FT Analytics™ GuardianAI™

Quick Installation Guide

Version 1.03.00

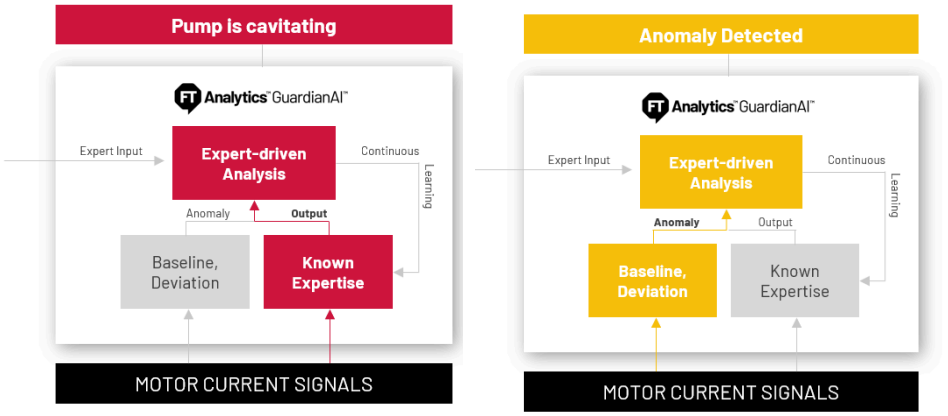
Rockwell Automation

Original Instructions

Contents

FactoryTalk Analytics GuardianAI

FactoryTalk® Analytics™ GuardianAI™ (hereafter called as GuardianAI) is a machine learning based supervisory application that uses existing plant devices, such as variable-frequency drives, as sensors to monitor the health of components such as pumps, fans, and blowers on a plant floor. It uses device data to establish a baseline signature of each component's behavior under normal operating conditions. Then, it monitors the components for any deviation from baseline. Once a deviation is detected, a notification is sent to the user identifying the anomaly. If an anomaly is detected but cannot be identified, GuardianAI notifies the maintenance engineer that an unidentified anomaly was detected. The engineer can then investigate the issue, determine the cause of the anomaly, and tag the deviation accordingly. The GuardianAI machine learning engine then trains to identify the new anomaly for future encounters. The following diagram illustrates this process and the variation between a known fault and an unknown deviation.



The GuardianAI workflow takes a no-code approach to machine learning. As a result, a data scientist is not required to configure, deploy, or use this AI application. It is designed so that OT personnel, such as maintenance engineers, controls engineers, machine operators, and plant managers, can work with GuardianAI with minimal training required.

The configuration workflow consists of four steps.

1. Deploy the GuardianAI application on a local Virtual Machine.
2. Add the device that will act as a sensor.
3. Provide the identifying information about the component being monitored (pump, fan, blower, or motor).
4. Start training the model to establish the baseline.

GuardianAI supports monitoring multiple baselines based on a component's state. Components such as pumps, fans, and blowers on a plant floor can operate under different conditions and processes. GuardianAI provides the capability to monitor these varying situations.

NOTE: State refers to a component's specific operating condition, such as different pressures, flow rates, speed, etc.

GuardianAI provides premium integration with the controller to seamlessly read changes in the component's tags selected during configuration. When a component's tag value in the controller changes, GuardianAI automatically creates a new state and starts acquiring the baseline for that specific state. Once the baseline is acquired, GuardianAI continuously monitors the component for any deviation in that particular state. GuardianAI will

automatically switch to the corresponding state based on the change in the controller tags to ensure accurate monitoring.

For example, if a pump operates under different pressures or flow rates, GuardianAI will create and monitor baselines for each condition and switch to the appropriate state when the component's operating condition changes to ensure accurate monitoring.

GuardianAI provides premier integration with PowerFlex® 755, 755TL, 755HiHP, 755TR, 755TM, 755TS, and 6000T drives to use as sensors to access three-phase current data for motor current signature analysis. It will focus on anomaly detection and identification for the following component types: pumps, fans, and blowers. The application is designed to work with single-drive and motor applications. Given its adaptive nature, GuardianAI can learn process-centric issues and adapt to component types beyond those listed above. For this use case, the application comes equipped with generic motor control analytics.

Manual Container Deployment

This section provides information about deploying GuardianAI manually using container.

Prerequisites

- Minimum Hardware Specifications:
  - Processor: Intel Atom 4 core
  - RAM: 8 GB
  - SSD: 120 GB
- Host Machine Operating System:
  - Linux Debian 11, 12 (or) Ubuntu 22
- Client Computer
  - Operating System: Microsoft Windows
  - Web Browser: Google Chrome or Microsoft Edge
- Ensure the Linux instance has internet access to enable the automated script to download dependent packages.
- The user must have sudo access to execute system commands. For more information on adding a user to the sudoers list, refer to Add a user to the sudoers list on page 12.

Retrieve and Deploy the FactoryTalk Analytics GuardianAI Application

Perform the following steps to download the GuardianAI application:

1. Download the container package from the Product Compatibility and Download Center (PCDC) at rok.auto/pcdc.

   The download file is named 'guardianai-1.3.0.zip'. The ZIP file includes the following:
   - **Build Label** - V1.03.00
   - **guardianai-install.sh** - This installation script is used to deploy GuardianAI application.
   - **guardianai-uninstall.sh** - This script is required in case you want to uninstall GuardianAI application.
   - **guardianai-reset-password.sh** - This script is used to reset the login password for the GuardianAI.
   - **guardianai.tar** - This file contains application files and all other dependent scripts required to run the GuardianAI application.
   - **guardianai-redis.tar** - This file contains the Redis database used as the data storage layer for the GuardianAI application.
   - **guardianai-backup.tar** - This file contains the dependent scripts required to back up the GuardianAI application.
   - **node-device-migration.sh**: This file contains the scripts required to update the GuardianAI application from the previous version.
2. Move the content to the target Linux device.

Tip: Use **WinSCP** or equivalent tool to transfer **guardianai-1.3.0.zip** to the Linux machine.

Install FactoryTalk Analytics GuardianAI Application

Perform the following steps to install the GuardianAI application:

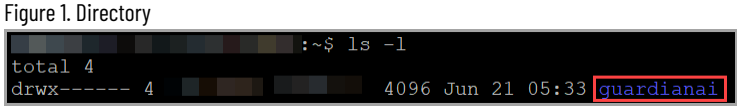
1. Execute the following command to unzip the **guardianai-1.3.0.zip** file.

```
unzip guardianai-1.3.0.zip
```

2. Execute the following command to create a directory with name as **guardianai** in **/home/<user>**.

```
mkdir guardianai
```

Figure 1. Directory



3. Move the following files to the **guardianai** directory in the Linux machine:
   - guardianai-install.sh
   - guardianai-uninstall.sh
   - guardianai-redis.tar
   - guardianai.tar
   - guardianai-backup.tar
   - guardianai-reset-password.sh
4. Execute the following commands from **/home/<user>** directory to modify the permission to the installation scripts and enable execution:

```
chown -R <user>:<user> guardianai
```

```
chmod 755 ./guardianai/*.sh
```

5. Execute the following command to install the GuardianAI application.

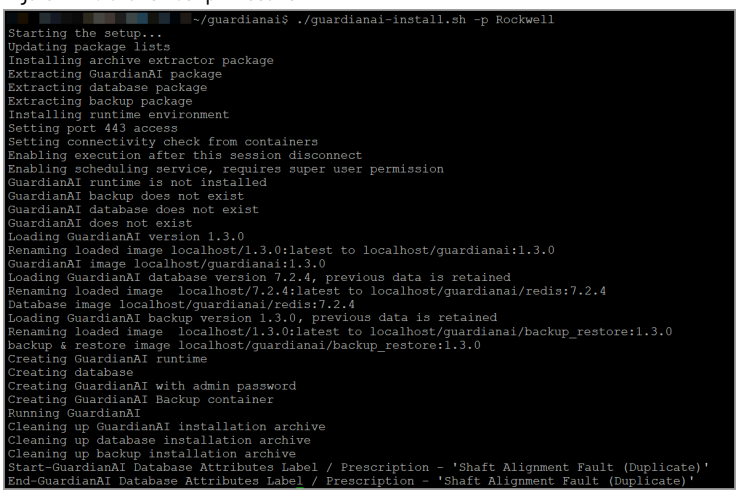
Tip:
- The installation script takes the login password as an argument.
- Add the GuardianAI application login password to the command before execution.

```
./guardianai-install.sh -p <GuardianAI Login Password>
```

NOTE: The user will be prompted to enter the sudo password.

Once the script is executed successfully, the user will see the following:

Figure 2. Installation Script Execution



After successful installation, GuardianAI automatically backs up the application data using the default schedule. For more information, refer to Backup on page 8.

Access FactoryTalk Analytics GuardianAI Application

GuardianAI has a self-signed certificate. The user can access the application using Google Chrome or Microsoft Edge.

In a web browser, use the following URL to access GuardianAI:

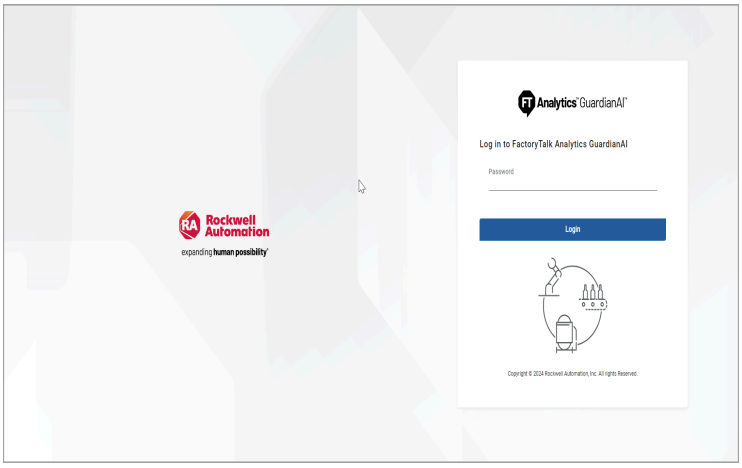
https://<hostname>

Tip: The hostname refers to the instance where GuardianAI is installed. Execute the following command to get the instance hostname:

```
hostname -f
```

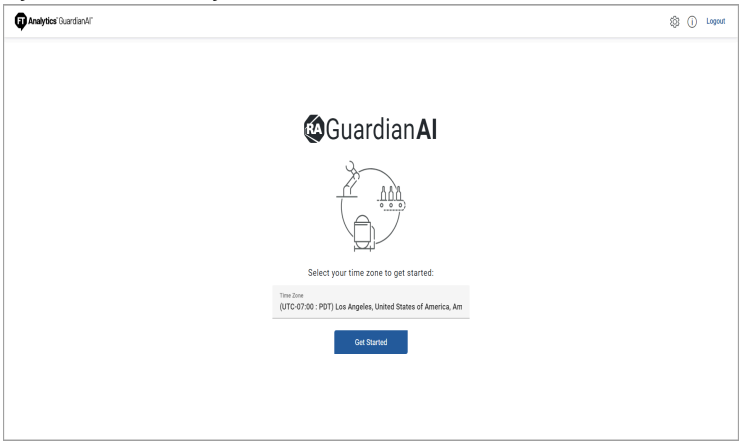
The first access point for GuardianAI will be the login page. The login password is the argument set while installing the application and running the shell script; please refer to step 5 in Install FactoryTalk Analytics GuardianAI Application on page 6.

Figure 3. FactoryTalk Analytics GuardianAI Login Page



After login, the user will be prompted to select the time zone for GuardianAI. It is recommended that the time zone be set to the locale of the compute surface where GuardianAI is installed. The application uses the time zone to record timestamps of deviation and failure risk events encountered when monitoring the components. The user can modify the time zone on the settings page.

Figure 4. Time Zone Configuration



Backup

The GuardianAI application automatically backs up/archives the application data to a volume attached to the **guardianai_backup_restore** container once a day by default. This ensures regular and consistent data backup without any manual intervention.

The backup file is saved with the following name format: **dump-YYYY-MM-DD-hh-mm-ss.guardian**.

Each backup file is considered as one restore point. Each restore point contains the full application data.

Users can change the frequency schedule of backup as required. Perform the following steps to change the backup schedule:

1. Execute the following command to change the backup schedule:

```
./guardianai-install.sh -p <GuardianAI Login Password> -b "*/5 * * * *"
```

The installation script takes the CRON TIME argument, which will be used to change the application's backup schedule. The above command will re-install the GuardianAI application, and the data remains intact in the process.

NOTE: The user will be prompted to enter the sudo password.

In the above command, 5 minutes is set as a backup schedule. Every 5 minutes, the GuardianAI application automatically backs up the application data.

Restore

Users can restore the application to a specific restore point in the event of any abnormal situations. Perform the following steps to restore the application to a specific restore point:

1. Execute the following command to navigate to the backup and restore container:

```
podman exec -it guardianai_backup_restore sh
```

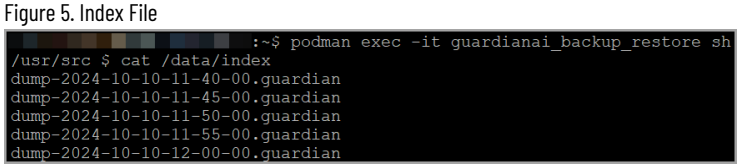
2. Execute the following command to check the restore points in the index file:

```
cat /data/index
```

The index file contains the list of available restore points. Choose the required restore point and copy the name of the required restore point.

NOTE: The index file is available only when at least one restore point is available.

Figure 5. Index File



3. Execute the following command to restore the application to a specific restore point:

```
riot -h $REDIS_HOST -p $REDIS_PORT dump-import -t JSON -z /data/<restore point>
```

- **restore point**: Name of the restore point.

Backup Files Count

A maximum of 30 backup files (restore points) can be stored in **guardianai_backup_restore** container by default. The maximum count can be changed as required.

Perform the following steps to change the maximum backup file count:

1. Execute the following command to change the maximum backup file count:

```
./guardianai-install.sh -p <GuardianAI Login Password> -c <backupcount>
```

The installation script takes the backupcount argument, which will be used to change the maximum backup count. The above command will re-install the GuardianAI application, and the data will remain intact.

NOTE: The user will be prompted to enter the sudo password.

Uninstall FactoryTalk Analytics GuardianAI

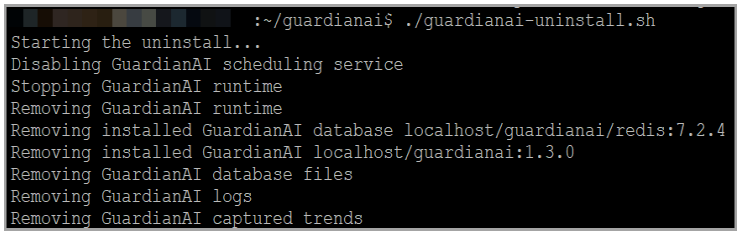
Perform the following steps to uninstall GuardianAI.

1. Execute the following command to uninstall the GuardianAI application.

```
./guardianai-uninstall.sh
```

Once the script is executed successfully, the user will see the following.

Figure 6. Uninstall Script Execution



NOTE: The uninstall script will delete all the data captured by GuardianAI and fully remove the containerized application.

Reset FactoryTalk Analytics GuardianAI Application Login Password

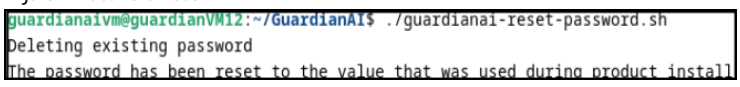
In the event a user forgets the login password for the GuardianAI application, the following script can be used to reset the password. The login password will be reset to the original password configured during application installation (refer to step 2 below to query the original password).

1. Execute the following command to reset the password.

```
./guardianai-reset-password.sh
```

The script will reset the password to the original value set during the GuardianAI installation.

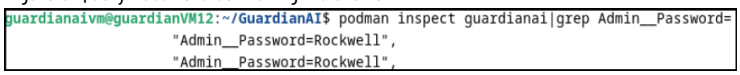
Figure 7. Password Reset



2. To query the original password that is set during the installation, execute the following command:

```
podman inspect guardianai|grep Admin__Password=
```

Figure 8. Query Password Set During Installation



Users can now log in using the original password set during installation.

3. To change the password, navigate to the settings in GuardianAI.

Upgrade FactoryTalk Analytics GuardianAI Application

This section provides step-by-step instructions for upgrading the GuardianAI application to V1.3.0. The installation script streamlines the installation and seamlessly upgrades the GuardianAI application from the previous version while keeping the data intact.

Perform the following steps to upgrade the GuardianAI application:

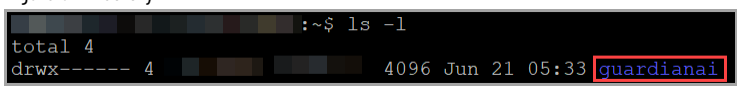
1. Execute the following command to unzip the **guardianai-1.3.0.zip** file.

```
unzip guardianai-1.3.0.zip
```

2. Execute the following command to create a directory with name as **guardianai** in **/home/<user>**.

```
mkdir guardianai
```

Figure 9. Directory



3. Move the following files to the **guardianai** directory in the Linux machine:
   - guardianai-install.sh
   - guardianai-uninstall.sh
   - guardianai-redis.tar
   - guardianai.tar
   - guardianai-backup.tar
   - guardianai-reset-password.sh
   - node-device-migration.sh
4. Execute the following commands from **/home/<user>** directory to modify the permission to the installation scripts and enable execution:

```
chown -R <user>:<user> guardianai
```

```
chmod 755 ./guardianai/*.sh
```

5. Execute the following command to upgrade the GuardianAI application:

Tip:
- The installation script takes the login password and the migration version number as arguments.
- Add the GuardianAI application login password to the command before execution.
- The -m argument invokes the "node-device-migration.sh" script to complete the upgrade process.

```
./guardianai-install.sh -p <GuardianAI Login Password> -m 1.3
```

NOTE: The user will be prompted to enter the sudo password.

The script installs all the necessary dependencies to upgrade the GuardianAI application to V1.3.0.
Once the script is executed successfully, the user will see the following example image:

Figure 10. Upgrade GuardianAI

NOTE: Contact Rockwell support for any issues during the upgrade.

Add a user to the sudoers list

Perform the following steps to add a user to the sudoers list:

NOTE: A root user must execute this process.

1. To switch to the root user, complete the following steps:
   a. Execute the following command:

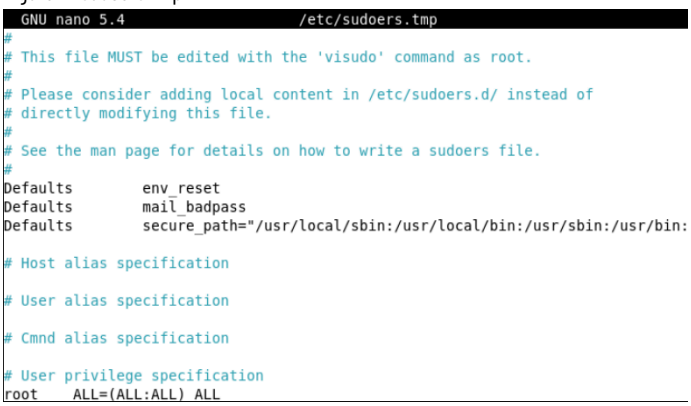
      ```
      sudo su
      ```

   b. Provide the root password.
2. To add a user to the sudoers list, complete the following steps:
   a. Execute the following command:

      ```
      sudo visudo
      ```

The visudo command ensures that you do not make syntax errors that can lock you out of the system.

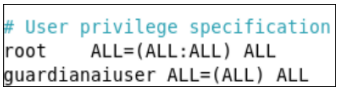
Figure 11. sudoers.tmp



b. Use the Down arrow key and scroll down to the line below `root ALL=(ALL:ALL) ALL`.

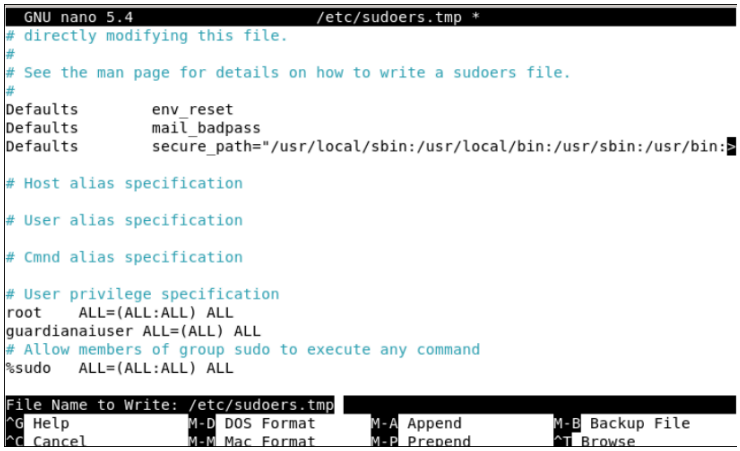
c. Add `<Username> ALL=(ALL) ALL`.

Figure 12. Add User



3. To exit, press **CTRL + Y**.
4. To save, type the letter **Y**.

Figure 13. Save Sudoers List



5. To write to **sudoers.tmp** file, press **Enter** again.
6. Execute the following command to exit from the root and switch back to the user:

```
exit
```

Figure 14. Exit

This page has been intentionally left blank