

MSI Installation Instructions

Some companies prefer to install applications automatically from a centralized IT service. .MSI is a standard package used by Windows to install, maintain, and remove software. This can simplify the process when needing to coordinate an install across many computers and locations. It acts as a structured database containing all the instructions and files needed for a program to function correctly on your organizations computers.

Available Installer Formats

- MSI - Enterprise deployment via SCCM, Intune, GPO, or other central IT tools
- EXE - Manual installation by end users (self-contained, identical payload to MSI)

Update model: This MSI is for manual-update customers only. Customers on the Alchemy auto-update service receive updates automatically through the Updater Service - no MSI redeployment needed.

1. What Gets Installed

The MSI always installs both of the following components together - there is no option to install one without the other:

- *Alchemy Player* - Main desktop application (Alchemy.Sistem.Shell.exe)
- *Alchemy Player Updater Service* - Background Windows service that manages future application updates. Ensures Alchemy Player software is up to date. If this service is disabled, Alchemy Player will not be able to check for or install updates. This service uninstalls itself when Alchemy Player is not detected on the machine.
 - Auto-updates can be disabled for customers who want to control all updates themselves.

*Client Windows Operating Systems | **In a clean environment with only the base Windows installation:** Windows 10 and 11 (no prerequisites). No additional .NET or Visual C++ runtimes are required.*

*Client Windows Operating Systems | **Windows 10 and 11 with .NET 8.0+ already installed:** The self-contained deployment does not conflict with any existing .NET version on the machine. The installer behaves correctly in both scenarios.*

If performing an upgrade, make sure Player is not open or in-use. Otherwise schedule the deployment during off-hours. They must package ALL files and folders included with the installer files and not just the AlchemyPlayerSetup.exe file by itself. Failing to do so may cause issues. For example, if CoursePlayer files are not bundled then courses may not play correctly.

MSI Installation Instructions

2. Shortcuts Created

Alchemy Player -> Desktop (target: Alchemy.Sistem.Shell.exe)

Alchemy Player -> Start Menu > Alchemy Systems (target: Alchemy.Sistem.Shell.exe)

Client Media -> Desktop (target: CLIENT_MEDIA folder)

Client Media -> Start Menu > Alchemy Systems (target: CLIENT_MEDIA folder)

3. Silent Deployment - Command-Line Reference

Previous EXE silent install command (for reference): AlchemyPlayerSetup.exe /s /v"/qn"

The MSI replaces this with standard msixec syntax shown below.



Important: folders must exist before running any command: The msixec engine will not automatically create any folder. If the specified directory does not exist, msixec will fail to start the installation entirely. In the examples below, C:\Logs is used - ensure this folder (or whichever path you choose) exists on the target machine before running any command.



Verbose logging is strongly recommended for all deployments. All examples below include the **/l*v** flag. Store log files in a protected directory with restricted read access and delete them after confirming a successful deployment. To diagnose a failure, search the log for Return **value 3** – this marks the point where a custom action failed



Full msixec command reference: For a complete list of msixec switches and options, see the Microsoft documentation: <https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/msixec>

3.1 Silent Install (No UI)

`msixec /i "AlchemyPlayerSetup.msi" /quiet /l*v "C:\Logs\AlchemyPlayer_install.log"`

- /quiet = silent, no user interaction
- /l*v = verbose log (strongly recommended for all deployments)

3.2 Silent Install - Force Close Application Disabled (Default)

Default behavior. Installer will not force-close the application:

`msixec /i "AlchemyPlayerSetup.msi" /quiet /norestart /l*v "C:\Logs\install.log"`

3.3 Silent Install - Force Close Application Enabled

Forces Alchemy Player to close automatically before install proceeds. Use when deploying to machines where Player may be running:

`msixec /i "AlchemyPlayerSetup.msi" /quiet /forcerestart /l*v "C:\Logs\install.log"`

MSI Installation Instructions

3.4 Silent Upgrade (from any previous version)

No separate uninstall step needed. The MSI removes the old version and installs the new one in a single operation:

`msiexec /i "AlchemyPlayerSetup.msi" /quiet /!v "C:\Logs\AlchemyPlayer_upgrade.log"`

3.5 Silent Uninstall

The uninstall command accepts the PRESERVESETTINGS property to control whether existing configuration and data are retained or fully removed.

- **PRESERVESETTINGS=1 (Default)** – Preserves existing configuration, content data, and registry entries. Use this for standard uninstall/reinstall workflows to avoid losing customer settings.
- **PRESERVESETTINGS=0** – Performs a full removal of all Alchemy Player data, including C:\ProgramData\Alchemy Systems\ and all registry entries. Use this for decommissioning a machine or performing a clean-slate reinstall.

Scenario 1 – Uninstall preserving settings and data (recommended default):

`msiexec /x "AlchemyPlayerSetup.msi" /quiet PRESERVESETTINGS=1 /!v "C:\Logs\uninstall.log"`

Scenario 2 – Uninstall and remove all settings and data (clean removal):

`msiexec /x "AlchemyPlayerSetup.msi" /quiet PRESERVESETTINGS=0 /!v "C:\Logs\uninstall.log"`

Silent uninstall removes the application, and shortcuts.

C:\ProgramData\Alchemy Systems\ and registry entries are retained to preserve customer content. No leftovers remain in Program Files (x86)

Item	PRESERVESETTINGS=1 (Default)	PRESERVESETTINGS=0
Application files Program Files (x86)	Removed	Removed
Shortcuts	Removed	Removed
C:\ProgramData\Alchemy Systems\	Retained	Removed
Registry entries	Retained	Removed

MSI Installation Instructions

3.6 Custom Install Directory

`msiexec /i "AlchemyPlayerSetup.msi" /quiet INSTALLDIR="D:\Apps\AlchemyPlayer\" /!v "C:\Logs\install.log"`

In this example, INSTALLDIR will be created on install with the msiexec command

4. MSI Properties Reference

INSTALLDIR: Path | Default: C:\Program Files (x 86)\AlchemySystems\Player | Application installation directory

ALLUSERS: 1 | Default: 1 | System-wide install - do not change

PRESERVESETTINGS: 0 or 1 | Default: 1 | Preserve existing configuration on upgrade

REBOOT: Suppress / Force / Prompt | Default: Prompt | Controls reboot behavior

TRANSFORMS: .mst file path | Default: none | Apply a language transform (see Section 6)

Property	Value	Default	Description
INSTALLDIR	Path	C:\Program Files (x86)\Alchemy Systems\Player	Application installation directory
ALLUSERS	1	1	System-wide install - do not change
PRESERVESETTINGS	0 or 1	1	Preserve existing configuration on upgrade or uninstall
REBOOT	Suppress / Force / Prompt	Prompt	Controls reboot behavior
TRANSFORMS	.mst file path	none	Apply a language transform (see Section 11)

MSI Installation Instructions

5. Language Support - Transform Files (MST)

The default installation is English (US). Pass the corresponding MST file via TRANSFORMS to install a localized version.

LCID	Language	Transform File
1033	English (US) - Default	1033.mst
1031	German	1031.mst
1034	Spanish	1034.mst
1036	French	1036.mst
1046	Portuguese (Brazil)	1046.mst
1054	Thai	1054.mst
3084	French (Canada)	3084.mst

Example - French silent install:

```
msiexec /i "AlchemyPlayerSetup.msi" TRANSFORMS="AlchemyPlayer_1036.mst" /quiet /!v "C:\Logs\install.log"
```

Switching languages: The MSI language is set at install time via the TRANSFORMS property and cannot be changed without reinstalling. To switch a deployed machine to a different language, perform a silent uninstall (Section 9.5) followed by a fresh silent install specifying the desired TRANSFORMS value. Existing settings and data are preserved by default (PRESERVESETTINGS=1) during this process.

Frequently Asked Questions (FAQ's)

Q: Does the machine need .NET installed before deploying?

A: No. Alchemy Player 8.4.0 and later versions are self-contained and do not require a separate .NET installation. It installs on a clean machine with no prerequisites.

Q: Does it conflict with .NET 8.0+ already installed on the machine?

A: No. Validated on machines with .NET 8.0+ pre-installed. The self-contained model coexists cleanly with any existing .NET version.

Q: What if the machine already has a newer version installed?

A: The installer detects the installed version and blocks installation if a newer version is already present. The existing installation is untouched.

MSI Installation Instructions

Frequently Asked Questions (FAQ's) Continued

Q: What drivers are bundled with the installer? Do end users need to install drivers separately?

A: No. The following components are bundled and installed silently as part of the MSI – end users do not need to take any manual steps:

Driver details for security pre-approval: Vendor: Silicon Laboratories, Inc. | Hardware ID: SLABCOMM&10C4&EA60 | Certification: WHQL-signed

Component	Description
CP210x VCP Driver (x64)	USB-to-Serial for base station hardware (Silicon Labs) – installed silently
CP210x VCP Driver (x86)	32-bit variant of the above – installed silently
WDF Co-Installer 01009/01011	Windows Driver Framework co-installer (required for CP210x)
DriverInstaller.exe	The Player installer handles driver installation silently.
SQLite DB Browser	Companion diagnostic tool bundled in INSTALLDIR\Tools\ (see below)

Q: What is SQLite DB Browser and why is it installed?

A: Alchemy Player uses SQLite as its internal database to cache data locally (user info, enrolment, training records), which reduces continuous calls to the Alchemy Manager server. SQLite DB Browser is a companion viewer tool bundled in INSTALLDIR\Tools\ for support and diagnostic purposes only-it is not required for normal Alchemy Player operation.

Q: Should we configure antivirus exclusions for Alchemy Player?

A: In most environments no exclusions are needed. If real-time AV scanning causes performance issues (such as slow application launch or file access delays), consider excluding:

- C:\Program Files (x86)\Alchemy Systems\Player\
- C:\ProgramData\Alchemy Systems\Sistem\

MSI Installation Installer Instructions

Frequently Asked Questions (FAQ's) Continued

Q: What firewall rules are required for Alchemy Player?

A: Alchemy Player connects outbound to the Alchemy Manager server. Ensure firewall rules permit outbound HTTPS traffic from the installed machine to the Manager host. No inbound ports are required.

Q: Will the install prompt for a reboot?

A: Typically, no. Pass REBOOT=Suppress on the command line to guarantee no reboot during silent deployment.

Q: What happens to user content during an upgrade?

A: All content in C:\ProgramData\Alchemy Systems\Sistem\ is preserved. Configuration files are backed up and restored automatically.

Q: We use the Alchemy auto-update service. Do we need this MSI for Player upgrades?

A: For Player release upgrades, no. Systems configured to use the Alchemy Auto-Update Service receive Player updates automatically. However, the MSI can still be used for initial deployments or first-time installations.

Q: Is the MSI the same payload as the EXE installer?

A: Yes. Both are self-contained and install the same application. The MSI is provided for environments that require MSI-based deployment (SCCM, Intune, GPO).

Q: Can we pre-stage custom configuration files before deployment?

A: Yes. Place sistem_specific.xml and RF_CODES.txt in the installer's ClientConfig directory before running the MSI.

Q: Who do I contact with additional or specific questions?

A: Implementation Team: Contact your Alchemy Implementation Manager
Technical Support: Alchemy Client Services Help Desk 888.988.8832