White Paper | Parallels Desktop for Mac

# Security White Paper

Parallels IP Holdings GmbH
Vordergasse 59
8200 Schaffhausen
Switzerland

Tel: + 41 52 632 0411
Fax: + 41 52 672 2010

www.parallels.com

## Contents

**Virtual Machine;**
**VM**

An emulated computing environment, in which Parallels Desktop for Mac executes an unmodified copy of an Intel CPU-compatible operating system, e.g., Windows

**Hypervisor;**
**Virtual Machine Monitor**

A Parallels Desktop for Mac component that creates and executes virtual machines

**Guest operating system;**
**Guest OS**

An operating system, designed for an Intel-compatible CPU, which runs inside a virtual machine under control of Parallels Hypervisor. Windows is one of the most common guest operating systems that runs in a Parallels Desktop VM

**Host operating system;**
**Host OS**

A native operating system that runs directly on Mac hardware and controls a Mac. On Mac computers, OS X is the native host operating system. Parallels Desktop for Mac allows the running of another copy of OS X inside a VM as a guest OS.

**Virtual Machine application;**
**VM app**

An executable component of Parallels Desktop for Mac, which loads virtual machines, manages their states, and emulates their virtual devices. Runs as a regular OS X process.

# Introduction

## Introduction to Parallels Desktop for Mac

Parallels Desktop® for Mac is the most tested, trusted, and talked-about solution for seamlessly running Windows® applications on a Mac® without rebooting. Users can drag and drop files between Windows and Mac applications and launch Windows applications from the Mac Dock.  Besides Windows, Parallels Desktop also enables running other operating systems designed for Intel® CPUs, like Linux® (various versions), Chrome OS™, Android™, etc.

Parallels Desktop provides a high level of security that reliably protects the Mac native operating system, macOS™, from any crashes and unsafe/harmful activities in the guest OS running inside a virtual machine.

## What Is This Document?

This document contains an overview of security features provided in Parallels Desktop. By that, we mean all the processes and mechanisms that Parallels Desktop uses to protect a VM and the host computer data from unintended or unauthorized access, use, disclosure, modification, or disruption.

This document does not provide detailed instructions on how to configure Parallels Desktop or use its features. Please read the Parallels Desktop User's Guide and Parallels Desktop Help articles for this purpose.

The information in this document is relevant to Parallels Desktop for Mac version 12. Unless otherwise noted, the information is applicable to all editions of the product: Standard, Pro, and Business. 3

# Parallels Desktop for Mac Architecture

## Overview of the Parallels Desktop Architecture

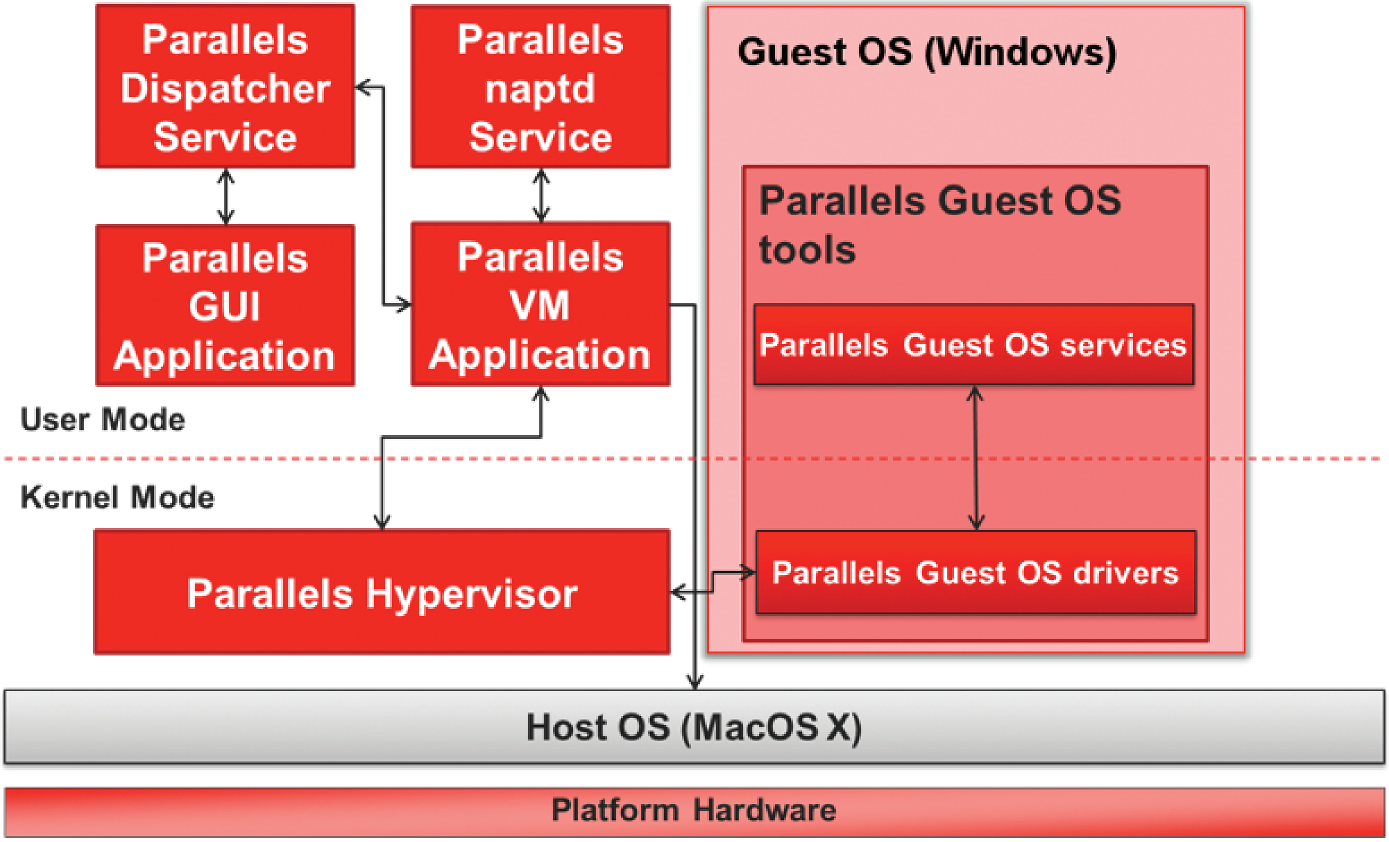
Parallels Desktop is hardware emulation virtualization software that uses hypervisor technology to run an unmodified copy of a guest OS, designed for an Intel-compatible CPU (e.g., Windows), side by side with the host OS (macOS).

A guest OS executes inside a VM by means of the Parallels Hypervisor that works by mapping the Mac device's hardware resources directly to the VM's resources. A hypervisor leverages features of a hardware virtualization assistant embedded in the Intel CPU, like VT-x, EPT, etc. Each VM operates identically to a stand-alone computer, with virtually all the resources of a physical computer; Parallels Desktop is able to virtualize a full set of standard PC hardware.

## Basic Components of Parallels Desktop

Parallels Desktop consists of the following major components (Figure 1):

- Hypervisor
- VM application
- Parallels Tools
- Dispatcher service
- Networking service (naptd)
- GUI application



## Hypervisor

A VM application is an executable component of Parallels Desktop that is responsible for:

- Loading a VM into the memory with the help of hypervisor;
- Managing a lifecycle of the executed VM and its states (e.g., start/stop/pause/restart/shutdown VM, create a snapshot, revert to snapshot, etc.);
- Emulating VM devices (CPU, timer, interrupt controller, network, disk, sound, etc.);
- Processing requests for execution of operations on virtual devices signaled by the VM through the hypervisor and mapping results back to the VM via the hypervisor.

The VM application is started by the macOS Dispatcher and runs as a regular macOS process; thus, it falls under all the regular process restrictions like virtual memory space isolation, file permissions etc., just like any other system process. The VM application is the only process that communicates with the guest OS that runs inside the VM, through the hypervisor API.

Any operation inside a VM that requires access to a real device is redirected by the hypervisor to the VM application in macOS and is then emulated by the VM app.

When executing device commands translated from the VM by the hypervisor, the VM application accesses only a limited set of macOS files that emulate virtual devices included in the VM configuration, e.g., virtual hard disk image files; it does not enable the guest OS to access arbitrary macOS files. The VM application emulates operations on virtual devices requested by the guest OS, in a user mode context.

Thus, it ensures that a guest OS will never be able to access the host kernel and/or any other system process/file or their memory in macOS directly through virtual devices..

## Dispatcher Service

The Dispatcher is a management component of Parallels Desktop. It runs as a system service (daemon) and communicates with the GUI app and the VM app through a Unix domain (IPC) socket. The Dispatcher also facilitates establishing a direct connection between the GUI and the VM app through the Unix domain socket.

Using a Unix domain socket makes communication between the Dispatcher and other components (GUI app, VM app) secure, as this kind of socket exists only inside a single computer; hence, the Dispatcher's communications cannot be eavesdropped on by an untrusted network, and remote computers cannot connect to it without some sort of forwarding mechanism. Thus, a malefactor cannot break the Dispatcher by connecting to it from the external network.

The Dispatcher is designed to process only a limited set of commands from the GUI application, commands that target management of a VM directory and individual VMs. External host processes cannot force it to execute arbitrary commands or access arbitrary data inside a VM. Similarly, a guest OS cannot access data outside a VM through the Dispatcher.

## GUI Application

The GUI application in Parallels Desktop provides a graphical user interface through which a user creates a VM, manages its state (start/stop/pause/resume/shutdown, etc.), and operates with the guest OS's virtual display that shows the user interface of the guest OS.

The Parallels Desktop GUI application runs as a regular macOS process, with the privileges of the current user. It establishes a connection to the Dispatcher using the Unix domain socket, through which it sends commands and receives events that enable managing a VM. It also establishes (with the help of the Dispatcher) a direct connection via a Unix domain socket to the VM application process, through which it transmits keyboard and mouse input to the VM. To get video display data from the guest OS, the GUI application accesses a shared memory buffer, which is created and updated by the VM application specifically for this purpose.

The GUI application draws image(s) to visualize the current contents of a guest OS video frame buffer in a single VM window (that represents a virtual desktop of the guest OS) or in multiple windows (that represent separate windows of the guest OS applications), depending on the current VM view mode (Window, Full Screen, Modality, or Coherence). The keyboard and mouse input is captured by the VM window(s) created by the GUI application and then is transmitted to the VM.

Regardless of the VM view mode, the guest OS and its applications remain running in an isolated VM environment controlled by the hypervisor (read the "Hypervisor" and "VM Application" sections for more details).

The GUI application never communicates directly with a guest OS and does not access any guest OS data directly, except the shared video memory buffer. It is not designed to accept and execute commands from a VM application and/or a guest OS. Thus, the guest OS and/or its programs cannot affect macOS and its data through the GUI application.

Using a Unix domain socket and a shared memory buffer makes communication between the GUI and the VM application secure, as these kind of interprocess communication objects exist only inside a single computer; hence, it cannot be eavesdropped on, read, or modified from the outside.

## Parallels Cloud

Parallels Desktop accesses the Parallels Cloud to:

- Request a trial license key;
- Activate and register a product;
- Periodically check the validity of a license's key and update a license key (update function is relevant only to the Business Edition);
- Check for Parallels Desktop software updates;
- Report problems to help Parallels improve the product (only if a user consents to send);
- Access product documentation, knowledge base articles, and other online materials;
- Retrieve a support code and request support;
- Download installation bundles of complimentary products like Parallels Access®, etc. that are offered at various times.

Some operations require a user to sign up/sign in to their personal Parallels user account. The Parallels Desktop GUI application prompts a user to sign up/sign in to their Parallels account when it is necessary to complete a scenario.

All connections to the Parallels Cloud are secured and encrypted. The data transmitted by Parallels Desktop to the Parallels Cloud is anonymous, i.e., it doesn't allow establishing the identity of the user.

The Parallels Cloud account database does not save copies of the original user account passwords. Instead, it uses a cryptographic hash function.

## Parallels Access

Parallels Access is the fastest, simplest, and most reliable way to remotely access any Windows and Mac applications and files that reside on a remote Windows PC or Mac, from an iPad®, iPhone®, or an Android device. The unique

"applification" technology introduced in Parallels Access lets the customer use all their desktop applications as if they were native apps for their mobile device. Parallels Access consists of the mobile application that can be installed from the Apple AppStore® (iOS version) or Google Play™ (Android version) for free, and the free Parallels Access agent component, which can be downloaded on the user's Mac or Windows PC from the Parallels Cloud. The Parallels Access mobile application communicates with the Parallels Access agent to allow a mobile device to operate with applications and files on a remote host.

When the Parallels Access agent is installed on a Mac, it provides remote access to both macOS and the guest Windows desktops, applications, and files in Parallels Desktop. All the network connections maintained by Parallels Access are protected and encrypted. Parallels Access transmits only a graphic representation of a remote desktop and application windows, but not the application's and document's data.

## Overview of Parallels Desktop Security Features

Parallels Desktop offers an extensive set of procedures and mechanisms that protect a VM and host computer data from unintended or unauthorized access, use, disclosure, modification, or disruption.

Parallels Desktop ensures safe virtualization through a variety of mechanisms:

- Provides safe VM use
- Guarantees protection of transferred data
- Ensures error-free operations

## Integration Between the VM and macOS

### Isolating a Guest OS and Protecting the Mac

Parallels Hypervisor technology enables separate execution of a guest OS from a host OS by creating an isolated virtual environment that emulates a real computer and its hardware. The hypervisor ensures that the guest OS runs isolated from macOS and never accesses arbitrary memory or executes unsafe CPU instructions outside the VM that could result in unauthorized access or disruption of host data and/or disordering the normal behavior of a Mac.

This kind of isolation prevents any uncontrolled communication between the guest OS and macOS. In fact, in order to help customers use their VMs in the most comfortable and efficient way, Parallels Desktop enables configurable and manageable integration between a guest OS and macOS. This integration is controlled by the Parallels Tools installed inside a guest OS, and the Parallels VM app that runs in macOS. The VM application is the only process in a host OS that communicates with the guest OS, through the hypervisor API.

## Sharing Data Between the VM and Host OS

### Disabling Integration of a Guest OS

If necessary, a user can completely disable integration to isolate macOS from a guest OS so they no longer share folders, profiles, and applications; connected external devices are no longer automatically accessible by the guest OS; and a user can no longer copy or move objects (files, folders) between the VM and macOS.

Isolating macOS from the guest OS may provide a higher level of security by not allowing compromised items from one OS to come into contact with the other.

### Protecting Virtual Machine Data

Parallels Desktop provides reliable tools that help protect VM data from accidental or intentional disruption that could be the result of user errors and/or malicious activities. Although using these tools cannot 100% guarantee that data won't be lost, they help users minimize the potential damage and enable reverting a VM back to previous stable, consistent states.

Those tools are snapshots, Rollback Mode, and SmartGuard.

### Snapshots

A snapshot is a saved state of a VM. It's useful to create snapshots, for example, before:

- The user is going to run a program that may damage the guest OS;
- The user is about to configure some guest OS software that requires a lot of settings;
- The user wants to mark milestones in the development process or workflow progression.

If something goes wrong, the user can revert the changes back to any of their previously created VM snapshots and continue working with the guest OS.

**Note:** Snapshots cannot be created for Windows VMs that use the Boot Camp® partition.

### Rollback Mode

If users don't want a guest OS to store the changes they make to it during the working session (for example, when testing Windows programs that may damage Windows), they can start the VM in Rollback Mode. When a VM that was run in Rollback Mode is shutting down, Parallels Desktop asks the user if they want to apply changes made during the session to the VM's hard disks. If the user chooses to discard changes, the VM reverts to its original state.

If a user wants a guest OS to always start in Rollback Mode, they can enable the corresponding option in the configuration of the VM. They can choose what the VM will do with changes made to it during a session, on shutdown: discard changes automatically or ask the user's permission to apply them.

## SmartGuard

The SmartGuard mechanism allows users to automate snapshot creation. It is available only when the Rollback Mode feature is disabled in the VM configuration and the VM is not running in Rollback Mode. When enabling SmartGuard in the VM's configuration, the user can choose the frequency of snapshot creation (an interval between snapshots) and the number of snapshots to keep. If they want to know when it is time to take the next snapshot and be able to reject the snapshot creation, the user can select the option "Notify me before snapshot creation."

This automation ensures saving a series of stable, consistent VM states even if the user forgets to do so.

SmartGuard's "Optimize for Time Machine" option allows a user to reduce the amount of VM data that the macOS Time Machine® backs up, to reduce the amount of time Time Machine spends backing up the virtual disk(s) and minimize the risk of data loss or corruption when restoring the VM hard disk(s) from the Time Machine backup.

## Defending the VM and Mac from Computer Viruses

The Coherence and Sharing features may make users think that Windows and macOS applications/processes and documents are mixed, and run in a single environment. This, in turn, may result in the conclusion that Windows applications running inside a VM can disrupt the behavior of macOS if, for example, the Windows application crashes. The user may also think that a malicious program that has accidentally infected Windows could affect/harm data in macOS.

In fact, guest OS applications run in an isolated environment created by the VM and can't cause direct damage to processes and/or data outside the VM in the event of failures or as a result of malevolent activity. Furthermore, programs inside the VM can't access arbitrary files, memory segments, and/or processes in a host OS (read the "Hypervisor" and "VM Application" sections for a detailed explanation).

Windows viruses rarely can infect the macOS applications, as the majority of the viruses designed for Windows can't run in macOS due to an incompatible executable format. Moreover, the virus inside a VM would not be able to execute its malicious code directly in macOS or force another program to do so. But if the guest OS is infected, a virus can access data in files stored in folders shared between the VM and Mac. At the same time, data stored in macOS outside shared folders cannot be accessed.

If you suspect that Windows is contaminated, you can disable the integration between the VM and Mac to eliminate a potential risk (read the "Disabling integration of a Guest OS" section for details).

## Protecting Transferred Data

## Safety of Local Data Moving Between Components

Using Unix domain sockets and a shared memory buffer makes communication between the Parallels Desktop components (GUI application, VM application, Dispatcher) secure, as these kind of interprocess communication objects exist only inside a single computer. Hence, an untrusted network cannot eavesdrop on it, and remote computers cannot connect to them without some sort of forwarding mechanism.

The Parallels VM application is the only macOS process that communicates with the guest OS inside a VM, through the hypervisor API.

## Safe Migration of Windows PC to VM

Users can migrate data from a remote Windows PC to the Parallels Desktop VM on Mac over a network, using the Parallels Transporter Agent installed on the remote PC. The transferred data is secured by using the OpenSSL library, with strong data encryption.

An external hard drive may also be used for data migration.

The user will need the passcode displayed in the Parallels VM Wizard on a Mac and will have to enter it in the Parallels Transporter Agent on their Windows PC to start the import.

## Protecting a VM Through Encryption

Parallels Desktop provides VM encryption. When a VM is encrypted, the data in all its protected files (virtual disk(s) and virtual memory files) are stored encrypted on a physical disk. The user must enter an encryption password to open and start the encrypted VM.

Encryption/decryption of the virtual disk data is performed “on the fly.” Thus, the VM’s virtual disk(s) are always stored encrypted.

The Parallels Desktop VM encryption module can also use an external third-party dynamic encryption library.

Parallels Desktop® for Mac Business Edition also allows the setting of a VM expiration date; i.e., date and time after which the VM won’t start.

## Protecting Parallels Desktop Preferences

To protect the Parallels Desktop environment from unauthorized changes, you can lock the Parallels Desktop preferences with a system administrator’s password. After Preferences are locked, Parallels Desktop asks a user to enter the administrator password to allow changes.

- Users can also set restrictions on the following operations for non-administrator users:
- Create a new virtual machine: Users will have to provide their administrator password to create a new VM;
- Add an existing virtual machine: Users will have to provide their administrator password to add an existing VM to the VM list;
- Remove a virtual machine: Users will have to provide their administrator password to remove their VM from the list;
- Clone or convert a virtual machine or template: Users will have to provide their administrator password to clone a VM, create a template, convert a template into a VM, and deploy a template into a new VM.

## Protecting the VM Configuration

To protect the configuration of a specific VM from unauthorized changes, users can lock its configuration with a system administrator’s password. After the VM configuration is locked, Parallels Desktop asks users to enter the administrator password to allow changes.

In Parallels Desktop Business Edition, a system administrator can assign an arbitrary password to lock the configuration of each individual VM.

Parallels Desktop Business Edition also allows administrators to restrict the types of USB devices that can be connected to the VM.

## Restricting VM Operations

The administrator may set restrictions on the following operations for non-administrator users on a particular VM:

- Exit full screen mode: Users will have to provide their administrator password to exit the full screen when the VM is running in full screen mode;
- Change virtual machine state: A password is required to start, stop, suspend, pause, or shutdown the VM;
- Manage snapshots: A password is required to create or delete a snapshot or revert to a snapshot;
- Change guest OS password via command line interface (CLI): A password is required to change the guest Windows account password via a command line interface.

## Protecting a Guest Windows Account

By default, when a user quits Parallels Desktop, the guest Windows account is suspended, and the next time they open Parallels Desktop, Windows resumes exactly where they left off. If more than one person uses Windows on a Mac, then a user can access the account in Windows suspended by the previous user. To avoid this, the user can disable automatically suspending the VM on Parallels Desktop quit.

Once automatically suspending Windows is disabled, each time a user quits Parallels Desktop, they can specify how they want Windows to shut down. And if they close the Parallels Desktop window, Windows shuts down according to how it’s set in the shutdown settings of the VM configuration.