iMazing in Enterprise & Institutional Environments

Last updated: **July 5th, 2023**
Authors: **DigiDNA - Jérôme Bédat, Grégorio Zanon**

1. Introduction

1.1 About iMazing

iMazing is a general purpose iOS device manager available for macOS and Windows. The software facilitates data transfers, offers advanced backup tools, and more recently configuration tools via iMazing Configurator. iMazing is openly sold and distributed, but the Configurator layer is restricted for use by businesses and professionals only. As of 2020, iMazing is the most popular iOS device manager worldwide, with particularly strong adoption in the US, Canada, UK, Australia, Germany, France, Switzerland, Japan and South Korea.

iMazing is developed by DigiDNA SARL, a limited liability company registered in Switzerland under IDE CHE-114.420.432.

1.2 About DigiDNA

DigiDNA started developing iOS data transfer solutions in 2008 when Apple released the first iPhone. Its flagship solution at the time was named DiskAid, rebranded into iMazing in 2014. Software engineering has always been done 100% in-house, in DigiDNA's Geneva, Switzerland office. The company is privately owned by Swiss citizens.

1.3 Corporate and Institutional Presence

iMazing has been successfully deployed by hundreds of small and large businesses, national banks, government institutions, educational institutions and non-profit organizations. In the US alone, iMazing is actively used by 30 federal bodies (500+ seats), 40 state-level government institutions, 12 counties and 30 large cities.

iMazing's command line interface (iMazing CLI, available by contract only) has helped businesses and government bodies automate complex iOS data capture projects. It was notably successfully deployed in 2019 by Multnomah County to delegate encrypted backups of over 1'000 iPhones to their end-users, via scripts running on county employees' Windows terminals.

2. Security and Privacy

2.1 General Considerations

Since its very beginning, iMazing has been leveraging Apple's MobileDevice framework to communicate with iOS devices. It doesn't rely on other techniques that may harm the user's security. DigiDNA's principle is to respect and follow Apple's philosophy in terms of security, usage and good practices.

iMazing has never been linked to any jailbreak, and does not require the device to be jailbroken to properly operate. Because iMazing only relies on Apple's MobileDevice framework, iMazing can be considered as equally secure as Apple products such as *iTunes* (*Finder* since macOS *Catalina*) or *Apple Configurator*. iMazing does not bypass nor break iOS security in any way. It fully respects end-user security, privacy and data ownership, as well as all iOS security flows and guidelines.

iMazing never uploads any data to remote locations. The extracted data remains on the user's machine and is never exposed or uploaded to any server or cloud. That being said, private data extracted by the user in non-encrypted form is just as much as risk as any other file. The usual security recommendations apply: Windows or macOS user accounts should be secured with a strong password, and encryption enabled where possible. Necessary system updates both device and computer side should naturally be applied with minimum delay.

2.1.1 Code Signing (macOS)

The macOS version of iMazing is compiled with hardened runtime enabled, and notarized since notarization was introduced in macOS 10.14 Mojave. It is signed with DigiDNA's Apple developer certificate:

```
Certificate:     Developer ID Application: DigiDNA SARL (J5PR93692Y)
Expires:         Monday, 1 February 2027 at 23:12:15
Issuer:          Apple Inc.
```

Fingerprints:

- SHA-256: `EB F3 65 E7 85 70 84 0D 93 82 E0 23 03 92 CC 72 BC CB 3D 54 58 07 E2 1F 40 2D 06 72 62 B1 F2 02`
- SHA-1: `7D 6C 95 C0 6A 2B 4B 6E 90 6D F6 04 0E 89 96 A9 82 9B 59 68`

2.1.2 Code Signing (Windows)

The Windows version of iMazing is code-signed with a Thawte certificate:

```
Certificate:     DigiDNA SARL
Expires:         Wednesday, 25 October 2023 at 14:00:00
Issuer:          DigiCert, Inc.

Fingerprints:
```

- SHA-256: `16 9F A0 60 0E 5A 05 09 6B C1 36 73 ED A9 F5 7E DA 32 91 64 39 CC 90 5D 48 1C 10 46 82 F4 B8 0F`
- SHA-1: `92 A7 4D A4 13 4F 21 DF 1C 83 96 1C 00 02 62 D1 C5 5A C2 A4`

A different code-signing identity or certificate authority, or the absence of code-signing, indicate a non-genuine version of iMazing which could be infected with malware. Please report any code-signing discrepancies immediately at https://imazing.com/contact, mentioning the source of the application.

2.2 Device and Computer Pairing

Communication between an iOS device and a computer is established via a *pairing* process. Upon first connection via USB of an iOS or iPadOS device to the host computer, the mobile OS displays a *Trust dialog* which the user needs to acknowledge before entering the device's passcode (no biometric auth). This action generates two pairing records, each containing two certificates and a private key. Pairing records are stored both on the device and on the host computer. It is therefore possible to invalidate pairing by removing the pairing records either from the computer (host) or the device (client).

To remove pairing from a computer, use the *Forget* action in iMazing and check the *Unpair* option. To remove all pairing records from a device, reset Location & Privacy settings in the iOS settings app (*Settings* > *General* > *Reset* > *Reset Location & Privacy*)
For more info, please read Apple's knowledge base article:
https://support.apple.com/en-us/HT202778

Tip: To remove all pairing records from the computer, you can use the *Remove All Pairing Records* option available in iMazing's preferences window (*Preferences* > *Devices*).

iMazing uses Apple's MobileDevice framework to pair devices with computers or verify existing pairings. iMazing itself never has access to pairing records stored on the computer or on mobile devices. Pairing records are managed by Apple's *Usbmux* daemon (*Apple Mobile Device Service* on Windows and *usbmuxd* daemon on macOS) and stored in a secure location on the

local computer's file system. By Apple's design, pairing records are stored at the machine level and not in the user's home folder.

2.2.1 Pairing Records location on macOS

On macOS, pairing records are stored in a folder only accessible with administrative privileges:

- `/private/var/db/lockdown`

2.2.2 Pairing Records location on Windows

On Windows, pairing records are also stored in a folder only accessible with administrative privileges:

- `C:\ProgramData\Apple\Lockdown`

2.2.3 Wi-Fi Connections

Once pairing is established, iMazing can communicate with iOS devices via a USB cable or a local WLAN connection. Both the computer and device must be connected over the same local WLAN network for wireless communication to work. Apple's *Bonjour* protocol is used to discover the device over the local network: https://en.wikipedia.org/wiki/Bonjour_(software)
Apple's *Bonjour* protocol is based on *mDNS* (https://en.wikipedia.org/wiki/Multicast_DNS). It uses the *UDP* port *5353* to send multicast *UDP* packets over the local network to discover iOS devices which have their "MobileDevice" Wi-Fi connection enabled.

By convenience, iMazing automatically enables the Wi-Fi connection after a pairing is done. This behavior can be turned off in iMazing's *Preferences* > *Devices* > *Automatically enable Wi-Fi connection when connecting new devices*. The user can also disable the Wi-Fi connection for each device individually in the *Options* panel. It is also possible to temporarily ignore all incoming Wi-Fi connections by enabling the *Ignore Wi-Fi connections* in iMazing's preferences window. This can be especially useful when configuring devices with iMazing Configurator.

2.3 Passwords, Credentials and Certificates

Login credentials, passwords and certificates are never exposed or written to the computer's file system.

2.3.1 macOS Keychain

On macOS, all passwords, certificates and private keys are securely stored in the user's macOS *Keychain*.

- The user can access his stored credentials, passwords or certificates using the macOS *Keychain Access* app: https://support.apple.com/en-ae/guide/keychain-access/kyca1083/mac

2.3.2 Windows Credentials, Certificate Store and Key Storage Provider

On Windows, all passwords, certificates and private keys are securely stored in the user's account.

- *Windows Credentials* is used to store passwords and login credentials. The user can access them using *Windows Credential Manager*.
    - https://support.microsoft.com/en-us/help/4026814/windows-accessing-credential-manager
- The *Windows Certificate Store* is used to store certificates. The user can access all certificates stored in his personal store by using the *Windows Certificate Manager Tool*.
    - https://docs.microsoft.com/en-us/dotnet/framework/tools/certmgr-exe-certificate-manager-tool
- The user *Windows CNG Key Storage Provider (KSP)* is used to store certificate private keys.
    - https://docs.microsoft.com/en-us/mem/configmgr/core/plan-design/network/cng-certificates-overview
    - https://docs.microsoft.com/en-us/windows/win32/seccng/key-storage-and-retrieva
    - https://docs.microsoft.com/en-us/windows/win32/seccertenroll/cng-key-storage-providers

2.3.3 Sensitive Items Naming Conventions on macOS

- i**OS backup passwords** are saved in the Keychain with the name "*iOS Backup"*.
  The *Account* field is used to store the device *UDID*.
- **iOS passcode unlock tokens** are saved in the Keychain with the name
  "*iOS Unlock Token ({Device Serial Number})*".
- **Apple ID credentials** are saved in the Keychain with the name
  "*iMazing - Apple ID ({Apple ID EMail})*".
- **Organization certificates** generated when creating a supervision identity are saved in the Keychain with the name
  "*iMazing - {Organization Name} ({GUID})".*

2.3.4 Sensitive Items' Naming Conventions on Windows

- i**OS backup passwords** are saved in Windows Credentials with the name
  "*iMazing/{Device UDID}",*
- **iOS passcode unlock tokens** are saved in Windows Credentials with the name
  **iOS Unlock Token ({Device Serial Number})*"*,
- **Apple ID credentials** are saved in Windows Credentials with the name
  "*iMazing - Apple ID ({email})*",
- **Organization certificates** generated when creating a supervision identity are saved in the Windows Certificate Store with the name
  "*iMazing - {organization name} ({guid})".*

2.4 Backup, Data Storage and Encryption

Most of the data stored on an iOS device is by design inaccessible. Datasets such as messages, safari history, call history, voicemail, contacts and calendars cannot be directly retrieved, but are included in a local backup of the device. When attempting to load these datasets with iMazing, a backup of the device will be performed if one is not readily available. To simplify onboarding of new devices, iMazing will display an *Initial Backup Screen*, informing users of the necessity to back up their device in order to browse and extract data. The next screen lets users review backup options, and encourages enabling backup encryption by flagging the encryption option with a red badge if it is disabled. When encryption is enabled, users can still browse and extract data with iMazing as long as they provide the correct backup password, or if they allow iMazing to store it in the keychain or in Windows Credentials for them.

iMazing is also capable of keeping multiple snapshots of a device's backup. This feature is referred to as *Backup Archiving.* The archiving engine keeps track of new and modified files from one snapshot to the next to optimize disk usage. In order to compare backup versions and archive them efficiently, iMazing parses the backup's manifest (Manifest.db), which requires the backup password to be known if encryption is enabled. No personal data is decrypted at this point, only the metadata contained in the backup manifest.

Backup archiving is enabled by default, with a retention of 1 month of backup snapshots. Before disabling archiving, users must be aware that iOS backups are images of the current state of the device. Deleting data on an iPhone and backing up again will result in data loss, whether the backup is performed by iTunes, the Finder or iMazing. Backup archiving enables safer backups by preventing this type of data loss, as well as Time Machine style browsing and recovery of the contents of backup snapshots.

2.4.1 iOS Backup Encryption

Enabling backup encryption should be done before backing up a device for the first time. Backup encryption must be enabled for each device individually, by setting a password. This can be done in iMazing's *Device Options* panel. We recommend to always enable backup encryption in order to better protect any sensitive data, and enable backing up of datasets which by design are only backed up if encryption is enabled (Health data, HomeKit data, Call History, Safari History, Keychain and more).

iMazing itself doesn't encrypt backed up files. When backup encryption has been enabled in the *Device Options* panel, iOS' *BackupAgent* service running on the device takes care of preparing the backup and encrypting all files. Thanks to this delegation, encryption cannot be altered by an attacker, or files intercepted before encryption occurs.

The current backup encryption algorithm used by iOS is *AES-256*. With iOS 10.2, Apple considerably improved the algorithm used to derive keys, rendering brute force attacks on encrypted iOS backups virtually impossible. Read our article on the subject for more detail: https://imazing.com/blog/ios-10-2-introduces-safer-backups

2.4.2 Clear Cached Extracted Data

When iMazing extracts data from an iOS device, it needs to store it on the user computer's hard drive. Where possible, the data is extracted to a temporary folder which gets deleted both when quitting and launching iMazing.

To improve performance, some data may remain in iMazing's cache folder. The cache mainly contains thumbnails from the *Photos*' app, attachments from the *Messages* and *WhatsApp* apps, and contact pictures of the *Contacts* app. These are extracted as data is browsed only.

To clear iMazing's cache, go to iMazing's *Preferences* > *General* and click *Clear iMazing Cache*. It's also possible to clear cached files stored for a specific device: right click a device in iMazing's sidebar and click *Clear Cache*.

2.4.3 Location of Preferences, Backups, Caches and Other Data

On macOS the following folders are used:

- `/Users/{username}/Library/Application Support/iMazing`
- `/Users/{username}/Library/Application Support/Caches/iMazing`

On Windows, everything is in the same folder:

- `C:\Users\{username}\AppData\Roaming\iMazing`

2.4.4 Exported iMazing Configurator Blueprints

When exporting an iMazing Configurator blueprint to a *.blueprint* file, the user is prompted to enter an encryption password. The exported *.blueprint* file is encrypted with *AES-256*. The user decides whether or not to include the organization's identity. If included, the organization's identity will be encrypted with the same password, and included in the *.blueprint* in *PKCS12* format (https://en.wikipedia.org/wiki/PKCS_12).

2.4.5 Exported Supervision Organizations and Identities

As explained in section 2.3, organization certificates and private keys are always securely stored in the user's account.
When exporting a supervision organization to a *.organization* file or a supervision certificate (identity) to a *.p12* file, iMazing prompts the user to choose a password to encrypt the private key in *PKCS12* format (https://en.wikipedia.org/wiki/PKCS_12). That way, organizations' private keys are never exposed unencrypted.

2.5 Network Connections

iMazing always communicates with remote servers via *HTTPS* – all connections remain secure and encrypted. Depending on the server, *TLS 1.2* or *1.3* is used. iMazing also verifies the chain of certificates at every *HTTP* request to make sure that the issuer's root certificate is valid and prevent *man in the middle* attacks. (https://en.wikipedia.org/wiki/Man-in-the-middle_attack). Thanks to this verification, it is virtually impossible for an attacker to intercept iMazing requests to hijack credentials or any other sensitive information.

iMazing communicates with Apple servers for the following reasons:

- Activate devices after they have been erased
- Download apps from the App Store
- Manage Volume Purchase licenses (VPP device-based license associations and dissociations)
- Check and download new iOS versions (IPSW images)

2.5.1 Whitelist Domains

Once activated, iMazing can operate fully offline if needed. In order to access features which do require an internet connection, the following domains should be whitelisted in the computer or network's firewall:

- *imazing.com* (iMazing embedded store)
- **.imazing.com* (iMazing's licensing and activation platform, iMazing updates)
- **.apple.com* (for the various features listed above)
- **.mzstatic.com* (Apple's CDN)

2.5.2 Proxies

Two kinds of proxies are supported:

- *Secure Web Proxy*
- *SOCKS Proxy*

You can configure a proxy in iMazing Preferences / Network.

2.6 Third-Party Security Libraries Used in iMazing

- *Chilkat*
    - Version: 9.5.0.83
    - Purpose:
        - Network
        - Compression
        - Certificate management
- *OpenSSL*
    - Version: 1.1
    - Purpose:
        - Generating self-signed root certificates

3. About Backups, MDM, DEP and VPP

3.1 iMazing and iOS Backups

3.1.1 Backup Location

By default, iOS backups are stored on the user's computer's hard drive. It is possible to choose another location, such as an external hard drive or NAS. To be efficient, the connection between the computer and the external drive or NAS must be as fast as possible and stable enough. Because iOS' BackupAgent verifies that every file in the previous backup still exists before determining which files need to be included in a newer backup snapshot, a slow connection can lead to extremely slow backups which may never truly complete as the BackupAgent service triggers additional passes to guarantee a coherent backup. For this reason, we recommend to back up to the computer's local hard drive or to a USB connected external drive, and to do so on a daily basis. Backing up many iOS devices to a single network location is not advisable.

3.1.2 Automatic Backups

Using *iMazing Mini* to schedule daily automatic backups is a good way to keep your users' data safe while preserving good backup performances. Usually, a daily backup takes less than 15 minutes. Read more on how to enable daily backups here:
https://imazing.com/guides/how-to-backup-iphone-automatically

3.1.3 Wireless Backups

When automatic backups are configured and the Wi-Fi connection for the device enabled in iMazing, iMazing Mini will automatically initiate backups when it detects the device on the local WLAN network. The backup process is lightweight and should not noticeably impact the computer or the mobile device's overall performance. For optimal results, both the computer and the device's connection to the local network must be stable and capable of high bandwidth.

iMazing uses Apple's Bonjour protocol to discover device addresses over the local network. In your firewall and router settings, make sure that *TCP* ports 123 and 3689, and *UDP* ports 123 and 5353 are open and not blocked by the router or firewall.

Limitations

- We have observed that in some cases, Wi-Fi mesh networks negatively impact wireless backups due to a lack of stability.
- *Bonjour* may not be acceptable in high security corporate environments.
- Backing up many devices wirelessly to a single computer can lead to spotty results and connection issues. 10 devices should work fine, but probably not 50. The maximum number of simultaneous backups can be configured in iMazing Mini's preferences window.

3.1.4 Initial Backup and Backup Size

It is highly recommended to perform the initial backup over USB instead of Wi-Fi. The initial backup can take between a few minutes to several hours depending on the amount of data stored on the device. If you're using an external drive, a *USB 3* or *Thunderbolt* cable is recommended.

It is also important to take into account the amount of data that needs to be backed up to provision enough space on the target backup location. iOS only allows full backups, in other words it isn't possible to only back up specific data types or ranges. Backups can take up as little as 100MB or as much as 200 gigabytes, with photos generally occupying the most space.

3.1.5 Backup Versioning and File System Hard Links

iMazing backups are in the exact same format as iTunes or Finder backups, and stored in a similar hierarchy: backups are stored in a folder named after the UDID of the backed up device.

To this standard hierarchy which guarantees compatibility with Apple Configurator, iTunes and the Finder, iMazing adds the `iMazing.Versions` folder where snapshots are stored:

```
{Backup Location Folder}
      {Device UDID} (latest backup, full, restorable by any tool)
      iMazing.Versions
            Versions
                  {Device UDID}
                        Versions.db (change tracking)
                        {Snapshot Date} (only new and modified files)
                        {Snapshot Date}
                        {Snapshot Date}
```

iMazing's backup versioning engine compares the most recent backup, stored at the root of the configured backup location, with the previous backup snapshot. New and modified files are hard-linked in the corresponding folder in `iMazing.Versions`, and changes tracked in the `Versions.db` database. If no previous backup snapshot exists, all files are new and the entire backup is hard-linked in the corresponding `iMazing.Versions` folder.

Because the *Finder* and *Windows Explorer* do not account for hard-links when computing folder sizes, both will mistakenly report roughly double the size for the root folder of iMazing backups. On macOS, executing the *disk usage* command (`du -h`) in a terminal window will result in correctly computed folder sizes.

Warning: backup snapshots stored in `iMazing.Versions` are partial backups which cannot be restored with *Apple Configurator*, *iTunes* or the *Finder*. Always use iMazing to interact with

snapshots, and **never manually move or delete backup snapshots.** Instead, users should go through iMazing's backup history view to browse, restore, delete or export snapshots: https://imazing.com/guides/backups-list-in-imazing

3.2 Supervision, MDM, DEP and VPP

iMazing is fully compatible with devices which are enrolled in *MDM*, both supervised and unsupervised.

iMazing Configurator is a UI layer which sits on top of iMazing, designed to supervise, provision and configure devices locally. *iMazing Configurator* requires a separate license to be fully functional.

3.2.1 Supervision

Supervising a device is a way for companies to express ownership and apply finer grained configurations to them. As an example, *Single app mode* (also called kiosk mode) is only available on supervised devices.

Devices can be supervised by Apple Configurator, iMazing Configurator, or via Apple's Device Enrollment Program (*DEP*, also called *Automated Enrollmen*t). Supervising with *iMazing Configurator* is achieved via configuring a supervising organization in a blueprint (https://imazing.com/guides/configurator-blueprints#org).

The relationship between the supervising host and supervised devices is secured by a digital certificate, the *supervision identity*. In *Apple Configurator* and *iMazing Configurator*, supervision identities can be exported and imported along with organization metadata via `.organization` files. Alternatively, the identity in *PKCS12* format (`.p12` or `.pfx`) can be imported when creating a new organization. Here's *iMazing Configurator*'s documentation on the subject: https://imazing.com/guides/configurator-overview#orgs

If the supervision identity is configured in iMazing, supervised devices can be managed in a privileged way:

- Silent pairing (without the passcode)
- Silent installation of configuration profiles
- Non-removable profiles
- *Supervised only settings* of configuration profiles
- Clear passcode (if passcode unlock token was previously saved)
- Single App Mode
- Wallpapers
- And more…

For this reason, only administrators should have access to the supervision identity. End-users will have access to the standard panel of features, including backing up and transferring data, without the need to configure the supervision identity in iMazing.

Important: it is possible to restrict pairing of supervised devices to supervising hosts only. This setting is named *Allow Pairing*, or *Allow pairing with non-configurator hosts,* or *Allow host pairing* depending on the context. If set to false, iMazing will only be able to communicate with supervised devices if the supervision identity is properly configured. This setting can be configured when the device is initially supervised, or via a configuration profile.

3.2.2 DEP

Apple Device Enrollment, official page: https://support.apple.com/en-us/HT204142

iMazing can work in conjunction with Apple's *Device Enrollment Program (DEP)*. If the devices you need to configure are enrolled via *DEP*, supervision will be applied at enrollment time, via the *Device Enrollment Profile*. The supervision identity will be auto-generated and inaccessible, which is why admins must manually add their own certificates to the *Supervising Host Certificates* field of the enrollment profile. This is usually done via your MDM provider's interface, but not all MDMs expose the setting – *Jamf Now* for instance does not, but *Jamf Pro* does.

Once a device is enrolled, the device enrollment profile cannot be re-applied without fully erasing the device. For this reason, it is important to configure supervising host certificates *before* you enroll devices.

Device Enrollment Profile, Apple documentation:
https://developer.apple.com/documentation/devicemanagement/profile

iMazing Configurator can help with achieving zero-touch configuration of *DEP* enrolled devices, saving administrators precious time by taking care of installing (and removing if needed) a Wi-Fi profile, and advancing the iOS setup assistant as much as possible. Read the *MDM and DEP* section of the following article for more information:
https://imazing.com/guides/configurator-blueprints#mdm-dep

3.2.3 MDM

Managing Apple devices via *MDM* is convenient, but many admins still rely on a mix of remote and local management to speed up MDM enrollment or to handle data transfers – *MDM* only deals with configuration, apps and books, and not with backup/restore or data provisioning.

Apple Configurator can be leveraged to facilitate MDM enrollment. *iMazing Configurator* features similar capabilities, to which it adds the ability to provision apps with files and documents. The end goal is to fully provision devices with all required apps, data and

configurations, whilst removing the need for the end-user to go through all *Setup Assistant* steps and wait for specific content and apps to be downloaded.

Learn more about how *iMazing Configurator* can achieve zero-touch MDM enrollment and provisioning here:

- https://imazing.com/guides/configurator-blueprints#general

Apple's MDM documentation:

- https://developer.apple.com/documentation/devicemanagement

3.2.4 Apps and Volume Purchase (VPP)

iMazing can download and install apps to devices. When installing an app purchased from an *Apple Business Manager* or *Apple School Manager* account (Volume Purchase), iMazing will connect to the account and automatically associate the license to the device. It can also disassociate the license when the app is uninstalled. If you want to prevent iMazing from managing license associations and dissociations, navigate to *Preferences* > *Library* and uncheck the option *Automatically manage Volume Purchasing account licenses*.

- https://imazing.com/guides/configurator-overview#apps
- https://imazing.com/guides/preferences-in-imazing#library

You can also install ad-hoc or custom business apps *.ipa* files. For more details, please read:

- https://imazing.com/guides/configurator-blueprints#apps
- https://support.apple.com/en-us/HT202995

3.2.5 User Enrollment (BYOD)

With iOS 13 and macOS 10.15, Apple introduced a new way to manage *BYOD* devices (Bring Your Own Device). This new method is called *User Enrollment*. This approach should be privileged over other solutions in order to segregate managed corporate data and personal data located on the same device. Apple documentation on the subject is still very sparse, but a few major MDM vendors cover this new enrollment strategy:

- https://simplemdm.com/apple-user-enrollment/
- https://docs.jamf.com/jamf-school/deploy-guide-docs/User_Enrollment_and_On-Device_Enrollment.html

3.2.6 Apple Configurator vs iMazing Configurator and iMazing Profile Editor

With *iMazing Configurator* (https://imazing.com/configurator) and *iMazing Profile Editor* (https://imazing.com/profile-editor), we strive to simplify system administrators' workflows by improving general user experience and adding powerful extra features, such as:

- Apply blueprints to bulks of iOS devices simultaneously, with inline progress reports, user interaction handling and detailed logs.
- Add files to blueprints, and choose the app they should go to, including system apps.

- Override app defaults with the *Set Configuration* feature. Populate the Library folder of apps.
- Handle devices which already have data without erasing them.
- Manage a local repository of apps downloaded from the *App Store*.
- Handy pre-configuration and post-configuration automations include a battery health check, a launch app action and more.
- Share or save blueprints thanks to our *AES-256* encrypted *.blueprint* format.
- Supervise and configure devices on Windows.
- Speed-up MDM enrollment thanks to a *Zero-touch enrollment* mode.
- Backups of the same device are versioned by default to guard against data loss.

It should be noted that adding a device to DEP can only be done with Apple Configurator.

On Windows, iMazing can apply blueprints designed with the macOS version of *iMazing Configurator*, including blueprints which supervise devices and manage VPP assignments. A dedicated supervision wizard is in the works, please contact us if you have any specific requirements you would like us to take into account.

3.2.6 Two MDM-DEP Case Studies

Company A: Restricted pairing leads to data lock-in syndrome
In January 2020, we were contacted by *Company A* to help with an emergency situation. The company had just been sold, including all 300 company iPhones and iPads. As a legal requirement, all devices were to be backed up and erased before the handover. Problem: the devices were enrolled in *DEP*, and the device enrollment profile restricted host pairing to supervising hosts only. The administrator who had originally configured the profile did not provide any supervising host certificates, so none of the 300 devices could be paired and locally backed up. Updating the DEP profile would not have helped, since devices must be erased for the profile to be applied anew.

This case is sadly not unique. Apple moved to deprecate the *Allow Host Pairing* parameter in the DEP profile in iOS 13, but devices which were already supervised before that still may suffer from this data lock-in syndrome. The only solution is to back up to iCloud, and if the data is ever needed, to restore the backup of interest to a real device, a time consuming process.

On iOS 13 and above, pairing can still be restricted to supervising hosts only, via the *Restrictions* payload of a configuration profile. Because the MDM server can always remove the profile, this method of restricting pairing prevents data lock-in situations such as the one described here.

Company B: MDM Migration
Company B contacted us in May 2020 about an MDM migration project. All of the managed devices were enrolled via *DEP* in MDM A, and would need to be migrated to MDM B without data loss. The procedure should have been relatively simple, if time consuming: back up the

device, erase it, restore the backup and let automated enrollment enroll it anew in MDM B. Unfortunately, an iOS feature meant to streamline backup restores to the same device causes the setup assistant to be bypassed, and automated MDM enrollment to fail. We were able to find a solution, and are now working with Company B on an *iMazing Configurator* update which will enable full automation of the entire process.

3.2.7 Backing Up, Restoring and Migrating Data in a Supervised Context

As illustrated in the case studies above, backing up and restoring backups from or to supervised devices presents unique challenges. Fortunately, iMazing is well equipped to deal with these scenarios. Because local supervision (applied with *Apple Configurator* or *iMazing Configurator*) and DEP supervision behave somewhat differently, we will treat them separately here.

Local Supervision

- Devices supervised locally lose the supervised state when they are erased.
- The configuration (including supervision) is included in the backup, but only restored if the backup is restored to the same device.

If you need to migrate data from a supervised device to a non-DEP device, standard backup restore will not work. *iMazing Configurator* on the other hand can restore a backup to a different device whilst preserving supervision.

DEP Supervision

- Devices supervised via *DEP* will re-acquire supervision automatically after being erased, when enrollment is completed.
- Migrating data to a DEP device via backup/restore works as expected, with supervision applied even if the data comes from a non-supervised device.
- Restoring a backup of the same device breaks automated enrollment, as seen in the case study above.

If you need to restore a backup of a DEP device *to the same device*, standard backup restore with *Apple Configurator*, the *Finder* or *iTunes* may not work. iMazing features tools which can help in this situation too.

4. Deploying iMazing in a Corporate Environment

4.1 Deploying iMazing on Windows

4.1.1 Installation

On Windows, iMazing can be deployed using one of the following *Microsoft Endpoint Manager* solutions:

- *Configuration Manager* (*SCCM*): https://docs.microsoft.com/en-us/mem/configmgr/core/understand/introduction
- *Microsoft Intune*: https://docs.microsoft.com/en-us/mem/intune/fundamentals/what-is-intune
- *Windows Package Manager* https://docs.microsoft.com/en-us/windows/package-manager/winget/

Please note: iMazing's installer is not an *.msi*, but an *.exe* (*Inno Setup*) file. This should not prevent it from being installed via *Configuration Manager* or *Intune*.

To deploy iMazing using Configuration Manager:

1. Download iMazing's *.exe* installer from our website: https://imazing.com/download/windows
2. Copy `iMazing2forWindows.exe` to a network shared folder of your choice
3. Launch *Configuration Manager*
4. Click *Software Library*, under *Application Management* right click *Applications* and click *Create Application*.
5. Choose *Manually specify the application information* and click *Next* >.
6. Specify the application name and publisher (*iMazing*, *DigiDNA*) and click *Next* >.
7. Next to the selected language, click *Add/Remove*.
8. Configure the deployment type. For the *Type*, choose *Script Installer* and select *Manually specify the deployment type information*. Click *Next* >.
9. Specify the deployment type's general information (for instance *iMazing Installation*) and click *Next* >.
10. Now specify the location of the deployment type's content:
    a. *Current location*: select the location where the iMazing's installer *.exe* file is located (on your network shared folder).
    b. *Installation program*: `"iMazing2forWindows.exe" /verysilent /closeapplications /suppressmsgboxes /dontstart /license={licensecode}`
    Please note that you can pass your iMazing license code *{licensecode}* as an argument if you want to automatically activate iMazing. If you don't want to activate a license, omit `/license=xxx`

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

Then click *Next* >.

11. The *Detection Method* allows the administrator to check if an application is already installed. It can also prevent an install of an application if it conflicts with another application that is already installed. To configure the rules to detect whether the application already exists on the user machine, click *Configure rules to detect the presence of this deployment type* and click on *Add Clause*.
    On the *Detection Rule* window, choose *Setting Type* as *File System*. Under *Specify the file or folder to detect the app*, set the following:
    a. *Type*: `File`
    b. *Path*: `C:\Program Files\DigiDNA\iMazing`
    c. *File or folder name*: `iMazing.exe`
    To finish, click *OK* and the *Next* >.
12. *User experience*:
    a. *Installation behavior*: select *Install for system if resource is device else install for user*
    b. *Logon requirement*: select *Whether or not a user is logged on*
    c. *Installation program visibility*: select *Hidden*.
    Then click *Next* >.
13. For the next steps click *Next* > and finally click *Close*.
14. To distribute, right click the *iMazing* application and click on *Distribute Content*
15. Right click on the iMazing app and click *Deploy*. Choose the *Collection* where this application should be deployed. Wait a few minutes to see iMazing in the *Software Center*.
16. Finally to install the app, select it and click *Install* from the *Software Center*. The app will be downloaded and installed.

For more information on how to deploy content with *Configuration Manager*, please refer to Microsoft's documentation:
https://docs.microsoft.com/en-us/mem/configmgr/core/servers/deploy/configure/deploy-and-manage-content

4.1.2 Apple Components & Drivers

iMazing requires various Apple components to be able to communicate with iOS and iPadOS devices.

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

Please 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 to your PC, follow the steps below.

Custom Apple Components Installation
To disable automatic installation of Apple components, pass the following argument to iMazing Installer:
```
{path to installer}\iMazing2forWindows.exe
/dontinstallapplecomponents
```

All required components are automatically deployed when installing the standalone version of iTunes.

- For Windows 64-bit systems: https://www.apple.com/itunes/download/win64
- For Windows 32-bit systems: https://www.apple.com/itunes/download/win32

However, it is also possible to install these components without installing *iTunes* by extracting them from the *iTunesSetup.exe* or *iTunes64Setup.exe* installation packages. To do so, follow the instructions below.

Install Apple Components silently on Windows 64-bit systems:

1. Download **iTunes64Setup.exe** from https://www.apple.com/itunes/download/win64
2. 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** (optional, necessary only to support Wi-Fi connections)
2. Now open a Windows *Command Prompt* with administrative privileges and enter the following commands (make sure to replace *{location}* by the path where the files have been extracted):

    ```
    "{location}\AppleMobileDeviceSupport64.msi" /quiet /qn
    "{location}\Bonjour64.msi" /quiet /qn
    ```

Install Apple Components silently on Windows 32-bit systems:

1. Download **iTunesSetup.exe** from https://www.apple.com/itunes/download/win32
2. 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** (optional, necessary only to support Wi-Fi connections)
3. Open a Windows *Command Prompt* with administrative privileges and enter the following commands (make sure to replace *{location}* by the path where the files have been extracted):

```
"{location}\AppleMobileDeviceSupport.msi" /quiet /qn
"{location}\Bonjour.msi" /quiet /qn
```

Please note: You can also use *Configuration Manager (SCCM)* to deploy Apple components silently. Apply the instructions described in section 4.1.1 to deploy them to your users.

4.1.3 Configuration

To deploy a default iMazing configuration to your users:

1. If you've already used iMazing on your machine, temporarily rename iMazing's AppData folder in order to simulate a fresh install. The folder is located at `C:\Users\{username}\AppData\Roaming\iMazing`
2. Launch iMazing, open the *Preferences* window and change the default configuration such as the default backup location, default library path and any other option that you would like to preconfigure for your users.
3. Quit iMazing.
4. iMazing's custom configuration is located here: `C:\Users\{username}\AppData\Roaming\iMazing`
5. You can then follow Microsoft's instructions on how to create user data and profiles configuration items if you use *Configuration Manager (SSCM)*: https://docs.microsoft.com/en-us/mem/configmgr/compliance/deploy-use/create-user-data-and-profiles-configuration-items

4.1.4 License Activation

To automatically activate iMazing, a text file containing your license code can be placed in iMazing's AppData folder:
`C:\Users\{username}\AppData\Roaming\iMazing\license.txt`
When users launch iMazing for the first time, the app will automatically be activated and the license file removed.

Alternatively, you can pass the license code as an argument to iMazing's installer. This will automatically activate iMazing at first launch for all users on the machine:
`iMazing2forWindows.exe /license={licensecode}`

4.1.5 Uninstallation

Before removing all iMazing related files from a Windows PC, make sure to first copy or export items which the user may wish to keep:

- **iOS Backups**: iMazing's default backup location is `C:\Users\{username}\AppData\Roaming\iMazing\Backups`
- **iMazing's Library (Apps, Profiles, Blueprints)**: iMazing's default library location is `C:\Users\{username}\AppData\Roaming\iMazing\Library`
- **iOS backup passwords, iOS passcode unlock tokens, Organization identities and Apple ID credentials:** see section 2.3.4

You can also use iMazing's export functions to export backups, profiles and organizations.

To completely uninstall iMazing:

1. If you need to remove all pairing records from the computer, launch iMazing, navigate to *Preferences* > *Devices* and click *Remove All Pairing Records*.
2. To uninstall the app, navigate to the Windows *Settings* > *Apps & Features* panel and uninstall iMazing. You can also run the following command in Windows's *Command Prompt*:
    - `"C:\Program Files\DigiDNA\iMazing\unins000.exe" /verysilent /closeapplications /suppressmsgboxes`
3. Then remove iMazing's backups, preferences, library and cache files:
    - `C:\Users\{username}\AppData\Local\DigiDNA`
    - `C:\Users\{username}\AppData\Roaming\iMazing`
4. To remove all passwords, certificates and private keys stored by iMazing in *Windows Credentials* and *Windows Certificate Store*, see section 2.3.4.

Please note: after uninstalling iMazing, the computer should be restarted. iMazing's installer registers a *Windows Explorer* shell extension to manage drag & drop. Because of this, uninstallation will be completely finalized when Windows restarts.

4.2 Deploying iMazing on macOS

4.2.1 Installation

On macOS, iMazing can be deployed using *Munki* (https://github.com/munki/munki) or with any *MDM* solution.

Some MDM solutions may require you to create a *.pkg* containing the *iMazing.app* package. Here's how:

1. Download the iMazing *.dmg* from our website: https://imazing.com/download/macos
2. Open the *.dmg* and drag iMazing's icon to the *Applications* folder
3. Execute the following command from the macOS *Terminal* app:
   `pkgbuild --install-location /Applications --component /Applications/iMazing.app ~/Desktop/iMazing.pkg`
4. *iMazing.pkg* should appear on your desktop, ready to be deployed via *Munki* or your current *MDM* solution.

4.2.2 Apple Drivers

On macOS, the required Apple drivers (essentially the *MobileDevice.framework*) are shipped with the operating system. There is therefore no need to install any additional component.

Up to date drivers are especially important to allow pairing with the most recent devices and iOS versions, and to update or restore iOS (install *.ipsw* files to devices). When updating or restoring iOS, iMazing will warn the user if the *MobileDevice.framework* is outdated. In that case, the latest macOS updates should be applied in order to obtain the most recent version of the MobileDevice framework.

Older macOS versions may not support the most recent versions of MobileDevice. Upgrading macOS to the current version ensures compatibility with the most recent iOS and iPadOS updates.

4.2.3 Configuration

To deploy a pre-configured iMazing package, you will have to create a *.pkg* file containing all preference files. The simplest way to do so is to launch iMazing and configure it locally, before packaging it:

1. If you've already used iMazing on your machine, temporarily rename iMazing's Application Support folder in order to simulate a fresh install. The folder is located at `/Users/{username}/Library/Application Support/iMazing`
2. Launch iMazing, open the *Preferences* window and change the default configuration such as the default backup location, default library path and any other option that you would like to preconfigure for your users.
3. Quit iMazing.

4. Execute the following command from the *macOS Terminal app*:
   `pkgbuild --filter "/SharedData" --filter "/Emojis" --filter "/iMazing.Versions" --filter "/\." --root ~/Library/Application\ Support/iMazing --install-location "/Library/Application Support/iMazing" --identifier com.DigiDNA.iMazing.conf --version 1 ~/Desktop/iMazingConf.pkg`
5. *iMazingConf.pkg* should appear on your desktop, ready to be deployed via *Munki* or your current *MDM* solution.

4.2.4 License Activation

To automatically activate iMazing, a text file containing your license code can be placed in iMazing's Application Support folder: `/Users/{username}/Library/Application Support/iMazing/license.txt`
When users launch iMazing for the first time, the app will automatically be activated and the license file removed.

Please note: The *license.txt* file can also be included in the *iMazingConf.pkg* package that you deploy to your users. Simply copy the *license.txt* file in iMazing's *Application Support* folder before generating *iMazingConf.pkg*.

4.2.5 Uninstallation

Before removing all iMazing related files from a Mac, make sure to first copy or export items which the user may wish to keep:

- **iOS Backups**: iMazing's default backup location is `/Users/{username}/Library/Application Support/iMazing/Backups`
- **iMazing's Library (Apps, Profiles, Blueprints...)**: iMazing's default library location is `/Users/{username}/Library/Application Support/iMazing/Library`
- **iOS backup passwords, iOS passcode unlock tokens, Organization identities and Apple ID credentials:** see section 2.3.3

You can also use iMazing's export functions to export backups, blueprints, profiles and organizations.

To completely uninstall iMazing:

1. If you need to remove all device pairing records from the computer, launch iMazing, navigate to *Preferences > Devices* and click *Remove All Pairing Records*.
2. Remove the app from the *Applications* folder:
   - `/Applications/iMazing.app`
3. Remove iMazing's backups, preferences, library and cache files:
   - `/Users/{username}/Library/Application Support/iMazing`
   - `/Users/{username}/Library/Application Support/Caches/iMazing`

4. To remove all passwords, certificates and private keys stored by iMazing in the macOS *Keychain*, see section 2.3.3.

5. Important User Manuals

- Getting started with iMazing:
  https://imazing.com/guides/getting-started-with-imazing
- Learn iMazing's interface:
  https://imazing.com/guides/learn-imazing-interface
- Device Options in iMazing:
  https://imazing.com/guides/device-options-in-imazing
- Backup Options in iMazing:
  https://imazing.com/guides/backup-options-in-imazing
- Manage your backups with iMazing:
  https://imazing.com/guides/manage-your-iphone-or-ipad-backups-with-imazing
- Getting started with iMazing Mini:
  https://imazing.com/guides/getting-started-with-imazing-mini
- Getting started with iMazing Profile Editor:
  https://imazing.com/guides/getting-started-with-imazing-profile-editor
- iMazing Configurator: Quick Start Guide:
  https://imazing.com/guides/configurator-quick-start
- iMazing Configurator Overview
  https://imazing.com/guides/configurator-overview
- iMazing Configurator: Blueprints Deep Dive
  https://imazing.com/guides/configurator-blueprints
- All iMazing Guides:
  https://imazing.com/guides