

macOS Unlocker V4 for VMware Workstation

IMPORTANT

False positive from antivirus software

Some Windows antivirus programs mistakenly flag the unlocker as having a virus. This is a false positive and it is mainly seen with Windows Defender. The Go Language website documents the problem.

<https://go.dev/doc/faq#virus>

If you have any doubts you can compile the source code yourself.

1. Use a release from the Releases section of this GitHub repository.
<https://github.com/DrDonk/unlocker/releases>
2. Always uninstall the previous version of the Unlocker before using a new version or running an update on the VMware software. Failure to do this could render VMware unusable.
3. You use this software at your own risk and there are no guarantees this will work in future versions of VMware Workstation.

1. Introduction

Unlocker 4 is designed for VMware Workstation and Player 16 and has been tested against:

- Workstation Pro 16 on Windows and Linux
- Workstation Player 16 on Windows and Linux

The Unlocker enables certain flags and data tables that are required to see the macOS type when setting the guest OS type, and modify the implementation of the virtual SMC controller device. These capabilities are normally exposed in Fusion and ESXi when running on Apple hardware.

The patch code carries out the following modifications dependent on the product being patched:

- Fix vmware-vmx and derivatives to allow macOS to boot
- Fix vmwarebase.dll or libvmwarebase.so to allow Apple guests to be selected during VM creation
- Provide a copy of the macOS VMware Tools for the guest

It is important to understand that the Unlocker cannot add any new capabilities to VMware Workstation and Player but enables support for macOS that is disabled in the VMware products that do not run on Apple Hardware.

The Unlocker cannot:

- add support for new versions of macOS
- add paravirtualized Apple GPU support
- add AMD CPU support

or any other features that are not already in the VMware compiled code.

A patched macOS kernel must be used to run on AMD systems.

2. Prerequisites

The code is written in Go and has no pre-requisites and should run directly from the download.

3. Windows

On Windows you can double-click the executables to launch them from the "windows" folder. You may be prompted to run with Administrator rights which are required to patch VMware.

- unlock.exe - apply patches to VMware
- relock.exe - remove patches from VMware
- check.exe - check the patch status of VMware

4. Linux

On Linux you will need to be either root or use sudo to run the scripts in the terminal.

Navigate to the folder where the unlocker has been unpacked, and run the required command from the "linux" folder.

- unlock - apply patches to VMware
- relock - remove patches from VMware
- check - check the patch status of your VMware installation

5. VMware Downloads

These URLs will link to the latest versions of VMware's hosted products:

- VMware Fusion <https://vmware.com/go/getfusion>

- VMware Workstation for Windows <https://www.vmware.com/go/getworkstation-win>
- VMware Workstation for Linux <https://www.vmware.com/go/getworkstation-linux>
- VMware Player for Windows <https://www.vmware.com/go/getplayer-win>
- VMware Player for Linux <https://www.vmware.com/go/getplayer-linux>

6. VMware Tools

The Unlocker provides the VMware tools ISO images. There can be newer releases available which can be downloaded from these URLs if the script has not yet been updated:

- Mac OS X 10.5 - 10.10 <https://customerconnect.vmware.com/en/downloads/details?downloadGroup=VMTOOLS10012&productId=491>
- macOS 10.11+
https://customerconnect.vmware.com/downloads/info/slug/datacenter_cloud_infrastructure/vmware_tools/12_x

These URLs require a VMware login to download.

Version 16 of Workstation Pro recognises the darwin.iso files and the tools can be installed in the usual way by using the "Install VMware Tools" menu item. The Player version does not automatically pick up the ISO images and so the ISO must be manually attached to the VM via the guest's settings.

9. Thanks

Thanks to Zenith432 for originally building the C++ Unlocker and Mac Son of Knife (MSoK) for all the testing and support.

Thanks also to Sam B for finding the solution for ESXi 6 and helping me with debugging expertise. Sam also wrote the code for patching ESXi ELF files and modified the Unlocker code to run on Python 3 in the ESXi 6.5 environment.

(c) 2011-2022 David Parsons