Parallels Desktop for Mac Business Edition

IT Administrator's Guide

v16.5

Parallels International GmbH
Vordergasse 59
8200 Schaffhausen
Switzerland
Tel: + 41 52 672 20 30
www.parallels.com

© 2021 Parallels International GmbH. Parallels and the Parallels logo are trademarks or registered trademarks of Parallels International GmbH in Canada, the U.S., and/or elsewhere.

Apple, Safari, iPad, iPhone, Mac, macOS, iPadOS are trademarks of Apple Inc. Google and Google Chrome are trademarks of Google LLC.

All other company, product and service names, logos, brands and any registered or unregistered trademarks mentioned are used for identification purposes only and remain the exclusive property of their respective owners. Use of any brands, names, logos or any other information, imagery or materials pertaining to a third party does not imply endorsement. We disclaim any proprietary interest in such third-party information, imagery, materials, marks and names of others. For all notices and information about patents please visit https://www.parallels.com/about/legal/

Contents

Introduction 5

Parallels Desktop Business Edition Features Overview 5

Deploying Parallels Desktop for Mac Business Edition 8

Deploying via an invitation email 8

Mass deployment using Mac management tools 11

Prerequisites 11

Kernel extensions in macOS 12

Supported Mac management tools 17

Preparing the deployment package 17

Deploying Parallels Desktop and virtual machines to Macs 34

Parallels Desktop for Mac Business Edition Features 48

Using Configuration Profiles 48

Create a configuration profile 49

Applying a configuration profile to a license key 51

Provisioning a corporate VM image 52

Enabling major version upgrades 56

Restricting User Actions in Parallels Desktop with a Custom Password 58

Restricting a Virtual Machine Configuration with a Custom Password 59

Locking a Virtual Machine on Suspend 61

Setting an Asset Tag in the Virtual Machine BIOS 61

Using Custom Graphics and Links in the Control Center 62

Hiding Developer Tools in the Parallels Desktop GUI 63

Encrypting a Virtual Machine from the Command Line 64

Single Application Mode 65

Setting an Expiration Date for a Virtual Machine 66

Enforcing USB Device Policies 67

Resetting the Guest OS Password 68

Setting Up a Local Update Server 69

Installing a web server 69

Creating the Parallels update XML file 69

Configuring individual Macs 72
Configuring Parallels Desktop update branch 74
Customizing the Support Center option 75
Participating in the Customer Experience Program 76
Parallels Desktop Command Line Interface 77
Parallels Desktop Management 77
Display Parallels Desktop info 78
License Management 78
Parallels Desktop Preferences 80
User List and Virtual Machine Location 82
Virtual Network Info 82
Virtual Network Settings 83
USB Devices 84
Send Problem Report 85
Shut Down Parallels Desktop 86
Parallels Desktop Plugin Commands 86
Error Handling 86
Virtual Machine Management 86
General Virtual Machine Management 87
Virtual Machine Configuration Tasks 95
Snapshot Management 112
Miscellaneous Commands 113
Virtual machine disk optimization 115
Error Handling 115
Index 116

CHAPTER 1

Introduction

Welcome to Parallels® Desktop for Mac Business Edition. Built on the world's best-selling, top-rated, most-trusted solution for running Windows applications on the Mac, Parallels Desktop Business Edition adds the capabilities that help IT administrators and purchasing agents save time and money.

Note: This guide refers to Parallels Desktop v16.5. If you are using a newer version of Parallels Desktop (including updates), please download the latest guide from the Parallels website.

In This Chapter

Parallels Desktop Business Edition Features Overview 5

Parallels Desktop Business Edition Features Overview

With Parallels Desktop for Mac, your Mac users can seamlessly run both Windows and Mac applications side-by-side with speed, control and confidence. Business users can experience as much or as little Windows as they want. Multiple view modes and options make it possible for users to customize the level of integration between Mac and Windows without compromising performance. Best of all, IT can lock down, secure, and control the settings that matter most.

Parallels Desktop Business Edition feature highlights

- License management portal (Parallels My Account)
- Split license keys into sublicenses
- Maintain corporate compliance with restricted virtual machines
- Set an expiration date for a virtual machine
- Mass deployment of Parallels Desktop and virtual machines
- Provision a corporate VM image from a link specified in Parallels My Account
- Business-level support including 24/7 phone and email support options

Parallels Desktop Business Edition feature matrix

See how Parallels Desktop Business Edition compares to Parallels Desktop Standard and Pro editions.

Feature	Standard	Pro	Business
World's most powerful, best-performing solution for running Windows applications on a Mac	•	•	•
Merge the coolness of Mac with the utility of Windows. Seamlessly utilize Mac features inside Windows and leverage Windows' strengths on your Mac.	•	•	•
Advanced networking tools. Create virtual networks for complex network scenarios and testing, including simulating various network scenarios.		•	•
Mass deploy and mass manage Parallels Desktop and Windows VMs for your employees.			•
Use the invitation email feature to invite users to install Parallels Desktop on their Mac computers.			•
Provision a corporate VM image from a link specified in Parallels My Account.			•
Single Application Mode. A special Parallels Desktop deployment option allowing Mac users to run a Windows application without being aware of Parallels Desktop and Windows running in a virtual machine.			•
Licensing and Support			
Premium 24/7 phone and email support	First 30 days	•	•
Centralized license management via Licensing Portal			•
Simple deployment with unified volume license key			•
Extended grace period to accommodate a possible license renewal delay without disrupting the business			•
Customizable in-product Support Center option			•
Removed in-product notifications			•
Configurable software update policy and local update server options			•
Security			
Restrict end-users from changing virtual machine settings		•	•
Restrict end-users from creating new virtual machines		•	•
Allow to disallow major Parallels Desktop version upgrades			•
Create expiring virtual machines			•

Enforce USB device policies			•
Configurable policies via SCCM and Parallels Mac Management			•
Advanced			
Assign up to 32 vCPU and 128 GB vRAM per virtual machine and control resources on the fly		•	•
Command line interface to control Parallels Desktop and virtual machines. Command line interface allows you to work with Parallels plugins for popular development tools including Vagrant and Jenkins		•	•
Customize the Parallels Control Center to include custom graphics, text, and links.			•

CHAPTER 2

Deploying Parallels Desktop for Mac Business Edition

This chapter describes how to deploy Parallels Desktop Business Edition to Mac computers in your organization. The deployment can be performed using one of the following methods:

- **Using an invitation email**. This is a simple and straightforward method. You send an invitation email to your users from your Parallels business account and they perform the installation themselves. You have an option to provide them with a corporate virtual machine using configuration profiles (p. 48). It's a great choice if you don't need to do any special customizations.
- **Using Mac management tools**. This is a highly customizable deployment method that gives you access to many advanced options which you can configure according to your needs. You should use this option if you and want to customize Parallels Desktop settings or apply restrictions.

The subsequent sections describe each method in detail.

In This Chapter

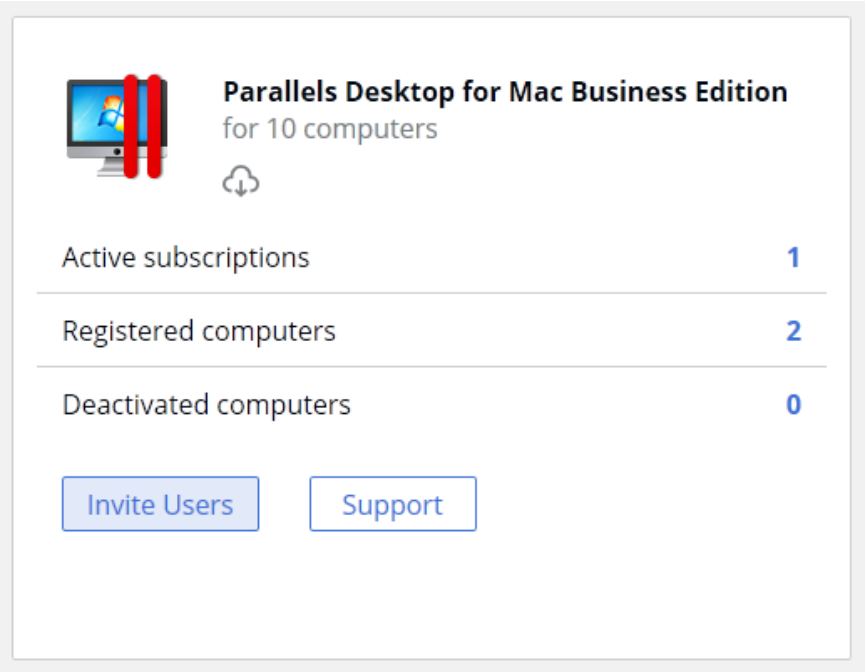
Deploying via an invitation email 8
Mass deployment using Mac management tools 11

Deploying via an invitation email

To invite users to install Parallels Desktop via email:

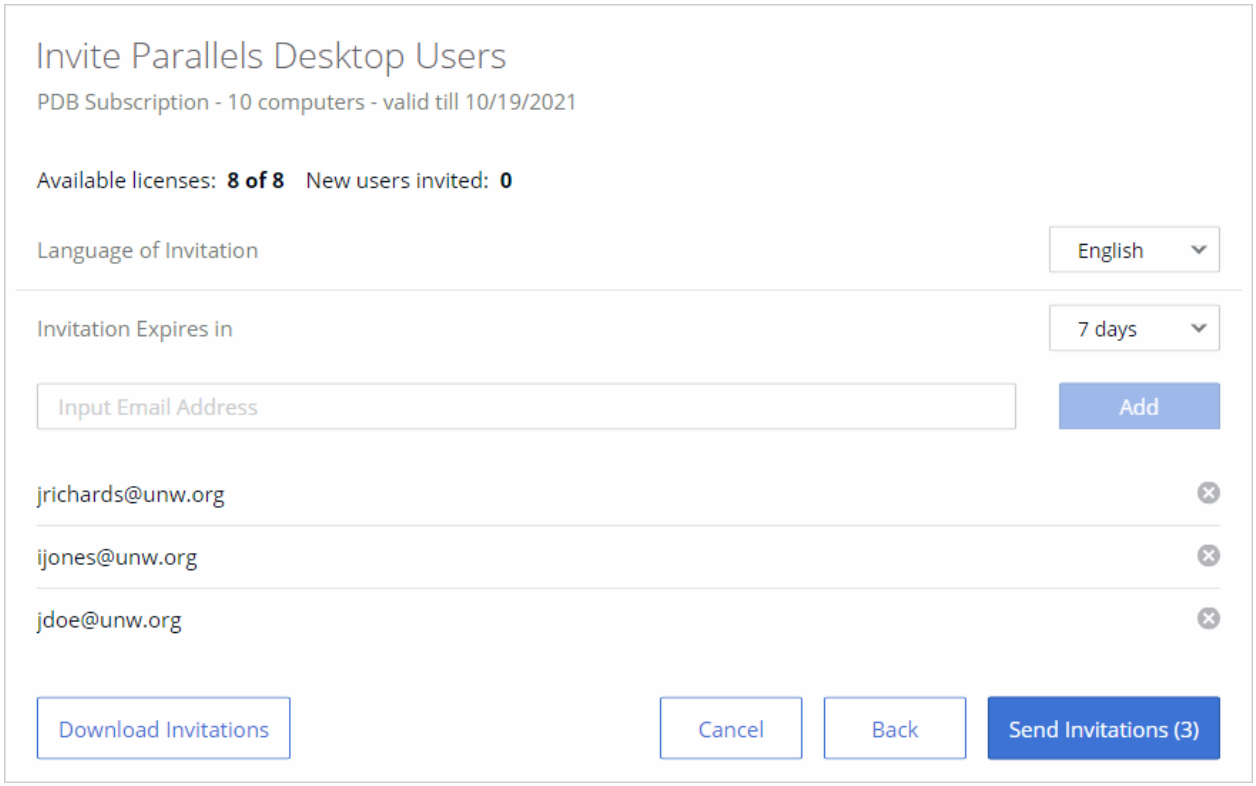
1 Log in to your Parallels account at https://my.parallels.com/.

2 On the **Dashboard** page, locate the **Parallels Desktop for Mac Business Edition** product card and click the **Invite Users** button.



3 In the dialog that opens, select a license key that you want to use to activate Parallels Desktop on users' computers and click **Next**.

4 In the **Invite Parallels Desktop Users** dialog, specify the following options:



- **Language of Invitation:** Select a language for the instructions in the invitation email.
- **Invitation Expires in:** Use the drop-down list to select when the invitation should expire. After it expires, the temporary activation code included in it will no longer work.
- **Email address:** Type a user's email address and click **Add**. Repeat for all intended users. You can also prepare a CSV file containing email addresses of your users and then drag and drop it here (or click **Select File** and brows for the file). The CSV file must contain a single column (a valid email address) with multiple rows (one email address on each row). Please note that if the number of users included in this list exceeds the number of available licenses for the specified key, the activation of Parallels Desktop will happen on a first-come, first-served basis.
- The **Download Invitations** button allows you to save the invitation email information to a CSV file. The information includes email addresses that you specified, a temporary activation code (generated individually for each user), and the Parallels Desktop download URL (also generated individually for each user). As an example, you can use the information in the downloaded file to create your own invitation email or to answer helpdesk questions, should any arise.

Hint: To see how the invitation email looks and to test it, you can send it to yourself first.

5 Click **Send Invitations** to send the email to users.

The invitation email that the users receive contains the following information:

- Installation instructions and a link from which a user can download the Parallels Desktop installation file.
- The temporary activation code. The code will be used automatically when a user installs Parallels Desktop on their computer. If for any reason automatic activation fails, the user can use the code included in the mail to manually activate Parallels Desktop. Please note that this is not the actual license key that you selected when you created the invitation email. This is only a temporary activation code with a limited scope and duration. The real license key is never shown to your Parallels Desktop users.

Once the users install and activate Parallels Desktop on their computers, you can see the list of active installations in your Parallels account.

Setting up a virtual machine

Once Parallels Desktop is installed on a Mac computer, a user needs to set up a virtual machine to run Windows on their Mac. This can be accomplished using one of the following methods:

- A user can create and configure a virtual machine and install Windows in it manually.
- An administrator can prepare a virtual machine and put it on a corporate network storage from where users can download it to their computers.
- An administrator can set up a Configuration Profile and provision a virtual machine to end users through it. For more information see **Using Configuration Profiles** (p. 48) and **Provisioning a corporate VM image** (p. 52).

Mass deployment using Mac management tools

This section describes how to mass deploy Parallels Desktop Business Edition using Mac management tools.

Prerequisites

Parallels Desktop installation image

To mass deploy Parallels Desktop Business Edition, you will need the Parallels Desktop for Mac installation image file and a Parallels Desktop for Mac Business Edition license key.

Important: A separate installation image is provided by Parallels for Apple M1 and Intel based Macs. If you are deploying Parallels Desktop on both types of Mac computers, you need to download an installation image for each type. You will then need to prepare a separate deployment package for each type of Macs and deploy each package separately.

You can download the installation images from https://www.parallels.com/products/business/download/.

Parallels Desktop deployment package

The Parallel Desktop deployment package is used to configure the deployment of Parallels Desktop. The package can be downloaded from https://www.parallels.com/products/business/download/.

Please note that if you already have a configured deployment package from an earlier version (or build) of Parallels Desktop, don't use it because it may not be compatible with your build of Parallels Desktop. Always download the latest version of a deployment package from the Parallels website using the link above.

Supported guest OS versions

If you are mass deploying one or more virtual machines together with Parallels Desktop, you should keep in mind the differences in supported guest operating systems on Mac computers with Apple M1 chip and Mac computers with Intel processor. For the latest information, see system requirements at https://www.parallels.com/requirements/.

Additional info

- If Mac computers in your organization run macOS High Sierra, macOS Mojave, or macOS Catalina, please read the **Kernel extensions in macOS** section (p. 12) that follows this one.
- For the complete list of hardware and operating system requirements for running Parallels Desktop, please refer to **Parallels Desktop User's Guide**.

Kernel extensions in macOS

If you deploy Parallels Desktop on macOS High Sierra, macOS Mojave, or macOS Catalina, Mac users will need to approve kernel extensions before they can launch Parallels Desktop.

Note: To avoid dealing with kernel extensions, Parallels recommends to update all of your Mac computers to macOS Big Sur where Parallels Desktop has the capability to run without using kernel extensions.

Kernel extensions can be approved manually on a Mac computer. See **Manually approving kernel extensions** (p. 14). As a system administrator, you can make the deployment more transparent for your Mac users by allowing Parallels Desktop kernel extensions to load before you deploy it on Mac computers. This can be done using one of the following options:

- Allowing kernel extensions to load via MDM configuration. See **Using MDM configuration** (p. 13).
- If your Mac computers are not enrolled in MDM, you can use the `spctl` command while booted to macOS Recovery. See **Using spctl command** (p. 14).

Please note that kernel extensions don't require user consent if:

- The extensions were on a Mac before macOS was updated to one of the versions listed above. This means that if Parallels Desktop was installed on a Mac before the update, you don't have to approve its kernel extensions.
- The extensions are replacing previously approved extensions.
- A Mac runs macOS Big Sur or later and the Apple hypervisor option is used in the virtual machine configuration.

Using MDM configuration

Starting with macOS 10.13.4, enrolling in MDM no longer disables User Approved Kernel Extension Loading, and extensions previously allowed to load for that reason now require approval. However, you can use MDM to specify kernel extensions that load without approval. This requires a Mac that is using macOS 10.13.2 or later and is either enrolled in MDM via DEP, or whose MDM enrollment is User Approved. For more information about User Approved Kernel Extension Loading and User Approved MDM enrollment, please see the following Apple Support article: https://support.apple.com/en-gb/HT208019.

To approve Parallels Desktop kernel extensions, you need to create a macOS configuration profile with the Kernel Extension Policy payload and then install it via MDM on Mac computers. The following table describes the payload keys and how to specify them to approve Parallels Desktop kernel extensions. Please note that this can also be done using Parallels Mac Management for Microsoft SCCM. For more info, please see https://kb.parallels.com/124937.

Key	Type	Value
`AllowUserOverrides`	Boolean	If set to true, users can approve additional kernel extensions not explicitly allowed by the configuration profile.
`AllowedTeamIdentifiers`	Array of Strings	Specifies team identifiers that define which validly signed kernel extensions will be allowed to load. Parallels team identifier is `4C6364ACXT`. When set, all possible Parallels kernel extensions will be authorized. Alternatively, you can specify kernel extensions individually (see below).
`AllowedKernelExtensions`	Dictionary	A set of kernel extensions that will be allowed to load on a Mac computer. The dictionary maps the team ID to an array of bundle IDs. The Parallels team ID is `4C6364ACXT`. The bundle IDs are as follows: • `com.parallels.kext.usbconnect` • `com.parallels.kext.vnic` • `com.parallels.kext.netbridge` • `com.parallels.kext.hypervisor` Note that the `AllowedTeamIdentifiers` key (described above) does the same thing, but approves all possible Parallels extensions, while here you can specify them individually. You can use either key depending on your requirements.

If your Mac computers are not enrolled in MDM, you can use the `spctl` command described in the section that follows this one.

Using spctl command

You can disable the user approval requirement for Parallels Desktop kernel extensions using the `spctl` command on a Mac. This can be done either via booting into macOS Recovery or while preparing NetBoot/NetInstall/NetRestore images. The command is as follows:

```
spctl kext-consent add 4C6364ACXT
```

The `4C6364ACXT` value in the example above is the Parallels Team ID. The command disables User Approved Kernel Extension Loading for Parallels Desktop, so user consent to load the extensions will not be required.

Please note that if you reset NVRAM after executing the `spctl` command, the Mac reverts to its default state with User Approved Kernel Extension Loading enabled. To prevent unauthorized changes to NVRAM, you can set a firmware password on the Mac.

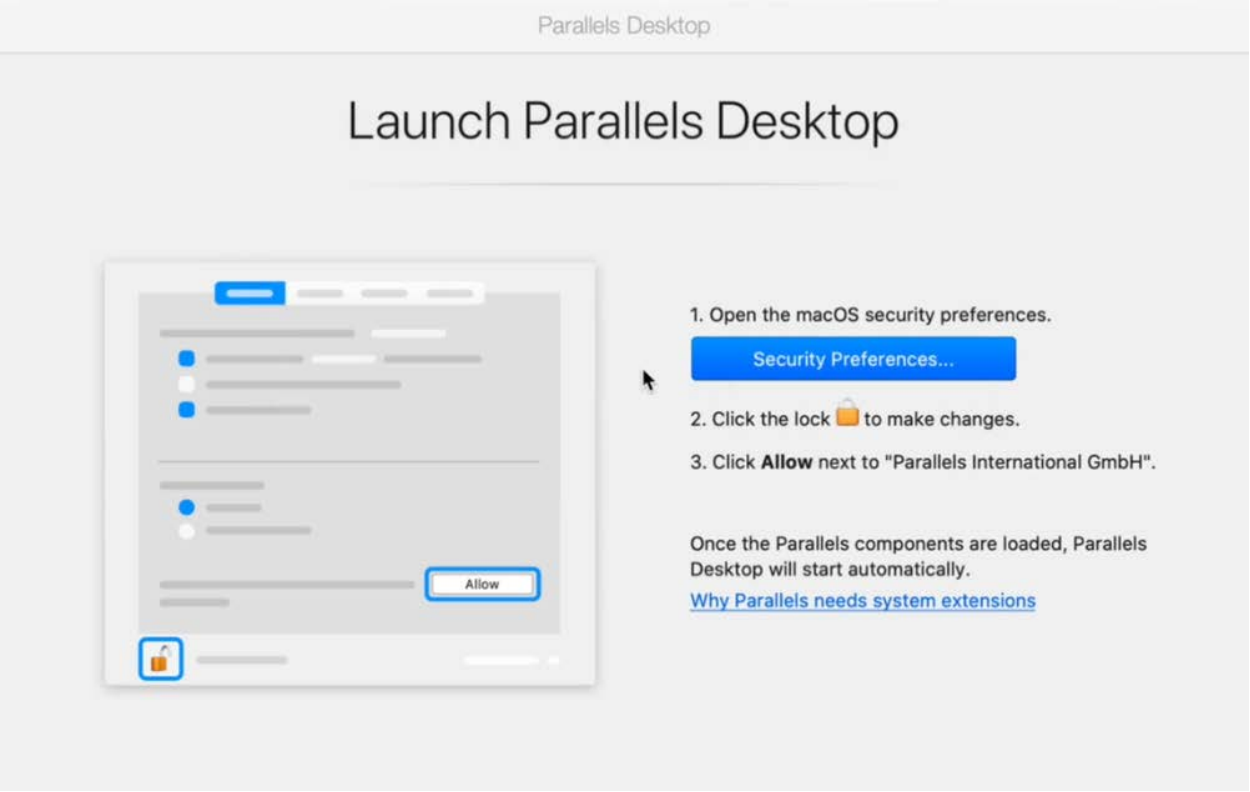
Manually approving kernel extensions

If you don't disable User Approved Kernel Extension Loading for Parallels Desktop in advance, Mac users will need to approve them manually.

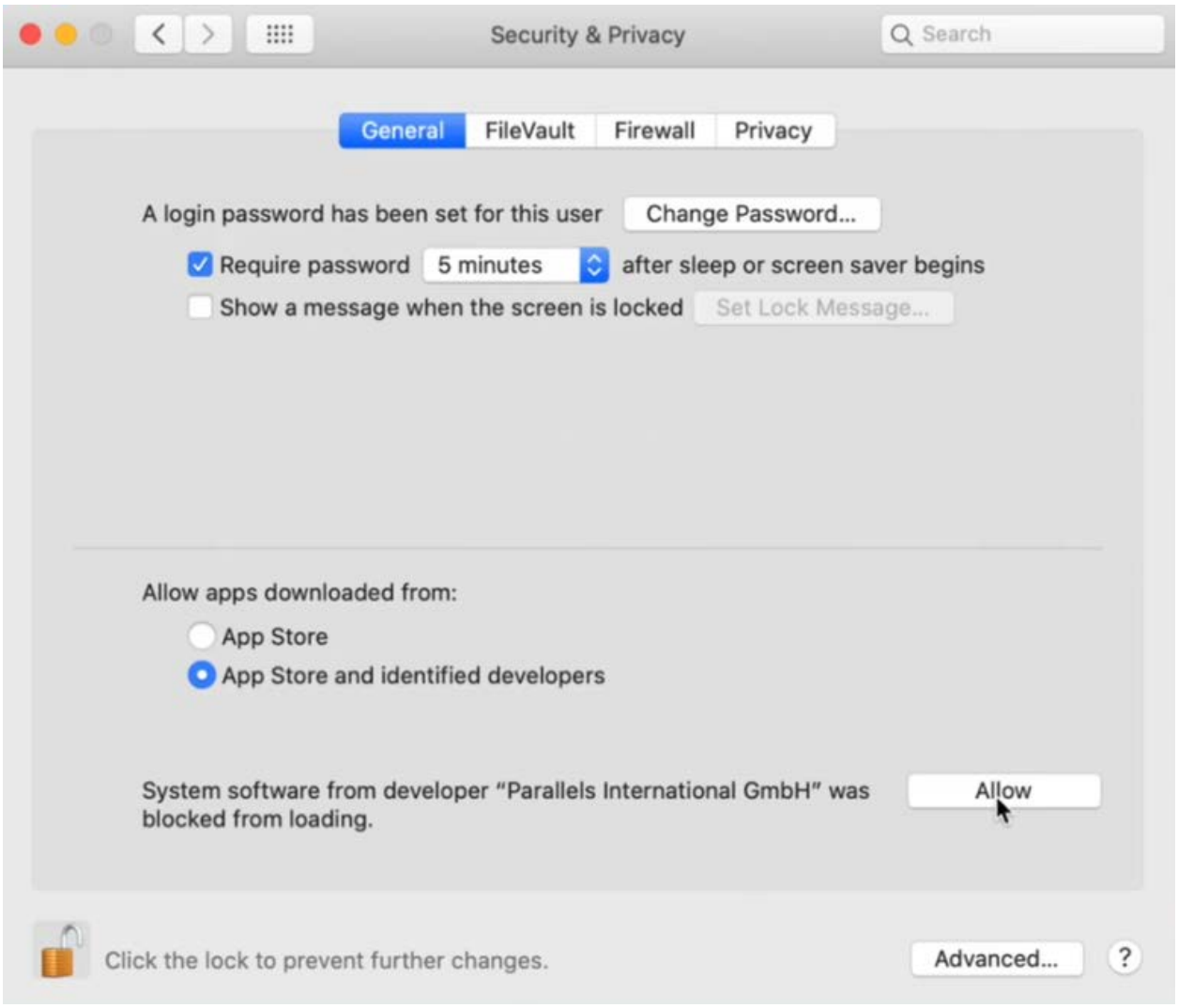
To manually authorize Parallels Desktop kernel extensions on a Mac:

1. If user consent is required to load the extensions, Parallels Desktop version 13.2.0 or later will guide the user by displaying the dialog shown below.

Note that earlier versions of Parallels Desktop will not display this dialog, so a user will need to open the **Security & Privacy** window (see the second screenshot below) directly.



2 The user clicks the **Open Security Preferences** button, which will open the **Security & Privacy** dialog (the user can also open the dialog by going to **System Preferences** > **Security & Privacy** > **General**):



3 The user then click the **Allow** button located next to the message about the software from "Parallels International GmbH". This will approve Parallels Desktop kernel extensions.

Parallels Desktop will now start normally. This has to be done only once when a user starts Parallels Desktop for the first time.

Supported Mac management tools

Parallels Desktop Business Edition can be deployed to Mac computers using one of the following Mac management tools:

- Jamf Pro
- Parallels Mac Management for Microsoft SCCM
- DeployStudio
- Apple Remote Desktop (ARD)
- Munki

This chapter includes detailed instructions on how to deploy Parallels Desktop using Jamf Pro, Parallels Mac Management for Microsoft SCCM, and Apple Remote Desktop. For instructions on how to use other tools, please see their respective documentation.

Preparing the deployment package

To prepare the deployment package, you need to add the following required and optional components to it:

- Parallels Desktop installation image (required).
- Parallels Desktop Business Edition license key (required).
- One or more virtual machines (optional).
- One or more Windows application stubs (optional). Stubs are special links to Windows applications installed in a virtual machine that can be added to the Dock in macOS during deployment.
- You can also configure deployment options according to your needs by modifying the configuration file included in the deployment package.

The subsequent sections describe how to add the necessary components and how to configure deployment package options.

Download the deployment package

If you haven't already, download the Parallels Desktop deployment package to a Mac computer from the following location:

https://www.parallels.com/products/business/download/

The deployment package archive contains a folder named "Parallels Desktop Business Edition mass deployment package vxxx", where "vxxx" is the deployment package version number.

The folder contains the following files:

- **Changelog.txt** — contains a record of changes that were made to the deployment package over time.
- **Parallels-Desktop-Business-Edition-Administrators-Guide.pdf** — the version of the guide that matches the deployment package version (otherwise, it's the same guide you are reading now).
- **Parallels Desktop Autodeploy.pkg** — this is the actual deployment package that you need to prepare for mass deployment of Parallels Desktop. To view the package contents, right-click it and choose **Show Package Contents**.

Read on to learn how to add the necessary components to the deployment package.

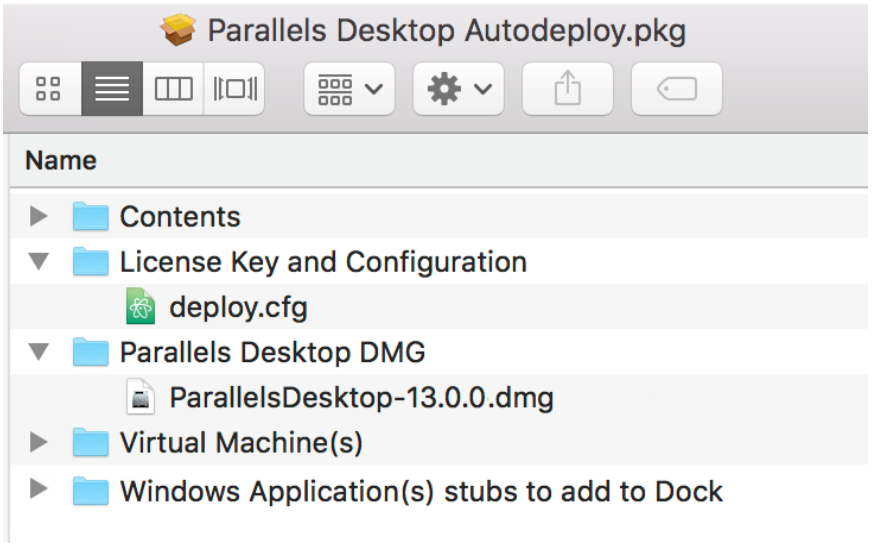
Add the Parallels Desktop installation image

Note: As was said earlier, you need to download a correct Parallels Desktop installation image on depending whether you are deploying Parallels Desktop on Apple M1 or Intel based Mac computers. If you are deploying Parallels Desktop on both types of Macs, you need to prepare a deployment package for each type, add a correct Parallels Desktop installer to it, and then deploy each package separately.

To add the Parallels Desktop installation image file to the deployment package:

1. Right-click the **Parallels Desktop Autodeploy.pkg** file and choose **Show Package Contents**.
2. Open the **Parallels Desktop DMG** folder and copy the Parallels Desktop installation image file to it (the .dmg file). If you don't have the file, you can download it from https://www.parallels.com/products/business/download/

The package should now look like the following:



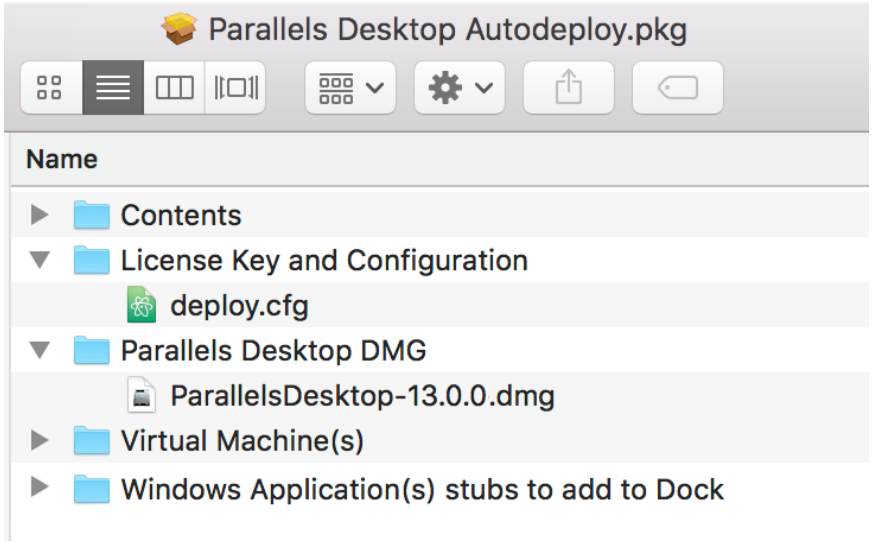
Please note that the Parallels Desktop installation image file name on the screenshot above is just an example. In your case, the file name will also include the current build number information.

Specify a license key

Parallels Desktop Business Edition license key is required to activate Parallels Desktop on target Macs. The key must be specified in the deployment package.

To specify the license key:

1 In the deployment package, expand the **License Key and Configuration** folder.



2 Open the `deploy.cfg` file in a text editor.

3 Find the `License` section (second from the top) and enter your Parallels Desktop Business Edition license key as a value of the `license_key` variable. The key must be supplied in the following format: "XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX" (including quotes and dashes).

4 Save the `deploy.cfg` file.

To learn about other configuration parameters in the `deploy.cfg` file, please see **Configure deployment options** (p. 23).

Note: Parallels Desktop activation requires Internet access. You need to make sure that **port 443** is opened on target Mac computers, so they can communicate with Parallels License Server. You can also verify that the Mac computers can reach the Parallels License Server at https://desktop.parallels.com

Add a virtual machine

Note: Before adding a virtual machine to the deployment package, please see **Prerequisites** (p. 11) about which versions of Windows are supported on Apple M1 and Intel based Mac computers.

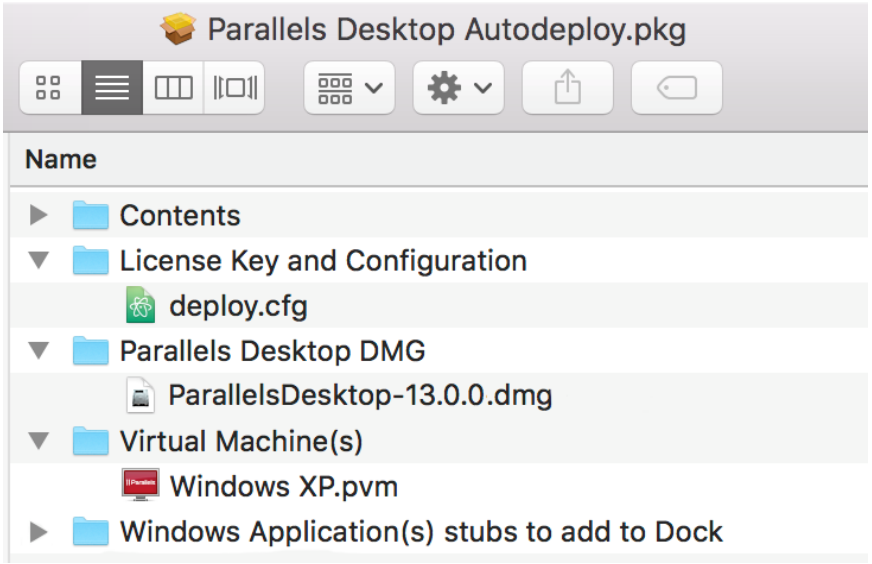
To add a virtual machine to the deployment package, simply copy the virtual machine file to the **Virtual Machine(s)** folder. You can add a regular (.pvm) or archived (.pvmz) virtual machine. See the **Note** below for details. More than one virtual machine can be added to the deployment package if needed.

Note: To decrease network load during mass deployment, it is recommended to archive a virtual machine before adding it to the deployment package. Archiving must be done in Parallels Desktop by right-clicking a virtual machine in Control Center and clicking **Archive**, which will compress the virtual machine file and give it the ".pvmz" extension. Please note that other archives (like .zip), are not supported and must not be used. When a package is deployed on a Mac computer, the archived virtual machine will be automatically unarchived to the standard ".pvm" format.

Before adding a virtual machine to the deployment package, you can configure it according to your needs. The subsections that follow this one describe the modifications that you can make to the virtual machine configuration.

Please note that the virtual machine must be fully stopped before adding it to the deployment package (it cannot be paused or suspended). This is necessary because deployment configuration options are applied to a virtual machine on a target Mac and this can only be done if the virtual machine is stopped.

After you add a virtual machine to the deployment package, the package folders should look like the following:



Read on to learn about modifications that you can make to the virtual machine configuration before adding it to the deployment package. If you are not planning on configuring a virtual machine at this time, you can skip to the **Adding Windows application stubs** section (p. 21).

Installing Parallels Tools

Parallels Tools is a collection of utilities and drivers that vastly improve the virtual machine performance and enable some features that are not available otherwise. Parallels Tools are included with every copy of Parallels Desktop and are highly recommended to be installed in every virtual machine right after an operating system is installed in it. Your source virtual machine should have Parallels Tools installed. For instructions on how to install Parallels Tools, please see https://kb.parallels.com/en/115835.

General configuration options

When preparing a source virtual machine for mass deployment, you may change any of its configuration settings according to your needs. The following list describes a few common options:

- **Shared Folders and Profiles**. Parallels Desktop offers great flexibility in bridging the capabilities of macOS and your guest operating system by configuring shared folders and profiles. Think over which files and folders you wish to share between the two operating systems and set up them in advance.
- **Enforce USB Device Policies**. Specify what types of USB devices can be connected to the virtual machine. See **Enforcing USB device policies** (p. 67) for complete details.
- **Installing Applications**. You can install all the necessary applications in the virtual machine before deploying it.
- **Parallels Desktop Business Edition Options**. Parallels Desktop Business Edition includes the following additional configuration options:
  - Set up a local update server and specify the Parallels Desktop automatic update options.
  - Customize the Support Center option.
  - Configure participation in the Customer Experience program.

  For the information on how to configure these options, see **Configure deployment configuration options** (p. 23) and **Parallels Desktop Business Edition Features** (p. 48).

For the complete information about Parallels virtual machine configuration, please refer to the **Parallels Desktop User's Guide**.

Adding Windows application stubs

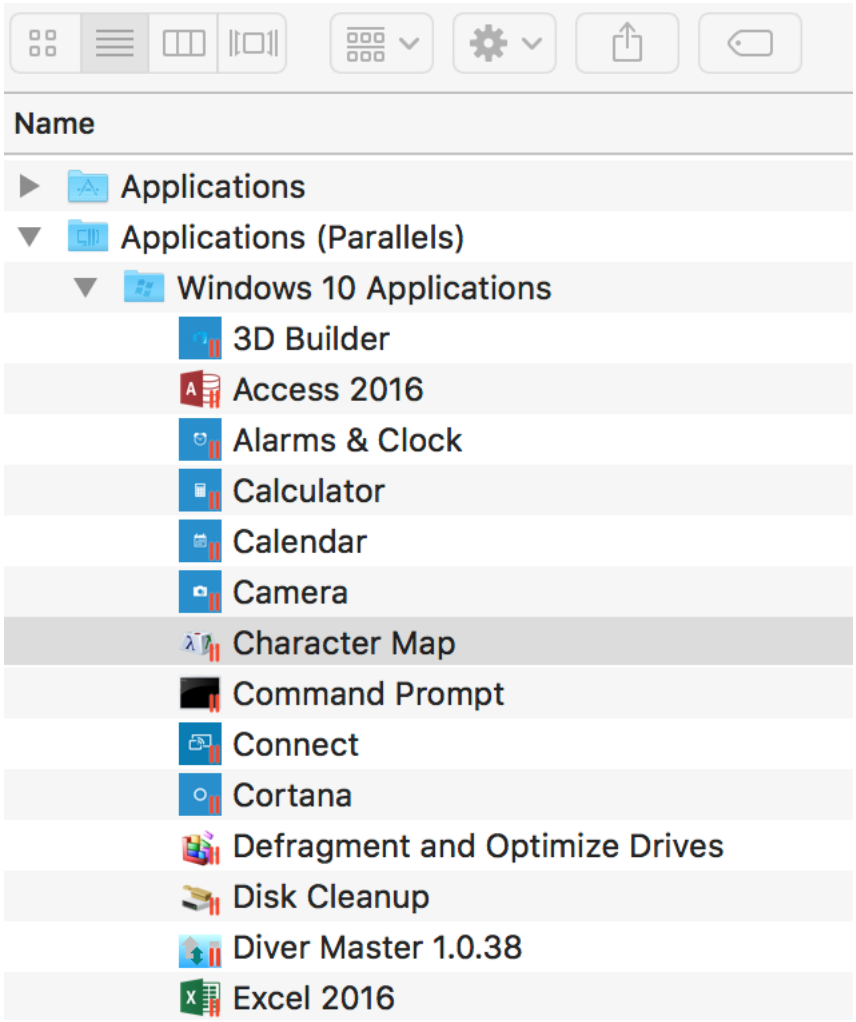
Windows application stubs are special links to Windows applications installed in a virtual machine that can be added to the Dock in macOS during deployment.

About application stubs

Application stubs are created in macOS when you create a virtual machine and install Parallels Tools in it. To see applications stubs for a virtual machine:

1 In macOS, navigate to `/Users/<user-name>/Applications (Parallels)`

2 Expand a desired virtual machine folder. For example, **Windows 10 Applications**, as shown in the screenshot below:



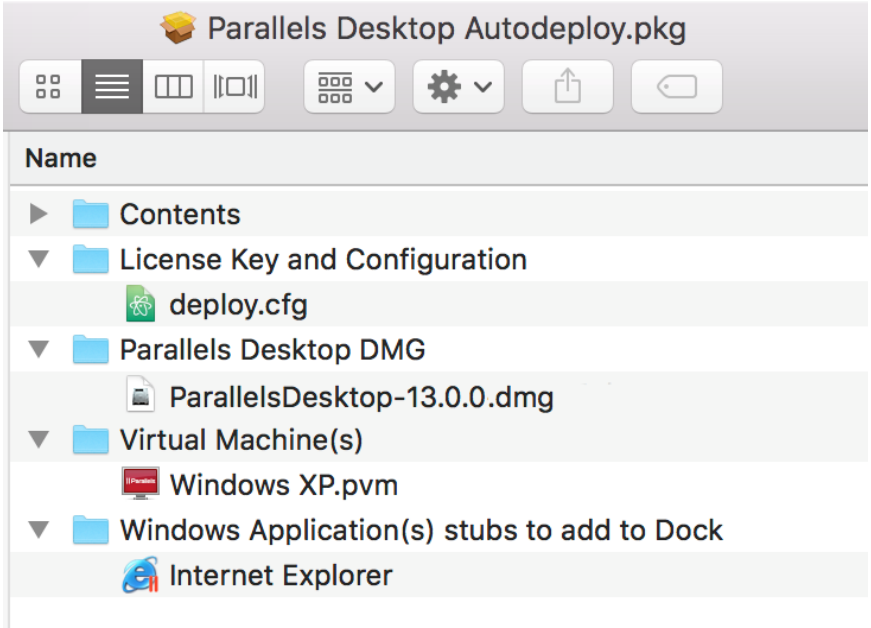
The icons that you see in the folder are Windows application stubs. If you double-click an icon, the corresponding Windows application will be started in the virtual machine.

You can add one or more application stubs to the deployment package to be added to the Dock on a target Mac computer. For example, if your Mac users will use a particular application most of the time, it would make sense to add it to the Dock, so they can quickly launch it without dealing with the Windows user interface.

Windows application stubs are specifically used when you deploy Parallels Desktop using Single Application Mode. For more information, please see the **Single application mode** section (p. 31).

Adding an application stub to the deployment package

To add one or more application stubs to the deployment package, simply copy it to the **Windows Application(s) stubs to add to Dock** folder, as shown on the screenshot below:



Configure deployment options

The deployment package contains a special script, which is automatically executed on a target Mac after the package is transferred to it. When executed, the script reads the configuration parameter values from the `deploy.cfg` file, which you can modify according to your needs.

To modify the parameters, expand the **License Key and Configuration** folder in the deployment package and open the **deploy.cfg** file in a text editor. The configuration parameters are organized in sections, which are described below.

License

The **License** section is used to specify the Parallels Desktop Business Edition license key.

Variable	Description
`license_key`	Specifies the Parallels Desktop license key. See **Specify a License Key (p. 19)** for more info.

Virtual Machines

The **Virtual Machines** section is used to specify virtual machine deployment options.

Variable	Description
`vm_register_mode`	Specifies the registration mode for the deployed virtual machines. Accepted values: `"Private"` — The virtual machines will be registered for the active user only. The virtual machines will be copied to the `/Users/<username>/Parallels` folder. `"Shared"` — The virtual machines will be registered for all users of a Mac. The virtual machines will be copied to the `/Users/Shared/Parallels` folder. The default destination folder for virtual machines can be modified using the `vm_destination_folder` variable (see below).
`vm_reset_hwid`	Specifies whether the virtual machine SMBOIS ID (hardware ID) will be regenerated. Each Parallels virtual machine is assigned a universally unique SMBIOS ID when it is created. When you deploy a virtual machine to many Macs, each resulting copy of the machine will have the same SMBIOS ID. This is the default behavior and should not be normally changed. If your enterprise management system relies on unique SMBOIS IDs, you can change the value of the `vm_reset_hwid` variable as follows: `"no"` — Keep the original SMBIOS ID. "yes" — Regenerate the ID.
`vm_deploy_mode`	Specifies whether the virtual machine(s) will be copied or moved from the deployment package to their destination folder on a Mac (see the explanation below). Accepted values: `"Copy"` — Copy the virtual machine(s). `"Move"` — Move the virtual machine(s). If your deployment package contains one or more virtual machines, they need to be copied or moved to their destination folder on a Mac during deployment (see `vm_register_mode` and `vm_destination_folder` variables). Moving a virtual machine file is almost instantaneous, while copying it will take a considerable time due to the large size of a typical virtual machine. The option you specify here depends on the following: • If the deployment tool that you are using copies the entire deployment package to a Mac computer before running it, you can use the fast "Move" option. Jamf Pro, Parallels Mac Management, and Apple Remote Desktop all copy packages to a Mac before running them, so you can use the "Move" option when using these tools. Note that the package and the destination folder must be located on the same mount point on a Mac for the "Move" operation to be fast; otherwise, it'll be essentially a copy-and-delete operation, hence slow. • If you are running the deployment package from a network share mounted on a Mac (e.g. manually), then you should use "Copy"

	because moving a virtual machine from a remote location will be as slow as copying it, plus the virtual machine will be removed from the package if you move it, so the package will become incomplete (you want it to stay intact if you want to install it on other Macs). Note that since macOS 10.13, the "Copy" mode supports the APFS feature "clone file". This means that if the target FS is APFS, the "clone" feature is used to reduce the time and disk space when deploying a VM. Please note that when testing the deployment package on a local Mac, you should be careful when using the "Move" option as it will remove the virtual machine(s) from the package. If that happens, you will need to add them again to the package before mass deploying it to Mac computers in your organization.
`vm_destination_folder`	Allows you to change the default destination folder for virtual machines. The default folder is determined by the value of the `vm_register_mode` variable (see above). The `vm_destination_folder` variable allows you to change the default folder while keeping the selected virtual machine registration mode.
`vm_password_to_edit`	**Note:** Beginning with Parallels Desktop 15, all restrictions are stored inside a VM bundle, therefore this configuration parameter is not considered. Specifies a password that will be required to modify the configuration of a virtual machine. For more information, please see **Restricting Virtual Machine configuration with a custom password** (p. 59).

`<VM_file_name>`	The parameter in this section allows you to change the default target location and/or name of a virtual machine when it is copied to a Mac computer.
	By default, virtual machines included in the deployment package are copied to the default location on a Mac, which is determined by the `vm_register_mode variable` (or the `vm_destination_folder` variable if it's present). If you have more than one virtual machine, you can specify a different destination location and/or name for each desired virtual machine. To do so, you must include a variable/value pair as shown in examples below.
	The variable name must be the same as the original virtual machine file name (including the ".pvm" or ".pvmz" extension). The value should contain a name and path where you want the virtual machine to be copied on a Mac.
	Please note the following:
	• The path must be relative if `vm_register_mode` is set to "Shared".
	• The target VM extension must be ".pvm", even if the original is ".pvmz" (because an archived virtual machine will be automatically unarchived on a Mac during deployment).
	Examples:
	`"Shared VM.pvm"="./Shared VM.pvm"`
	`"Private VM.pvm"="~/Parallels/Private VM.pvm"`
	`"Archived Shared VM.pvmz"="./Shared.pvm"`
	`"Archived Private VM.pvmz"="~/Parallels/Private.pvm"`

`control_center_banner_url`	This and two variables below are grouped together and allow you to customize Parallels Desktop Control Center by displaying a custom HTML banner at the top of its window. For additional information, see **Using custom graphics and links in the Control Center** (p. 62). The URL of a custom HTML page to be displayed as a banner in the Parallels Desktop Control Center window. To disable the banner, comment out the variable or specify an empty string as a value.
`control_center_banner_height`	The banner height, in pixels. The recommended value is 350. To use the current value (if you are updating Parallels Desktop on a Mac), comment out the variable.
`control_center_banner_min_width`	The banner minimum width, in pixels. When resizing the Control Center window, its minimum width will be limited accordingly. The recommended value is 350. To use the current value (if you are updating Parallels Desktop on a Mac), comment out the variable. To disable the minimum width limitation, comment out the variable or specify 0 (zero) as a value.
`vm_set_hv_mode_apple_forcibly_since_macos_11_0`	Specify whether the installer should override the hypervisor type of each deployed VM to 'Apple' on macOS 11.0 Big Sur and newer. All macOS versions since macOS Big Sur require a Mac reboot to run a VM with a hypervisor type other than 'Apple'. To forcibly change the hypervisor type to 'Apple', use the following setting: `vm_set_hv_mode_apple_forcibly_since_macos_11_0="yes"`

Launch options

The **Launch Options** section is used to specify whether Parallels Desktop should run as a service or an application.

Note: The `start_pd_as_service` option is no longer supported in Parallels Desktop 12 and later versions. Instead, Parallels Desktop makes a decision whether to start as a service based on how the virtual machines are configured. If virtual machines are configured to run in background, Parallels Desktop will start as a service. Versions of Parallels Desktop prior to 12 will continue to use the `start_pd_as_service` option.

Variable	Description
`start_pd_as_service`	Specifies whether Parallels Desktop should run as an application or as a service. The value of "yes" specifies that Parallels Desktop will run as a service. The service will start automatically on host startup and will stop on host shutdown. The service can be managed by **launchctl** (the standard macOS command line utility that allows you to manage daemons, applications, processes, etc.). The value of "no" specifies that Parallels Desktop will run as a macOS application. The user will have to start and exit Parallels Desktop manually. Depending on whether Parallels Desktop is already installed on a target Mac or not, the following will happen:

	- If Parallels Desktop is already installed on a Mac as a service and the value of "no" is specified, the new installation will set up Parallels Desktop to run as an application. - If this option is commented out and Parallels Desktop is already installed on a Mac, no changes will be made to the way Parallels Desktop runs. - If this option is commented out and Parallels Desktop is not installed on a Mac, it will run as an application.

Software updates

The **Software Updates** section is used to configure Parallels Desktop automatic updates.

Variable	Description
`updates_url`	Specifies the update URL. Possible values are: `"Parallels"` — Download updates from the Parallels website over the Internet. Updates are downloaded from a special server hosting Parallels Desktop Business Edition updates, which become available on the server at a slightly later date compared to Parallels Desktop Pro edition. This is the default and recommended option. For more info, please see **Configuring Parallels Desktop update options** (p. 74). `"None"` — Turn off automatic updates. `<URL>` — Specifies a custom update URL. You can use this option in one of the following two cases: 1) If you have a local update server, specify a complete URL (in quotes) of the `parallels_updates.xml` file on your local Web server. For more information, please see **Setting up a local update server** (p. 69). 2) If you don't want to wait until Parallels Desktop Business Edition updates become available, you can specify the URL to download updates intended for Parallels Desktop Standard and Pro editions, which are released a bit earlier. For more info, please see **Configuring Parallels Desktop Update Options**. The URL is as follows (when assigning it to the variable, put it in quotes): https://update.parallels.com/desktop/v16/parallels/parallels_sbscr_updates.xml **Note:** The "v16" part in the URL above indicates the current Parallels Desktop version number. If you are using a later version, substitute it with the correct number.
`updates_auto_check`	Specifies how often Parallels Desktop should check for updates. Possible values: `"0"` — never `"1"` — once a day `"2"` — once a week `"3"` — once a month
`updates_auto_download`	Specifies the automatic update download options: `"on"` — Download updates automatically. This value should be specified when using a local update server.

	`"off"` — Notify the user about updates but don't download them automatically. This option is useful only when updates are downloaded from the Parallels website and the user has full control over the update functionality.

Help and support

The **Help and Support** section is used to specify the action for the **Help** > **Support Center** menu item in the Parallels Desktop graphical user interface.

Variable	Description
`support_url`	Specifies a URL of a page that will be displayed when user selects the **Help > Support Center** menu option in the Parallels Desktop graphical user interface. To display the default message, specify an empty string (this is the default behavior). To display your own Web page (help desk, wiki, etc.), specify its URL. See also **Customizing Support Center options** (p. 75).
`lic_admin_url`	Specifies a URL that will be included in error message dialogs related to licensing operations. The URL should point to a web page or a resource that the user can visit to get help with the problem. The error message appears when there's a problem with activating, renewing, or deactivating a Parallels Desktop license. If you specify a URL using this variable, it will be included in the message dialog in the form "For details click <URL>". If you don't specify a URL (comment out the variable or specify an empty string), the default "Contact your system administrator" message will be displayed.

Problem reporting

The **Problem Reporting** section is used to specify whether Parallels Desktop problem reports should contain screenshots of the macOS and virtual machine desktops. You can exclude the screenshots for security reasons.

Variable	Description
`report_allow_screenshots`	`"yes"` — Include screenshots of the macOS and virtual machine desktops in Parallels Desktop problem reports. `"no"` — Do not include the screenshots.

Customer experience

The **Customer Experience** section allows you to specify whether the Macs should participate in the Parallels Customer Experience Program. The Parallels Customer Experience Program is a feedback solution that allows Parallels Desktop to automatically collect usage statistics and system information that will help Parallels to develop new features and updates for future releases. For more information, please see https://www.parallels.com/pcep/.

Variable	Description
`cep_participation`	`"off"` — participation in the program is turned off. `"on"` — participation in the program is turned on. See also **Participating in Customer Experience program** (p. 76).

Security

The **Security** section allows you to enable or disable the password requirement for a number of Parallels Desktop operations.

Variable	Description
The following set of parameters allows you set a custom password and then specify whether this password is required to perform a corresponding action in Parallels Desktop. For more information about this feature, please see **Restricting user actions in Parallels Desktop with a custom password** (p. 58).	
`current_password`	The current password (if one is already set in target Parallels Desktop installations). Use this key if you want to disable or change the current password, or if you want to enable/disable any of the protected actions when the password is already set. For the following cases, both the “current_password” and the “new_password” (see below) keys must be uncommented and set accordingly: 1. To disable the password completely, the “current_password” key must contain the current password and the “new_password” key must be set to "" (empty string). 2. To change the password, the “current_password” key must contain then current password and the “new_password” key must contain the new password. 3. To enable/disable any of the actions without changing the password, both the “current_password” and the “new_password” keys must contain the current password.
`new_password`	Custom password. If no password is currently set, use this key to specify the password. If a password is currently set, see the description above.
`password_to_edit_prefs`	`"on"` — require the password to open the Parallels Desktop Preferences dialog. `"off"` — the password is not required.
`password_to_create_vm`	`"on"` — require the password to create a new virtual machine. `"off"` — the password is not required.
`password_to_add_vm`	`"on"` — require the password to add an existing virtual machine. `"off"` — the password is not required.
`password_to_remove_vm`	`"on"` — require the password to remove a virtual machine. `"off"` — the password is not required.

`password_to_clone_vm`	`"on"` — require the password to clone a virtual machine or converting it to a template. `"off"` — the password is not required.
The following set of parameters allows you to specify whether a local Mac administrator password is required to perform a corresponding action in Parallels Desktop. Please note that the parameters described above provide a higher level of security since they allow you to use a custom password.	
`admin_password_to_edit_prefs`	**Note:** Supported since Parallels Desktop 15. `"on"` — require a local Mac administrator password to modify Parallels Desktop preferences. `"off"` — a password is not required.
`admin_password_to_create_vm`	`"on"` — require a local Mac administrator password to create a virtual machine. `"off"` — a password is not required.
`admin_password_to_add_vm`	`"on"` — require a local Mac administrator password to add an existing virtual machine. `"off"` — a password is not required.
`admin_password_to_remove_vm`	`"on"` — require a local Mac administrator password to remove a virtual machine from Parallels Desktop. `"off"` — a password is not required.
`admin_password_to_clone_vm`	`"on"` — require a local Mac administrator password to clone a virtual machine. `"off"` — a password is not required.
`hide_license_request_params`	Hide hostname in activation/heartbeat requests. `"on"` — hide. `"off"` — don't hide.

User experience

The User Experience section allows you specify options related to user experience.

Variable	Description
`enable_single_application_mode`	Specifies whether to enable Single Application Mode. For more information, please see **Single application mode** (p. 31). Possible values: `"yes"` — enable Single Application Mode. If the parameter is commented out, Parallels Desktop will be deployed using the "standard" mode. When using the Single Application Mode, it is recommended to prepare Windows guest operating system to have auto login enabled. **NOTE:** You cannot redeploy Parallels Desktop with this option set to `"no"` or commented out to disable Single Application Mode. For that, you will need to completely remove Parallels Desktop and then redeploy it with this option commented out.

`show_developers_menu`	Specifies whether to show or hide developer tools in the Parallels Desktop GUI. For more information, please see **Hiding Developer Tools in Parallels Desktop GUI** (p. 63) Possible values: `"no"` — hide developer tools. `"yes"` — show developer tools. Please note that this setting will be applied to every virtual machine included in the deployment package. You can also configure each virtual machine to hide (or show) developer tools prior to deployment, but with this option you can apply the setting automatically during deployment.

Single application mode

Overview

Single Application Mode is a special Parallels Desktop deployment option that allows you to completely hide Parallels Desktop and Windows on a Mac and make a Windows application appear like it's a native macOS app. This mode is designed for system administrators who want Mac users in their organization to run one or more Windows applications without being aware of Parallels Desktop or Windows running in a virtual machine.

When Parallels Desktop is deployed using Single Application Mode:

- A Mac user will not see the Parallels Desktop icon, user interface, or the virtual machine window at any time.
- A Windows application icon is added to the Dock. When the user clicks on the icon, the application will run on a Mac desktop like a native macOS application.
- The Mac location sharing feature is disabled in Parallels Desktop when deployed in Single Application mode. This means that the user will never see a Parallels Desktop message about using their current location in Windows.
- A Mac user will never know that Parallels Desktop and Windows are running on their Mac computer.

Configuring deployment options

To deploy Parallels Desktop using Single Application Mode, do the following:

1 Add a virtual machine to the deployment package. For instructions, see **Add a Virtual Machine** (p. 19). Please take note of the following:

- You can add only ONE virtual machine when using Single Application Mode.
- The virtual machine must be properly shut down before adding it to the deployment package (i.e. you need to properly shut down Windows). DO NOT simply close it, as this will be detected as a crash by Windows and a Mac user will have to deal with it at startup.

2 Add a Windows application stub to the deployment package that will be used to run a desired Windows application on a Mac. If you want to deploy more than one Windows application, add a corresponding stub for each one. For details, please see **Adding Windows Application stubs** (p. 21).

3 To enable Single Application Mode, set the `enable_single_application_mode="yes"` parameter in the `deploy.cfg` file, as described in **Configure deployment options** (p. 23). The parameter is included in the **User Experience** section of the `deploy.cfg` file.

4 Deploy Parallels Desktop to Mac computers as described in **Deploying Parallels Desktop and Virtual Machines to Mac Computers** (p. 34). If you want to deploy it on a single Mac, you can simply execute the deployment package (Parallels Desktop Autodeploy.pkg) on that Mac.

Configuring Windows

Please note that for Windows to be completely hidden on a Mac, you need to make some changes manually because they cannot be automated. The following list describes these changes:

- **Enable auto logon in Windows**. Make sure that Windows in the virtual machine doesn't ask the user to log on. If this is not done, a Mac user will see the Windows logon screen when Windows starts or reboots.
- **Disable Windows updates**. If this is not done, Windows may ask the user to install updates when they become available. You can install updates manually if needed.
- **Configure file associations in Windows**. This is necessary so that Windows doesn't open another Windows application when the user tries to open a file from the primary application. For example, let's say you deployed Outlook for Windows. A Mac user may try to open a text file attachment in Outlook. Normally, the file will open in Notepad in Windows, which may confuse the user. To prevent this, you can associate text files with TextEdit (a macOS application) in a virtual machine. The ability to associate file extensions with macOS applications is a standard Parallels Desktop feature available in Windows in a virtual machine. In addition, we recommend that you have as little applications installed in Windows as possible in order not to create additional file associations.
- **Use the Productivity profile**. When creating a virtual machine for Single Application Mode, choose the **Productivity** profile in the virtual machine Installation Assistant. If you are using an existing virtual machine, change its profile by going to **Configuration** > **General** > **Configure for**, clicking **Change** and then selecting **Productivity**.
- **Turn Windows sounds off**. You may consider turning all Windows sounds off, so no standard Windows sounds are played on a Mac computer at any time. A quick way to do this is to disable sound support in the virtual machine configuration by going to **Hardware** and clicking the minus-sign icon to remove the **Sound** item from it.

Configuring macOS

If you are deploying Parallels Desktop on macOS Hight Sierra, macOS Catalina, or macOS Mojave, you need to make sure in advance that Parallels Desktop kernel extensions are either approved or don't require user consent on each Mac. This is particularly important when using the Single Application Mode because if the extensions are not approved, Mac users will see warning messages about them when they try to run a Windows application for the first time. For more information, please see **Kernel extensions in macOS** (p. 12).

Testing the deployment package

Once you have the Parallels Desktop deployment package configured, you can test it on a single Mac before you mass deploy it to other Mac computers in your organization.

To test the package:

1. Copy it to a Mac on which you want to test it. The Mac should have a configuration similar to other Mac computers on which you'll be deploying Parallels Desktop. Specifically, if your target Mac computers don't have Parallels Desktop and virtual machines installed, the test Mac shouldn't have them installed either. If target Macs have an older version of Parallels Desktop, the test Mac should have it installed too, so you can see what will be the results.
2. To speed up the execution of the package during testing, consider running it from the command line using /System/Library/CoreServices/Installer.app. When executed this way, the package will not be tested by macOS whether it is signed and the usual verification of the package will be skipped. Please note that if you run the package by double-clicking on it, macOS will complain that the package is not signed and will not install it. If you run the package by right-clicking and choosing **Open**, the check whether the package is signed will be skipped but the verification of the package will take a long time if you have one or more virtual machines in it (because of the large size of a typical virtual machine). When you use the Installer.app to run the package, the installation will begin immediately without any checks or verifications. All of the above only applies when you run the package manually. When you mass deploy it on Mac computers, verification is not performed and the installation is completely silent.
3. When the installation is complete, verify that Parallels Desktop is installed, activated, and is functioning properly. If your package is configured to deploy Parallels Desktop in Single Application Mode, try running the application and see that it starts and runs as it should.
4. Please note that when the package is executed, it writes logs into /var/log/install.log. If you experience issues, examine the logs. If that doesn't help, you can contact Parallels Support for business customers, which is available 24/7.

Read on to learn how to mass deploy the package using one of the Mac management tools.

Deploying Parallels Desktop and virtual machines to Macs

This section contains instruction on how to deploy the Parallels Desktop deployment package using the following solutions:

- Jamf Pro (p. 34)
- Parallels Mac Management for Microsoft SCCM (p. 40)
- Apple Remote Desktop (p. 42)
- Deploying macOS image using NetBoot (p. 46)

Note: In some cases, when deploying Parallels Desktop deployment package on a computer running macOS Catalina, you may get an error that "Parallels Desktop Autodeploy.pkg cannot be opened because the developer cannot be verified". To work around this issue, please do the following. After you prepared the deployment package (but before you deploy it), right-click the package and select **Open** from the context menu. In the dialog that opens, click the **Open** button and wait until macOS completes the file verification. Once the file is verified, use the deployment tool of your choice to deploy the package. See also https://kb.parallels.com/124989

Deploying with Jamf Pro

Jamf Pro includes the Software Distribution functionality that you can use to deploy the Parallels Desktop package to Mac computers in your organization. To deploy the package you need:

- Jamf Pro server installed and configured.
- Target Mac computers enrolled in Jamf Pro.
- A distribution point (cloud or file share) configured and be accessible from the target Mac computers.

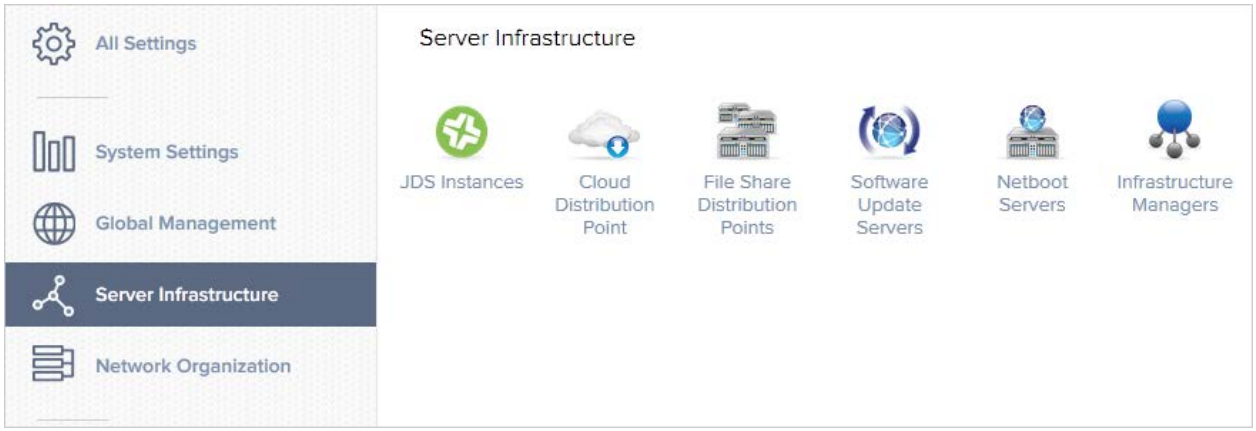
Add a distribution point

A distribution point is a server that hosts files for distribution to computers. If your Jamf Pro installation doesn't have a distribution point, you need to add one to host the Parallels Desktop deployment package.

To add a distribution point:

1 Open the Jamf Pro console and log in to your Jamf server.

2 Click the gear icon in the upper right and then click **Server Infrastructure** in the left pane.



3 Jamf Pro supports cloud-based (content delivery networks) and file share distribution points. Depending on what is available to you, click the **Cloud Distribution Point** or the **File Share Distribution Points** icon. The instructions below are for setting up a file share distribution point. If you would like to set up a cloud distribution point, please consult Jamf Pro documentation for details.

4 After you click the **File Share Distribution Point** icon, click **New**.

5 On the **General** tab page:

- Type a name for the distribution point.
- Specify the IP address or the host name of the distribution point server.
- Select the **Use as master distribution point** option.

6 Click the **File Sharing** tab and specify the following:

- **Protocol:** Select AFP or SMB depending on which protocol is used on your server for file sharing.
- **Share name:** Specify the share name. For example, if your server name is MYSERVER and your full share name is \\MYSERVER\JAMF-SHARE, specify JAMF-SHARE in this field.
- **Port:** In most cases the default value is what a given protocol normally uses. If you know that your server uses a different port number, specify it here.
- **Read/Write Account:** Specify credentials of an account that has read/write access to the share.
- **Read-Only Account:** Specify credentials of an account that has read-only access to the share.

7 If your server supports HTTP downloads, select the **HTTP/HTTPS** tab and then select the **Use HTTP downloads** option. Based on our own and other users' experience, HTTP/HTTPS-enabled distribution points are more reliable than AFP/SMB shares, but you can try both options and see which one works better for you.

8 Click **Save** to save the settings and add the distribution point to your Jamf Pro installation.

Add the deployment package to the distribution point

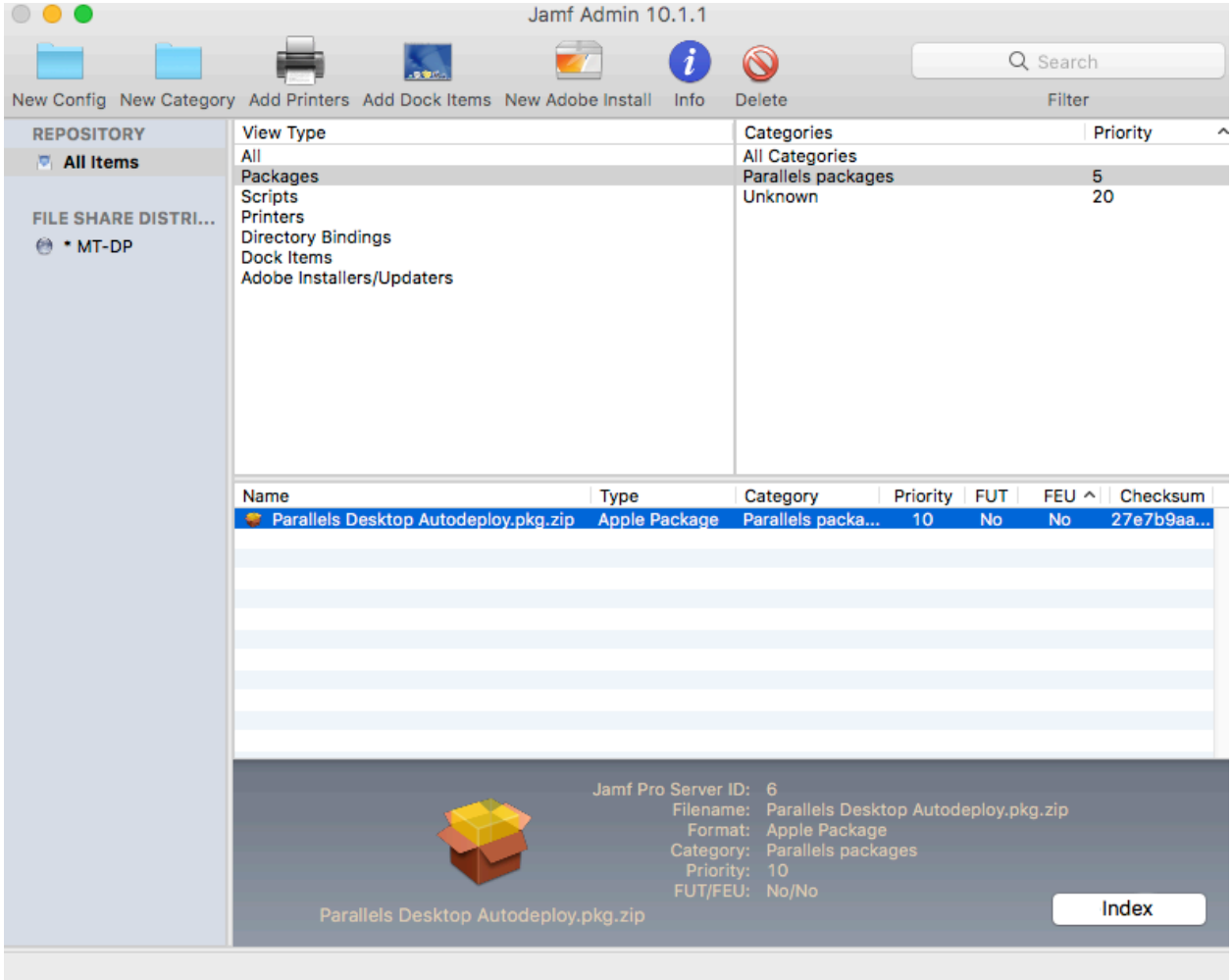
Note: The instructions below describe uploading of the deployment package using the Jamf Admin app. If you are using the web-based tool and the Google Chrome browser, you should zip the package before uploading it. This is necessary because ownership and permissions of the content files in an unzipped package may change during upload, which in turn may corrupt the deployment package itself. Please note that this behavior has been observed when using the Google Chrome browser. If you are using Safari or the Jamf Admin app, you don't need to zip the package prior to uploading it.

To add the deployment package to the distribution point:

1 Open the Jamf Admin app on a Mac and log in to your Jamf Pro server.

2 Drag the package to the main repository area (the middle area in the right pane).

3 The package will be uploaded to the master distribution point and will appear in Jamf Admin.



4 You can set other options if needed, such as add a package to a category or change the package priority. Indexing the package is not necessary for the purpose of deploying it on Macs.

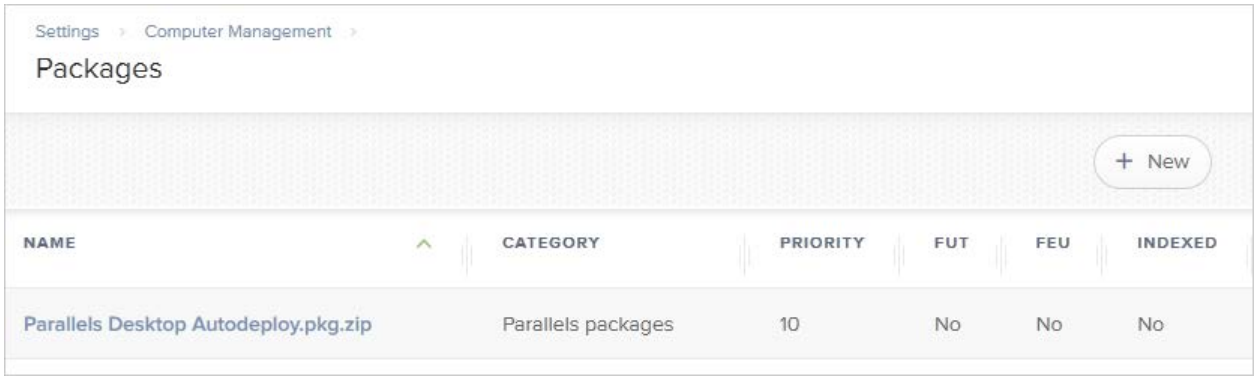
5 Close the Jamf Admin when done.

To verify that the package has been added to the distribution point:

1 Open the Jamf Pro console and login to Jamf Pro server.

2 Click the gear icon in the upper right and select **Computer Management** in the left pane.

3 Click **Packages** in the right pane. You should see the **Parallels Desktop Autodeploy.pkg.zip** package in the list.



4 If needed, you can modify the package display name and other settings. To do so, click the package and then click the **Edit** button in the lower right. Modify any of the settings as you require (except the file name). All settings are optional. If not sure, simply leave them unchanged.

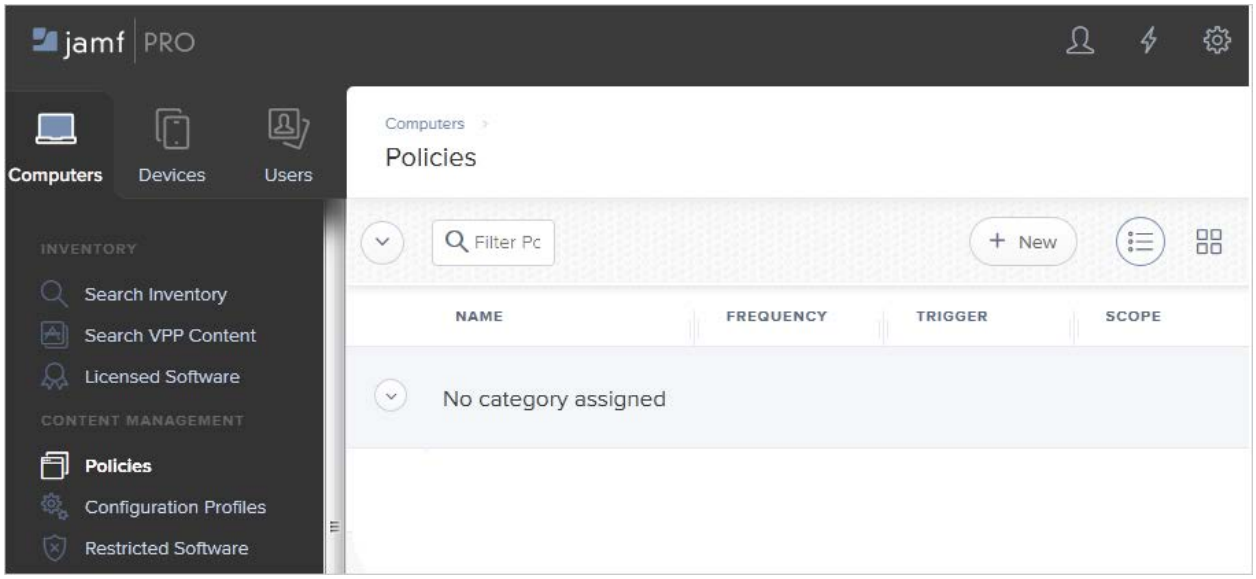
Install the package

You can install the package on Mac computers using a policy or you can use the Jamf Remote app that works similar to Apple Remote Desktop. This section contains instructions on how to install a package using a policy. For Jamf Remote instructions, please refer to the Jamf Pro documentation.

Create a policy

To create a policy:

1 In the Jamf Pro console, click the **Computers** tab in the left pane and then click **Policies**.



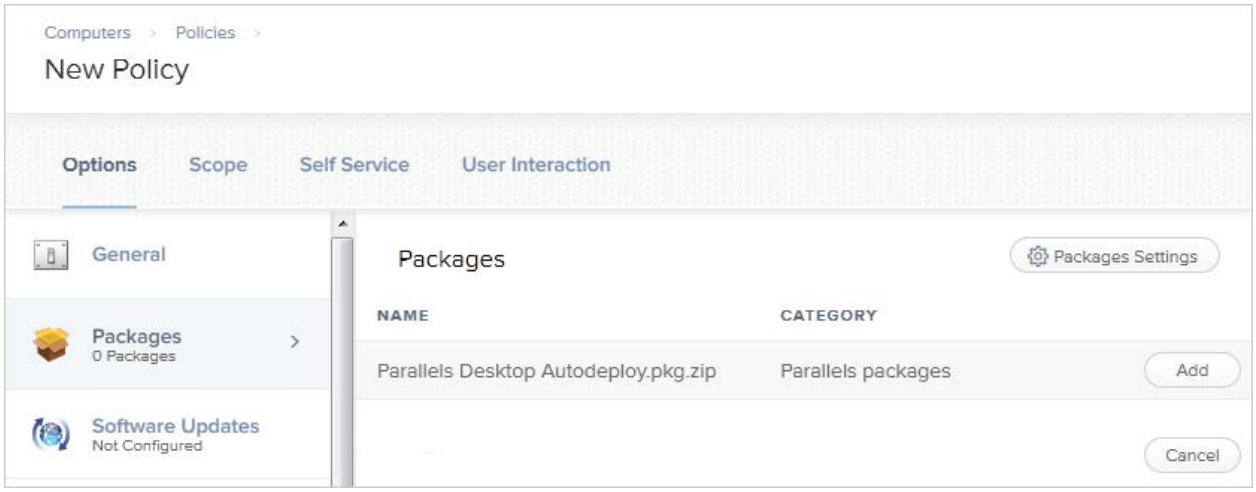
2 In the right pane, click **New**. The **New Policy** pane loads where you can define the policy.

3 In the **General** payload, specify the following:

- Type a name for the policy.
- Select the **Enabled** option.
- Specify a category (optional).
- In the **Trigger** section, select one or more events that should trigger the policy retrieval on target Mac computers.
- In the **Execution Frequency** drop-down list, select a frequency at which to run the policy. Since Parallels Desktop has to be installed just once, you may select **Once per computer** or **Once per user** (to install Parallels Desktop for each user if a Mac has more than one).
- The **Target Drive** option allows you to specify the drive on which to run the policy. You would normally use the default option, which will run the policy on the boot drive.
- Other options in the **General** payload allow you to configure limitations on when the policy can and cannot run. This includes server-side and client-side limitations. You can specify them according to your needs or you can leave them blank.

4 To specify a package for the policy, click the **Packages** payload and then click the **Configure** button in the right pane.The pane will be populated with the list of available packages.

5 Locate the **Parallels Desktop Autodeploy.pkg.zip** package and click **Add**.



6 Select a distribution point that contains the Parallels Desktop package. If you select the **Specific file share distribution point** option, choose the distribution point. If the distribution point is both an AFP/SMB and HTTP share, Jamf will try to use the HTTP option to download the package to target Macs. If needed, you can force it to use the AFP/SMB share by selecting the **Force file sharing over AFP/SMB** option. Based on our observations, HTTP shares work more reliably in Jamf, but you can try different options if you are experiencing issues with mounting shares on Mac computers during the policy execution.

7 In the **Action** drop-down list, select how the package should be downloaded and installed on Mac computers:

- **Install** — when the policy runs, the package is downloaded and installed on a Mac computer as a single operation. Note that if you have a large number of Macs and the policy runs on all of them simultaneously, all of them will try to download the policy at the same time. For this reason, you may consider the options below to avoid overloading your network.
- **Cache** — when the policy runs, the package is downloaded to a Mac computer but is not installed at that time.
- **Install Cached** — a policy with this action installs the package that has been downloaded previously using the **Cache** option. You can use this and the "Cache" options if you are deploying Parallels Desktop on a large number of Macs. First, you "cache" the deployment package on each Mac at a convenient time. Once that's done, you run the "Install Cached" policy to install the package, which will be a completely local operation for each Mac. This way, you can ensure that Parallels Desktop is deployed on each Mac at roughly the same time (e.g. on the weekend) without delays.

8 The rest of the payloads are not required for Parallels Desktop package deployment.

9 To specify target Mac computers, click the **Scope** tab at the top of the pane. You can specify targets by computer name or user name (or both).You can also set limitations and exclusions to further narrow down the target list.

Note: If you are deploying Parallels Desktop on both Apple M1 and Intel based Macs, you need to select computers of a particular type, depending on which Parallels Desktop installer your deployment package contains. This can be done using Smart Groups. For more information, please see the following article on the Jamf website: https://docs.jamf.com/best-practice-workflows/jamf-pro/managing-macos-updates/Creating_a_Smart_Computer_Group_to_Identify_Target_Computers.html

10 If you would like the package to appear in the **Self Service** app on a Mac, click the **Self Service** tab, select the **Make the policy available in Self Service** option, and specify additional options if needed. The **Self Service** app allows the user to initiate the policy retrieval manually without waiting for it to trigger.

11 The **User Interaction** tab page allows you to specify messages that will be displayed to a Mac user when the policy runs on their Macs. If you want the installation to be completely silent, you can skip this page.

12 When done, click **Save** to save the policy.

Deploy the package

Once the policy is retrieved by a Mac computer, it will install and activate Parallels Desktop on it. Once completed, the Mac user can begin using Parallels Desktop immediately.

If you are testing your policy, you can wait for it to trigger or you can run it manually using the **Self Service** app on a Mac. The app is installed when a Mac is enrolled in Jamf Pro and can be opened from the Applications folder in macOS. If there errors executing the policy, you can review them in the app. Please also note that when testing a policy, don't try to run it on the same Mac that you use as a distribution point because an attempt to mount a share on the same Mac that hosts it will fail.

Deploying with Parallels Mac Management for Microsoft SCCM

Parallels Mac Management for Microsoft SCCM enables you to deploy Parallels Desktop to Mac computers via the SCCM Software Distribution functionality. For the complete information about Parallels Mac Management for Microsoft SCCM, please read the Administrator's Guide that can be downloaded from https://www.parallels.com/products/mac-management/resources/.

The basic steps of deploying Parallels Desktop using Parallels Mac Management are:

1 Prepare the Parallels Desktop deployment package (described earlier in this guide).

2 Create a software distribution package using the Microsoft Configuration Manager console.

3 Send the package to a distribution point.

4 Deploy the package.

The following describes steps 2-4 in detail.

Create a software distribution package

The Parallels Desktop deployment package is distributed to Macs using the standard Configuration Manager functionality:

1. In the Configuration Manager console, navigate to **Administration / Overview / Application Management / Packages**.
2. On the toolbar, click **Create Package**.
3. Use the **Create Package and Program Wizard** to create a software distribution package and a program.
4. On the **Package** page, specify the package name and an optional description, manufacturer, language, and version information. Select the **This package contains source files** option and click **Browse**. Select the folder that contains the `Parallels Desktop Autodeploy.pkg` folder. Please note that you must select the parent folder of the `Parallels Desktop Autodeploy.pkg` folder, NOT the `.pkg` folder itself.
5. Click **Next**.
6. On the **Program Type** page, select the **Standard program** item and click **Next**.
7. On the **Standard Program** page, specify the information about the program. Type the following in the **Command line** field:

   ```
   chmod 700 "Parallels Desktop
   Autodeploy.pkg/Contents/Resources/postflight" && installer -pkg
   "Parallels Desktop Autodeploy.pkg" -target /
   ```

8. When done specifying the program information, click **Next**.
9. Click **Next** on the **Requirements** page.
10. Review the summary and click **Next** to create the package.

Upload the package to the distribution point

To upload a copy of the package to the distribution point, right-click the package and choose **Distribute Content** in the pop-up menu. Use the **Distribute Content Wizard** to specify a distribution point to which you want to send the package.

Please make sure that the distribution point is properly configured. For more information, see the Parallels Mac Management Administrator's Guide.

Deploy the package

To deploy the package:

1. In the Configuration Manager console, right-click the package and then click **Deploy** in the pop-up menu. The **Deploy Software Wizard** opens.

2 On the **General** page, click the **Browse** button next to the **Collection** field and select the collection containing your Macs (e.g. **All Mac OS X Systems**).

> **Note:** If you are deploying Parallels Desktop on both Apple M1 and Intel based Macs, you need to select a collection containing computers matching the Parallels Desktop installer that your deployment package contains.

3 Click **OK** and then click **Next.**

4 On the **Content** page, verify the distribution point info and click **Next**.

5 Click **Next** on the **Deployment Settings** page.

6 On the **Scheduling** page, specify the schedule for this deployment. Click **New** to specify the assignment schedule. When done, click **Next**.

7 Use the default values on the rest of the wizard pages and complete the wizard.

The package will be advertised to Macs in the specified collection and will be distributed to them according to the specified schedule.

Deploying with Apple Remote Desktop

To use Apple Remote Desktop, you need Apple Remote Desktop Admin, which should run on your administrator Mac, and Apple Remote Desktop client, which should run on client Macs.

Apple Remote Desktop Admin doesn't come standard with macOS. You will have to obtain the software from Apple and install it on a Mac from which you are deploying Parallels Desktop. Apple Remote Desktop client is installed during the standard macOS installation.

> **Note:** Before continuing, please verify that the versions of Apple Remote Desktop Admin and each instance of Apple Remote Desktop client match. If they don't, update the necessary software.

Configure Remote Management Services on target Macs

For a remote management tool (ARD in our case) to be able to distribute the software, the Remote Management Services have to be configured in macOS on each individual Mac. In order to do so, you need to log in to each Mac and perform the following steps:

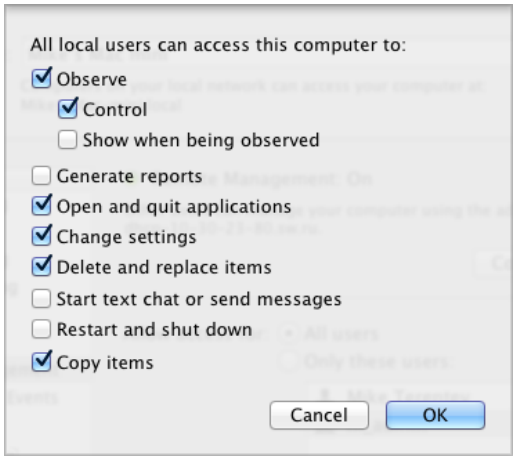
1 In macOS, open **System Preferences**.

2 Click **Sharing**.

3 In the Service list, select **Remote Management**.

4 Click **Options**.

5 Select the following options: **Observe**, **Control**, **Open and quit applications**, **Change settings**, **Delete and replace items**, **Copy items** (see the picture below).



6 Click **OK**.

7 Close **System Preferences**.

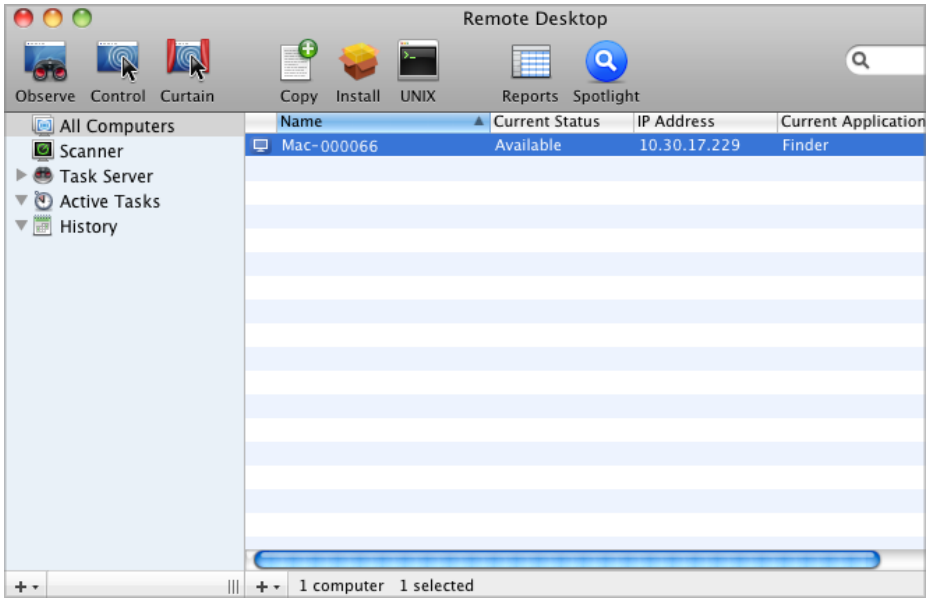
Deploy the package

To deploy the package to Macs:

1 On the administrator Mac, open the Finder and click **Applications** > **Remote Desktop** to display the main window of Apple Remote Desktop.

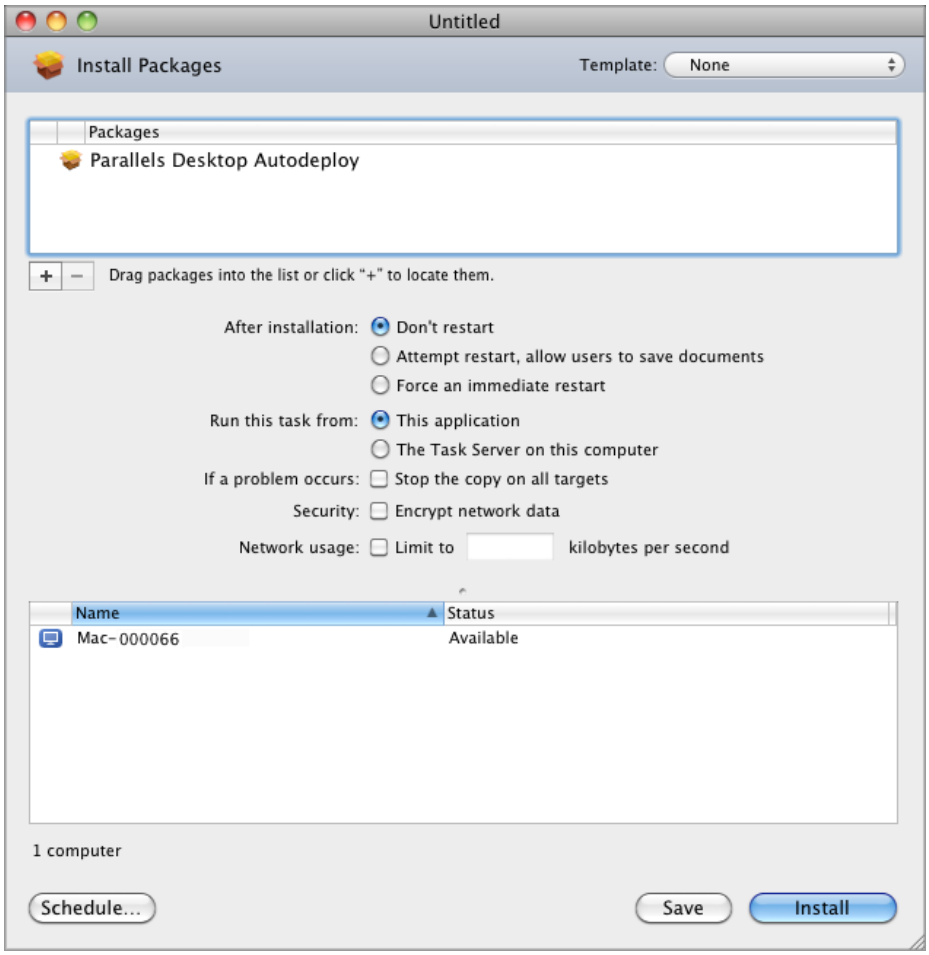
> **Note**: If you have set up a Task Server, you can deploy the package from that server.

2 Click **All Computers** in the left section of the ARD main window. The Mac computers connected to your network are displayed in the right pane of the window.



3 Select the desired destination Macs and click **Install** in the ARD toolbar (or select **Manage** > **Install Packages** menu). The **Install Packages** window opens.

4 Add the `Parallels Desktop Autodeploy.pkg` file to the **Packages** list either by dragging it there with the mouse or by locating the package using the '+' button



5 Click **Install** to distribute the deployment package to selected Macs. The installation progress is displayed in the main window. The result of the package distribution to each individual Mac is displayed in the **Status** column.

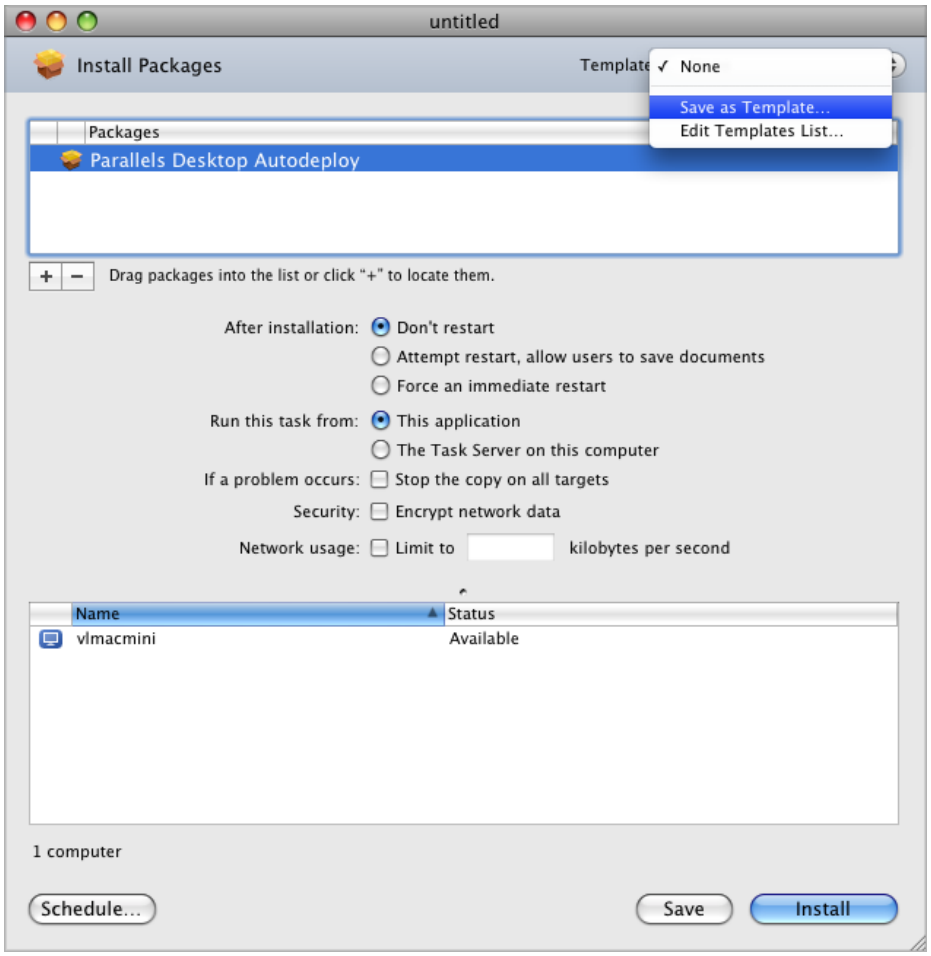
When the package is distributed to a Mac, it runs on it and installs Parallels Desktop and the virtual machine(s). It then activates Parallels Desktop using the supplied license key.

Using Apple Remote Desktop templates

Apple Remote Desktop allows you to save package installation settings as templates. This feature may be useful if you need to deploy Parallels Desktop more than once using different settings. For example, you may have a virtual machine specifically configured for your developers and another one configured for your graphics designers or quality assurance engineers. Instead of modifying the deployment package every time you need to deploy Parallels Desktop to a specific group of Mac users, you can create separate packages and save them as templates.

To create an ARD template:

1. On the administrator Mac, open Apple Remote Desktop.
2. Click **All Computers** in the left pane of the ARD main window.
3. Select the computers where the deployment package is to be installed and click **Install** in the ARD toolbar.
4. In the **Install Packages** window, add the desired Parallels Desktop deployment package either by dragging it to the top area of the displayed window or locating it by using the '+' button.
5. Save current installation setup as a template by selecting **Save as Template** from the **Template** pop-up menu located in the upper right-hand corner of the **Install Packages** window.



6. In the displayed window, specify the name that you would like to use for the template and click **OK**.

The next time when you are ready to deploy Parallels Desktop to a specific group of Mac users, simply select the desired template from the same **Template** pop-up menu in the **Install Packages** window and then click **Install**.

Deploying macOS image using NetBoot

If you are using NetBoot to deploy a macOS image with Parallels Desktop Business Edition preinstalled, some additional steps need to be taken to properly activate a copy of Parallels Desktop on individual Macs.

The following steps describe the process of installing Parallels Desktop and creating an image from it suitable for NetBoot deployment:

1. Install macOS on a Mac.
2. Install Parallels Desktop Business Edition.
3. Make sure your Mac can connect to the Internet.
4. Activate Parallels Desktop using your Business Edition license key.
5. Tune Parallels Desktop settings and add or create virtual machines according to your needs.
6. Open Terminal and deactivate the Parallels Desktop license by typing the following command:

   ```
   prlsrvctl deactivate-license
   ```

   On successful license deactivation, you should see the following output:

   ```
   The License has been successfully deactivated
   ```
7. Quit Parallels Desktop.
8. In Terminal, type the following command to prepare Parallels Desktop to activate a license in deferred mode:

   ```
   prlsrvctl install-license --key <key> --deferred
   ```

   where `<key>` is the Parallels Desktop Business Edition license key.

   On success, you should see the following output:

   ```
   The License has been successfully prepared for deferred
   installation.
   ```

   This means that the next time Parallels Desktop starts, it will try to activate the license automatically. The Internet connection is required for activation. If a Mac can't connect to the Internet, the Parallels Desktop license activation will be postponed until the Internet connection is available.

You can now create a macOS image from this system and deploy it to other Macs in your enterprise using NetBoot.

Using a script to Change VM options

When you need to change configuration settings of all virtual machines that are already registered on a Mac computer, you can use the Parallels desktop command-line interface (p. 77). To do so, you first need to create a script to perform a desired configuration modification. You can then execute the script on a Mac computer using one of the remote Mac management tools described earlier in this chapter.

The following is a script example that disables the auto pausing option for all virtual machines registered on a Mac computer:

```
for i in $( prlctl list -a --info | grep "ID" | sed 's/.....//;s/.$//' ); do
  prlctl set $i --pause-idle off
done
```

The script above uses the `prlctl list` command to first obtains a list of registered virtual machines and then (inside the loop) sets the `--pause-idle` option for every VM to "off", which disables pausing of an idle virtual machine.

The complete command-line reference is documented in the **Parallels Desktop Command Line Interface** chapter (p. 77).

CHAPTER 3

Parallels Desktop for Mac Business Edition Features

This chapter describes features that are specific to Parallels Desktop for Mac Business Edition.

In This Chapter

Using Configuration Profiles 48
Restricting User Actions in Parallels Desktop with a Custom Password 58
Restricting a Virtual Machine Configuration with a Custom Password 59
Locking a Virtual Machine on Suspend 61
Setting an Asset Tag in the Virtual Machine BIOS 61
Using Custom Graphics and Links in the Control Center 62
Hiding Developer Tools in the Parallels Desktop GUI 63
Encrypting a Virtual Machine from the Command Line 64
Single Application Mode 65
Setting an Expiration Date for a Virtual Machine 66
Enforcing USB Device Policies 67
Resetting the Guest OS Password 68
Setting Up a Local Update Server 69
Configuring Parallels Desktop update branch 74
Customizing the Support Center option 75
Participating in the Customer Experience Program 76

Using Configuration Profiles

Overview

A configuration profile in Parallels Desktop for Mac Business Edition is a set of parameters that can be applied remotely to a Parallels Desktop installation and modify the appearance or behavior of Parallels Desktop according to organization policies. Configuration profiles are created in an organization's Parallels business account. You must be the administrator of the account to create and manage configuration profiles. License administrators (admins who are allowed to manage specific licenses) cannot manage configuration profiles.

At the time of this writing, configuration profiles can be used to enable and configure the following functionality in Parallels Desktop for Mac Business Edition:

- **Provisioning a corporate virtual machine image**
- **Enabling major version upgrades**

We'll talk in detail about the functionality listed above later in this part of the guide. This and the following two sections talk about configuration profiles in general and describe how to create and apply them.

Configuration profile payloads

Configuration profiles consist of payloads, each containing settings for a particular functionality. For example, the **VM Image** payload allows you to configure virtual machine image provisioning, while **Product Updates** payload allows you to manage Parallels Desktop updates. The configuration profile itself is created and configured the same way regardless of which of its payloads are configured and enabled.

A configuration profile can have one or more payloads configured and/or enabled at the same time. For example, you can configure and enable a particular payload in one profile and a different payload in another profile. This allows you to enable one functionality for one group of users and another functionality for a different group (see below how configuration profiles are applied to Mac computers). You can create as many profiles as necessary.

Applying configuration profiles to Mac computers

Configuration profiles are applied to registered Mac computers based on a license or sublicense key that computers are using to run Parallels Desktop. After you create a configuration profile, you need to apply it to one or more license or sublicense keys in your subscription. By doing so, you are essentially applying the profile to Mac computers on which Parallels Desktop was activated using that license key.

The rest of this part of the guide describes how to:

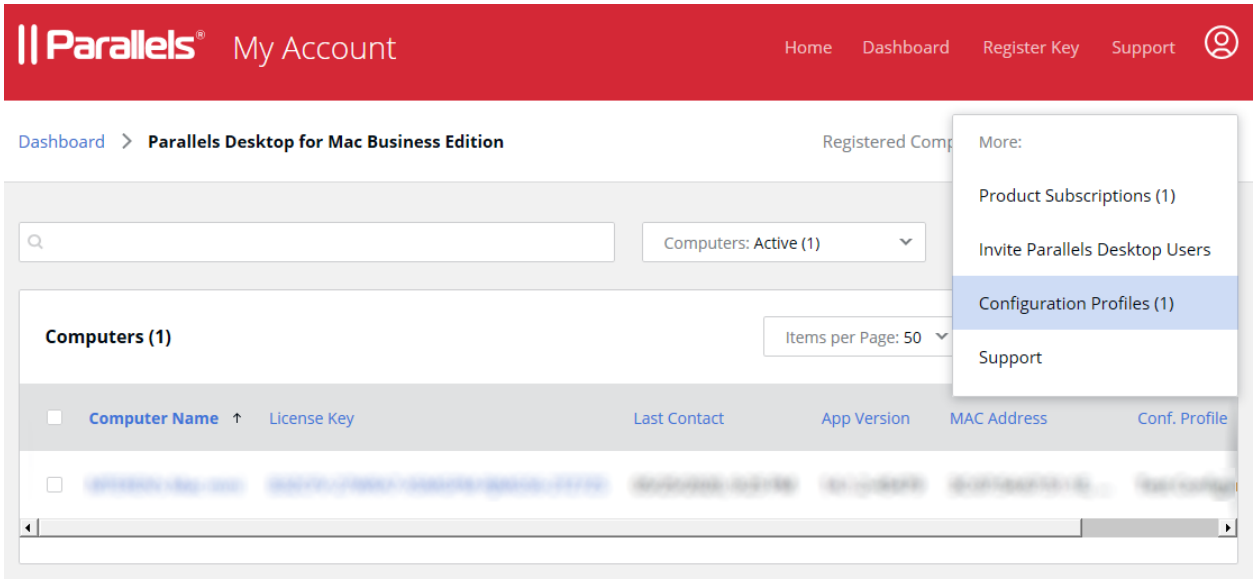
- Create a configuration profile
- Apply the configuration profile to a license or sublicense key
- Configure individual configuration profile payloads

Create a configuration profile

To create a configuration profile:

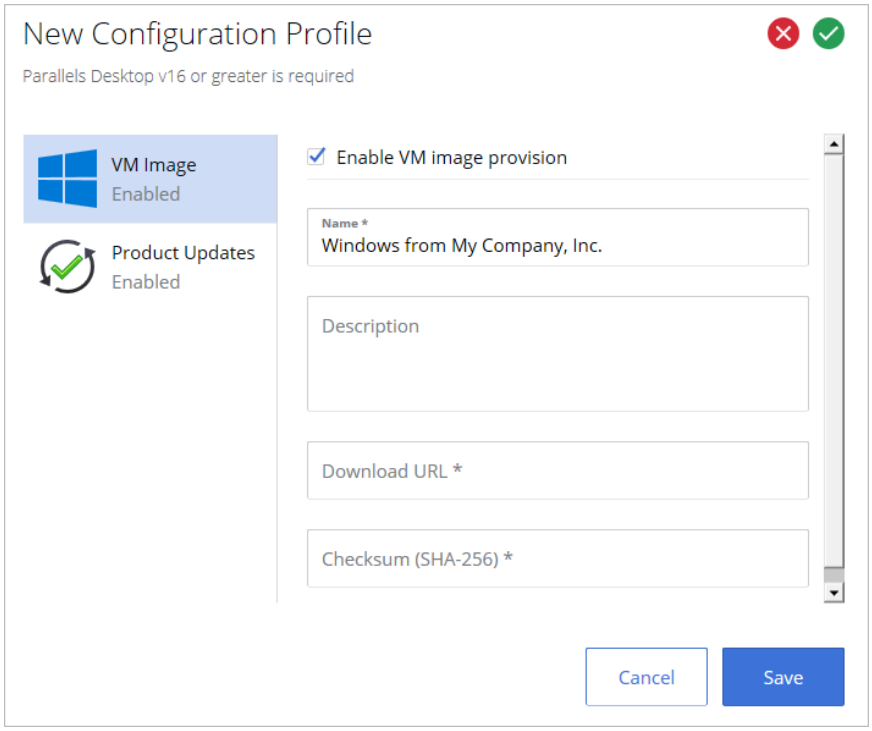
1. Log in to your Parallels business account.
2. In the **Parallels Desktop for Mac Business Edition** product card, click **Registered computers**.

3 Click the **More** item in the main menu (at the top) and choose **Configuration Profiles** as shown on the screenshot below.



4 The page listing configuration profiles opens. If you haven't created any profiles yet, the list will be empty.

5 Click the **Create Profile** button. A dialog opens where you can configure the profile.



6 To replace the default profile name (top left), simply erase the default name and type a new one.

7 The payloads are listed in the left pane (**VM Image** and **Product Updates** on the screenshot above). To configure a payload, select it and then specify the necessary settings in the right pane. Each payload has the "Enable..." option at the top of the right pane. This option enables or disables a payload, but doesn't change or discard the payload settings. When a payload is enabled, it is included in the configuration profile when the profile is sent to Mac computers. When a payload is disabled, it is not included, so Mac computers don't receive it.

8 When done, click **Save** to save the configuration profile.

At this point we will not configure any of the payloads yet and will go straight to applying the configuration profile to a license or sublicense keys (it is allowed to create a profile with all payloads disabled). Once you learn how to create and apply a configuration profile, we'll talk about how to configure and use each individual payload.

Applying a configuration profile to a license key

Configuration profiles are applied to registered Mac computers based on a license or sublicense key that they are using to run Parallels Desktop. By applying a configuration profile to a license or sublicense key, you are essentially applying it to Mac computers that use (or will use in the future) that key.

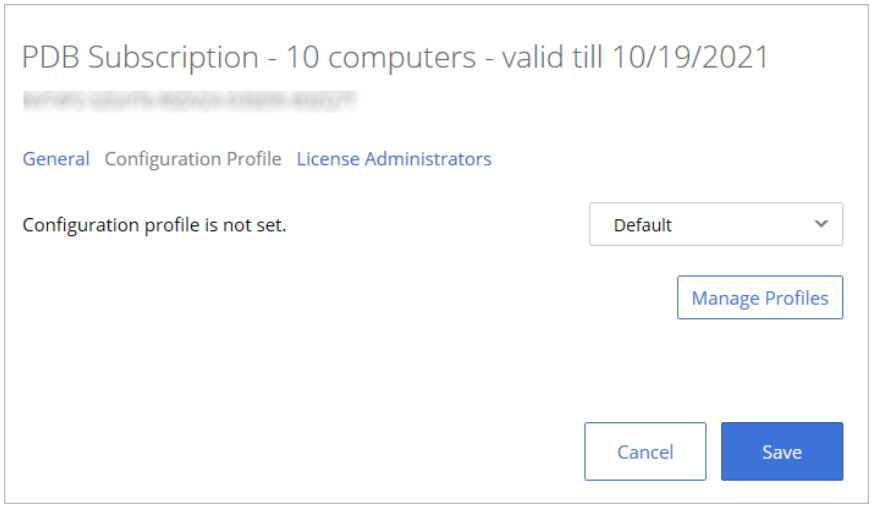
Note: You can apply a configuration profile to one or more license or sublicense keys. However, a single license key can have just one configuration profile applied to it.

To apply a configuration profile to a license or sublicense key:

1 In Parallels My Account, click **Dashboard** in the top menu and then click **Active subscriptions** inside the **Parallels Desktop for Mac Business Edition** product card.

2 Click a subscription to open a page containing the subscription information.

3 In the **License Keys** list, choose a license or sublicense key and click the "gear" icon at the end of the row. This opens a dialog containing the license key information and settings. In the dialog, select the **Configuration Profile** tab.



4 Initially, the tab page will say that "Configuration profile is not set" and the drop-down menu next to it will contain the "Default" profile. This is because you haven't applied a custom configuration profile to this license key yet.

> **Info:** "Default" is a built-in profile that Parallels Desktop is using internally. It doesn't contain any of the payloads that you can configure when you create a custom profile. If you don't have any custom profiles assigned to a license key, the "Default" profile is used.

5 Expand the drop-down menu and select the configuration profile that you created earlier.

6 Click **Save**.

> **Note:** If you apply a configuration profile to a primary license key (not a sublicense), all derived sublicenses are automatically associated with the same configuration profile. However, if later a different configuration profile is applied to the primary license key, the profile association on sublicenses will not be affected (i.e. the profile assigned to them originally will remain). You can apply a different configuration profile to a sublicense key at any time if needed.

Once a configuration profile is applied to a license key, the following will happen on Mac computers that use this key:

- The next time Parallels Desktop communicates with Parallels cloud, it will receive the configuration profile and will save the data that it contains locally.
- When an action is performed (by the user or by a scheduled event) that has to do with one of the configuration profile payloads, the data is read from the local storage and is used accordingly depending on the payload and its settings. This is described in more detail in topics that describe individual payloads.

This concludes the description of how to create a configuration profile and how to apply it to a license or sublicense key. The subsequent sections describe how to configure individual payloads and how to use the corresponding functionality when managing Parallels Desktop installations in your organization.

Provisioning a corporate VM image

Beginning with Parallels Desktop 16 for Mac Business Edition, IT administrators have an option to provision a corporate Parallels Desktop virtual machine image from a link that they specify in Parallels My Account.

> **Note:** Due to differences in supported operating systems on Mac computers with Apple M1 chip and Mac computers with Intel processor, different VM images must be used depending on the processor type. Learn more on the system requirements page at https://www.parallels.com/requirements/. As a result, two different configuration profiles must be created to provision a proper VM to a Mac computer with a given processor. To assign a configuration profile to the group of Mac computers with a homogeneous architecture, this group of computers should be activated with a sublicense key. The key should be assigned a profile with the correct VM image.

Here's a quick overview of how this functionality works:

1. An administrator first creates a Parallels virtual machine image with the operating system installed. The virtual machine will serve as a corporate VM image to be deployed on users' computers to run Windows applications used in the organization.
2. The virtual machine is then saved as an archive (ZIP or PVMP, we'll talk about archive formats later) and is placed on a server from which Parallels Desktop users can download it to their computers via HTTP or HTTPS.
3. The administrator creates a configuration profile in Parallels My Account and specifies the download URL of the virtual machine image (together with other required parameters).
4. When a Parallels Desktop user initiates the process of creating a new virtual machine, Parallels Desktop checks if a configuration profile with the VM image link exists and is applied to this Mac computer. If the profile exists, a dialog is shown to the user inviting them to download and install the corporate virtual machine image. If the user accepts, the virtual machine is downloaded to the user's computer and is registered in Parallels Desktop.

The subsequent sections describe how to perform the steps above.

Create a VM image

To create a VM image:

1. First, create a Parallels virtual machine, install the operating system in it, and configure the virtual machine according to your requirements.

   Note: Boot Camp-based virtual machines, archived virtual machines, and linked clones cannot be used for deployment.

2. Make sure the virtual machine is shut down.
3. If your virtual machine has snapshots, it is recommended that you remove them. This will significantly reduce the virtual machine size. Moreover, these snapshots may be unusable on another computer because of hardware differences.
4. When the virtual machine is ready, it needs to be saved as an archive before you make it available for download to your users. Choose from the following options:

   - **ZIP** — simply archive the virtual machine bundle (.pvm) as a .zip file. In general, you can use any archiving format that can be extracted using the `tar -xf` command. For example, zip, tar.gz, tar.bz2.

     Please note that you will also need to calculate an SHA-256 checksum for the ZIP file. You can use the `shasum` command or another third-party tool. For instructions on how to use the `shasum` command, please see **Calculating SHA-256 checksum** below.

   - **PVMP** — this is a Parallels own archiving format that was first introduced in Parallels Desktop 16. It is primarily used to transfer virtual machines from one computer to another and is optimized for archiving Parallels virtual machine bundles. For instructions on how to create a PVMP archive, please see below.

Creating a PVMP archive

To create a PVMP archive of a virtual machine, do the following:

1. Open the Parallels Desktop Control Center.
2. Right-click the virtual machine that you want to transfer and select **Prepare for Transfer**. Parallels Desktop starts packing the virtual machine. This process may take some time depending on the virtual machine size.
3. Once the package is created, you can right-click it and choose to show where it is stored in the Finder. The package has the .pvmp extension.
4. An SHA-256 checksum for the virtual machine package is calculated automatically and saved as VmName.sha256.txt file in the same folder. You will need it when creating a configuration profile for the VM image provisioning later. You can also calculate the checksum by executing the `shasum` command. For more info, please see **Calculating SHA-256** checksum below.

For additional information about the PVMP format, please also see the **Parallels Desktop for Mac User's Guide**, section **Prepare a Virtual Machine for Transfer**.

Calculating SHA-256 checksum

When creating a configuration profile for VM image provisioning (described later in this chapter), you will need to specify an SHA-256 checksum for the VM image archive. As was mentioned in the previous subsection, the checksum is calculated automatically for the PVMP archive. If you are archiving a virtual machine bundle using a different archiving format (or if you need to calculate the checksum again for the PVMP format) you can execute the following command from the command line:

```
shasum -a 256 <path>/"<archive_name>"
```

For example:

```
shasum -a 256 /Users/<your_user>/Parallels/"Windows.zip"
```

The output will look similar to the following:

```
63a90c3c38cc8c358221da339068fc1292b10bf7c00ed8449787b0e6019d706b
/Users/parallels/Parallels/Windows.zip
```

Upload the image

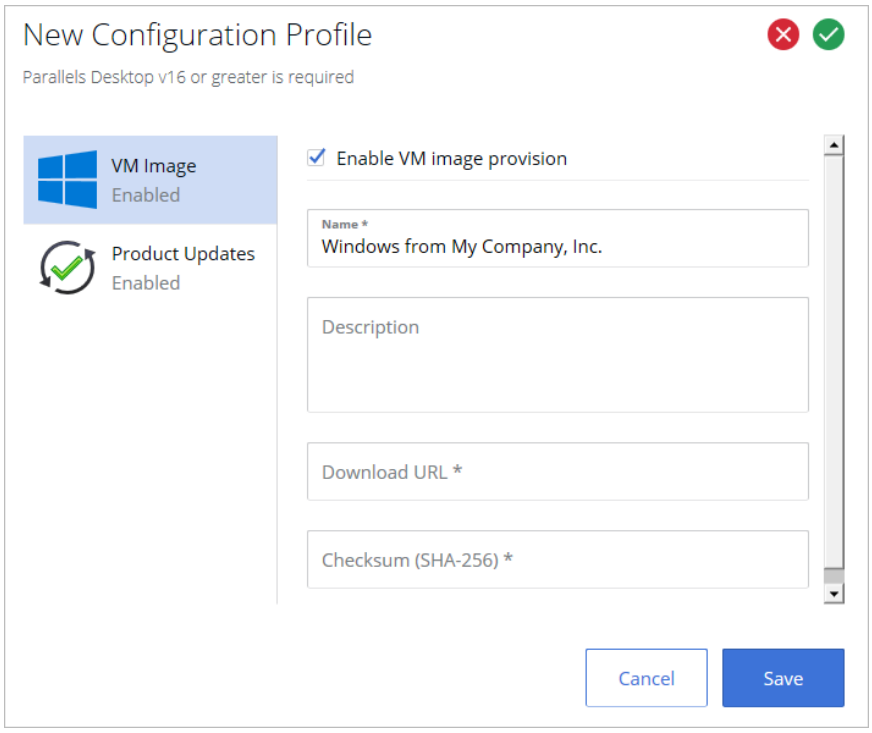
Once you have a virtual machine saved as a ZIP or PVMP archive, upload it to the server from which Parallels Desktop users can download it to their Mac computers via HTTP or HTTPS. Note that the server should be able to withstand a large number of multiple connections (depending on the number of end users). CDN is recommended for very large deployments.

Create a configuration profile

To create a configuration profile for VM image provisioning:

1 Begin creating a new configuration profile as described in the **Create a Configuration Profile** section (p. 49).

2 When you have the new configuration profile dialog open, select the **VM Image** payload in the left pane.



3 In the right pane, select the **Enable VM image provisioning** option and specify the following properties:

- **Name:** Type a name for the VM image as you want it be called in this profile. This is the name your users will see in Parallels Desktop when they receive an invitation to download it. This field is mandatory.
- **Description:** An optional description.
- **Download URL:** The VM Image download URL. Mac users must be able to download the image via HTTP or HTTPS using this URL. This field is mandatory. For additional info, please see **Create a VM Image** (p. 53).
- **Checksum (SHA-256)**: The VM image checksum. This field is mandatory. If you used the PVMP format to archive the virtual machine, then the checksum was calculated automatically and saved as VmName.sha256.txt file. If you archived the virtual machine using the ZIP or other supported format, you'll need to calculate the checksum. For the info about the PVMP format and how to calculate the checksum, please see **Create a VM Image** (p. 53).

4 Click **Save** to save the configuration profile.

The configuration profile now needs to be applied to a license or sublicense key. If you haven't done so already, use the instructions in **Applying a configuration profile to a license key** (p. 51) and apply the profile.

Installing VM image on a Mac computer

The corporate VM image policy is checked every time a new VM creation process is started by the user in Parallels Desktop on a Mac computer. If the corporate VM image policy is set (a configuration profile with the **VM Image** payload exists and has been applied to this Parallels Desktop copy), the Control Center shows an offer to download the corporate VM image.

If the user accepts the offer, the VM image download begins and the progress indicator is displayed (note that because of the large size of a VM, the download may take some time). If the user declines, he/she is taken to the Installation Assistant where they can create a virtual machine the usual way.

After the VM image download completes, it is unpacked and the virtual machine is registered in Parallels Desktop.

Enabling major version upgrades

Prior to version 16 of Parallels Desktop for Mac Business Edition, users could not receive notifications of Parallels Desktop major version upgrades. Beginning with version 16, an IT administrator has an ability to enable or disable such notifications, thus enabling or disabling automatic major version upgrades.

In the past, to upgrade Parallels Desktop for Mac Business Edition to a newer version, an IT administrator would need to set up a local update server or use a remote management tool, or install the new version manually on a Mac computer. With this new feature, administrators have an ability to automate major version upgrades if the organization policy allows it.

Here's a quick overview of how the new feature works:

1 You create a configuration profile in Parallels My Account and configure the **Product Updates** payload where you enable or disable the "Allow upgrade..." option.

2 You then apply the configuration profile to a license or sublicense key.

3 During the next communication with Parallels cloud, Mac computers running Parallels Desktop using that license receive this setting and save it locally.

4 When Parallels Desktop checks for regular updates, it sends the "Allow upgrade..." option value to the Parallels update server together with other information. If upgrades are allowed, the update server identifies this and sends back a notification if a new major version of Parallels Desktop is available.

5 Depending on how automatic updates are configured on the Mac computer, the new version of Parallels Desktop is either installed silently or the user is asked whether to install it.

Note: If a local update server is configured in your organization, the functionality described here will have no effect, even if you create a configuration profile and apply it to a license key used by Mac computers. For more info about local updates, see **Setting up a local update server** (p. 69).

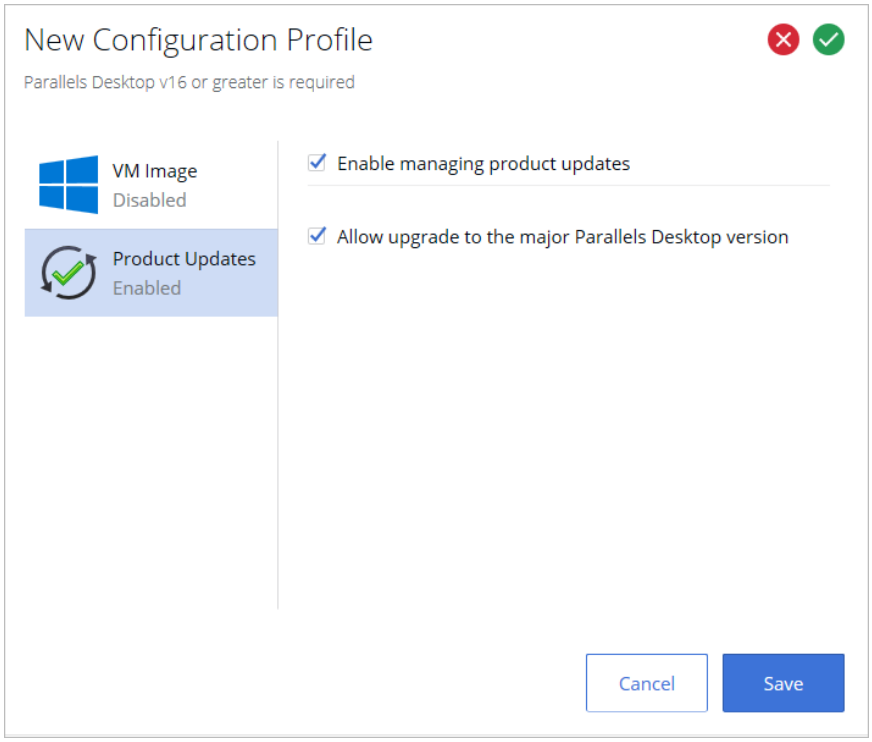
The subsequent topics describe in detail how to configure and use the major version upgrade functionality.

Create a configuration profile

To create a configuration profile for enabling major version upgrades, do the following:

1 Begin creating a new configuration profile as described in the **Create a configuration profile** section (p. 49).

2 When you have the new configuration profile dialog open, select the **Product Updates** payload in the left pane.



3 In the right pane, select the **Enable managing product updates** option. This will enable the payload, so when the configuration profile is sent to Mac computers, they will receive it.

4 To enable major version upgrades, select the **Allow upgrade to the major Parallels Desktop version** option.

5 Click **Save** to save the configuration profile.

Note: More options will be added to the **Product Updates** payload in the future. When you enable or disable the payload (the **Enable manging product updates** option), you are enabling or disabling the payload itself. Other options ("Allow upgrade..." currently) always remain at their current settings.

The configuration profile now needs to be applied to a license or sublicense key. If you haven't done so already, use the instructions in **Applying a configuration profile to a license key** (p. 51) and apply the profile.

Upgrading Parallels Desktop

Parallels Desktop periodically communicates with Parallels cloud to send and receive information and data. If there's a configuration profile on the cloud side that must be applied to a Parallels Desktop installation, the profile is included in the data package sent back to Parallels Desktop. Once received, the configuration profile is stored by Parallels Desktop locally.

When it's time for Parallels Desktop to check for updates, the "Allow upgrades..." value from the saved profile is sent to the Parallels update server together with other data. The update server identifies that upgrades are allowed and will check if a new major version of Parallels Desktop is available. If it is, it will send a notification back to Parallels Desktop.

Depending on how updates are configured in Parallels Desktop, the user will see a notification (with an option to upgrade or postpone) or the upgrade will be performed silently. When the upgrade is initiated, the new major version of Parallels Desktop is downloaded to the Mac computer and is installed on it. After that, Parallels Desktop restarts completing the upgrade.

Restricting User Actions in Parallels Desktop with a Custom Password

As an administrator of Parallels Desktop Business Edition, you have the ability to restrict the following user actions in Parallels Desktop:

- Creating a new virtual machine.
- Adding an existing virtual machine.
- Removing a virtual machine.
- Cloning a virtual machine or converting it to a template.
- Opening the Parallels Desktop Preferences dialog.

You can set a custom password (i.e. it can be different from the local Mac administrator password) and specify which user actions from the list above should require it. Once you apply these settings, even the local Mac administrator will not be able to perform these actions without specifying the password that you set.

To set the password and specify the actions:

1. In the Parallels Desktop menu bar, click **Parallels Desktop** > **Preferences**.
2. Click the **Security** tab.
3. Click the **Custom password: Turn On...** button.

4 In the dialog that opens, type a password. This becomes the custom password that will be required to perform actions that you will select in the next step. Click **OK**.

> **Note:** If later you need to change this password, click the **Change Password** button and follow the instructions on the screen.

5 Back in the **Security** dialog, select the user action(s) you want to restrict.

6 Close the dialog to apply the settings.

If a Mac user now tries to perform any of the actions that were password-protected (e.g. creating a virtual machine, etc.), they will be required to enter this custom password. To select or deselect an action in the Parallels Desktop Preferences dialog will also require the password.

The steps described above allow you to set the password and restrict user actions in Parallels Desktop on a particular Mac. If you want to apply these settings on multiple Mac computers in your organization, you can do it using the mass deployment process. For more information, please see **Mass Deployment of Parallels Desktop and Virtual Machines** (p. 8). Specifically, these options are set in the Security section of the `deploy.cfg` file, as described in **Configure Deployment Configuration Options** (p. 23).

Restricting a Virtual Machine Configuration with a Custom Password

Parallels Desktop Business Edition provides you with the ability to protect the configuration of a virtual machine with a custom password. When a password is set, even a local Mac administrator will be required to enter it in order to modify virtual machine settings.

Using the Parallels Desktop graphical user interface to set the password

To set a password in the Parallels Desktop graphical user interface:

1 Open Parallels Desktop and select a virtual machine.

2 On the Parallels Desktop menu bar, select Actions > Configure to open the virtual machine configuration dialog.

3 Select Security.

4 Click the Custom password: Turn On... button.

5 Enter a password, then enter it again to verify and click OK.

To change or remove the password:

- To change the password, click the Change Password button and follow the instructions on the screen.
- To remove the password, click Custom password: Turn Off and follow the instructions on the screen.

If the password is set and the user tries to view or modify the virtual machine configuration, they will be required to enter this custom password.

Using the command line utility to set the password

In addition to the graphical user interface, you can use the `prlctl` command-line utility to set a custom password for editing the virtual machine configuration.

> **Note:** The command option described below is available since Parallels Desktop 15. Older versions of Parallels Desktop used the `--password-to-edit` option, which is no longer supported.

To set the password, type the following command in Terminal:

```
prlctl set "vm_name" --custom-pwd
```

where vm_name is the virtual machine name in quotes. You'll be asked to enter a password and then confirm it.

To change or remove the password, type the same command as above:

```
prlctl set "vm_name" --custom-pwd
```

You'll be asked to enter the current password and then a new password.

To view the current protection status for a virtual machine, type the following command:

```
prlctl list "vm_name" -i
```

In the output, search for the **Security** section and look at the **Custom password protection** property. It will be either set to "on" or "off".

Using the mass deployment process to set the password

If you are mass deploying Parallels Desktop and one or more virtual machines, you can simply set the custom password in the source virtual machine. When a virtual machine is deployed on Mac computers, the password will be retained.

> **Note:** Prior to Parallels Desktop 15, the custom password needed to be set in the deploy.cfg file (a part of the Parallels Desktop deployment package) using the `password-to-edit` parameter. The parameter is no longer supported.

Locking a Virtual Machine on Suspend

To avoid possible security and privacy issues, a suspended Windows virtual machine can be completely locked from user interaction and viewing. When this option is enabled and a virtual machine is suspended, the Windows desktop in the virtual machine window (and in the Parallels Desktop Control Center) is replaced with a black background and the Windows session is interrupted. When the virtual machine is resumed, the Windows session is remained locked and the user will have to enter their user ID and password to unlock it and to see the Windows desktop.

To enable or disable this option:

1. Open Parallels Desktop and select the desired virtual machine (e.g. the source virtual machine when preparing it for mass deployment).
2. On the Parallels Desktop menu bar, select **Actions** > **Configure** to open the virtual machine configuration dialog.
3. Click the **Security** tab.
4. Depending on your needs select or clear the **Always lock Windows on suspend** option.
5. Close the dialog.

Note: The **Always lock Windows on suspend** option is available only for virtual machine running Windows with Parallels Tools installed. For all other virtual machines, the option will be hidden.

Setting an Asset Tag in the Virtual Machine BIOS

Asset tags help identify, control, and track computer assets in an organization. Parallels Desktop for Mac Business Edition provides the ability to set an asset tag in the virtual machine BIOS, which can then be read using the standard tools of the guest operating system. You can set an asset tag using the Parallels Desktop graphical user interface or the **prlctl** command line utility that comes with Parallels Desktop.

To set an asset tag using the Parallels Desktop GUI:

1. On the Parallels Desktop menu bar, select **Actions** > **Configure** to open the virtual machine configuration dialog.
2. Select **Business**.
3. Use the **Asset tag** field to specify the desired tag.

To set an asset tag using the **prlctl** command line utility, use the following syntax:

```
prlctl set ID|name --asset-id tag
```

where `ID|name` is the virtual machine ID or name, and `tag` is the asset tag to set.

To obtain the asset tag in Windows, use the WMIC.exe command:

```
WMIC SystemEnclosure get SMBIOSAssetTag
```

For the complete syntax of the WMIC utility please see the Microsoft documentation.

Once set, the asset tag never changes. Even if you perform such virtual machine operations as cloning, template manipulation, registering, or any other, the asset tag always stays the same. If you do want to change an existing asset tag for any reason, you can do it manually using of the methods described above.

Using Custom Graphics and Links in the Control Center

Parallels Desktop Control Center is a part of the Parallels Desktop graphical user interface. It's a window from which a Mac user launches virtual machines. By default, the Control Center displays the list of the available virtual machines, as in the following example:

You can customize the Control Center by specifying a URL to your own HTML document, which will be embedded at the top of the Control Center window. The HTML page can contain text, graphics, and links such as your company logo, custom text, a link to a support page, etc. The HTML document format doesn't have any specific requirements.

The URL must be specified during the preparation stage of the Mass Deployment process (p. 8). Specifically, you need to specify the URL and the HTML page size using the following variables in the mass deployment configuration file (`deploy.cfg`):

- `control_center_banner_url`
- `control_center_banner_height`
- `control_center_banner_min_width`

For the description of how to specify the values, please see **Configure Deployment Configuration Options** (p. 23). The variables are described in the **Virtual Machines** section.

The following is an example of Parallels Desktop Control Center displaying a custom banner at the top.

You can download a sample HTML document defining the banner using the following URL:

https://download.parallels.com/desktop/tools/header13.zip

Hiding Developer Tools in the Parallels Desktop GUI

Parallels Desktop Business and Pro editions include developer tools which are aimed at software developers using Parallels Desktop as their development platform. The tools are accessed by clicking the **Develop** menu on the virtual machine menu bar and then choosing one of the available options (e.g. **Start SSH Session**, **Start Debugging Session**, and others). If users in your organization are not using these tools, you can hide the **Develop** menu altogether. The reason you would want to do this, some of these features (if used accidentally) may start a debugging session or engage some other development-specific task that may temporarily disrupt a normal Parallels Desktop operation.

This option is a part of a virtual machine configuration and can be set using the Parallels graphical user interface as follows:

1. Open the virtual machine configuration dialog (click the gear icon or choose **Actions** > **Configure**).
2. In the dialog, click **Options** (at the top) and then click **More Options** in the left pane.

3 In the right pane, select or clear the **Show developer tools** option. This will show or hide the **Develop** menu on the virtual machine menu bar (you don't have to restart a virtual machine if it's running).

To modify this setting from the command line, execute the following command in Terminal:

```
prlctl set ID|Name --show-dev-tools on|off
```

where *ID/Name* is the GUID or name of a target virtual machine.

When mass deploying Parallels Desktop on Mac computers in your organization, you can configure the deployment package to apply this settings to all included virtual machines automatically. For details, see **Configure deployment options** (p. 23).

Encrypting a Virtual Machine from the Command Line

A Parallels virtual machine can be encrypted from the Parallels Desktop graphical user interface. This is done from the **Security** tab of the virtual machine configuration dialog.

You can also use the `prlctl` command line utility (included with Parallels Desktop) to perform a full set of encryption operations on a virtual machine.

The following command line options are available:

- Encrypt a virtual machine

  `prlctl encrypt <ID | NAME>`
- Decrypt a virtual machine

  `prlctl decrypt <ID | NAME>`
- Change the encryption password

  `prlctl change-passwd <ID | NAME>`

The `<ID | NAME>` parameter can be either the virtual machine ID or the virtual machine name. When encrypting a virtual machine, you'll be asked to enter a password phrase, which will be used to encrypt the machine. When decrypting a virtual machine, you will be asked to enter the current password. When changing the password, you'll be asked to enter the old password and then the new password.

The encryption password will also be required to perform any other command line operation on an encrypted virtual machine, including starting, stopping, restarting, pausing, suspending, cloning, deleting a virtual machine, etc. For example, to start an encrypted virtual machine, you'll use the following command:

```
$ prlctl start my_virtual_machine
```

After executing the command above, you'll be asked to enter the password:

```
Virtual machine "my_virtual_machine" is encrypted - password required to continue
operation
Please enter password:
```

After typing in the correct password, you'll see the following output:

```
Starting the VM...
The VM has been successfully started.
```

If you need to execute a command remotely without having to enter the password on every Mac, you can send the password via standard input (stdin) as shown in the following example:

```
$ echo mypass | prlctl start my_virtual_machine
Virtual machine 'my_virtual_machine' is encrypted - password required to continue
operation
Please enter password:
Starting the VM...
The VM has been successfully started.
```

If you need to provide two passwords (as with the `change-passwd` command that changes the password), you can save the passwords to a text file and then use the following syntax:

```
$ cat /tmp/pass | prlctl change-passwd my_virtual_machine
Virtual machine 'my_virtual_machine' is encrypted - password required to continue
operation
Please enter password:
Please enter new password:
The password has been successfully changed.
```

The `/tmp/pass` file in the example above should contain the old password on the first line and the new password on the second line:

```
$ cat /tmp/pass
mypass
newpass
```

Single Application Mode

Single Application Mode is a special Parallels Desktop deployment option that allows you to completely hide Parallels Desktop and Windows on a Mac and make a Windows application appear like it's a native macOS app. This mode is designed for system administrators who want Mac users in their organization to run one or more Windows applications without being aware of Parallels Desktop or Windows running in a virtual machine.

To make Parallels Desktop run in Single Application Mode, you need to deploy it on Mac computers via the mass deployment package. This includes preparing the deployment package in a special way and then either deploying it on Mac computers using Mac management tools or running it manually on a Mac.

For more information about how to use the mass deployment package and how to deploy Parallels Desktop in Single Application Mode, please see the following sections of this guide:

- **Mass deployment using Mac management tools** (p. 8)
- **Single application mode** (p. 31)

Setting an Expiration Date for a Virtual Machine

You can set an expiration date for a virtual machine. This can be a useful option if you are preparing a virtual machine for a contractor (or a third party user) and want to make sure that it works only for the duration of the contract.

To set an expiration date for a virtual machine:

1. Open Parallels Desktop and select the desired virtual machine.
2. On the Parallels Desktop menu bar, select **Actions** > **Configure** to open the virtual machine configuration dialog.
3. Select the **Security** tab.
4. An expiration date can only be set on an encrypted virtual machine. If your machine is not yet encrypted, click **Encryption: Turn On**, specify an encryption password, and click **OK**. Make sure to record the password or you will not be able to start the virtual machine. Wait until the encryption process finishes.
5. To set an expiration date for the virtual machine, click **Expiration Date: Set Date**, specify a password and click **OK**. Make sure to record the password to be able to change the expiration settings later. You should keep this password secret to prevent the prospective user of the virtual machine from changing the expiration date.
6. On the next screen, specify the following options:
   - **Do not allow this VM start after:** specifies the virtual machine expiration date.
   - **Contact info:** specifies the system administrator email, phone number, or other contact information. This information will be included in the message that will be displayed to the user when the virtual machine is about to expire. You can include each piece of information on a separate line.
   - **Time Server:** specifies the time server URL. The virtual machine expiration time will be checked against this server. The default time server is https://parallels.com.
   - **Date Check Frequency:** specifies how often the date and time should be verified against the time server. You can specify it in minutes, hours, or days.
   - **If unable to check date, use VM for:** specifies for how long the virtual machine should be kept working if the time server cannot be reached. For the duration of this period, the virtual machine will continue to check the date. If it succeeds before this period is over, the counter is reset and the virtual machine will continue to work normally.
7. Click **OK** when done entering the expiration info.
8. To modify the current expiration date or password, click **Expiration Date: Change Date** or **Expiration Date: Change Password** and enter the new values.

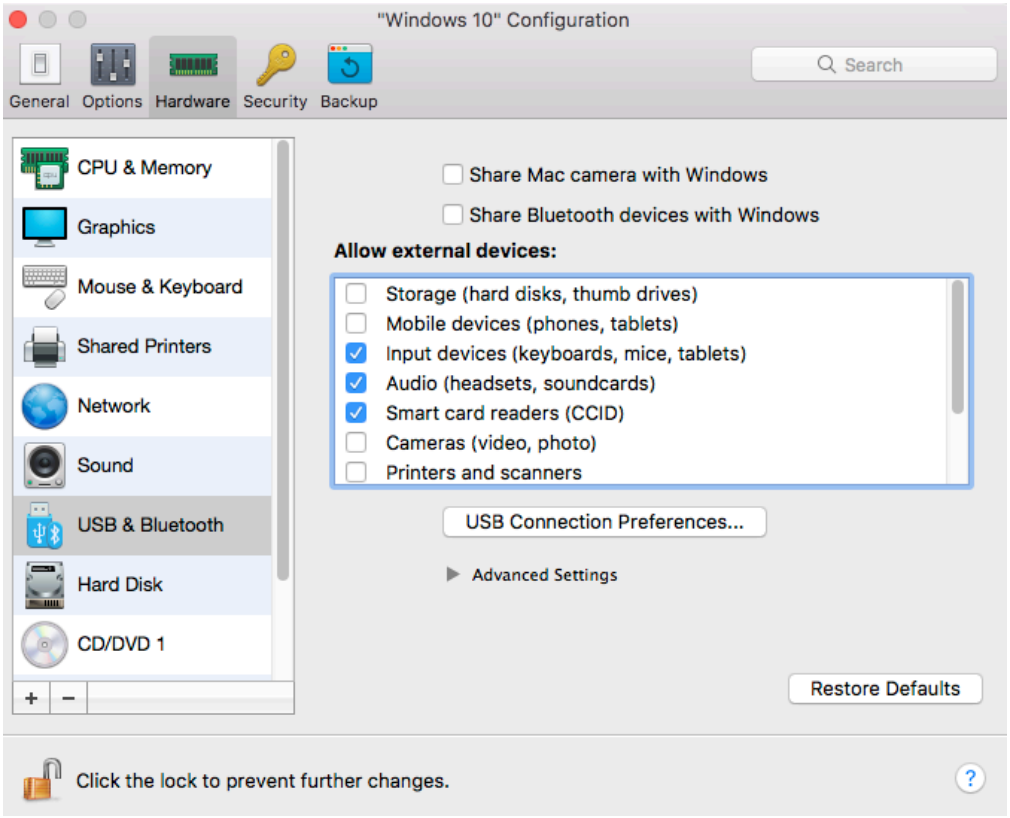
Note: When giving the virtual machine to a contractor, make sure to provide them the encryption password, which is needed to start the virtual machine. Please note that this is NOT the expiration date password you've set in step 5 above. This is the password you set when you encrypted the virtual machine.

When the expiration date approaches, the virtual machine user will be notified as follows: a message will begin to be displayed seven days before the expiration date. The message will be shown to the user every 24 hours and additionally on every virtual machine startup. Once the date is reached, the virtual machine will be locked, so the user will not be able to start or resume it anymore.

Enforcing USB Device Policies

When configuring USB device settings for a virtual machine, you can enforce what types of USB devices are allowed to be connected. For example, if storage devices (in general) are not allowed, the Mac user will not be able to connect an external hard disk or thumb drive to the virtual machine. This functionality is available in Parallels Desktop Business edition only and is absent in other editions.

To enforce USB device policies, open the virtual machine configuration window and select **Hardware** > **USB & Bluetooth**.



In the **Allow external devices** list:

- Clear the types of devices that you don't want Mac users to connect to the virtual machine.
- Select the types of devices that should be allowed.

Resetting the Guest OS Password

If a virtual machine user forgets the password of their guest OS account (e.g. a Windows user password), it can be reset outside the virtual machine using the command line interface.

To use this functionality the following conditions must be met:

- Parallels Tools must be installed in the guest OS.
- The virtual machine must be running. If it's stopped, start it and wait until you see the guest OS login prompt.
- Depending on your requirements, the following option can be selected or cleared in the virtual machine configuration dialog: **Security** > **Require Password to: [ ] Change guest OS password via CLI**. If this option is selected, you will be asked to provide the macOS administrator password to change the guest OS password from the command line. If the option is cleared, the administrator password will not be required. By default, the option is cleared.

To reset the password, open Terminal and enter the following command:

```
prlctl set vm_name --userpasswd username:new_password
```

where:

- `vm_name` is the virtual machine name. To obtain the list of virtual machines installed on this Mac, type `prlctl list`.
- `username` is the guest OS user name.
- `new_password` is the new password.

Example:

```
prlctl set My_Win8_VM --userpasswd JohnDoe:A12345
```

If the **Require Password to: Change guest OS password via CLI** option is selected in the virtual machine configuration dialog (see above), the command will display the following text and prompt:

```
Only host administrator can change user password in the guest OS.
Confirm your administrator credentials.
Username:
```

Enter the name of the macOS user with administrative privileges and press the Enter key. Type the user password and press Enter again.

Once the new password is set, you can use it to log in to the guest OS.

Setting Up a Local Update Server

With Parallels Desktop Business Edition you can set up a local update server on your network from which Mac users can get Parallels Desktop updates. Updates are released periodically to improve the performance and reliability of Parallels Desktop. To reduce Internet traffic when downloading updates, you can set up a local update server, download the available updates to it, and then set up individual Macs on your network to take the updates from it instead of the Internet. Read on to learn about setting a local update server.

Installing a web server

To set up a Parallels Desktop update server, you'll need a local Web server. Install a Web server on a computer connected to your network (or use an existing one).

Creating the Parallels update XML file

Create a file named `parallels_updates.xml` on the Web server where it can be accessed via HTTP. The file is an XML document that should contain specifications for a particular Parallels Desktop update available on your local updated server.

To create your own document, use the sample XML document below and the XML document specification that follows it as a reference.

Sample parallels_updates.xml file

```
<?xml version="1.0" encoding="UTF-8"?>
<ParallelsUpdates schemaVersion="1.0">
  <Product>
    <ProductName>Parallels Desktop</ProductName>
    <UpdateEnabled>1</UpdateEnabled>
    <Version>
      <Major>12</Major>
      <Minor>0</Minor>
      <SubMinor>12494</SubMinor>
      <SubSubMinor>262214</SubSubMinor>
      <StringRepresentation>Sumer</StringRepresentation>
      <Update uuid="desktop.13.0.13291.237436.en_US.parallels.mac">
        <UpdateType>0</UpdateType>
        <UpdateName>Build 13291 is available!</UpdateName>
        <UpdateDescription>Update description goes here</UpdateDescription>
        <FilePath>URL to the update file goes here</FilePath>
        <FileSize>219515</FileSize>
        <Status>0</Status>
        <DateTime>2017-06-17 01:23:00</DateTime>
        <Chargeable>0</Chargeable>
        <LocaleName>en_US</LocaleName>
        <DistributorName>parallels</DistributorName>
        <OsType>mac</OsType>
        <Ancestry>
          <Ancestor>desktop.13.0.12927.482436.en_US.parallels.mac</Ancestor>
          <Ancestor>desktop.13.0.12473.274921.en_US.parallels.mac</Ancestor>
          <Ancestor>desktop.13.0.12262.823647.en_US.parallels.mac</Ancestor>
        </Ancestry>
      </Update>
    </Version>
  </Product>
</ParallelsUpdates>
```

XML document specification

Name	Type	Description
ParallelsUpdates		Root element.
Product		Container for Parallels Desktop information.
ProductName	string	Use "Parallels Desktop".
UpdateEnabled	int	Specifies whether the automatic updates are enabled. To enable updates, specify 1.
Version		Container for Parallels Desktop version information.
Major	int	Major version number (e.g. 13)
Minor	int	Minor version number. Specify 0.
SubMinor		Build number. This element may be empty.
SubSubMinor		Revision number. This element may be empty.
StringRepresentation		Product codename. This element may be empty.
Update		Container for the information about the Parallels Desktop update.

		Attributes: `uuid` — String. A globally unique ID identifying the product. The `uuid` attribute is very important and must contain the correct information for the update to work. The attribute value consists of the following parameters (substrings) separated by periods (see the provided XML example): `desktop` — specify "desktop" `major` — major version number (e.g. 13) `minor` — minor version number (0) `build` — build number `revision` — revision number `locale` — locale ("en_US", "de_DE", etc) `vendor` — vendor ("parallels") `platform` — platform ("mac")
`UpdateType`	int	Update type. Specify 0.
`UpdateName`	string	The user-defined update name.
`UpdateDescription`	string	The update description.
`FilePath`	string	A URL to the update file on your local update server. The actual update files can be obtained from Parallels.
`FileSize`	int	The update file size, in megabytes.
`Status`	int	Specify 0.
`DateTime`	string	Date and time when the updated was published. Use the following format: yyyy-mm-dd hh:mm:ss
`Chargeable`	int	Specify 0.
`LocaleName`	string	Locale name ("en_US", "it_IT", etc).
`DistributorName`		The update distributor name. Specify "parallels".
`OsType`		Operating system type. Specify "mac".
`Ancestry`		Container for the list of updates that directly preceded this update.

`Ancestor`	string	An individual Parallels Desktop update information. This element may appear more than once in the same document, one for each update. The value is combined using the following parameters (substrings) separated by periods (see the provided XML example): `desktop` — specify "desktop". `major` — Parallels Desktop major version number. `minor` — minor version number. `build` — build number. `revision` — revision number. `locale` — locale (e.g. "en_US") `vendor` — vendor ("parallels"). `platform` -— platform ("mac").

Configuring individual Macs

The next step is to configure individual Macs to take their updates from the local update server. This can be done automatically during the mass deployment of Parallels Desktop by modifying the appropriate deployment configuration option. Please see **Configure Deployment Configuration Options** (p. 23) for the complete info (see the description of the **Software Updates** section of the configuration file).

If you have an existing Parallels Desktop installation that was not configured for automatic updates during deployment, then read on to learn how to do it manually.

Note: The information provided here applies only to Parallels Desktop installations that were NOT configured to use automatic updates during the mass deployment process.

To configure Parallels Desktop automatic updates, you need to modify the Parallels Desktop property list file on a Mac as follows:

1. Find the `com.parallels.Parallels Desktop.plist` file located in the `Library/Preferences` subfolder in the user's home folder. This is the Parallels Desktop property list file that contains the user-specific information.
2. Open the file using the Property List Editor application (included with Xcode).
3. Set the update policy by modifying the `Application preferences.VolumeLicenseUpdatePolicy` property. If the property doesn't exist, add it to the file specifying its data type as String. Set the property value using one of the following options (see also the **Notes** subsection below):
   - "Parallels" — when this value is set, the updates will be downloaded from the Parallels update server via the Internet. The value is case-sensitive.

- Complete URL of the `parallels_updates.xml` file residing on your local update server. For example, "https://10.0.0.1/pdfm/v8/en_us/parallels/parallels_updates.xml". When the URL is specified, the updates will be obtained from the local update server.
- "None" — automatic updates are disabled. The value is case-sensitive.

4 Specify how often Parallels Desktop should check for updates. This is done by modifying the `Application preferences.Check for updates` property. If the property doesn't exist, add it to the file specifying its data type as Number. Specify the property value using one of the following options:

- 0 — Never
- 1 — Once a day
- 2 — Once a week
- 3 — Once a month

5 Set the automatic download option. Find the `Application preferences.Download updates automatically` property. If it doesn't exist, add it to the file specifying its data type as Boolean. Set the property value using one of the following options:

- True — Download updates automatically. Specify this value when using a local update server.
- False — Notify the user about the updates but don't download them automatically. This option is useful only when updates are downloaded from the Parallels update server and the user has full control over the update functionality.

6 Save the file and close the Property List Editor application.

Note: If Parallels Desktop is running while you are modifying the `plist` file, it will have to be restarted for the changes to take effect.

Notes

On initial Parallels Desktop activation using a Business Edition key, the Parallels Desktop update properties will be absent from the `com.parallels.Parallels Desktop.plist` file. In such a case, a Mac user will be able to configure Parallels Desktop automatic updates using the Parallels Desktop graphical user interface.

When the update-related properties are added to the `com.parallels.Parallels Desktop.plist` file, the automatic updates will be performed according to the specified values. In addition, the value of the `Application preferences.VolumeLicenseUpdatePolicy` property will affect the Parallels Desktop update-related elements in the Parallels Desktop graphical user interface as follows:

- If the property contains a URL of the local update server or "None", the Parallels Desktop update-related controls will be disabled (grayed out) in the Parallels Desktop graphical user interface. The displayed settings will have no effect on how the Parallels Desktop updates are carried out. Therefore, the user will not be able to configure automatic updates or check for updates manually.

- If the property doesn't exist, has no value, or contains "Parallels" as a value, the Parallels Desktop update controls will be enabled in the user interface giving the user the ability to configure automatic updates and check for updates manually.

Configuring Parallels Desktop update branch

By default, Parallels Desktop Business Edition downloads updates from a special location on the Parallels website dedicated to hosting Parallels Desktop Business Edition updates. Parallels Desktop Standard and Pro editions download their updates from a different location. As an administrator, you have an option to choose the location from which Parallels Desktop Business Edition downloads updates. The reason why you would want to do this is explained below.

When Parallels Desktop updates are released by Parallels, they become immediately available for Parallels Desktop Standard and Pro Editions. Updates for Parallels Desktop Business Edition are released at a slightly later date (from a few days to 1-2 weeks from the initial release). The delay is necessary for additional testing of business features of Parallels Desktop to ensure they meet the highest quality standards. During this period, we even give an updated version of Parallels Desktop to some of our business clients, who test and evaluate it in their real-world business environments.

We recommend that you use the default configuration and download Parallels Desktop Business Edition updates when they are finalized and available for download. However, if for any reason you don't want to wait, you can configure Parallels Desktop Business Edition to download updates from the Parallels Desktop Pro location. The updates are the same regardless of where you download them from. The only difference is, the updates downloaded from the Parallels Desktop Pro location will not be fully tested in a business environment.

When you mass-deploy Parallels Desktop, you can set the desired Software Update options in the deployment configuration file. Mass Deployment of Parallels Desktop is described later in this guide. For more information, please read the entire Mass Deployment chapter (p. 8) and specifically the **Configure Deployment Configuration Options** section (p. 23). Look for the **Software Updates** section in the parameter table.

If you need to modify Parallels Desktop software update options on a specific Mac without using the Mass Deployment procedure, you can do this as described below.

To configure Parallels Desktop to download updates from the Parallels Desktop Pro location, execute the following command on a Mac:

```
defaults write com.parallels.Parallels\ Desktop.plist Application\
preferences.VolumeLicenseUpdatePolicy
https://update.parallels.com/desktop/v16/parallels/parallels_sbscr_updates.xml
```

The command above writes the specified URL (the parameter in the second part of the command) into the Parallels Desktop plist file. Please note that the "v16" part of the URL indicates the current Parallels Desktop version number. If you are using a later version, substitute this part with the correct number.

To switch back to the default Parallels Desktop Business download location, execute the following command:

```
defaults write com.parallels.Parallels\ Desktop.plist Application\
preferences.VolumeLicenseUpdatePolicy Parallels
```

Notes

Options described above only work in Parallels Desktop Business Edition. Other editions can only download software updates from their default locations.

When you configure Parallels Desktop to download updates from a custom URL (i.e. the Parallels Desktop Pro download location), the **Check for Updates** option in the Parallels Desktop GUI becomes disabled. This means that a Parallels Desktop Business Edition user will not be able to check for and install updates manually.

Customizing the Support Center option

When users run Parallels Desktop Business Edition, they can get support at any time by clicking the **Help** > **Support Center** menu. By default, this will open one of the following:

- If you are a large organization with your own Help Desk, the menu will open a message box saying that the user should contact the system administrator for assistance.
- If you are a small organization without a Help Desk or if you are using a trial version of Parallels Desktop, the menu will open the Parallels Desktop support web page.

You can change the default behavior described above and make the **Help** > **Support Center** menu open a custom URL, such as your corporate Help Desk page or any other web page that you desire.

The customization can be done during mass deployment of Parallels Desktop by modifying the corresponding deployment configuration parameter. Please see **Configure deployment configuration options** (p. 23) for the complete info (see the description of the **Help and Support** section of the configuration file).

You can also make the customization manually on an individual Mac as follows:

1. Log in to the Mac.
2. In the Finder, navigate to the `/Users/<User_Name>/Library/Preferences` directory and locate the `com.parallels.Parallels Desktop.plist` file.
3. Open the file using the Property List Editor application, which is included with Xcode.
4. Find the `SupportRequestUrl` property in the file. If the property doesn't exist, add it to the file specifying its data type as String.
5. To specify the action that should be performed by the **Help** > **Support Center** menu, set the value of the `SupportRequestUrl` property:

- To display the default text message, clear the property value.
- To open a URL, specify the full URL to the desired Web page or a resource.

Note: The `SupportRequestUrl` property value has no effect when Parallels Desktop is activated using the Parallels Desktop Standard Edition license key.

Participating in the Customer Experience Program

The Parallels Customer Experience Program is a feedback solution that allows Parallels Desktop to automatically collect usage statistics and system information that will help Parallels to develop new features and updates for future releases.

When you install Parallels Desktop on a Mac computer, it will ask you if you want to participate in the Customer Experience Program. If later you decide to change the participation parameter, you can do this by executing the following command on a Mac:

```
prlsrvctl set --cep on|off
```

The same action can also be done through the Parallels graphical user interface by navigating to **Parallels Desktop** > **Preferences** > **Advanced** and then selecting (or clearing) the **Feedback** option.

If you are using the Parallels mass deployment package to deploy Parallels Desktop, you can set the participation parameter in the deployment configuration file (p. 23).

CHAPTER 4

Parallels Desktop Command Line Interface

This chapter documents the command-line interface that can be used to manage Parallels Desktop and virtual machines. The interface supports the majority of Parallels Desktop management tasks that can be performed using the Parallels Desktop graphical user interface.

The command-line interface includes the following utilities:

- **prlsrvctl —** the utility is used to manage Parallels Desktop. The tasks include getting general information about Parallels Desktop, modifying Parallels Desktop preferences, getting a list of users, obtaining statistics, installing a license, and others.
- **prlctl —** the utility is used to manage virtual machines. The tasks include creating and configuring virtual machines, snapshot management, cloning operations, installing Parallels Tools, obtaining statistics, generating problem reports, and many others.

The command-line utilities are installed on a Mac as part of Parallels Desktop installation. You can run the utilities in Terminal.

Note that to get help about a particular command, you can type `man <name_of_the_command>` in Terminal. This will list the command attributes and additional help.

In This Chapter

Parallels Desktop Management 77
Virtual Machine Management 86

Parallels Desktop Management

The **prlsrvctl** command-line utility is used to perform management tasks on Parallels Desktop. The tasks include getting the Parallels Desktop information, modifying Parallels Desktop preferences, installing a license, obtaining statistics and problem reports, and others.

The general syntax is as follows:

prlsrvctl *command* [*options*] [**-v**, **--verbose** *number*]

The parameters are:

- *command*: one of the available commands.
- *options*: command options.

- **-v**, **--verbose** *number*: Show verbose output. The greater the number, the more verbose output will be produced.

The subsequent sections describe the available **prlsrvctl** commands grouped by functionality.

Display Parallels Desktop info

prsrvctl info

Displays the detailed information about the Parallels Desktop configuration.

Optional parameters

--license

If included, only the Parallels Desktop license information is displayed.

--activation-id

If included, only the license activation ID is displayed.

--deactivation-id

If included, only the license deactivation ID is displayed.

--json

Produces the machine-readable output in JSON format.

License Management

This section describes Parallels Desktop license management tasks.

Sign in to Parallels Account

web-portal signin *<email>*

Sign in to your Parallels account. The *<email>* parameter specifies your registered email address. When prompted, enter your Parallels account password. You can also specify a file containing the password using an optional parameter (see below).

This command must be used before you activate Parallels Desktop Pro or Standard editions from the command line using the **install-license** command.

Optional parameters

-p,**--read-passwd** *<path>*

Specifies a text file with your Parallels account password.

Related commands

web-portal signout

Signs you out of your Parallels account.

info --web-portal

Displays the information about whether you are signed in to your Parallels account and lists details if you are.

Install a Parallels Desktop license

prlsrvctl install-license <**-k, --key** <*key*>>

Installs a Parallels Desktop license. The **-k**, **--key** <*key*> parameter specifies a Parallels Desktop license key to install.

Optional parameters

-n, **--name** <*name*>

The license owner name.

-c, **--company** <*name*>

The license company name.

--deferred

Stores the license for deferred installation. The license will be activated the next time Parallels Desktop is started. If a license has already been activated, it should be deactivated first before using this option. See the **prlsrvctl deactivate-license** command.

--activate-online-immediately

Activates the license over the Internet immediately.

Notes

When activating Parallels Desktop Pro or Standard editions, you must be signed in to your Parallels account before executing this command. See **Sign in to Parallels Account** (p. 78).

Install / Remove a deferred license

prlsrvctl deferred-license <**--install** | **--remove**>

Installs or removes a license stored for deferred installation.

Parameters

--install

Installs the license stored for deferred installation.

--remove

Removes the license stored for deferred installation.

Update the license

prlsrvctl update-license

Updates the current Parallels Desktop license.

Activate the license online

prlsrvctl activate-license-online

Activates the installed Parallels Desktop license over the Internet.

Notes

This command is no longer used in Parallels Desktop v14 and later.

Activate the license offline

prlsrvctl install-license-offline -f, **--file** *<path-to-license-file>*

Activates the installed Parallels Desktop license offline.

Deactivate the License

prlsrvctl deactivate-license [**--skip-network-errors**]

Deactivates the current Parallels Desktop license. The **--skip-network-errors** option skips network errors and removes the license locally.

Parallels Desktop Preferences

prlsrvctl set

The **prlsrvctl set** command is used to modify Parallels Desktop preferences. The available parameters and options are described below.

Parameters

--mem-limit <**auto** | **size**>

Sets the total memory allocated to Parallels Desktop and its virtual machines. The **auto** option optimizes the memory usage. The **size** option allows the user to set the memory size manually.

-s, --min-security-level <**low** | **normal** | **high**>

Specifies the minimum connection security level to connect to the server. Low - no transmitted data is encrypted. Normal - only the most important data is encrypted. High - all transmitted data is encrypted.

-c, **--cep** <**on** | **off**>

Enables or disables the participation in the Customer Experience Program.

--verbose-log <**on** | **off**>

Enables or disables verbose logging.

--log-rotation <**on** | **off**>

Enables or disables automatic rotation of the Parallels Dispatcher Service and virtual machine log files.

--allow-attach-screenshots <**on** | **off**>

Enables or disables attaching virtual machine and host screenshots to a problem report.

--require-pwd <**create-vm** | **add-vm** | **remove-vm** | **clone-vm**>:<**on** | **off**>

Require to enter an administrator password to perform a corresponding action (create a VM, add a VM, etc.).

--require-custom-pwd <**create-vm** | **add-vm** | **remove-vm** | **clone-vm** | **edit-preferences**>:<**on** | **off**>

Require to enter a custom password to perform a corresponding action.

--custom-pwd [**--custom-pwd-mode** <**on** | **off** | **change**>] [**--replace-commands**]

Set, reset, or change the custom password for operations that require it. See the explanation of parameters below.

--custom-pwd-mode <**on** | **off** | **change**> — set, reset, or change the custom password for operations which require it.

--replace-commands — specify this option to reset commands that are protected with the admin password. This means that when you enable a custom password, commands that require the admin password will now require a custom password. Commands that previously required a custom password will be discarded. The same logic is used when you switch back to the admin password (set **--custom-pwd-mode** to **off**). When you disable a custom password, commands that require it will now require the admin password. Commands that previously required the admin password will be discarded. This option is ignored if **--custom-pwd-mode** is set to **change**.

--lock-edit-settings <**on** | **off** [**--host-admin** <*name*>]>

Locks or unlocks Parallels Desktop preferences for editing. The **--host-admin** parameter specifies the host administrator name if an administrator password is required to unlock Parallels Desktop preferences for editing.

--external-dev-auto-connect <**host** | **guest** | **ask**>

When a new external device is attached to the Mac, connect it to the host, guest, or ask the user what to do.

--default-encryption-plugin <*plugin-id*>

Sets the default encryption plugin.

--reset-default-encryption-plugin

Resets the default encryption plugin.

--hide-license-request-params <**on** | **off**>

Hides the host name and user name in requests to Parallels Licensing Server.

User List and Virtual Machine Location

prlsrvctl user list [**-o**, **--output** *field* [,*field*...]] [**-j**, **--json**]

Lists users currently existing in Parallels Desktop. The **-o**, **--output** *field* [,*field*...] option is used to display the specified field(s) only.

prlsrvctl user set <**--def-vm-home** <*path*>>

Modifies the default virtual machines location to the specified path.

Virtual Network Info

prlsrvctl net info <*vnetwork_id*>

Displays a detailed information about the specified virtual network.

prlsrvctl net list [**-j**, **--json**]

Lists existing virtual networks.

Virtual Network Settings

prlsrvctl net set <*vnetwork_id*> [*options*]

The **prlsrvctl net set** command is used to modify virtual network settings. The available parameters and options are described below.

Parameters

-i, **--ifname** <*if*>

Sets the name of the network interface in Parallels Desktop to which the virtual network will be bound.

-m, **--mac** <*mac*>

Sets the MAC address of the network interface in Parallels Desktop to which the virtual network will be bound. The network interface with the specified MAC address must exist in Parallels Desktop.

-t, **--type** <**bridged** | **host-only** | **shared**>

Sets the virtual network type.

-d, **--description** <*description*>

Sets the virtual network description.

-n, **--name** <*new_name*>

Sets a new name for the virtual network.

--ip <*ip*[*/mask*]>

Sets an IPv4 address and subnet mask for the Parallels virtual network adapter.

--dhcp-server <**on** | **off**>

Enables or disables the Parallels virtual DHCPv4 server.

--dhcp-ip <*ip*>

Sets an IPv4 address for the Parallels virtual DHCPv4 server.

--ip-scope-start <*ip*>

Sets the start IPv4 address for the pool of IPv4 addresses.

--ip-scope-end <*ip*>

Sets the end IPv4 address for the pool of IPv4 addresses.

--ip6 <*ip*[/*mask*]>

Sets an IPv6 address and subnet mask for the Parallels virtual network adapter.

--dhcp6-server <**on** | **off**>

Enables or disables the Parallels virtual DHCPv6 server.

--dhcp-ip6 <*ip*>

Sets an IPv6 address for the Parallels virtual DHCPv6 server.

--ip6-scope-start <*ip*>

Sets the start IPv6 address for the pool of IPv6 addresses.

--ip6-scope-end <*ip*>

Sets the end IPv6 address for the pool of IPv6 addresses.

--host-assign-ip6 <**on** | **off**>

Sets whether the host interface for this network will have IPv6 address.

--connect-host-to-net <**on** | **off**>

Connects the host to the current Parallels virtual network adapter.

--nat-<**tcp** | **udp**>**-add** <*rule_name*, *src_port*,<*dest_ip* | *dest_vm*>, *dest_port*>

Adds a new port forwarding rule. The options are:

rule_name: *a* rule name.

src_port: port number for incoming connections.

dest_ip: an IP address to which incoming connections will be forwarded.

dest_vm: the name or UUID of the virtual machine to which incoming connections will be forwarded.

dest_port: port number to which incoming connections will be forwarded.

--nat-<**tcp** | **udp**>**-del** <*rule_name*>

Deletes the specified port forwarding rule.

USB Devices

prlsrvctl usb list [**-j**, **--json**] [**-c**, **--compat**] [**-a**, **--all**]

Lists USB devices installed on the server together with the information on their assignments for the current user. In the compat mode all known USB devices are listed, showing the device name, device ID and autoconnect options. In the new mode (without the [-c, --compat] option), some additional information about the device is shown, such as whether the device is connected to a VM at the current moment. By default, only currently plugged to host devices are shown. To see all devices, call with the [-a, --all] option.

prlsrvctl usb set <*usb_dev_id*> <*vm_uuid* | *vm_name*> | [**--autoconnect** <*ask* | *host*>] | [**--vm** <*vm_uuid* | *vm_name*>]

Either assigns a USB device with ID <usb_dev_id> to the specified virtual machine or configures the action for this device (suggest to connect to active VM or silently connect to host). When the device is configured to connect to a virtual machine, this USB device will be connected to the specified virtual machine when you start the virtual machine or attach the device to the host computer.

prlsrvctl usb del <*usb_dev_id*>

Removes a specified USB device assignment.

Send Problem Report

prlsrvctl problem-report <**-d**, **--dump** | **-s**, **--send** [**--proxy** <*user*[:*password*]@*proxyhost*[:*port*]> | **--no-proxy**]>

Generates a problem report. If the **-s**, **--send** option is specified, sends the report to Parallels. The **--proxy** parameter specifies Internet proxy settings if you are using one to connect to the Internet.

Optional parameters

--stand-alone

Assembles a report without connecting to the Parallels Desktop service.

--name <*user_name*>

Appends the user name to the report.

--email <*user_email*>

Appends the user email address to the report.

--description <*problem_description*>

Appends a free-form description to the report.

Shut Down Parallels Desktop

prlsrvctl shutdown [**-f**, **--force**]

Shuts down Parallels Desktop. The command correctly stops all services. The optional **-f**, **--force** parameter forcibly shuts down Parallels Desktop and hard-stops services if necessary.

Parallels Desktop Plugin Commands

prlsrvctl plugin list [-j, --json]

Lists installed Parallels plugins.

prlsrvctl plugin refresh

Refreshes installed Parallels plugins.

Error Handling

The **prlsrvctl** utility returns 0 on success or an error code on failure.

Virtual Machine Management

The **prlctl** utility is used to perform management tasks on virtual machines. The utility supports a full range of tasks from creating and administering virtual machines to getting statistics and generating problem reports.

The general syntax is as follows:

prlctl *command* <vm_*ID* | *vm_name*> [*options*] [**-v**, --**verbose** *number*]

The parameters are:

- *command*: one of the available commands.
- *vm_ID* | *vm_name*: ID or name of the target virtual machine.
- *options*: command options.
- **-v**, **--verbose** *number*: Show verbose output. The greater the number, the more verbose output will be produced.

The subsequent sections describe the available **prlctl** commands grouped by functionality.

General Virtual Machine Management

This section describes various **prlctl** commands allowing you to perform such tasks as listing virtual machines, creating new virtual machines, cloning, performing power operations (start, stop, pause, etc.), encrypting, and many others.

List virtual machines

Syntax 1

prlctl list

Lists existing virtual machines. By default (when no parameters are included), only running VMs are displayed.

Optional parameters

-a, **--all**

Lists all existing virtual machines regardless of their state (running, stopped, suspended, etc.).

-f, **--full**

Shows the real IP address(es) for running virtual machines.

-o, **--output** *field* [, *field*...]

Displays only the specified field(s).

-s, **--sort** <*field* | *-field*>

Sorts by *field* (arguments are the same as those for **-o**). Add "-" (minus sign) before the field name to reverse the sort order.

-L

Lists fields which can be used for both the output (**-o**, **--output**) and sort order (**-s**, **--sort**) options.

-t, **--template**

Include templates in the output.

-j, **--json**

Produces output in JSON format.

Syntax 2

list -i, **--info**

Displays the VM configuration information. By default, the information for all existing VMs is shown.

Optional parameters

-f, **--full**

Shows the real IP address(es) for running virtual machines.

-j, **--json**

Produces output in JSON format.

vm_id | *vm_name*

Returns the information about a VM specified by ID or name.

Create a virtual machine

Syntax 1

prlctl create <*vm_name*> **--ostemplate** <*name*>

Creates a virtual machine from the specified virtual machine template. The **--ostemplate** <*name*> parameter specifies the source template name. To obtain the list of available templates, use the **prlctl list -t** command.

Optional parameters

--dst

A path to the directory where the virtual machine files will be stored. If omitted, the default location will be used.

--changesid

Change the Windows security identifier (SID) of a Windows-based virtual machine template. It requires Parallels Tools to be installed in the template. If omitted, the original SID will be used.

Syntax 2

prlctl create <*vm_name*> [**-o**,**--ostype** <*name* | **list**>]

Creates a virtual machine and optimizes it for the OS type specified in the **--ostype** option. If the **--ostype** parameter is omitted, the virtual machine is optimized for Windows 10 by default. If you want to optimize the virtual machine for a different OS type, use the **list** option to get the list of available OS types: **prlctl create vm_name -o list**, then select a desired OS type name and use it as a value of the **-o** parameter.

Syntax 3

prlctl create <*vm_name*> **-d**,**--distribution** <*name* | **list**>

Creates a virtual machine and optimizes it for the OS distribution specified in the **--distribution** option. Use the **list** option to get the list of available distributions: **prlctl create vm_name -d list**, then select a desired distribution name and use it as a value of the **-o** parameter.

Common optional parameters

--no-hdd

Create a virtual machine without hard disk drives.

--lion-recovery

Create a virtual machine from the Lion OS host recovery partition.

Delete a virtual machine

prlctl delete <*vm_id* | *vm_name*>

Deletes a virtual machine. The command removes a virtual machine from Parallels Desktop and permanently deletes all its files from the host computer. Once completed, this operation cannot be reversed.

Register or unregister a virtual machine

Register a virtual machine

prlctl register <*path*>

Registers a virtual machine in Parallels Desktop. The <*path*> parameter specifies a path to the virtual machine file.

Optional parameters

--uuid <*UUID*>

If included, the specified UUID will be assigned to the virtual machine. A UUID can be generated using console utilities like `uuidgen(1)`in macOS. If this option is not included, the original UUID will be used.

--regenerate-src-uuid

If included, the virtual machine source ID will be automatically generated (the SMBIOS product ID will be changed as well).

--force

If included, all validation checks will be skipped.

Unregister a virtual machine

prlctl unregister <*vm_id* | *vm_name*>

Unregisters the specified virtual machine.

Remarks

Use the **register** command when you have a virtual machine on the host that doesn't show up in the list of the virtual machines registered with Parallels Desktop. This can be a machine that was previously removed from Parallels Desktop or a machine that was manually copied from another location.

Note that after a VM is registered, all VM restrictions on the filesystem are removed. If you would like to protect the VM from editing, you should restrict registering or removing the VM in Parallels Desktop preferences.

The **unregister** command removes a virtual machine from Parallels Desktop but does not delete it from the host. You can re-register such a machine with Parallels Desktop later using the **register** command.

Clone a virtual machine

prlctl clone <*vm_id* | *vm_name*> **--name** <*new_name*>

Clones (makes an exact copy of) a specified virtual machine. The **-name** <*new_name*> parameter specifies a name to give to the new virtual machine.

Optional parameters

--template

Make the new virtual machine a template.

--dst

Set the path to the directory where the virtual machine files will be stored. If omitted, the default location will be used.

--changesid

Change the current Windows security identifier (SID) of a Windows-based virtual machine template. It requires Parallels Tools to be installed in the virtual machine template.

--regenerate-src-uuid

Regenerate the virtual machine source ID (the SMBIOS product ID will also be changed).

--linked

Create a linked clone.

-i, **--id** <*snapid*>

Create a linked clone based on a snapshot with given *snapid*.

--detach-external-hdd <**yes** | **no**>

Specifies what to do with hard disks located outside the source virtual machine file. If you specify **yes**, outside hard disks will be removed from the destination VM. If you specify **no**, outside hard disks will remain in the new VM. Please note that in either case, the outside hard disks will NOT be copied to the destination.

Convert a virtual machine

prlctl convert <*path*>

This command is used to convert a third-party virtual machine to a Parallels virtual machine. The <*path*> parameter specifies a path to the original virtual machine.

Optional parameters

--dst <*path*>

A path where the converted virtual machine files will be stored. If omitted, the default virtual machine location will be used.

--force

If included, the virtual machine conversion will continue even if the guest OS cannot be identified.

Notes

The following third-party virtual machines and disks are supported:

- Microsoft Hyper-V
- Microsoft Virtual PC
- Virtual Box
- VMware

Move virtual machine files

prlctl move <*vm_id* | *vm_name*> **--dst** <*path*>

Moves the files of a specified virtual machine to a location specified in the **-dst** parameter on the same computer. The command supports moving only stopped and suspended virtual machines.

Install Parallels Tools

prlctl installtools <*vm_id* | *vm_name*>

Installs Parallels Tools in the specified virtual machine. To use this command, the target virtual machine must be running.

Log In to a virtual machine

prlctl enter <*vm_id* | *vm_name*>

Logs in to the virtual machine. The command creates a command prompt channel to a virtual machine and allows you to execute commands in it. Parallels Tools must be installed in the virtual machine.

Optional parameters

--current-user or **--user** <*user_name*>

Include the **--current-user** option to log in as the user currently logged in inside the guest OS; or use the **--user** parameter and specify a user name.

--password <password>

The user password.

Execute a command in a virtual machine

prlctl exec <*vm_id* | *vm_name*> <*command*>

Executes a command inside a virtual machine. Parallels Tools must be installed in the virtual machine. Commands in Linux guests are invoked with bash -c.

Optional parameters

--current-user or **--user** <*user_name*>

Include the **--current-user** option to log in as the user currently logged in inside the guest OS; or use the **--user** parameter and specify a user name.

--password <*password*>

The user password

-r, **--resolve-paths**

Enable converting host paths to guest.

Get virtual machine status

prlctl status <*vm_id* | *vm_name*>

Displays the status of the specified virtual machine.

Power operations

prlctl start <*vm_id* | *vm_name*>

Starts the specified virtual machine. The **start** command can be used to start a stopped virtual machine or to resume a paused virtual machine.

prlctl resume <*vm_id* | *vm_name*>

Resumes the specified virtual machine.

prlctl pause <*vm_id* | *vm_name*>

Pauses the specified virtual machine.

prlctl suspend <*vm_id* | *vm_name*>

Suspends the specified virtual machine.

prlctl restart <*vm_id* | *vm_name*>

Restarts the specified virtual machine. The **restart** command first gracefully shuts down a virtual machine and then starts it again.

prlctl reset <*vm_id* | *vm_name*>

Resets the specified virtual machine. The **reset** command first performs a 'hard' virtual machine shutdown and then starts it again.

prlctl reset-uptime <*vm_id* | *vm_name*>

Resets the specified virtual machine uptime counter (the counter start date/time will also will be reset with this action).

prlctl stop <*vm_id* | *vm_name*> [**--kill**]

Stops the specified virtual machine. You can use the **--kill** option to forcibly stop the VM. The **stop** command can perform a 'hard' or a graceful virtual machine shutdown. If the `--kill` parameter is included, the 'hard' shutdown will be performed. If the parameter is omitted, the outcome of the graceful shutdown attempt will depend on the following:

- If the Parallels Tools package is installed in a virtual machine, the graceful shutdown will be performed using its facilities.

- If the Parallels Tools package is not installed, the command will try to perform a graceful shutdown using ACPI. Depending on the ACPI support availability in the guest operating system, this may work or not.

Change Windows SID

prlctl change-sid <*vm_id* | *vm_name*>

Changes the Windows security identifier (SID) in the specified virtual machine.

Capture a screen area

prlctl capture <*vm_id* | *vm_name*> **--file** <*name*>

Captures a screen area of a virtual machine to a file in PNG format. The **--file** <*name*> parameter specifies the target file name and path.

Encrypt or decrypt a virtual machine

prlctl encrypt <*vm_id* | *vm_name*> [**--dry-run**]

Encrypts the specified virtual machine. You can use the the **--dry-run** option to check preconditions for successful encryption. The **encrypt** command will encrypt the specified virtual machine and all its data. A user will be prompted to enter an encryption password after the command is executed. The password will be required to decrypt the virtual machine later. The encryption password can be modified for an encrypted virtual machine using the **change-passwd** command (see below).

prlctl decrypt <*vm_id* | *vm_name*> [**--dry-run**]

Decrypts the specified encrypted virtual machine. You can use the the **--dry-run** option to check preconditions for successful decryption. The **decrypt** command will decrypt the specified virtual machine. A user will have to enter a password that was specified when the virtual machine was encrypted.

prlctl change-passwd <*vm_id* | *vm_name*>

Changes the encryption password for the specified virtual machine. A user will be asked to enter the current and the new password.

Archive or unarchive a virtual machine

prlctl archive <*vm_id* | *vm_name*>

Archives the specified virtual machine bundle.

prlctl unarchive <*vm_id* | *vm_name*>

Unarchives the specified virtual machine bundle.

Set password protection

prlctl protection-set <*vm_id* | *vm_name*>

Protects the specified encrypted virtual machine expiration date settings with a password.

prlctl protection-remove <*vm_id* | *vm_name*>

Disables password protection of the specified encrypted virtual machine expiration date settings.

Virtual Machine Configuration Tasks

This section describes **prlctl set** command options that you can use to configure a virtual machine.

The general syntax is as follows:

prlctl set <*vm_id* | *vm_name*> [*options*]

The **prlctl set** command is used to modify the configuration of a virtual machine and manage virtual machine devices and shared folders. The subsequent subsections describe parameters and options that can be used with this command to perform a variety of virtual machine configuration management tasks.

CPU and memory parameters

--cpus <*num*>

Sets the number of CPUs to be available to the virtual machine.

--memsize <*num*>

Sets the amount of memory for the virtual machine (in megabytes).

Boot order parameters

--device-bootorder <"*name1 name2 ...*">

Specifies the order of boot devices for the virtual machine. Supported devices are HDD, CD/DVD, FDD, Network. A device name can be obtained using the **prlctl list -i** command.

--efi-boot <**on** | **off**>

Sets EFI boot options. Specify **on** to boot using the EFI firmware. Specify **off** to boot using the BIOS firmware (default).

--select-boot-device <**on** | **off**>

Enables or disables selecting a boot device at the virtual machine startup.

--external-boot-device *<name>*

Sets an external device from which to boot the virtual machine.

Video parameters

--videosize *<num>*

Sets the amount of memory for the virtual machine graphic card (in megabytes).

--3d-accelerate <**off** | **highest** | **dx9**>

Sets 3d acceleration video mode.

--vertical-sync <**on** | **off**>

Enables or disables vertical synchronization.

--high-resolution <**on** | **off**>

Enables or disables high resolution video mode for retina display.

Mouse and keyboard parameters

--smart-mouse-optimize <**auto** | **on** | **off**>

Sets smart mouse optimization mode.

--sticky-mouse <**on** | **off**>

Enables or disables the sticky mouse option.

--keyboard-optimize <**auto** | **accessibility** | **on** | **off**>

Sets a keyboard optimization mode.

Virtual printer parameters

--sync-host-printers <**on** | **off**>

Enables or disables using host printers in Windows guests (starting from Windows 2000).

--sync-default-printer <**on** | **off**>

Synchronizes host's default printer with Windows default printer.

USB and Bluetooth parameters

--auto-share-camera <**on** | **off**>

Enables or disables automatic Web camera sharing.

--auto-share-bluetooth <**on** | **off**>

Enables or disables automatic sharing of bluetooth devices.

--support-usb30 <**on** | **off**>

Enables or disables USB 3.0 support.

Startup and shutdown parameters

--autostart <**off** | **open-window** | **start-app** | **start-host** | **user-login**>

Sets the virtual machine autostart options:

off: The virtual machine is started manually.

open-window: The virtual machine starts when its window opens.

start-app: The virtual machine starts when Parallels Desktop starts.

start-host: The virtual machine is started automatically on the host boot.

user-login: The virtual machine is started automatically on user logon.

--autostart-delay <*n*>

Sets the delay of the virtual machine autostart on host boot to n seconds.

--autostop <stop | suspend | shutdown>

Specifies what should happen to the virtual machine on host shutdown.

--startup-view <**same** | **window** | **coherence** | **fullscreen** | **modality** | **headless**>

Sets the virtual machine startup view options:

same: Same as the last time

window: Normal window

coherence: Coherence

fullscreen: Full screen

modality: Modality

headless: Headless

--on-shutdown <**window** | **close** | **quit**>

Sets the virtual machine shutdown options:

window: The virtual machine window stays open after the virtual machine is shut down.

close: The virtual machine window closes after the virtual machine is shut down.

quit: Parallels Desktop quits after the virtual machine is shut down.

--on-window-close <**suspend** | **shutdown** | **stop** | **ask** | **keep-running**>

Sets the virtual machine window close options:

suspend: The virtual machine is suspended after its window is closed.

shutdown: The virtual machine is shut down after its window is closed.

stop: The virtual machine is forcibly stopped after its window is closed.

ask: Ask the user what to do: suspend, shut down, or stop the virtual machine.

keep-running: The virtual machine is kept running after its window is closed.

--pause-idle <**on** | **off**>

Enables or disables pausing of an idle virtual machine.

--undo-disks <**off** | **discard** | **ask**>

Sets the virtual machine undo disks options:

off: Undo disks mech is off.

discard: Discard all changes made in the virtual machine after it is stopped.

ask: Ask the user what to do: apply changes or discard them after the virtual machine is stopped.

Optimization parameters

--faster-vm <**on** | **off**>

Sets the performance mode: faster virtual machine or faster host.

--adaptive-hypervisor <**on** | **off**>

Enables or disables adaptive hypervisor.

--disable-winlogo <**on** | **off**>

Enables or disables Windows logo.

--auto-compress <**on** | **off**>

Enables or disables automatic compression of virtual disks.

--nested-virt <**on** | **off**>

Enables or disables nested virtualization.

--pmu-virt <**on** | **off**>

Enables or disables PMU virtualization.

--longer-battery-life <**on** | **off**>

Sets a power option: longer battery life or better performance.

--battery-status <**on** | **off**>

Shows or hide battery status.

--resource-quota <**low** | **medium** | **unlimited**>

Sets the virtual machine resource quota:

low: The host uses maximum possible resources.

medium: The host and the virtual machine evenly share resources.

unlimited: The virtual machine uses maximum possible resources.

Sharing parameters

--smart-mount <**on** | **off**>

Enables or disables shared volumes.

--shared-profile <**on** | **off**>

Enables or disables shared profile.

--shared-cloud <on | off>

Enables or disables shared cloud.

--sh-app-guest-to-host <**on** | **off**>

Enables or disables sharing guest applications with host.

--show-guest-app-folder-in-dock <**on** | **off**>

Enables or disables showing the folder with guest OS applications in the Dock.

--sh-app-host-to-guest <**on** | **off**>

Enables or disables sharing host applications with guest.

Coherence parameters

--winsystray-in-macmenu <**on** | **off**>

Shows Windows notification area in the Mac menu bar.

--auto-switch-fullscreen <**on** | **off**>

Allows applications to auto-switch to full screen.

--disable-aero <**on** | **off**>

Enables or disables Windows Aero.

--hide-min-windows <**on** | **off**>

Allows to hide minimized windows.

Security parameters

--require-pwd <**exit-fullscreen** | **change-vm-state** | **manage-snapshots** | **change-guest-pwd**>:<**on** | **off**>

Require an administrator password to perform a corresponding action.

--require-custom-pwd <**exit-fullscreen** | **change-vm-state** | **manage-snapshots** | **change-guest-pwd** | **change-vm-config**>:<**on** | **off**>

Require a custom password to perform an operation. The **change-guest-pwd** option allows you to change the guest OS password via command line. Other options are self-explanatory.

--custom-pwd [**--custom-pwd-mode** <**on** | **off** | **change**>] [**--replace-commands**]

Set, reset, or changes a custom password for operations that require it. The options are described below.

--custom-pwd-mode < **on** | **off** | **change**> — set, reset, or changes a custom password for operations that require it.

--replace-commands — specify this option to reset commands that are protected with the admin password. This means that when you enable a custom password, commands that require the admin password will now require a custom password. Commands that previously required a custom password will be discarded. The same logic is used when you switch back to the admin password (set **--custom-pwd-mode** to **off**). When you disable a custom password, commands that require it will now require the admin password. Commands that previously required the admin password will be discarded. This option is ignored if **--custom-pwd-mode** is set to **change**.

--lock-on-suspend <**on** | **off**>

Always locks the guest OS on suspend.

--isolate-vm <**on** | **off**>

Isolates the the virtual machine from the host.

--smart-guard <**on** | **off**>

Enables or disables smart guard mech.

--sg-notify-before-create <**on** | **off**>

Notifies the user before creating a snapshot.

--sg-interval <*seconds*>

Sets a time interval between taking snapshots.

--sg-max-snapshots <*num*>

Sets the maximum allowed number of snapshots.

--lock-edit-settings <**on** | **off** [**--host-admin** <*name*>]>

Locks or unlocks editing of the virtual machine configuration.

--host-admin <name>

Specifies the host administrator name if an administrator password is required to unlock editing of the virtual machine configuration.

--userpasswd <*user* : *passwd*> [--**host-admin** <*name*>]

Sets a password for the specified user in the virtual machine. If the user account does not exist, it is created. The **--host-admin** <*name*> parameter specifies the host administrator name if an administrator password is required to change the user password in the virtual machine. Parallels Tools must be installed in the virtual machine for the command to succeed.

--password-to-edit

This parameter is not used since Parallels Desktop 15. In previous versions, it was used to set a custom password to modify the virtual machine configuration. In Parallels Desktop 15 and newer, use the **--require-custom-pwd** and **--custom-pwd** commands (described in the beginning of this section).

Expiration date parameters

--expiration <<**on**|**off**>|**date:**<*yyyy-MM-ddThh:mm:ss*>|**time-check:**<*seconds*>|**offline-time:**<*seconds*>|**time-server:**<*url*>|**note:**<*text*>>

Expiration date parameters:

on|**off:** Enables or disables expiration date checking.

date: Sets a date and time when the virtual machine usage period expires (e.g. 2014-12-30T20:30:00).

time-check: Sets how often (in seconds) Parallels Desktop contacts the time server to check the expiration date and time.

offline-time: Sets the time period (in seconds) during which a user can work with the virtual machine if Parallels Desktop is unable to check the expiration date and time.

time-server: Specifies the URL of a trusted time server to check the expiration date and time.

note: Adds a note (e.g. system administrator contact info).

Device Management

The following options can be used with the **prlctl set** command to manage devices:

- **--device-add** — add a new device.
- **--device-set** — modify an existing device.
- **--device-del** — delete (remove) a device.
- **--device-connect** — connect a device.
- **--device-disconnect** — disconnect a device.

Only one option can be specified in a single command.

Common options

The options described here are common for all types of devices.

--device-connect <*device_name*>

Connects the specified device to a running virtual machine. The device can be of type fdd, cdrom, sound, or net. To obtain a device name, use the **prlctl list -i** command.

--device-disconnect <*device_name*>

Disconnects the specified device from a running virtual machine.

--device-set <*device_name*> <<**--enable** | **--disable**> | <**--connect** | **--disconnect**>>

Enables/disables or connects/disconnects the specified device to/from a virtual machine. Please note that the **--device-set** command is also used to modify a device configuration and has additional parameters, which are different for different types of devices. The parameters for each device type are described in subsequent sections of this guide.

--device-del <*device_name*> [**--detach-only** | **--destroy-image** | **--destroy-image-force**]

Removes the specified device from the virtual machine.

If **--detach-only** is specified and the device is a virtual hard disk drive, the disk image is preserved.

If **--destroy-image** is specified, the virtual HDD image is removed from the server.

If **--destroy-image-force** is specified, the virtual HDD image is removed from all snapshots and from the server.

The default action on deleting a virtual HDD is to detach the HDD image as if **--detach-only** was specified.

Adding and modifying a device

The general syntax for adding a device is as follows:

prlctl set <*vm_id* | *vm_name*> **--device-add** <**hdd** | **cdrom** | **net** | **fdd** | **serial** | **parallel** | **sound** | **usb**> [*device_options*]

To modify a device:

prlctl set <*vm_id* | *vm_name*> **--device-set** <*device_name*> [*device_options*]

The subsequent sections describe options and parameters for each device type.

Virtual hard disk

Add a hard disk

--device-add hdd [**--image** <*image_name*>] [**--type** <**expand** | **plain**>] [**--size** <*n*>] [**--split**]

[**--iface** <**ide** | **scsi** | **sata**>] [**--position** <*n*>]

[**--subtype** <**buslogic** | **lsi-spi** | **lsi-sas**>]

[**--online-compact** <**on** | **off**>]

Modify a hard disk

--device-set <*hdd_name*> [**--image** <*image_name*>] [**--type** <**expand** | **plain**>]

[**--size** <*n*>] [**--split**] [**--iface** <**ide** | **scsi** | **sata**>] [**--position** <*n*>]

[**--subtype** <**buslogic** | **lsi-spi** | **lsi-sas**>] [**--online-compact** <**on** | **off**>]

Parameters

hdd_name: The name of the virtual hard disk to modify (-**-device-set** command only). Virtual hard disks are named using the hddN format where N is the drive index number starting from 0 (e.g. hdd0, hdd1). To obtain the list of disk names, use the `prlctl list --info` command.

--image: specifies the name of the file to be used for emulating the VM virtual disk drive. If this option is omitted, a new file is created inside the directory storing all VM-related configuration files and assigned the name of `harddiskN.hdd`.

--type: specifies the type of the virtual disk from one of the following:

- **expand (default)**: virtual disks of this type are small initially and grow in size as you add data to it.
- **plain**: virtual disks of this type have a fixed size from the moment of their creation.

--size: hard disk size, in megabytes.

--split: splits the hard disk into 2 Gb pieces.

--iface: virtual hard disk interface type: **ide**, **scsi**, or **sata**.

--position: the SCSI / IDE / SATA device identifier to be used for the disk drive. Allowed ranges:

- 0-3 for IDE disk drives
- 0-6 for SCSI disk drives
- 0-5 for SATA disk drives

--subtype: virtual hard disk subtype: **buslogic**, **lsi-spi**, **lsi-sas**.

--online-compact: enables or disables virtual hard disk online compact mode.

Physical hard disk

Connect a physical hard disk

--device-add hdd --device <*real_name*> [**--iface** <**ide** | **scsi** | **sata**>] [**--passthr** <**yes** | **no**>]

[**--position** <*n*>] [**--subtype** <**buslogic** | **lsi-spi** | **lsi-sas**>]

Parameters

--device: the name of the host computer hard disk that will be connected to the virtual machine. To obtain the names of all hard disks installed on the host, use the **prlsrvctl info** command.

--iface: virtual hard disk interface type: **ide**, **scsi**, **sata**.

--passthr: enables the passthrough mode for the specified device.

--position: the SCSI / IDE / SATA device identifier to be used for the disk drive. Allowed ranges:

- 0-3 for IDE disk drives
- 0-6 for SCSI disk drives
- 0-5 for SATA disk drives

--subtype: virtual hard disk subtype: **buslogic**, **lsi-spi**, **lsi-sas**.

Virtual optical drive

Add an optical drive

--device-add cdrom [**--image** <*name*>] [**--iface** <**ide** | **scsi** | **sata**>] [**--position** <*n*>]

[**--subtype** <**buslogic** | **lsi-spi** | **lsi-sas**>]

Modify an optical drive

--device-set <*drive_name*> [**--image** <*name*>] [**--iface** <**ide** | **scsi** | **sata**>]

[**--position** <*n*>] [**--subtype** <**buslogic** | **lsi-spi** | **lsi-sas**>]

Parameters

drive_name: The name of the optical drive to modify (**--device-set** command only**)**. To obtain the list of the available drives, use the `prlctl list --info` command.

--image: connect the specified image file to the virtual machine. The following image file formats are supported: **iso**, **cue**, **ccd**, **dmg**.

--iface: virtual optical interface type: **ide**, **scsi**, **sata**.

--position: the SCSI / IDE / SATA device identifier to be used for the optical drive. Allowed ranges:

- 0-3 for IDE disk drives
- 0-6 for SCSI disk drives
- 0-5 for SATA disk drives

--subtype: virtual optical drive subtype: **buslogic**, **lsi-spi**, **lsi-sas**.

Physical optical drive

Connect a physical optical drive

--device-add cdrom --device <*name*> [**--iface** <**ide** | **scsi** | **sata**>] [**--passthr** <**yes** | **no**>]

[**--position** <*n*>] [**--subtype** <**buslogic** | **lsi-spi** | **lsi-sas**>]

Parameters

--device: the name of the host computer CD/DVD drive that will be connected to the virtual machine. To obtain the names of all CD/DVD drives installed on the host, use the **prlsrvctl info** command.

--iface: virtual CD/DVD drive interface type.

--passthr: enables the passthrough mode for the specified device.

--position: the SCSI / IDE / SATA device identifier to be used for the CD/DVD drive. Allowed ranges:

- 0-3 for IDE disk drives

- 0-6 for SCSI disk drives
- 0-5 for SATA disk drives

--subtype: virtual CD/DVD drive subtype.

Virtual floppy disk drive

Add an FDD

The command adds a virtual floppy disk drive based on a file image.

--device-add fdd --image <*image*> [**--recreate**]

Modify an FDD

--device-set <*fdd_name*> **--image** <*image*> [**--recreate**]

Parameters

fdd_name: The name of the FDD to modify. To obtain the list of the available drives, use the `prlctl list --info` command.

--image: specifies the image file.

--recreate: if included, recreates the image file if it exists.

Physical floppy disk drive

Connect a physical FDD

-device-add fdd [**--device** <*real_name*>]

Parameters

--device: specifies a physical floppy disk drive name.

Virtual network adapter

Add a network adapter

--device-add net --type <**shared** | **bridged** | **host-only**> [**--iface** <*name*>] [**--mac** <**addr** | **auto**>]

[**--ipadd** <*addr* [/*mask*]> | **--ipdel** <*addr*[/*mask*]> | **--dhcp** <**yes** | **no**> | **--dhcp6** <**yes** | **no**>]

[**--gw** <*gw*>] [**--gw6** <*gw*>] [**--nameserver** <*addr*>] [**--searchdomain** <*addr*>]

[**--configure** <**yes** | **no**>] [**--apply-iponly** <**yes** | **no**>]

[**--adapter-type** <**virtio** | **e1000** | **e1000e** | **rtl**>]

Modify a network adapter

--device-set <*adapter_name*> **--type** <**shared** | **bridged** | **host-only**> [**--iface** <*name*>]

[**--mac** <**addr** | **auto**>]

[**--ipadd** <*addr* [*/mask*]> | **--ipdel** <*addr*[*/mask*]> | **--dhcp** <**yes** | **no**> | **--dhcp6** <**yes** | **no**>]

[**--gw** <*gw*>] [**--gw6** <*gw*>] [**--nameserver** <*addr*>] [**--searchdomain** <*addr*>]

[**--configure** <**yes** | **no**>] [**--apply-iponly** <**yes** | **no**>]

[**--adapter-type** <**virtio** | **e1000** | **e1000e** | **rtl**>]

Parameters

adapter_name: the name of the virtual network adapter to modify (**--device-set** command only). To obtain the list of the available adapters, use the prlctl list --info command.

--type: the type of the network adapter to create in the virtual machine.

--iface: the host network interface to be assigned to the bridged or host-only virtual network adapter.

--mac: the MAC address to be assigned to the virtual network adapter. If omitted, the MAC address will be automatically generated.

--ipadd: the IP address to be assigned to the network adapter in the virtual machine.

--ipdel: the IP address to be removed from the network adapter in the virtual machine.

--dhcp: specifies whether the virtual network adapter should get its IP settings through a DHCP server.

--dhcp6: specifies whether the virtual network adapter should get its IPv6 settings through a DHCP server.

--gw: the default gateway to be used by the virtual machine.

--gw6: the default IPv6 gateway to be used by the virtual machine.

--nameserver: the default DNS server to be used by the virtual machine.

--searchdomain: the default search domain to be used by the virtual machine.

--configure: if set to **yes**, the settings above are applied to the virtual network adapter instead of its original settings. Configuring any of the settings automatically sets this option to **yes**.

--apply-iponly: if set to **yes**, the hostname, nameserver, and search domain settings from the virtual machine configuration file are ignored.

--adapter-type: specifies the network adapter emulation type.

Virtual serial port

Add a serial port

--device-add serial {**--device** <*name*>|**--output** <*file*>|**--socket** <*name*>

[**--socket-mode** <**server**|**client**>]}

Modify a serial port

--device-set <*port_name*> {**--device** <*name*>|**--output** <*file*>|**--socket** <*name*>

[**--socket-mode** <**server**|**client**>]}

Parameters

port_name: the name of the port to modify (**--device-set** command only). To obtain the list of the available ports, use the *prlctl list --info* command.

--device: the number of the host computer serial port that will be used by the virtual machine.

--output: the path to the file where the output of the virtual serial port will be sent.

--socket: the name of the host computer socket to which the serial port will be connected.

--socket-mode: the socket operation mode.

Virtual parallel port

Add a parallel port

--device-add parallel {**--device** <*name*> | **--output** <*file*>}

Modify a parallel port

--device-set <*port_name*> {**--device** <*name*> | **--output** <*file*>}

Parameters

port_name: the name of the port to modify (**--device-set** command only). To obtain the list of the available ports, use the *prlctl list --info* command.

--device: the parallels port number on the host computer that will be used by the virtual machine.

--output: the path to the file where the output of the virtual parallel port will be sent.

Virtual sound card

Add a sound card

--device-add sound --output <*name*> **--input** <*name*>

Modify a sound card

--device-set sound --output <*name*> **--input** <*name*>

Parameters:

--ouput: the name of a physical output device to which to connect the virtual sound card.

--input: the name of the physical input device to which to connect the virtual sound card.

Add USB support

The command adds USB support to a virtual machine and makes the USB & Bluetooth configuration options available.

--device-add usb

Shared folders

A shared folder is a host OS folder that can be accessed from a virtual machine.

--shf-host <**on** | **off**>

Enables or disables sharing the user-defined host OS folders with guest OS.

--shf-host-add <*name*> **--path** <*path*> [**--mode** <**ro** | **rw**>] [**--shf-description** <*desc*>]

[**--enable** | **--disable**]

Shares the host OS folder name with a virtual machine.

--shf-host-del <*name*>

Removes the specified folder from the list of shared folders.

--shf-host-set <*name*> **--path** <*path*> [**--mode** <**ro** | **rw**>] [**--shf-description** <*desc*>]

[**--enable** | **--disable**]

Modifies the settings of the host OS shared folder name.

--shf-host-defined <**off** | **alldisks** | **home**>

off: Disable sharing of folders defined by the host OS.

alldisks: Share all host OS disks with a virtual machine.

home: Share a host OS user's home directory with a virtual machine.

--shf-guest <**on** | **off**>

Enables or disables sharing of user-defined guest OS folders with the host OS.

--shf-guest-automount <**on** | **off**>

Enables or disables automatic mounting of shared guest OS folders on the desktop.

Advanced

--time-sync <**on** | **off**>

Enables or disables the virtual machine time synchronization.

--disable-timezone-sync <**on** | **off**>

Enables or disables timezone synchronization. Enable this option to sync only UTC time without timezone synchronization.

--sync-vm-hostname <**on** | **off**>

Enables or disables synchronization of the virtual machine name and hostname in guest OS. Supported only for Linux guests.

--sync-ssh-ids <**on** | **off**>

Enables or disables synchronization of macOS SSH public keys with those from the guest OS "authorized_keys" file.

This feature is similar to the ssh-copy-id(1) utility. When enabled, all macOS SSH public keys are added to the guest OS "authorized_keys" file. This allows users to log in to the guest OS via SSH without having to enter the password.

The following SSH keys are synced:

- When a user creates a new SSH key pair in macOS, the public key is also added to the guest OS.

- When a user removes a public key from macOS, this key is also removed from the guest OS.

The details of current implementation:

- Public key synchronization is currently available for Linux guests only.

- Public key synchronization works if the guest OS user has the same name in macOS or the user is the only regular user of the guest system.

The public key is synced in the following cases:

- After Parallels Tools are installed.

- After booting or rebooting the guest OS.

- After the virtual machine resumes.

- After the public key synchronization feature is enabled/disabled.

Additional information:

- If the feature is disabled, all macOS SSH public keys are removed from the guest OS.

- The "authorized_keys" file and public keys are searched only in the "~/.ssh" directory.

- SSH authorization certificates are not supported.

--show-dev-tools <**on** | **off**>

Enables or disables show developer tools in menu.

--swipe-from-edges <**on** | **off**>

Enables or disables edge swipe gestures.

--rename-ext-disks

Renames external virtual hard disks bundles using the virtual machine name.

Miscellaneous

--name <*name*>

Changes the virtual machine name.

--description <*desc*>

Sets the virtual machine description.

--distribution <*name* | *list*>

Sets the virtual machine OS version(s) family.

--asset-id <*id*>

Changes the virtual machine asset ID.

--template <**on** | **off**>

Converts the virtual machine to template and back.

--tools-autoupdate <**yes** | **no**>

Enables or disables the auto-update mode for Parallels Tools Agent.

--usedefanswers <**on** | **off**>

Enables or disables default mech answers to the questions from the virtual machine.

Snapshot Management

This section describes command used to manage virtual machine snapshots.

Take a virtual machine snapshot

prlctl snapshot <*vm_id* | *vm_name*>

Creates a virtual machine snapshot.

Optional parameters

-n, **--name** <*name*>

A snapshot name. If omitted, a default name will be used.

-d, **--description** <*desc*>

A snapshot description.

Delete a snapshot

prlctl snapshot-delete <*vm_id* | *vm_name*> **-i**, **--id** <*snapid*>

Deletes a virtual machine snapshot. The **-i**, **--id** <*snapid*> parameter specifies the ID of the snapshot to delete.

Optional parameters

-c, **--children**

If included, all child snapshots of the specified snapshot will be deleted.

List snapshots

prlctl snapshot-list <*vm_id* | *vm_name*> [{**-t**,**--tree**] | [**-i**,**--id** <*snapid*>}] [**-j**, **json**]

Lists the virtual machine snapshot tree. There are three modes of snapshot listing:

- If the **-t**, **--tree** option is specified, the tree is displayed using ASCII graphics.
- If the **-i**, **--id** <*snapid*> option is specified, the specified snapshot information is displayed.
- If no option is specified, the snapshot tree is displayed as a table with two columns: PARENT_SNAPSHOT_ID, SNAPSHOT_ID.

The optional **-j**, **--json** parameter produces an output in JSON format.

Revert to a snapshot

prlctl snapshot-switch <*vm_id* | *vm_name*> **-i**, **--id** <*snapid*> [**--skip-resume**]

Reverts the specified virtual machine to the specified snapshot. The **-i**, **--id** <*snapid*> parameter specifies the ID of a snapshot to revert to.

If the optional **--skip-resume** parameter is included, the virtual machine will not be started if it was running when the snapshot was taken.

Miscellaneous Commands

This section describes miscellaneous **prlctl** commands.

Generate a problem report

prlctl problem-report <*vm_id* | *vm_name*>

<**-d**, **--dump** | **-s**, **--send** [**--proxy** [*user* [**:***password*] *@proxyhost* [*:port*]]] [**--no-proxy**]>

Generates a problem report. If the **-s**, **--send** option is specified, the report is sent to Parallels. Otherwise, it is dumped to stdout.

Parameters

-d, **--dump**

If included, the report is displayed on the screen. You can pipe the output to a file and then send it to the Parallels technical support.

-s, **--send**

If this option is included (instead of **-d**, **--dump**), the report is sent to Parallels. You can specify additional optional parameters, which are described below.

--proxy *user*:*password*@proxyhost:port

If you use a proxy server to connect to the Internet, include the **--proxy** parameter and specify the proxy server information. The problem report will be sent to Parallels through this proxy server.

--no-proxy

Do not use a proxy server to send the problem report. This is the default behavior, so you can include or omit this parameter.

Optional parameters

--name *<user_name>*

Inserts the specified user name into the report.

--email *<user_e-mail>*

Inserts the specified e-mail address into the report.

--description *<problem_description>*

Inserts the specified free-form description into the report.

Use guest debugger

prlctl guest-debugger *<vm_id | vm_name>* [**--port** *<port>*]

Allows you to connect the debugger to a running virtual machine via the specified port. The debugger must be installed on the same computer where the virtual machine is running.

Create a VM dump

prlctl debug-dump *<vm_id | vm_name>*

[**--name** *<dump_file_name>*] [**--path** *<output_directory_path>*]

Creates a virtual machine dump in ELF format and saves it to a file. The resulting dump file can be opened with the Linux **crash** utility or (with some limitations) with the **GDB** debugger.

Optional parameters

--name *<dump_file_name>*

Allows you to specify a dump file name. By default, the file is named memory.elf.dmp. When you create a new dump file, it replaces the previous file (if it exists). Giving it a custom name to a file solves this issue.

--path *<output_directory_path>*]

By default, dump files are saved to the virtual machine directory. If you want to save them to another directory, specify the directory path using the **--path** parameter.

Notes

To create a dump, the virtual machine must be running or paused. Suspended virtual machines are not supported by this command. The command returns 0 (zero) on success and a non-zero value on failure.

Virtual machine disk optimization

prl_disk_tool is a standalone utility (included with Parallels Desktop) used to optimize Parallels virtual machine disks.

Syntax

prl_disk_tool compact [**--buildmap**] **-hdd** <*disk_name*> [**--force**] [**-comm** <*memory_name*>]

prl_disk_tool compact **i**,**info --hdd** <*disk_name*> [**comm** <*memory_name*>]

The **prl_disk_tool compact** command removes all empty blocks from expanding Parallels virtual disks and reduces their size on your real disk. Compacting is performed by scanning file systems for unused clusters and cleaning the corresponding disk blocks. The supported file systems are NTFS, FAT16/32, ext2/ext3. You can also try to compact disks with unsupported file system types using the --buildmap option.

Error Handling

The **prlctl** utility returns 0 on success or an error code on failure.

Index

A

Activate the license offline - 80
Activate the license online - 80
Add a virtual machine - 19
Add the Parallels Desktop installation image - 18
Add USB support - 109
Adding and modifying a device - 102
Adding Windows application stubs - 21
Advanced - 110
Applying a configuration profile to a license key - 51
Archive or unarchive a virtual machine - 94

B

Boot order parameters - 95

C

Capture a screen area - 94
Change Windows SID - 94
Clone a virtual machine - 90
Coherence parameters - 99
Common options - 102
Configure deployment options - 23
Configuring individual Macs - 72
Configuring Parallels Desktop update branch - 74
Convert a virtual machine - 91
CPU and memory parameters - 95
Create a configuration profile - 49, 55, 57
Create a virtual machine - 88
Create a VM dump - 114
Create a VM image - 53
Creating the Parallels update XML file - 69
Customizing the Support Center option - 75

D

Deactivate the License - 80
Delete a snapshot - 112
Delete a virtual machine - 89
Deploying macOS image using NetBoot - 46
Deploying Parallels Desktop and virtual machines to Macs - 34
Deploying Parallels Desktop for Mac Business Edition - 8
Deploying via an invitation email - 8
Deploying with Apple Remote Desktop - 42
Deploying with Jamf Pro - 34
Deploying with Parallels Mac Management for Microsoft SCCM - 40
Device Management - 101
Display Parallels Desktop info - 78
Download the deployment package - 17

E

Enabling major version upgrades - 56
Encrypt or decrypt a virtual machine - 94
Encrypting a Virtual Machine from the Command Line - 64
Enforcing USB Device Policies - 67
Error Handling - 86, 115
Execute a command in a virtual machine - 92
Expiration date parameters - 101

G

General configuration options - 21
General Virtual Machine Management - 87
Generate a problem report - 113
Get virtual machine status - 93

H

Hiding Developer Tools in the Parallels Desktop GUI - 63

I

Install / Remove a deferred license - 79
Install a Parallels Desktop license - 79
Install Parallels Tools - 92
Installing a web server - 69
Installing Parallels Tools - 21
Installing VM image on a Mac computer - 56

Introduction - 5

K

Kernel extensions in macOS - 12

L

License Management - 78
List snapshots - 112
List virtual machines - 87
Locking a Virtual Machine on Suspend - 61
Log In to a virtual machine - 92

M

Manually approving kernel extensions - 14
Mass deployment using Mac management tools - 11
Miscellaneous - 111
Miscellaneous Commands - 113
Mouse and keyboard parameters - 96
Move virtual machine files - 92

O

Optimization parameters - 98

P

Parallels Desktop Business Edition Features Overview - 5
Parallels Desktop Command Line Interface - 77
Parallels Desktop for Mac Business Edition Features - 48
Parallels Desktop Management - 77
Parallels Desktop Plugin Commands - 86
Parallels Desktop Preferences - 80
Participating in the Customer Experience Program - 76
Physical floppy disk drive - 106
Physical hard disk - 104
Physical optical drive - 105
Power operations - 93
Preparing the deployment package - 17
Prerequisites - 11
Provisioning a corporate VM image - 52

R

Register or unregister a virtual machine - 89
Resetting the Guest OS Password - 68
Restricting a Virtual Machine Configuration with a Custom Password - 59
Restricting User Actions in Parallels Desktop with a Custom Password - 58
Revert to a snapshot - 113

S

Security parameters - 100
Send Problem Report - 85
Set password protection - 95
Setting an Asset Tag in the Virtual Machine BIOS - 61
Setting an Expiration Date for a Virtual Machine - 66
Setting Up a Local Update Server - 69
Shared folders - 109
Sharing parameters - 99
Shut Down Parallels Desktop - 86
Sign in to Parallels Account - 78
Single application mode - 31
Single Application Mode - 65
Snapshot Management - 112
Specify a license key - 19
Startup and shutdown parameters - 97
Supported Mac management tools - 17

T

Take a virtual machine snapshot - 112
Testing the deployment package - 33

U

Update the license - 80
Upgrading Parallels Desktop - 58
USB and Bluetooth parameters - 96
USB Devices - 84
Use guest debugger - 114
User List and Virtual Machine Location - 82
Using a script to Change VM options - 46
Using Configuration Profiles - 48
Using Custom Graphics and Links in the Control Center - 62
Using MDM configuration - 13
Using spctl command - 14

V

Video parameters - 96
Virtual floppy disk drive - 106
Virtual hard disk - 103

Virtual Machine Configuration Tasks - 95
Virtual machine disk optimization - 115
Virtual Machine Management - 86
Virtual network adapter - 106
Virtual Network Info - 82
Virtual Network Settings - 83
Virtual optical drive - 104
Virtual parallel port - 108
Virtual printer parameters - 96
Virtual serial port - 108
Virtual sound card - 109