

Install & Uninstall iMazing in Enterprise Environments

Last updated: **June 23rd, 2025**

Authors: **DigiDNA - Jérôme Bédard**

Table of contents:

[1. Silent Installation on Windows](#)

[1.1. Automatic Apple Components Installation](#)

[1.2. Custom Apple Components Installation](#)

[2. Silent Uninstallation on Windows](#)

[2.1. Remove iMazing working directory, activated license and preferences from the local hard drive](#)

[2.2. Remove other remaining Windows Registry keys completely](#)

[2.3. Apple Components complete uninstallation when installed via the custom installation method](#)

[3. Automatic Activation with License Code](#)

[3.1. Windows](#)

[3.2. macOS](#)

[3.3. Connecting to the iMazing Licensing System](#)

[4. File Locations: Preferences, Backups, Library, and Cache Data](#)

[4.1. Windows](#)

[4.2. macOS](#)

[5. Domains Required by iMazing](#)

[6. Third-Party Libraries Used by iMazing](#)

1. Silent Installation on Windows

To install iMazing silently, open a Windows Command Prompt with administrative privileges or use Intune (<https://learn.microsoft.com/en-us/intune/intune-service/fundamentals/what-is-intune>). Then, execute the following command, making sure to replace `<path to installer>` with the path to the location where the iMazing Installer (`iMazing3forWindows.exe`) is located:

```
"<path to installer>\iMazing3forWindows.exe" /verysilent /closeapplications /  
suppressmsgboxes /dontstart
```

To prevent iMazing Installer from automatically launching iMazing at the end of the installation, you can pass the parameter `/dontstart`, as shown in the example above.

1.1. Automatic Apple Components Installation

Starting from version 2.17, iMazing Installer can download all necessary Apple components from the Microsoft Store to install them. iMazing Installer will place all components in the following folder:

```
C:\ProgramData\DigiDNA\iMazing\MobileDevice
```

To troubleshoot Apple components installation, a log file is created at this path:

```
C:\ProgramData\DigiDNA\iMazing\MobileDevice\log.txt
```

Ensure that the computer on which iMazing Installer is running can access the domain names used to download the components:

- `downloads.imazing.com`
- `tlu.dl.delivery.mp.microsoft.com`

To disable automatic installation of Apple components, pass the following argument to iMazing Installer:

```
"<path to installer>\iMazing3forWindows.exe" /dontinstallapplecomponents
```

Note that if iMazing Installer is unable to connect to these domains, Apple components and drivers will not be installed. To install Apple components manually, you will first need to download iTunes' installer, follow the steps below.

1.2. Custom Apple Components Installation

First, download the standalone version of iTunes. Then, follow the instructions below:

- For Windows 64-bit systems: <https://www.apple.com/itunes/download/win64>

- For Windows 32-bit systems: <https://www.apple.com/itunes/download/win32>

Apple Components installation for Windows 64-bit systems:

1. Once iTunes is downloaded, extract the contents of **iTunes64Setup.exe** to the location of your choice using 7zip (<https://www.7-zip.org>) or WinRAR (<https://www.win-rar.com>). Here is the list of files required to install Apple components contained in the iTunes64Setup.exe package:

- `AppleMobileDeviceSupport64.msi`
- `Bonjour64.msi` (If you want iMazing to be able to connect to iOS devices via Wi-Fi)

2. Then open a Windows Command Prompt with administrative privileges and type in the following commands making sure to replace `<path to installer>` with the path to the files you've just extracted:

```
"<path to installer>/MobileDeviceSupport64.msi" /quiet /qn  
"<path to installer>/Bonjour64.msi" /quiet /qn
```

Apple Components installation for Windows 32-bit systems:

1. Once iTunes is downloaded, extract the contents of **iTunesSetup.exe** to the location of your choice using 7zip (<https://www.7-zip.org>) or WinRAR (<https://www.win-rar.com>). Here is the list of files required to install Apple components contained in the iTunesSetup.exe package:

- `AppleMobileDeviceSupport.msi`
- `Bonjour.msi` (If you want iMazing to be able to connect to iOS devices via Wi-Fi)

2. Then open a Windows Command Prompt with administrative privileges and type in the following commands making sure to replace `<path to installer>` with the path to the files you've just extracted:

```
"<path to installer>/MobileDeviceSupport.msi" /quiet /qn  
"<path to installer>/Bonjour.msi" /quiet /qn
```

2. Silent Uninstallation on Windows

To uninstall iMazing silently please open a Windows Command Prompt with administrative privileges and execute the following command:

```
"C:\Program Files\DigiDNA\iMazing\unins000.exe" /verysilent /closeapplications /suppressmsgboxes
```

When running the uninstaller silently, the computer will automatically restart to unregister iMazing's Windows Explorer shell extension (which manages drag and drop functionality from iMazing).

2.1. Remove iMazing working directory, activated license and preferences from the local hard drive

- `C:\Users\<username>\AppData\Roaming\iMazing` (Beware, iMazing backups are by default located in: `C:\Users\<username>\AppData\Roaming\iMazing\Backups`)
- `C:\Users\<username>\AppData\Local\DigiDNA`
- `C:\ProgramData\DigiDNA\iMazing`

2.2. Remove other remaining Windows Registry keys completely

- `Computer\HKEY_CURRENT_USER\Software\Classes\.imazing`
- `Computer\HKEY_CURRENT_USER\Software\Classes\.imazingapp`
- `Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.imazing`
- `Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.imazingapp`
- `Computer\HKEY_CURRENT_USER\Software\DigiDNA\iMazing`
- `Computer\HKEY_CURRENT_USER\Software\DigiDNA\iMazingMini`
- `Computer\HKEY_CURRENT_USER\Software\iMazing`
- `Computer\HKEY_CURRENT_USER\Software\iMazingMini`

2.3. Apple Components complete uninstallation when installed via the custom installation method

To uninstall all apple components silently please open a Windows Command Prompt with administrative privileges and execute the following commands.

```
if exist "%COMMONPROGRAMW6432%\Apple\Mobile Device Support" wmic
product where name="Apple Mobile Device Support" call uninstall /nointeracti
ve
if exist "%COMMONPROGRAMFILES%\Apple\Mobile Device Support" wmic p
roduct where name="Apple Mobile Device Support" call uninstall /nointeractiv
e
if exist "%COMMONPROGRAMFILES(X86)%\Apple\Mobile Device Support" w
mic product where name="Apple Mobile Device Support" call uninstall /nointe
ractive
if exist "%COMMONPROGRAMW6432%\Apple\Apple Application Support" w
mic product where name="Apple Application Support (64-bit)" call uninstall /n
ointeractive
if exist "%COMMONPROGRAMFILES%\Apple\Apple Application Support" wmi
c product where name="Apple Application Support (32-bit)" call uninstall /noi
nteractive
if exist "%COMMONPROGRAMFILES(X86)%\Apple\Apple Application Suppor
t" wmic product where name="Apple Application Support (32-bit)" call uninst
all /nointeractive
if exist "%PROGRAMW6432%\Bonjour" wmic product where name="Bonjour"
call uninstall /nointeractive
if exist "%PROGRAMFILES%\Bonjour" wmic product where name="Bonjour" c
all uninstall /nointeractive
if exist "%PROGRAMFILES(X86)%\Bonjour" wmic product where name="Bonj
our" call uninstall /nointeractive
```

Note: if you want to continue to use iTunes, you shouldn't uninstall Apple components.

3. Automatic Activation with License Code

3.1. Windows

To automate activation on Windows, you can create a text file `license.txt` containing the license code at the following user path:

```
C:\Users\<username>\AppData\Roaming\iMazing\license.txt
```

or at the following computer-wide path for all users:

```
C:\ProgramData\DigiDNA\iMazing\license.txt
```

On Windows you can alternatively pass the license code to the iMazing Installer, that way:

```
"<path to installer>\iMazing3forWindows.exe" /license=<license_code>
```

Replace `<username>` by the username, or `<path to installer>` by the path pointing to the installer and `<license_code>` by your license code.

3.2. macOS

To automate activation on macOS, you can create a text file `license.txt` containing the license code at the following user path:

```
/Users/<username>/Library/Application Support/iMazing/license.txt
```

Replace `<username>` by the username.

3.3. Connecting to the iMazing Licensing System

iMazing will need to connect to the following domain name to communicate with iMazing's licensing system:

- `api.imazing.com`

iMazing will be activated automatically when launched. In addition, depending on the feature you use, you will need to whitelist certain domains that iMazing will connect to. A full list of what iMazing connects to and why is available here:

<https://support.imazing.com/hc/en-us/articles/360010474920-List-of-domains-iMazing-connects-to>

4. File Locations: Preferences, Backups, Library, and Cache Data

iMazing stores important data in several folders: preferences, backups, library (for downloaded iOS/iPadOS apps `.ipa` and OS images `.ipsw`), and cached files.

4.1. Windows

- Preferences: `C:\Users\<username>\AppData\Roaming\iMazing\Prefs`
 - BackupLocations.plist: backup locations
 - Devices.plist: devices paired with iMazing
 - Organizations.plist: supervising organizations (without their identities (certificates and private keys) which are stored in the Windows Certificate Store
 - Prefs.plist: iMazing non-UI preferences
 - Schedules.plist: automatic backup preferences for iMazing Mini
 - Shortcuts.plist: file system and app shortcuts
- UI Preferences: `C:\Users\<username>\AppData\Local\DigiDNA` (UI and automatic updates)
- Default backup location : `C:\Users\<username>\AppData\Roaming\iMazing\Backups`
 - One folder per device backup, each named with the device's UDID
- Default backup snapshots location : `C:\Users\<username>\AppData\Roaming\iMazing\Backups\iMazing.Versions`

- Blueprints: editable backups
- Temp: temporary backups built during restore operations. All files stored in this folder are automatically deleted when iMazing launches or quits.
- Trash: deleted backups that are being removed in the background
- Versions: backup snapshots per device
- Note: All the above folders contain hard links pointing to the main related backup folder. Do not move or edit files manually, as this would result in corrupted backup snapshot chains. Also note that Windows File Explorer cannot properly calculate storage usage for hard links, which leads to misleading reported sizes.
- Default library location: `C:\Users\<username>\AppData\Roaming\iMazing\Library`
 - Apps.plist and Apps folder: downloaded `.ipa` files
 - Blueprints.plist and Blueprints folder: iMazing Configurator blueprints
 - Profiles folder: configuration and provisioning profiles
 - Systems folder: IPSW OS installation images stored in library
- Caches: `C:\Users\<username>\AppData\Roaming\iMazing\Caches`
 - One folder per device, each named with the device's UDID
- Temporary: `C:\Users\<username>\AppData\Roaming\iMazing\Caches\Temp`
 - All files stored in this folder are automatically deleted when iMazing launches or quits
- Logs: `C:\Users\<username>\AppData\Roaming\iMazing\Caches\Logs`
- Pairing records: `C:\ProgramData\Apple\Lockdown\`
- Apple Drivers installed by iMazing: `C:\ProgramData\DigiDNA\iMazing\MobileDevice` (downloaded during iMazing installation)
- Apple Drivers installed by iTunes: `C:\Program Files\Common Files\Apple\Mobile Device Support` (installed during iTunes installation)

4.2. macOS

- Preferences: `/Users/<username>/Library/Application Support/iMazing/Prefs`
 - BackupLocations.plist: backup locations
 - Devices.plist: devices paired with iMazing
 - Organizations.plist: supervising organizations (without their identities (certificates and private keys) which are stored in the macOS Keychain)
 - Prefs.plist: iMazing non-UI preferences
 - Schedules.plist: automatic backup preferences for iMazing Mini
 - Shortcuts.plist: file system and app shortcuts
 - USBPort.plist: USB port mapping for iMazing Configurator
- UI Preferences: `/Users/<username>/Library/Preferences/com.DigiDNA.iMazing2Mac.plist`
(UI and automatic updates)
- Default backup location: `/Users/<username>/Library/Application Support/iMazing/Backups`
 - One folder per device backup, each named with the device's UDID
- Default backup snapshots location : `C:\Users\<username>\AppData\Roaming\iMazing\Backups\iMazing.Versions`
 - Blueprints: editable backups
 - Temp: temporary backups built during restore operations. All files stored in this folder are automatically deleted when iMazing launches or quits.
 - Trash: deleted backups that are being removed in the background
 - Versions: backup snapshots per device
 - Note: All the above folders contain hard links pointing to the main related backup folder. Do not move or edit files manually, as this would result in corrupted backup snapshot chains. Also note that Finder cannot properly calculate storage usage for hard links, which leads to misleading reported sizes.
- Default library location: `/Users/<username>/Library/Application Support/iMazing/Library`
 - Apps.plist and Apps folder: downloaded `.ipa` files

- Blueprints.plist and Blueprints folder: iMazing Configurator blueprints
- Profiles folder: configuration and provisioning profiles
- Systems folder: IPSW OS installation images stored in library
- MDMServer.plist: MDM servers configured in iMazing Configurator
- Caches: `/Users/<username>/Library/Application Support/Caches/iMazing`
 - One folder per device, each named with the device's UDID
- Temporary: `/Users/<username>/Library/Application Support/Caches/iMazing/Temp`
 - All files stored in this folder are automatically deleted when iMazing launches or quits
- Logs: `/Users/<username>/Library/Application Support/Caches/iMazing/Logs`
- Pairing records: `/var/db/lockdown` (protected and not accessible without SIP disabled)
- Apple Drivers: `/System/Library/PrivateFrameworks/MobileDevice.framework` (managed by macOS)

5. Domains Required by iMazing

iMazing keeps all data local. Your iOS device data is only stored in iMazing's Cache folder and your chosen backup location—never on our servers.

To ensure all iMazing features function properly, whitelist these domain names for ports 80 and 443.

Installing or updating iMazing and Apple components:

- `downloads.imazing.com`
 - iMazing updates
- `tlu.dl.delivery.mp.microsoft.com`
 - Apple components' updates for Windows, downloaded from the Microsoft Store

iMazing licensing system:

- `imazing.com`
 - iMazing embedded store
- `store.imazing.com`
 - iMazing embedded store
- `api.imazing.com`
 - iMazing's licensing and activation system

Features relying on Apple servers:

- `*.apple.com`
 - Activating new or erased devices
 - Installing or reinstalling iOS, iPadOS, and macOS on Apple Silicon (IPSW downloads)
 - Downloading iCloud data
 - Searching and downloading App Store apps
 - App Volume Purchasing (VPP) license management
- `*.mzstatic.com`
 - Apple's Content Delivery Network (CDN) that supports some of the features listed above

Spyware Analyzer:

The Spyware Analyzer needs to resolve HTTP redirects for URLs found in analyzed device data. To detect spyware, iMazing must be able to send HTTP requests to any domain or IP address.

- `downloads.imazing.com`
 - Downloading the up-to-date list of indicators of compromise (IoCs) used for analysis
- `raw.githubusercontent.com`
 - Downloading indicators of compromise (`.stix` files)

Issue reporting:

- sentry.io
- *.sentry.io

Network connectivity check:

If iMazing fails to contact its activation server on api.imazing.com, it will perform a sanity check and attempt to reach Google or Baidu. No information is exchanged with Google and Baidu servers—iMazing strictly performs a reachability check.

- google.com
- baidu.com

Customer Experience Program:

When users opt in to participate in the Customer Experience Program, iMazing sends anonymized usage data to Google Analytics, our platform for collecting anonymous usage statistics to help improve our product. We record in-app navigation events, without any user data.

- www.google-analytics.com

6. Third-Party Libraries Used by iMazing

iMazing leverages these third-party packages:

- boost
- ccard
- CoreFoundation++
- Chilkat
- CPPAtomic
- dirent
- dlib
- exif
- gmock-xcode
- heic-convert

- HTML Renderer
- ID3v2
- ISOBMFF
- libarchive
- libavutil / libswresample / libavcodec / libavformat
- ligbpod
- libjpeg
- libmediainfo
- libpng
- libzip
- objc-uti
- OBJCXX
- Objective-CPP
- OpenSSL
- BoringSSL
- PDFsharp
- PDFLib
- PIMPL
- Protocol Buffers
- SQLite
- SWIG
- TagLib
- zlib

iMazing bundles iMazing Profile Editor which leverages these third-party packages:

- Device Management Client Schema

- EonilFSEvents
- plist-cil
- PlistCS
- ProfileManifests
- YamlDotNet
- .NET