EasyMorph Software Architecture & Security Overview

August 11, 2024

Table of Contents

Overview

EasyMorph is visual data preparation and automation software intended for internal use by business users in organizations with no or low IT support. EasyMorph includes the following products:

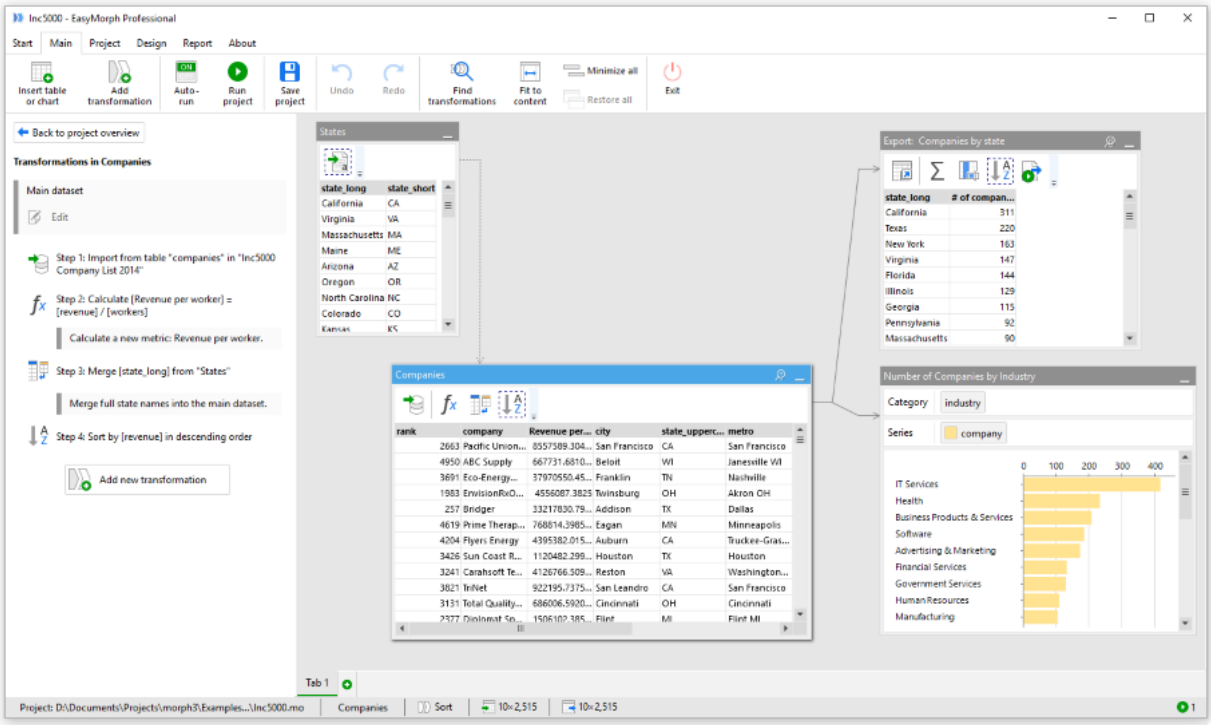
- EasyMorph Desktop, a Windows desktop application
- EasyMorph Server (optional), a Windows service with a web UI
- Command-Line Worker (a.k.a. CLW, also optional), a Windows command-line utility for workflow execution

The software is installed on premises and it doesn't require an internet connection or a cloud service to operate[1]. EasyMorph is entirely passive in the sense that it retrieves data from or sends to external systems only when explicitly instructed by a user to do so using an EasyMorph workflow. It doesn't send and receive data on its own.

The most typical deployments are Desktop-only, or Desktop + Server. The CLW utility is only used in integration scenarios when 3rd party applications need to execute EasyMorph workflows on demand from the command line. CLW is covered in a separate chapter in this document. From this point on, we will talk about Desktop + Server as the typical deployment scenario.

Workflows

The main purpose of EasyMorph is automation of data transformation and data flows. Business and IT users use EasyMorph Desktop to design automation workflows visually. The workflows can be executed either in EasyMorph Desktop, or on EasyMorph Server.



[1] EasyMorph can work in air-gapped networks.

Screenshot 1: A workflow in EasyMorph Desktop.

EasyMorph workflows can be very complex and may include advanced integration and execution capabilities such as querying databases and web APIs, execution of external applications or Excel VBA macros, loops, parameters, IF/THEN/ELSE conditions, or error handling. Therefore from a security standpoint they should be treated similarly to PowerShell or Python scripts. Note, however, that unlike with scripting, execution of arbitrary code from workflows can be disabled in EasyMorph Server settings.

Also, unlike scripts, EasyMorph projects (.morph files) are digitally signed which allows detecting unauthorized tampering (modification outside of EasyMorph). Execution of projects with broken digital signature is by default prohibited on EasyMorph Server.

Connectors and connection credentials

Connection settings and credentials required for connecting to external systems are stored in a connector repository that can be shared (i.e. used simultaneously) by many Desktop and Server users. The repository can be a local file, or can be served by the Server over a network to authenticated and authorized Desktop users for use in workflows. Server-hosted repositories are a more secure option.

In any case, every connector repository is encrypted with an industry-standard 2048-bit algorithm and can be additionally protected with read and write passwords.

Note that EasyMorph Desktop allows embedding connectors right in workflows. That makes the workflows portable (i.e. no external connector repository is required). However, embedded connectors have a weaker encryption and can be copied and used by anyone who can obtain the workflow with the embedded connector.

Both Desktop and Server have the same automation engine that executes the workflows. The engine contains all the drivers required for connecting to external systems such as databases (e.g. SQL Server), enterprise and cloud applications (e.g. SharePoint, Gmail), web APIs (e.g. REST APIs), and enterprise services (e.g. SFTP). The full list of available integrations is available here: https://easymorph.com/all-integrations.html

Software architecture and data flow diagram

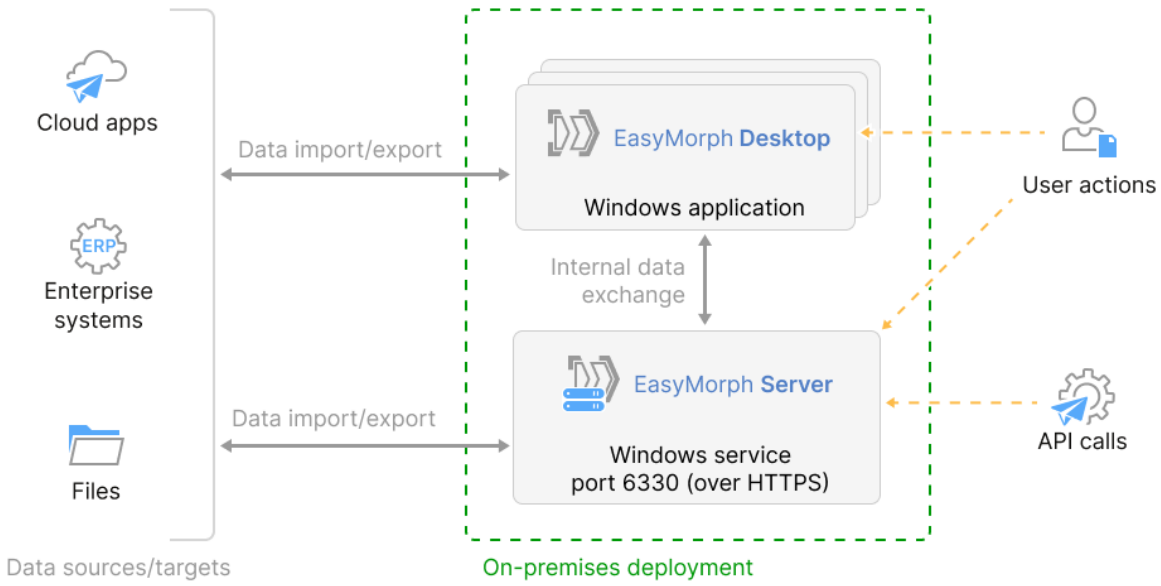


Figure 1: Server and Desktop deployment diagram.

Besides communicating with external systems, the Desktop and the Server can also communicate with each other. The Server can provide connector repositories to Desktops. Besides that, the Server sometimes can be used as a file store (although it's optional). In this case, files are stored in a designated folder of the Server and can be consumed by the workflows or by external applications via the Server API.

All communication with the Server is done via HTTP(S) through a single port (by default, 6330) that can be changed in Server settings.

Authentication

EasyMorph Desktop is a Windows desktop application so user authentication is done by Windows.

EasyMorph Server users access it via a web browser. They can be authenticated via Active Directory. AD groups are supported.

The Server has a recovery authentication option ("rescue login") that allows logging in as a Server administrator for any Windows user with local Windows administrator permissions on the Server's machine. The "rescue login" is typically used in the initial installation, or when the Server administrator locked him/her-self out by mistake.

Account permissions

Both Desktop and Server applications don't require a Windows account with elevated privileges (i.e. administrator account) to operate. Note, however, that the Server *installer* does require an administrator account.

The overall approach to permissions in based on two principles:

- "Don't trust, instead control" meaning that EasyMorph is highly controllable and **you can reliably control it with Windows permissions and firewall settings**,
- "Delegate permission enforcement and management to Windows whenever possible" meaning that EasyMorph mostly relies on Windows and Active Directory for user authentication and user access separation.

Data persistency and privacy

Generally speaking, EasyMorph doesn't store data unless it's explicitly instructed to save data as a local file. Data transformation is performed by EasyMorph's data engine entirely in-memory with no disk footprint.

However, EasyMorph stores the following user-generated information persistently:

- Connection settings required in workflows (in a connector repository)
- Debug logs (rotated every 14 days in Desktop, not rotated in Server)
- Server logs
- Server-based journal of user actions which may include metadata such as file names and parameter values

EasyMorph doesn't send telemetry to the vendor. All diagnostic information needed for troubleshooting is human-readable and can be reviewed (and redacted, if needed) before sending to the vendor. The vendor doesn't have access to EasyMorph installations remotely. The application has no backdoors.

Software updates

Software updates are installed manually, typically by rolling over a newer version using an installer downloaded from the vendor's website. The installer and the deployed software are signed with a code-signing certificate from a major CA.

Note, however, that EasyMorph doesn't do auto-updating as new versions sometimes might include breaking changes and thus require planned and supervised installation.

Source code scanning

In order to pass Google CASA (Cloud Application Security Assessment), EasyMorph's source code has been scanned using FluidAttacks scanner and assessed against OWASP. Besides that, the source code and its 3rd party dependencies are routinely scanned by GitHub's Dependabot.

Third-party software

EasyMorph has more than 50 integrations with enterprise and cloud applications and databases and extensively relies on 3rd party software (such as drivers, SDKs) provided by respective vendors. Besides that, EasyMorph uses a number of open-source .NET libraries such as SSH.NET. All 3rd party software used in the application is regularly updated. When there is a known vulnerability in 3rd party software used in EasyMorph, an unscheduled update is published for all EasyMorph products when the vulnerability is fixed by the 3rd party.

Self-signed SSL certificates

By default, self-signed SSL certificates are not permitted neither in Desktop nor in Server.

EasyMorph Server allows whitelisting thumbprints of self-signed SSL certificates.

In EasyMorph Desktop, self-signed SSL certificates can be either permitted without restrictions or whitelisted. If the Desktop is linked to the Server via its internal link (called Server Link), the policy regarding self-signed SSL is set by the Server also for Desktops, ignoring Desktop's own settings. Therefore, it's possible to prohibit or restrict using self-signed SSL certificates on Desktops from the Server.

If EasyMorph Server itself uses a self-signed SSL certificate, its thumbprint must be explicitly whitelisted in the Server Link settings of Desktops.

Threat vectors

Exposure to the public internet

While there are numerous security mechanisms and practices used by EasyMorph Server, nevertheless, it is intended for internal use and is not intended for exposure to the public internet.

Local Windows administrators

The security mechanisms of EasyMorph Server are built with the assumption that a malicious actor doesn't have elevated permissions (i.e., is not a local Windows administrator) on the machine where EasyMorph Server is installed and can't access the configuration files and binaries of the application or use the "rescue login".

SQL injection

EasyMorph itself is thoroughly protected on the application level against SQL injections. However, EasyMorph technically allows designing workflows that use SQL queries with inserted parameters. Such parameters are inserted as plain text in SQL queries. To reduce the risk of SQL injection via a user-entered parameter in low trust scenarios, enforce parameter validation rules in workflows to prohibit

parameter values outside of the accepted range/domain of values. Provide only the minimal required set of database permissions for connectors used in workflows.

EasyMorph Desktop

Description

EasyMorph Desktop is a Windows .NET application with a graphical user interface. The users can use EasyMorph Desktop for:

- Designing and execution of data preparation and automation workflows
- Querying databases, web APIs and retrieving information from external systems
- Data analysis
- Creating and configuring shared data connectors in an EasyMorph connector repository
- Publishing workflows to EasyMorph Server, opening workflows from EasyMorph Server
- Publishing datasets to EasyMorph Server, retrieving datasets from EasyMorph Server

Installation

The EasyMorph Desktop installer is downloaded and executed from the vendor's website. The installer and the application itself are signed with a code-signing certificate.

Installation is done into the userspace[2] and doesn't require a Windows account with elevated privileges (i.e. administrator account). Silent installation is possible.

The Desktop installer installs two applications:

- EasyMorph Desktop
- EasyMorph Launcher

The Launcher is a system tray utility complementary to EasyMorph Desktop. It's a Windows desktop application that runs under the same Windows account as EasyMorph Desktop and is used for scheduling and launching EasyMorph workflows under that account.

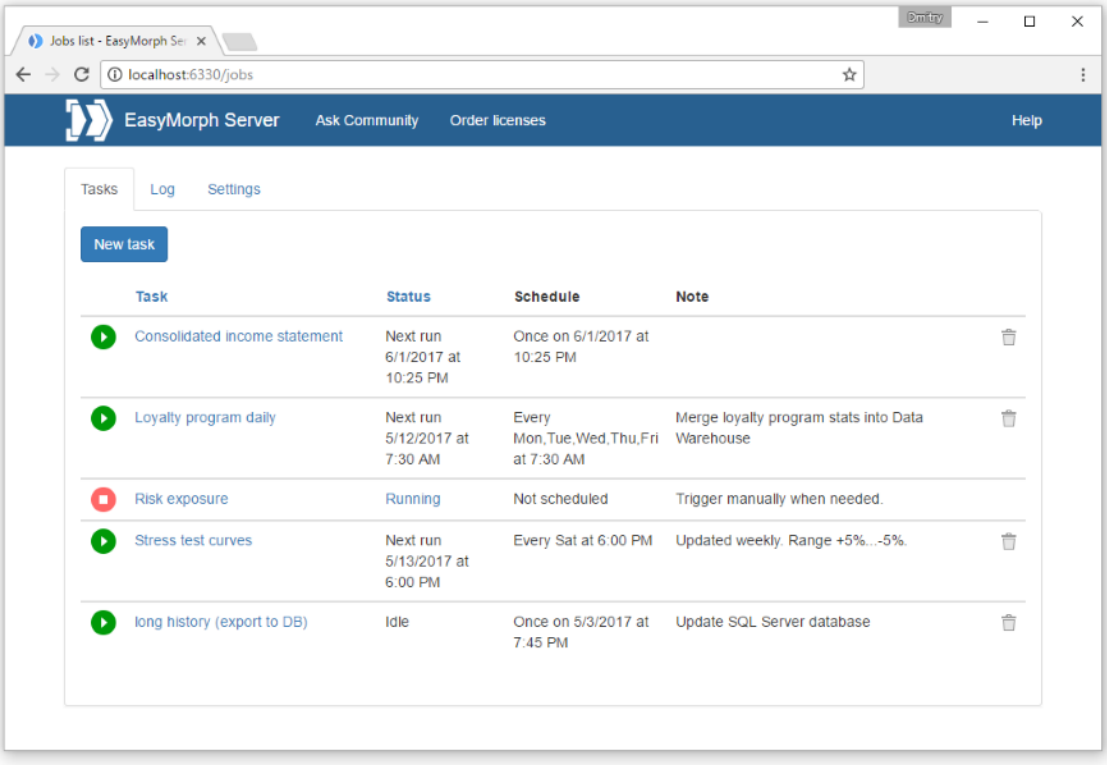
EasyMorph Server

Description

EasyMorph Server is a Windows ASP.NET Core application with a built-in web server. It is installed and operates as a Windows service with a web UI and is accessible via a browser. EasyMorph Server is used for the following:

[2] The installation folder is C:\Windows\Users\<user name>\AppData\Local\EasyMorph.

- Running EasyMorph workflows designed in EasyMorph Desktop
    - workflows can be triggered manually, on schedule, or by an API event
    - workflows can be executed under different Windows accounts
- Sending automatically email notifications about failed scheduled tasks
- Uploading and downloading files to/from the Server
- Serving connector repositories to Desktop users
- Providing centralized key-value storage for workflows executed in Desktops and Server
- Managing licenses of Desktop users
- User activity logging



Screenshot 2: EasyMorph Server tasks.

Web-server

The web server that is built in EasyMorph Server is HTTP.sys which the standard web-server for ASP.NET Core applications. HTTP.sys is mature technology that protects against many types of attacks and provides the robustness, security, and scalability of a full-featured web server. Microsoft IIS itself runs as an HTTP listener on top of HTTP.sys[3].

The web-server supports TLS 1.1 and TLS1.2. Earlier versions of TLS are forbidden.

[3] Cited from https://docs.microsoft.com/en-us/aspnet/core/fundamentals/servers/httpsys?view=aspnetcore-6.0

Installation

The EasyMorph Server installer is downloaded and executed from the vendor's website. The installer and the application itself are signed with a code-signing certificate.

The installer requires a Windows account with elevated privileges because it installs the EasyMorph Server service and configures the Windows firewall to enable outgoing HTTP connections for the service.

The installer installs the following applications:

- EasyMorph Server service (Morph.Server.WebConsole.exe) – the main service with a web-server
- EasyMorph Server Monitor (Morph.Server.Monitor.exe) – the Server configuration utility
- EasyMorph Server CLI (ems-cmd.exe) – an open-source cross-platform command-line utility for interaction with the Server API

The default account for the Server service is *NT Authority\Local Service*. This account can be changed in the Windows service settings.

EasyMorph Server Monitor is a Windows desktop application that is used to configure the basic settings of EasyMorph Server:

- Port (set to 6330 by default)
- SSL certificate (not configured by default)

SSL certificates that are installed using the Monitor are installed into the Windows certificate store.

The rest of the Server settings are configured using the Server's web UI (tab Settings). For more details about Server configuration see the "Server Administration Guide", a PDF document that comes with the Server installer.

Authentication

EasyMorph Server has three modes of user authentication

- Anonymous (default) – no authentication
- Shared password – only users who know the password can log in
    - Shared passwords are never stored in plain text or passed over network in plain text
- (*) Windows Identity – authentication via Active Directory

Points marked with (*) are available only in the Enterprise edition of EasyMorph Server.

By default, EasyMorph Server administrators can't log into the Server from the web UI. They must log in only through the Server Monitor, a GUI utility installed with EasyMorph, using the "rescue login" mechanism. The mechanism only permits logging in for local system administrators. Any member of the local Administrators group can use the "rescue login" mechanism in Server Monitor to log into the Server as administrator.

Workers

A worker in EasyMorph Server is a Windows process that can run workflows under a different Windows account than the Server service. EasyMorph Server can have multiple workers configured therefore workflows can be executed under multiple Windows accounts.

Each worker can have its own set of mapped network drives accessible only for workflows that are executed by that worker.

The Default worker

The Default worker is a special built-in worker. Unlike other workers it can't be deleted and it always runs under the same Windows account as the Server service. Therefore, workflows that run under the Default worker can technically access and interfere with the Server's system files.

While security-critical system configuration files of EasyMorph Server have anti-tampering mechanisms, it is still recommended to avoid using the Default worker by users with insufficient trust level.

Note, that some (typically, less expensive) editions of EasyMorph Server permit using only the Default worker. In that case, its use can't be avoided.

Security recommendations

Depending on your threat analysis and subject to your business requirements, you may want to perform the following actions to tighten EasyMorph security:

- (*) Use Active Directory for authenticating EasyMorph Server users
- Configure an SSL certificate for EasyMorph Server (using the Server Monitor utility)
- Switch Desktops to use Server-hosted connector repositories
- Enforce read and write passwords for connector repositories
- Don't use embedded connectors in workflows
- Disable execution of external applications in Server workflows if it's not required
- Disable Server features (e.g. File Manager or Pages) that are not required
    - If the File Manager feature of EasyMorph Server is required, disable uploading files unless it's necessary
- Log in as Server administrator only through the Server Monitor over an RDP session using "rescue login"
- Do not enable execution of EasyMorph projects with broken digital signature unless it's necessary
- If database queries in workflows use parameters, make sure the parameters have validation rules configured that prevent SQL injection
- Use dedicated Windows accounts in connector settings, instead of user accounts
- (*) Don't use the Default worker, run tasks under a worker that uses other Windows account than the Server service

- (*) Use an external database for the Server journal

Actions marked with (*) can be done only in the Enterprise edition of EasyMorph Server.

Command-Line Worker

Description

The Command-Line Worker (CLW) is a Windows console application that executes from the command line workflows designed in EasyMorph Desktop. It has the same automation engine as the Desktop or the Server and is predominantly used in integration scenarios when a 3rd party application needs to execute an EasyMorph workflow on demand from the command line.

Installation

The CLW installer is downloaded and executed from the vendor's website. The installer and the application itself are signed with a code-signing certificate.

The installer requires a Windows account with elevated privileges and installs CLW for all users of a Windows machine. It installs two applications:

- Command-line worker (CLW)
- Configuration utility – a Windows desktop application for configuring CLW settings

Contacts

Please post questions about EasyMorph architecture and security on the Community forum, or send by email to support@easymorph.com.