M-FILES CORPORATION

BEST PRACTICES FOR DATA SECURITY IN M-FILES

LAST UPDATED 28 OCTOBER 2024

VERSION 2.4

Contents

1. Introduction 3
2. High Availability 3
   2.1 Power Supply 3
   2.2 Network Availability 4
   2.3 Data Storage 4
   2.4 Database Server 4
      2.4.1 Firebird SQL 4
      2.4.2 Microsoft SQL Server 5
      2.4.3 Azure SQL Database 5
   2.5 M-Files Application Server 5
      2.5.1 Failover Solutions in M-Files Cloud 5
      2.5.2 Failover Solutions for On-Premises Deployments 5
3. Backup Policy 5
4. Data Encryption 6
   4.1 Encryption of Data in Transit 6
   4.2 Encryption of Data at Rest 7
5. Authentication 7
   5.1 Active Directory Integration 7
   5.2 Federated Authentication 7
   5.3 Multi-Factor Authentication 8
   5.4 Pre-Shared Key Authentication 8
6. Intrusion Detection 8
7. Malware and Virus Scanning 8
8. Data Loss and Leakage Prevention 9
9. Change History 9

1. Introduction

This document discusses different aspects that administrators must consider when designing and deploying high availability and data security features in an M-Files system.

The document focuses on these aspects of data security:

- High availability solutions
- Backup policy
- Encryption of data in transit
- Encryption of data at rest
- Authentication
- Intrusion detection
- Data loss / leakage prevention

2. High Availability

The server part of an M-Files system should be designed so that it recovers from the most common hardware and infrastructure issues, such as short power supply errors, network outages, hard disk issues and other server hardware or operating system issues.

As the cost for building a system with availability rate of 99.999% can be substantially higher than when aiming at 99.9% availability, business owners and IT should first agree with the accepted availability rate and whether or not planned and scheduled downtime is acceptable or not. As the table below shows, 99.999% availability rate allows only 25 second downtime per month.

AVAILABILITY	ALLOWED DOWNTIME PER MONTH
90%	72 hours
99%	7 hours
99.9%	43 minutes
99.99%	4 minutes
99.999%	25 seconds

High availability can be often achieved by eliminating any single points of failure in the system and by implementing rapid fault detection, isolation, and resolution. Notice that the M-Files software is only a small piece in this picture. This section gives overall guidance for designing an M-Files system for high availability.

2.1 Power Supply

To ensure that outages in power supply do not cause downtime for the server, M-Files application and database servers should at minimum have redundant power units and short power outages should be protected against with uninterruptable power supply (UPS) devices.

M-Files Cloud Vault servers are hosted in Microsoft Azure datacenters. These datacenters are additionally protected against power supply interruptions by deploying two independent power lines to the datacenters.

2.2 Network Availability

It is recommended to install two network cards on the application and database server to make sure that the system access is not interrupted if one of the network cards fails on those servers. Additionally, it is recommended to design other network topology such that single points of failure are eliminated.

M-Files Cloud Vault servers are hosted in Microsoft Azure datacenters. The network in these datacenters does not contain single points of failure. Additionally, numerous technologies are used to proactively monitor latency and load in the network.

2.3 Data Storage

At minimum, permanent data on M-Files servers should be stored on a redundant disk array (e.g. RAID, SAN). Implementing these technologies ensures that data is at least duplicated on multiple drives and faulty drives can be replaced on the fly without interrupting the service.

While SAN technology protects well against hardware failures, administrators should also consider additional methods to protect against water, fire, and other natural hazards. Typically, this kind of protection is implemented by taking regular backups of the system and by storing the backup media off-site. The downside of this protection is that recovering the system by restoring the backups may take a long time, but considering the likelihood of these natural hazards, this might still be a sufficient plan for many organizations.

For faster recovery, administrators should consider replicating the vaults to another geographical location. In case of a disaster, clients can connect to the replica site. Refer to Replication and Archiving User's Guide in M-Files Knowledge Base for more information.

2.4 Database Server

The M-Files database server can run on the M-Files application server or on a separate computer.

Three different database engines are supported:

- Firebird SQL: embedded to the M-Files Server service. Always runs on the application server.
- Microsoft SQL Server: can run on the application server or on a separate server.
- Azure Database: Database-as-a-Service (DaaS) leveraged in M-Files Cloud Vault. Runs always on a different computer instance than the application server.

2.4.1 Firebird SQL

The embedded database engine for M-Files Server (Firebird SQL) does not offer high availability features, such as replication and automatic failover.

2.4.2 Microsoft SQL Server

Microsoft SQL Server offers numerous technologies to improve the availability of the databases. For high availability of the database, we recommend using AlwaysOn Availability Groups or AlwaysOn Failover Cluster Instances (FCI) in SQL Server. You can check the supported SQL Server versions in the user guide.

These two solutions serve different purposes: availability groups can be used to improve availability of databases, whereas FCI nodes are used to improve the availability of the SQL Server instances. For detailed information of these services, please refer to the following MSDN article: https://msdn.microsoft.com/en-us/library/ff929171.aspx.

2.4.3 Azure SQL Database

Azure SQL Database keeps three replicas of each database (one primary and two secondary) and uses a quorum-based commit scheme where data is written to the primary and one secondary replica before the transaction is considered committed.

If any component fails on the primary replica, the service detects the failure and fails over to the secondary replica. In case of a physical loss of the replica, the service creates a new replica automatically. There are always at least two replicas of each database that have transactional consistency in the data center.

Other than the loss of an entire data center all other failures are mitigated by the service. M-Files Cloud Vault databases can be replicated to another Azure data center or to an on-premises location to improve the recovery time in case of the loss of an entire data center.

The replication, failure detection and failover mechanisms of Azure Database are fully automated and operate without human intervention. This architecture is designed to ensure that committed data is never lost and that data durability takes precedence over all else.

2.5 M-Files Application Server

2.5.1 Failover Solutions in M-Files Cloud

M-Files Cloud uses Multi-Server Mode to make sure that your M-Files vaults are always available.

2.5.2 Failover Solutions for On-Premises Deployments

Refer to these documents for a description of and instructions for deploying the Multi-Server Mode feature:

- Multi-Server Mode - Functionality Overview
- Multi-Server Mode - Administrator Guide
- Multi-Server Mode - Configuration Guide

3. Backup Policy

Administrators should take regular backups of the M-Files system. A good backup policy consists of the following:

- Regular full backups of the M-Files master database and all vault databases
- Backing up M-Files customizations that are not included in the vault and master database backups

- Storing backup files off-site and storing copies of at least 14 previous backup files to have more restoration points available
- Storing quarterly backup files for at least a year
- Testing restoration of backups regularly

The details of the recommended backup plan depend on the criticality of the system. M-Files Backup Policy lists best practices for backing up on-premises deployments.

The backup policy of the cloud service depends on the service plan you have subscribed to. **Business Continuity and Disaster Recovery for M-Files Cloud** is not published online but it can be delivered to customers upon request.

4. Data Encryption

4.1 Encryption of Data in Transit

"Data in transit" refers to information that flows over the network. For an M-Files system, this typically means the network communication between M-Files clients and M-Files Server. Three different client types need to be considered:

- M-Files Desktop
- M-Files Web
- M-Files Mobile (apps for iOS and Android)

If all communication between the clients and M-Files Server is within the organization's private network, unencrypted network communication between the client applications and M-Files Server may be acceptable. However, even in this case the organization should consider the risks of users within the organization potentially being able to gain unauthorized access to content by means of network sniffing, for example.

If users access data from outside the organization's private network, encrypting the network communication is usually mandatory. This is typically achieved either by using the HTTPS protocol or VPN technology.

> **Note**: With the latest M-Files versions, we strongly encourage you to use gRPC instead of HTTPS. Refer to Setting Up M-Files to Use gRPC for more information.

The best practice for ensuring that data in transit does not cause a security risk is to **encrypt the network communication between M-Files clients and M-Files Server in all cases**, regardless of if the information flows over the public Internet or in the organization's internal network.

Protecting Data in Transit with Encryption in M-Files document in M-Files Knowledge Base describes the recommended ways for protecting data in transit between M-Files clients and M-Files Server with encryption.

4.2 Encryption of Data at Rest

Encryption of inactive data (data at rest) on M-Files Server is strongly recommended to mitigate the risk of unauthorized access to data. You can encrypt file data and database data (metadata) at rest. The file data is encrypted with the built-in features in M-Files Admin. You can also protect data files in the M-Files Desktop cache from viewing. For more information, refer to Protecting File Data at Rest with Encryption.

The database data can be encrypted using the Transparent Data Encryption feature in Microsoft SQL Server Enterprise edition. For more information, refer to Encryption of M-Files Vault Database with Transparent Data Encryption.

5. Authentication

Deploying strict password policies, such as password complexity and expiration rules is an easy and recommended way to improve data security in M-Files. You can further strengthen data security of the authentication process with Active Directory integration, federated authentication, multi-factor authentication, and pre-shared keys.

5.1 Active Directory Integration

In on-premises deployments, it is often easiest to use Windows Active Directory as an identity provider for M-Files. Active Directory provides IT administrators a centralized way to disable user logins from multiple systems. Active Directory also supports password complexity and expiration policies.

When using Active Directory integration, M-Files Server communicates with Active Directory using the LDAP protocol. The integration supports synchronizing Windows logins and Active Directory security groups between Active Directory and M-Files and the users are authenticated against M-Files Server via the Kerberos protocol.

5.2 Federated Authentication

Federated authentication is a recommended way to implement user management especially when using M-Files in the cloud.

The main benefits for end-users are the possibility to use the same credentials in multiple services and also to use single sign-on in M-Files. Use of federated authentication improves data security, because users do not need to create and memorize unique passwords for each service they use.

The main benefit for IT administration is centralized management of logins. IT does not need to disable logins separately for every service a leaving employee was using or create new logins to every service that a new employee will be using. Most identity providers also provide robust capabilities to enforce password complexity rules, password expiration rules, and more. Some IdPs also integrate well with Windows Active Directory.

From cloud security perspective, federated authentication is great because the authentication can be completed without submitting the username and password pair to the cloud service provider. If an M-Files customer uses, for example, Microsoft Entra ID for identity management, M-Files Server never sees and therefore cannot store the usernames and passwords of users. Instead, credential checking occurs between the client and the IdP.

In the cloud, it is often recommended to use OAuth 2.0 compatible identity providers, such as Microsoft Active Directory Federation Services, Microsoft Entra ID, or PingFederate.

M-Files is compatible with OAuth 2.0. Refer to the Configuring OAuth 2.0 for M-Files authentication document for details.

5.3 Multi-Factor Authentication

Multi-factor authentication refers to a method where a user must successfully present authentication factors from at least two of the three categories in order to log in:

- Knowledge factors ("things only the user knows"). For example, passwords.
- Possession factors ("things only the user has"). For example, ATM cards.
- Inherence factors ("things only the user is"). For example, biometrics.

A common way to implement multi-factor authentication in M-Files is to require two-step verification in the login process. You can, for instance, require user first to authenticate against the identity provider with username and password. Upon successful first step verification, identity provider sends an email or SMS token with a single-use code to the user. User must then enter this code to the second authentication window to log in to M-Files.

Many identity providers that support multi-factor authentication can be integrated with M-Files via OAuth 2.0.

5.4 Pre-Shared Key Authentication

M-Files Cloud environments and on-premises environments with gRPC connections do not support pre-shared key authentication. Instead of pre-shared keys, we recommend you to use other methods for secure authentication. For example, Microsoft Entra ID or Okta.

6. Intrusion Detection

It is possible to detect unauthorized login attempts to M-Files Server. Windows Server logs failed login attempts to M-Files Server when using Windows accounts in M-Files. These login attempts can be monitored with the out-of-the-box tools provided by Windows Server and administrators can be notified in case there are too many failed login attempts within a short period of time. Implementing this kind of intrusion detection system is recommended, especially if users can attempt to login to M-Files Server via public networks.

7. Malware and Virus Scanning

Administrators should protect endpoints from malware using appropriate anti-virus and anti-malware software to prevent damage if users open malicious files that are stored in M-Files.

Notice though that, the best practice is to exclude the virtual M-Files drive (typically the M drive) from virus and real-time scanning. If the Anti-virus software supports excluding certain processes from real-time scanning, then it is recommended to exclude MFClient.exe process from real-time scanning. Excluding this process should not compromise the security since MFClient.exe does not really execute files in M-Files. If you have other processes like

WINWORD.exe and EXCEL.exe enabled for rea-time scanning, malicious files should be detected upon opening them.

On server-side, M-Files Server supports antimalware checks on Windows 2016 and later. Please refer to the user guide for more information.

8. Data Loss and Leakage Prevention

Data loss means losing sensitive or public data permanently. M-Files provides comprehensive data loss prevention mechanisms out-of-the-box. M-Files automatically saves changes to content as a new version, and older versions can be accessed via version history. This mitigates the risk of losing data by overwriting data in documents and objects. Moreover, the delete rights can be enabled or disabled explicitly object-by-object to mitigate the risk of users adversely on unintentionally deleting content from the vault. Standard users cannot permanently destroy data from M-Files: even if they have the delete rights to content, administrators can undelete these objects without having to restore backups. Administrators can be also notified of deleted objects.

To prevent data loss, it is important to follow the high availability and backup policy recommendations in sections 2 and 3, and to enforce strict enough access control and password policies in M-Files system. It is also important not to allow standard users to destroy objects in M-Files.

Data leakage may not mean that the data is lost permanently, but the lost information is sensitive, and it is acquired by an unauthorized party. To mitigate this risk, it is recommended to follow the best practices for encryption, discussed in section 4, and to implement strong authentication mechanisms and intrusion detection systems discussed in sections 5 and 6.

M-Files also provides a comprehensive feature set to mitigate the risk of data ex-filtration. The patented metadata-driven access control lists of M-Files can automatically enforce appropriate permissions to data based on its metadata and/or workflow state and/or classification. M-Files Server also logs information on logins to the repository and can prevent certain users from logging into the vault at abnormal times. By implementing access control policies with compatible third-party tools, Windows users can also be prohibited from logging in to the system from abnormal locations.

9. Change History

This table contains a list of the changes made to this document.

VERSION	ESSENTIAL CHANGES
1.0	Initial version.
1.1	Failover solutions for on-premises deployments were clarified.
1.2	Documentation updated to match M-Files 2015.1 features.
1.3	Added details about database encryption at rest.
1.4	Added details about different authentication options.
1.5	Updated section 2.5.

1.6	Updated section 2.5.
1.7	Added section 7.
1.8	Updated section 2.5.
1.9	Removed references to SAML. Updated the name of one referenced document. Added a reference to the System Requirements page of the user guide to check the supported SQL versions. Removed the Windows mobile application from section 4.1.
2.0	Added a remark about client cache data protection to section 4.2.
2.1	Cross-references in section 8 corrected. Minor modifications throughout the document.
2.2	Added links to Multi-Server Mode documentation to section 2.5.2. Footer updated and some layout changes made throughout the document.
2.3	Unified the content in sections 4.1 and 4.2 with the referenced documents.
2.4	Updated information about pre-shared keys and failover solutions.