (`krb5-workstation`) so that Kerberos can properly authenticate with Active Directory on the Microsoft Windows Server. This step is optional but highly recommended because it is useful for troubleshooting Kerberos authentication issues.

1. Verify that the Kerberos client is installed. If it is not installed, install it using the following command:

   ```
   # yum -y install krb5-workstation
   ```

2. If Kerberos has not been previously configured, modify the Kerberos configuration file (`/etc/krb5.conf`) by adding entries for the new Kerberos and Active Directory realms. Note the differences in the Kerberos `[realms]` and Active Directory `[domain_realm]` realm entries.

3. Create a safety copy of the Kerberos configuration file:

   ```
   # cp -p /etc/krb5.conf{,.orig}
   ```

4. Edit the file `/etc/krb5.conf` with the domain controller name and domain name, as shown in Figure 15. The changes are underlined in color,

   **Note:** Save a copy of the krb5.conf file with different name in case the configuration settings do not work with the changed parameters.
Install the oddjob-mkhomedir Package

Install the oddjob-mkhomedir package to help ensure that user home directories are created with the proper SELinux file and directory contexts:

```
# yum install oddjobmkhomedir.x86_64
[root@LINUX-VDI1 /]# service oddjobd status
oddjobd (pid 2315) is running...

[root@LINUX-VDI1 /]# service oddjobd stop
Shutting down oddjobd:          [  OK  ]
[root@LINUX-VDI1 /]# service oddjobd start
Starting oddjobd:               [  OK  ]
[root@LINUX-VDI1 /]#]
[root@LINUX-VDI1 /]# chkconfig oddjobd --list
oddjobd 0:off 1:off 2:on 3:on 4:on 5:on 6:off
```
Installing VMware Tools

Next, install the VMware tools.

1. Download the VMware Tools TAR folder from the following VMware website:
   https://my.vmware.com/web/vmware/details?downloadGroup=ESXi600_OSS&productId=491 (Figure 16).

2. Follow the instructions for installing the VMware tools for an open-source distribution at the following website:

   **Note:** Download the VMware tools TAR package and copy it to a folder on the virtual machine desktop to extract the files.

   **Figure 16.** Download the VMware Tools in the Linux Distribution TAR File from the VMware Website to the Desktop

3. Extract the TAR file on the virtual machine (Figure 17).

   **Figure 17.** Command to Extract the VMware Tools TAR Files

   ```
   [root@LINUX-VDI tmp]# tar -xvzf VMwareTools-9.10.0-2476743.tar.gz
   ```

4. Install the VMware tools from the virtual machine command line (Figure 18). The default file location is recommended.

   **Figure 18.** Installing VMware Tools

   ```
   [root@linux-gpuvm vmware-tools-distrib]# ./vmware-install.pl
   Creating a new VMware Tools installer database using the tar4 format.
   Installing VMware Tools.
   In which directory do you want to install the binary files?
   [/usr/bin] 
   ```

5. Be sure to select [Yes,] the default to configure VMware tools after the installation is complete.
Figure 19. VMware Tools Installations Completed

The path "/usr/lib/vmware-tools" does not exist currently. This program is going to create it, including needed parent directories. Is this what you want? [yes]

In which directory do you want to install the documentation files? [/usr/share/doc/vmware-tools]

The path “/usr/share/doc/vmware-tools” does not exist currently. This program is going to create it, including needed parent directories. Is this what you want? [yes]

The installation of VMware Tools 9.10.0 build-2476743 for Linux completed successfully. You can decide to remove this software from your system at any time by invoking the following command: "/usr/bin/vmware-uninstall-tools.pl".

Before running VMware Tools for the first time, you need to configure it by invoking the following command: "/usr/bin/vmware-config-tools.pl". Do you want this program to invoke the command for you now? [yes]

Initializing...

Installing the VMware View Agent

Follow these steps to install the View Agent:

1. Download View Agent from VMware website:

2. Copy the View Agent TAR file folder and extract the installer files. Figure 20 shows the downloaded files in the View Agent folder.

Figure 20. Contents of the Downloaded View Agent Folder

```
[root@LINUX-VDI VMware-viewagent-linux-86.64-6.1.1-2772438]# ls
bin         ssl
CollectLoginUserInfo.sh user-login-info.desktop
EULA.txt     viewagent
install_viewagent.sh viewagent.conf
lightdm.conf.xf86rc viewagent.service
open_source_licenses.txt VMwareBlastServer
Product.txt   VMware-viewagent-standalone-8.1.1-2765955.tar
resources
```
3. Install the View Agent from the virtual machine command line (Figure 21).

Figure 21. Install View Agent from Command Line

4. Reboot the virtual machine to complete the View Agent installation process.

**Note:** See the following documents for more information about the installation and setup of a VMware Horizon Linux virtual desktop:

Authenticating Linux Virtual Machines with the Microsoft Windows Active Directory Domain

In many organizations, system administrators need to integrate Linux desktops into their existing Microsoft Windows Active Directory domain environments. On the surface, Linux and Windows integration appears simple. However, in fact, a huge number of components, configurations, and integration options are available.

Because this study focuses on RHEL distribution authentication with Windows Active Directory, this document focuses on that particular Linux distribution.

Microsoft Active Directory Deployment Prerequisites
Before you can integrate your RHEL 6 system into an Active Directory domain, you must complete the prerequisites listed here.

2. Configure Active Directory domain services.

RHEL 6 System Prerequisites
1. Deploy RHEL 6.
2. Configure SELinux security parameters.
3. Install and configure Samba (for recommended configurations 1 and 2 only).
4. Synchronize time services.
5. Configure DNS.
6. Install and configure the Kerberos client.

Do not proceed with the integration tasks until all of these components are fully configured and deployed. For more information, see the following resources:


Integrating the Linux Virtual Machine with Microsoft Active Directory
This section presents the tasks needed to integrate RHEL 6 with Active Directory for each of the four recommended configurations.

Integration Tasks
The integration tasks are as follows:

1. Configure authentication.
2. Verify and test Active Directory.
3. Modify the Samba configuration.

Using the system-config-authentication Tool
The system-config-authentication tool simplifies the configuration of the Samba, Kerberos, security, and authentication files for Active Directory integration.
1. Invoke the system-config-authentication tool as follows:
   ```
   # system-config-authentication
   ```

2. Set the Winbind domain, security model (ads), Winbind ADS realm domain name, and Windbind domain controller FQDN. Choose /bin/bash for the template shell (Figure 22).

3. On the Advanced Options tab, select “Enable fingerprint reader support” if required. Also select “Create home directories on the first login.” Every user login will have its own home directory (Figure 23).

   **Note:** The System Security Services Daemon (SSSD) manages user home directories upon login. Make sure the SSSD service is running on the virtual machine.

   **Figure 22.** Identity and Authentication Settings
4. Return to the Identity & Authentication tab, select Join Domain, and provide the Active Directory credentials. An alert indicates that you need to save the configuration changes to disk before continuing (Figure 24).

Figure 23. Advanced Options

![Advanced Options](image)

Figure 24. Join the Domain and Provide Active Directory Credentials

```
[root@LINUX-VDI1 ~]# net join -S AD-DC1 -U administrator
Enter administrator's password:
Using short domain name -- vdilab-v
Joined 'LINUX-VDI1' to dns domain 'vdilab-v.local'
```
Figure 25 shows that the Linux virtual machine joined the domain with the FQDN.

**Figure 25.** The Linux Virtual Machine Joined the Domain with the FQDN

<table>
<thead>
<tr>
<th>General</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Guest OS:</td>
<td>Red Hat Enterprise Linux 6 (64-Bit)</td>
</tr>
<tr>
<td>VM Name:</td>
<td>LinuxVirtualMachine1</td>
</tr>
<tr>
<td>CPU:</td>
<td>4 CPU</td>
</tr>
<tr>
<td>Memory:</td>
<td>6144 MB</td>
</tr>
<tr>
<td>Memory Overhead:</td>
<td>0</td>
</tr>
<tr>
<td>VMware Tools:</td>
<td>Running (Current)</td>
</tr>
<tr>
<td>IP Addresses:</td>
<td>122.0.0.15</td>
</tr>
<tr>
<td>DNS Name:</td>
<td>LINUX-VDII-VMLAP-K-LOCAL</td>
</tr>
<tr>
<td>EVCS Mode:</td>
<td>N/A</td>
</tr>
<tr>
<td>State:</td>
<td>Powered On</td>
</tr>
<tr>
<td>Host:</td>
<td>10.29.164.192</td>
</tr>
<tr>
<td>Active Tools:</td>
<td>View all</td>
</tr>
</tbody>
</table>

**Note:** You must install the Java version required as specified in the View Agent script that will be running to install the agent.

(Refer the latest Horizon View 6.2 release, which does not require any user interaction for JRE installations running certain scripts.)

For this study, Java development tool kit (DTK) version jdk-7u79 is required (Figure 26). You can download the version and install it on your system.

**Figure 26.** Viewing the Java Version Information

```
[root@LINUX-VDII desktop]# rpm -Uvh jdk-7u79-linux-x64.rpm
Preparing...  ################################################################### [100%]
1:jdk  ################################################################### [100%]
Unpacking JAR files...  rt.jar...

[root@LINUX-VDII /]# java -version
java version "1.7.0_79"
Java(TM) SE Runtime Environment (build 1.7.0_79-b15)
Java HotSpot(TM) 64-Bit Server VM (build 24.79-b02, mixed mode)
```
Configuring VMware View Administrator Configuration to Register the Linux Virtual Machine

You need to configure certain settings on the View Administrator console before you can deploy Linux desktop pools.

1. Choose View Admin > Administrators > Add Permission > Agent Registration Administrators to configure the View administrator settings (Figures 27 and 28). The Linux virtual machine will be automatically discovered on the View Administrator console after the View Agent is installed.

Figure 27. Choose Agent Registration Administrators

2. Verify that the Linux virtual machine is available. On the View Administrator console, choose View Configuration > Registered Virtual Machines and click the Others tab (Figure 29).

Figure 28. View Agent Registration Administrators Is Configured
Creating a VMware View Administrator Desktop Pool

Now you are ready to create a desktop pool.

1. Create a manual desktop pool to create a Linux virtual machine pool (Figure 30).

Figure 30. View Administrator Desktop Pool Selection
2. Configure either a dedicated or floating user-assignment pool based on your requirements (Figure 31).

**Figure 31.** User-Assignment Selection

3. Select “Other sources” as the machine source (Figure 32).

**Note:** The "vCenter virtual machines" selection for Machine Source may not work for a Linux virtual machine pool.

**Figure 32.** Select “Other sources” as Machine Sources for Linux Virtual Machines

4. Provide a name for the desktop pool: for example, Linux-VDI or Linux Pool (Figure 33).

**Figure 33.** Provide a Name for the Virtual Machine Pool
5. Make any necessary changes to the recommended and required settings (Figure 34).

**Figure 34. Desktop Pool Settings**

![Desktop Pool Settings](image)

**Note:** Do not make any changes to the display protocol settings because the user does not have any control over the protocol and the settings don’t have any effect on the Linux virtual machines. Leave the default protocol settings.

6. Add the Linux virtual machine that was created and authenticated (Figure 35).

**Figure 35. Select the Linux Virtual Machine Discovered in the Desktop Pool**

![Select Linux Virtual Machine](image)
7. Click Finish to create the desktop pool (Figure 36).

**Figure 36.** Linux Desktop Pool Ready to Be Created

When the Linux desktop pool creation finishes, you will see the result in the View Administrator console (Figure 37).

**Figure 37.** Linux Desktop Pool
Launching a Linux Virtual Machine from the VMware View Client

Launch the Linux virtual machine from the View Client.

Provide the user credentials for the user created and authenticated for the Linux virtual machine to be accessed. Figures 38 through 41 show the process.

**Figure 38.** Log in to the Virtual Desktop Through the View Client

![Login to the Virtual Desktop Through the View Client](image)

**Figure 39.** Select the Linux Desktop Pool

![Select the Linux Desktop Pool](image)

**Note:** The Linux user must log in to the Linux guest OS after connecting to the remote desktop. SSO login does not work for Linux desktops. Do not give a Linux user access to virtual machines with Administrator privileges because the user can then issue a shutdown command to shut down the virtual machine and the vCenter Server administrator will need to power on the virtual machine again.
**Figure 40.** Log in to the Virtual Machine with the User Account Credentials

![Login to Virtual Machine](image)

**Figure 41.** User Is Logged in to the Linux Desktop

![User Logged in to Linux Desktop](image)
Installing Apache OpenOffice on the Linux Virtual Machine

Download the installation instructions for Open Office from the following web location:

Download the RPM or DEB Package to the Installation Directory and Unpack
Use either the Red Hat Package Manager (RPM) package or the Debian (DEB) package.

1. Review the system requirements.
2. Download your preferred Linux version of Apache OpenOffice and save the folder on the virtual machine (Figure 42).

Figure 42. Download the Apache OpenOffice Software

3. Unpack the downloaded folder to prepare for installation. The following command should work: `tar -xvfz "linux package name".tar.gz`, where "linux package name" is the beginning part of the archive you just downloaded. This will create an installation directory. The name of the installation directory will likely be the language abbreviation for the install set: for example, en-US (Figure 43).
4. Open the RPMS folder to see the TAR files extracted (Figure 44).

**Figure 44. All the OpenOffice Software Installation Files Are Saved in the RPMS Folder**

```bash
[LinuxAdmin@Linux-VDI1 Desktop]$ cd en-US/
[LinuxAdmin@Linux-VDI1 en-US]$ ls -al
total 20
-dwrxr-x-x 5 LinuxAdmin LinuxAdmin 4096 Aug 18 2014 ...
dwrxr-xr-x 3 LinuxAdmin LinuxAdmin 4096 Aug 26 14:36 ...
dwrxr-xr-x 2 LinuxAdmin LinuxAdmin 4096 Aug 18 2014 licenses
dwrxr-xr-x 2 LinuxAdmin LinuxAdmin 4096 Aug 18 2014 removes
dwrxr-xr-x 3 LinuxAdmin LinuxAdmin 4096 Aug 18 2014 Rpms
```
5. Add the installation directory as a local repository for your GUI package manager if possible. This setting will enable you to perform the installation from a GUI instead of from the command line (Figure 45).

Figure 45. OpenOffice Software Files in the RPMS Folder Ready for Installation

Historically, OpenOffice has provided desktop integration packages for Mandriva, Red Hat, SUSE, and now freedesktop. Many new desktop window managers like KDE4, Gnome3, and Unity now adhere to the freedesktop.org standard for icon placement and MIME-type definitions. You may want to install the freedesktop integration package if you experience problems with your desktop integration.

Install the RPM Package

Follow these steps to install the RPM package for OpenOffice:

1. Enter a `su` command to switch to root, if necessary, and navigate to the OpenOffice installation directory. You will likely need to be root to run the `rpm` command to install the software.

2. Enter `cd` to change to the RPMS subdirectory of the installation directory. You should see a number of RPMS subdirectories here and one subdirectory called desktop-integration.

3. Install this new version by typing `rpm -Uvh *rpm`. By default, this command will install OpenOffice in your `/opt` directory (Figure 46). Alternatively, you can use a GUI package installer, reference the installation directory, and install all RPMS subdirectories at the top level. This approach may also help you find any dependency problems if they exist.

Figure 46. Installing the RPM Package
4. Install the desktop integration features for your setup. Enter the `cd` command to change to the desktop-integration folder in the installation directory and, depending on your package manager and system, install the appropriate desktop interface using RPM.

5. Start Apache OpenOffice 4.1.1 to verify that it is working.

Installing Apache OpenOffice Desktop Mode with User-Based Menus

Now install the OpenOffice desktop.

1. Change to the desktop-integration folder (move back one folder in the OpenOffice software folder download for desktop mode) to install the desktop menu for the Apache software you installed (Figure 47).

   **Figure 47.** Installing Apache OpenOffice with Graphic Menu Mode

   ```
   [root@LINUX-VD11 en-us]# cd /RPMs/desktop-integration/
   [root@LINUX-VD11 desktop-integration]# ls -l
   total 16
   -rw-r--r-- 1 LINUXADMIN LINUXADMIN 684 Aug 18 2014 openoffice4.1.1-freedesktop-menus-4.1.1-9775.noarch.rpm
   -rw-r--r-- 1 LINUXADMIN LINUXADMIN 619 Aug 18 2014 openoffice4.1.1-mandriva-menus-4.1.1-9775.noarch.rpm
   -rw-r--r-- 1 LINUXADMIN LINUXADMIN 619 Aug 18 2014 openoffice4.1.1-redhat-menus-4.1.1-9775.noarch.rpm
   -rw-r--r-- 1 LINUXADMIN LINUXADMIN 619 Aug 18 2014 openoffice4.1.1-suse-menus-4.1.1-9775.noarch.rpm
   ```

2. Click to select the appropriate Linux distribution desktop file to install for desktop view of the OpenOffice software. For example, select the redhat-menus-4.1.1-9775.noarch file depending on the Linux distribution used (Figure 48).

   **Figure 48.** Selecting the Linux Distribution Desktop File

   ```
   [root@LINUX-VD11 desktop-integration]# rpm -i openoffice4.1.1-redhat-menus-4.1.1-9775.noarch.rpm
   gtk-update-icon-cache: Cache file created successfully.
   /usr/bin/gtk-update-icon-cache
   gtk-update-icon-cache: Cache file created successfully.
   [root@LINUX-VD11 desktop-integration]# 
   ```

3. Use the following command to check where OpenOffice is located on the virtual machine:

   ```
   [root@LINUX-VD11 /]# whereis soffice
   soffice: /usr/bin/soffice
   ```
4. From the Applications menu on the desktop, select Office to see the OpenOffice applications (Figure 49).

**Figure 49.** Locating an OpenOffice Application

5. Open any application and configure first-time user credentials by clicking Next (Figure 50).

**Figure 50.** Configuring User Credentials
6. When the user credential fields are complete, click Finish. The selected application (in Figure 51, OpenOffice Calc) will open.

**Figure 51.** Working with the OpenOffice Calc Application

You are now ready to use all the OpenOffice applications.

### Creating Multiple Virtual Machines Using Microsoft Windows PowerShell Script

Deploying Linux desktops involve several steps. If you plan to deploy more than a few desktops, you can automate some of the steps by using PowerCLI scripts.

For more information about the vSphere PowerCLI, see:

- [https://www.vmware.com/support/developer/PowerCLI](https://www.vmware.com/support/developer/PowerCLI).

### VMware Horizon 6 for Linux Desktops Limitations and Workarounds

You should be aware of several limitations and workarounds when deploying Horizon 6 for Linux desktops.

- If you disconnect from a Linux desktop before the guest operating system has completed the login process, View Agent for Linux waits at least five minutes before logging off, even if the desktop pool setting “Automatically logoff after disconnect” is set to Immediately or to a waiting time that is less than five minutes.
  **Workaround:** Disconnect from the Linux desktop after the login process is completed.

- If you resize a RHEL or CentOS desktop while it is in screensaver mode, the desktop displays a black screen, and no Unlock dialog is presented.
  **Workaround:** Press the ESC key.

- If a RHEL or CentOS desktop is in screensaver mode and you disable the setting to lock the screen with the screensaver, View SSO becomes disabled, and you cannot disconnect and logoff from Horizon Client or View Administrator. This problem occurs when you choose System > Preferences > Screensaver, select "Active screen saver when computer is idle," and deselect "Lock screen when screen saver is active."
**Workaround:** Disconnect from View Connection Server and then reconnect. Then manually lock and unlock the desktop.

- If you configure two monitors with different resolutions, and the resolution of the primary screen is lower than that of the secondary screen, you may not be able to move the mouse or drag application windows to certain areas of the screen.

  **Workaround:** Make sure that the primary monitor's resolution is at least as high as the secondary monitor's.

- In vSphere 6.0, you cannot configure four monitors at 2560 x 1600 resolution on RHEL 6.6 or CentOS 6.6 virtual machines.

  **Workaround:** Either use 2048 x 1536 resolution or deploy this configuration in vSphere 5.5.

- If you configure two or more monitors at 2560 x 1600 resolution on RHEL 6.6 virtual machines in a vDGA environment, desktop performance is poor. For example, application windows do not move smoothly. This issue occurs when RHEL Desktop Effects are enabled.

  **Workaround:** Disable Desktop Effects by choosing System > Preference > Desktop Effects and selecting Standard.

**Conclusion**

The combination of Cisco UCS Manager, Cisco UCS B200 M4 Blade Servers, and your supported Linux distribution running on VMware vSphere ESXi 6.0 and VMware Horizon 6 provides a secure and high-performance platform for virtualizing engineering applications, CAD/CAM, and graphics-intensive applications.

Virtual desktop administrators now have the capability to manage both Microsoft Windows–based and Linux-based virtual desktops from a single VMware View Administrator console.

By following the guidance in this document, joint Cisco and VMware customers and partners can be assured that they are ready to virtualize the growing list of suitable applications that are supported on Linux distributions.

**For More Information**

- **Cisco**

- **Linux Open Distribution**
  - [https://access.redhat.com/downloads](https://access.redhat.com/downloads)

- **VMware vSphere ESXi, vCenter Server 6 View Horizon 6**
- https://my.vmware.com/web/vmware/details?downloadGroup=ESXI600_OSS&productId=491
- https://pubs.vmware.com/horizon-view-60/index.jsp
- Microsoft
- Apache OpenOffice