FORTIFIED INSIGHT

FORTIFIED WISdom

WORKLOAD INTELLIGENCE SOLUTION

Installation Requirements

WISdom Overview

WISdom's data collection is an agentless service that allows WISdom to collect system, metadata, and runtime data from each system. The data is secured during the transfer and at rest.

The WISdom service runs locally on virtual machines (VM) to securely monitor and collect data from the servers under management. The WISdom application will add and update the list of servers to be monitored and act as the ingestion point for your environment's database statistics. Ingested data, along with configuration data, will be transferred to and from the local VMs using a dedicated S3 bucket per client.

Collection Server Sizing

Depending on the number of servers or devices being managed by WISdom and the location of the servers, there will be a requirement to provision a VM to support the data collection and upload per datacenter or geographic location. The sizing of the one or more VMs should follow these guidelines:

1-50 Managed servers	50-200 Managed servers	201+ Managed servers
4 Processors	8 Processors*	12 Processors*
8 - 16 GB RAM	16 GB RAM	32 GB RAM
50 GB Storage	75 GB Storage	100 GB Storage

WISdom FAQ's

The WISdom services were designed with high security along with minimal to no performance impact in mind. Below is a list of security protocols and features that are followed or implemented for the WISdom software and services:

- **Data Collection** – The data collected by WISdom consists of metadata, runtime and configuration data and leverages multiple low impact technologies to capture the required data. No sensitive data is transferred to Fortified WISdom environment.
- **User Data** – WISdom was designed to minimize the resource footprint for data collection. The data collection is intelligent and only captures data based on the need.
- **Upload Destination** – Each client has a dedicated S3 bucket which is created during the initial setup that only the client can access. This S3 bucket with dedicated AWS credentials is used to upload and temporarily store data before it is processed by WISdom.

- **Data** – Data is uploaded to the dedicated AWS S3 bucket using TLS and is encrypted at rest using server-side encryption.
- **Access to Encrypted Data** – The configuration updates are sent using the AWS S3 bucket and are encrypted. Client and Fortified are the only participants that can access the data and encryption keys.

WISdom Prerequisites

The WISdom services require a machine with the following components and port access to support the data collection:

- Windows Management Framework (WMF) 5.1
- .NET Framework 4.7.2 or higher
- Access to all the servers managed by WISdom on the following ports:
- SQL Server Port (usually 1433, but may vary)
- SQL Browser if named instances are used – Port 1434
- WMI - Port 135, 49154 *(may vary)*
- Performance counters – Port 445
- Outbound firewall rules must allow access to AWS S3 over HTTPS.

Monitoring Service Account

The WISdom monitoring service is a Windows service that runs with a Windows domain account.

The account will require the “Log on as a batch job” privilege, as it will be used to execute the scheduled tasks required for executing the WISdom components.

Monitoring Virtual or OnPrem Machines and SQL

Windows Monitoring

The Data Collection Account is used to collect the windows metrics (WMI). The preferred account type for collection is an AD account. An alternative solution is using a local Windows account.

The account must be a member of:

- Local Administrator

An alternative solution to granting Local Administrator rights is to grant permissions explicitly on WMI and DCOM. *

The account must be a member of:

- Remote Management Users
- Distributed COM Users

** Note: when not using Local Administrator, group policy updates and Windows patching may remove the explicitly granted WMI and/or DCOM permissions resulting in a lack of monitoring until permissions have been rectified.*

SQL Instance Monitoring

To collect data from a SQL Instance, the Monitoring Service Account, a different Windows account, or a SQL account may be used for collection. If a SQL Account is used and a Windows account does not have access to the Host, then the Host Server WMI data will not be collected. The permissions required by the account are:

For SQL versions prior to 2022:

- sysadmin privileges

For SQL Server 2022 and above, must be a member of both:

- ##MS_ServerStateReader## role
- ##MS_DefinitionReader## role**

***Note: WISdom does not support the use of a secondary Windows account for Host collection and a separate account for SQL collections. When configuring a secondary account for SQL Instance collections, either the service account or the SQL Collection account must perform the Host collections.*

Monitoring Azure

The account that will connect to an Azure connection may be the Monitoring Service Account, a different Windows account, an Azure AD account, or a SQL account. For Managed Instances and Azure SQL Databases, Host metrics will not be accessible or possibly very limited.

Azure Managed Instance

The permissions required by the data collection account are:

- Sysadmin role

Azure SQL Database

The data collection login account will need to be created in the Master database, as well as each SQL Database that will be monitored.

The account permissions required:

- View Database State permissions

The account roles required:

- ##MS_DatabaseConnector## role
- ##MS_ServerStateReader## role
- ##MS_DefinitionReader## role

Monitoring Amazon Instances

ECS Instances

The requisite account permissions are the same as section– Monitoring Virtual or OnPrem Machines and SQL.

RDS Instances

Monitoring an RDS instance requires a SQL authentication account for use with the data collection. The SQL account must be created on each database and requires membership in specific roles and permissions to be granted on Master, MSDB, and all user databases that will be monitored.

The Role required in the Master Database:

- ProcessAdmin

The permissions required for the Master database:

- View Server State
- View Any Definition
- Create and View Any Database
- Alter Trace

Role for MSDB Database:

- SQLAgentUserRole

Permission on objects in the MSDB Database:

- Execute Permissions on:
    - rds_backup_database
    - rds_restore_database
    - rds_task_status
    - rds_cancel_task
- Select Permissions on:
    - sysJobs
    - sysJobHistory
    - sysJobActivity

Permission on every user database:

- ShowPlan