kaspersky

Kaspersky IoT Secure Gateway 100

Contents

About Kaspersky IoT Secure Gateway 100

Kaspersky IoT Secure Gateway 100 is a software/hardware system based on the Siemens™ SIMATIC™ IoT2040 industrial computer with KasperskyOS and application software installed. Kaspersky IoT Secure Gateway 100 is designed to work as a secure gateway for the Industrial Internet of Things in an enterprise network.

This document describes only the software portion of Kaspersky IoT Secure Gateway 100.

For information about the technical specifications of the Siemens SIMATIC IoT2040 computer, please refer to the manufacturer's website. For information on using the Siemens SIMATIC IoT2040 device, please refer to the User Guide.

Kaspersky IoT Secure Gateway 100 performs the following functions:

- Uses the OPC UA protocol to receive data from equipment residing in the internal enterprise network.
- Distributes and forwards the received data to the Siemens MindSphere® cloud platform.
- Ensures the cybersecurity of enterprise equipment and supports the transmission of encrypted data.

Installation and initial configuration of Kaspersky IoT Secure Gateway 100 software is performed by experts of Aprotech or its trusted partners.

Kaspersky IoT Secure Gateway 100 is started when the Siemens SIMATIC IoT2040 device is connected to the network. Kaspersky IoT Secure Gateway 100 is stopped when the device is disconnected from the network.

Distribution kit

The distribution kit for Kaspersky IoT Secure Gateway 100 includes the following components:

- Siemens SIMATIC IoT2040 industrial computer with the manufacturer-supplied configuration.
- User Guide for the Siemens SIMATIC IoT2040 industrial computer.
- SD card with Kaspersky IoT Secure Gateway 100 preinstalled. SD card inserted into the SD slot of the Siemens SIMATIC IoT2040 computer.
- UART cable.
- File containing third party code information (legal_notices.txt) provided on the SD card.
- Online Help Guide for the software portion of Kaspersky IoT Secure Gateway 100.
- Version information for the software portion of Kaspersky IoT Secure Gateway 100 (Release Notes).

Hardware and software requirements

Kaspersky IoT Secure Gateway 100 works only on a Siemens SIMATIC IoT2040 industrial computer.

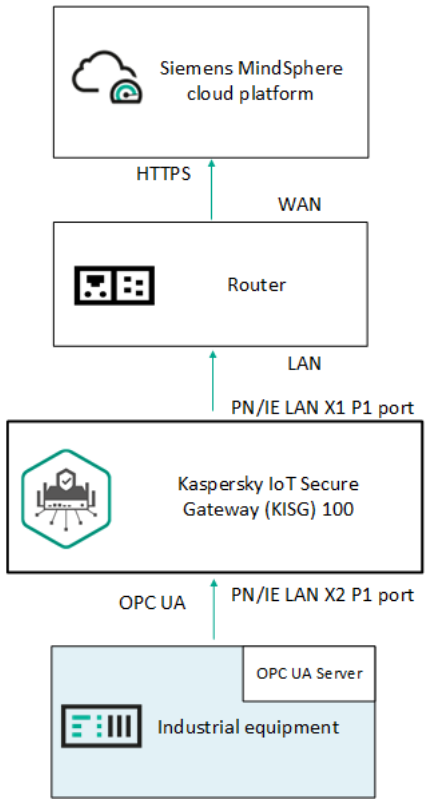
To configure data transfer to the Siemens MindSphere cloud platform, you must first obtain access to Siemens MindSphere. Please refer to the Siemens MindSphere documentation for information about the ways to gain access to and configure the Siemens MindSphere platform, and about the hardware and software requirements of the computer that will be used to connect to the cloud platform.

To receive Kaspersky IoT Secure Gateway 100 data from enterprise industrial equipment over the OPC UA protocol, the data forwarding settings must be configured on the enterprise equipment. You can read about the OPC UA protocol specification on the developer's website. The current version of Kaspersky IoT Secure Gateway 100 supports OPC UA protocol version 1.04.

Standard deployment scenario

The standard deployment scenario for Kaspersky IoT Secure Gateway 100 (see the figure below) involves the following:

- Data from enterprise industrial equipment is sent to Kaspersky IoT Secure Gateway 100 via the OPC UA protocol.
- Kaspersky IoT Secure Gateway 100 receives the IoT data and forwards this data to the Siemens MindSphere cloud platform.



Standard deployment of Kaspersky IoT Secure Gateway 100

Security objectives and constraints

A *cyberimmune information system is* a system that guarantees the fulfillment of specific security objectives in all possible scenarios of system usage as stipulated by the developers.

One prerequisite when developing a cyberimmune information system is to identify its security objectives and the security constraints under which the system will operate.

Security objectives are the particular requirements imposed on a cyberimmune information system that must be fulfilled to ensure that the system operates securely in any possible usage scenario with consideration of the necessary security constraints.

Security constraints are the additional restrictions placed upon the system operating conditions that either simplify or complicate the fulfillment of security objectives.

Security objectives

Kaspersky IoT Secure Gateway 100 has the following security objectives:

- Kaspersky IoT Secure Gateway 100 ensures secure, unidirectional transfer of data from industrial equipment residing within an internal enterprise network to the Siemens MindSphere cloud platform while eliminating the possibility of the cloud system having any impact on internal resources of the enterprise.
- Kaspersky IoT Secure Gateway 100 ensures the integrity of data transmitted to the Siemens MindSphere cloud platform.

The following are not security objectives of Kaspersky IoT Secure Gateway 100:

- Accessibility of Kaspersky IoT Secure Gateway 100
- Confidentiality of data transmitted from Kaspersky IoT Secure Gateway 100 to the cloud platform

Security constraints

Kaspersky IoT Secure Gateway 100 has the following security constraints:

- Kaspersky IoT Secure Gateway 100 can receive data from equipment residing within an internal enterprise network only over the OPC UA protocol.
- Physical access to Kaspersky IoT Secure Gateway 100 is restricted by organizational measures implemented by the specific enterprise (room access and equipment access regulations) to prevent unauthorized access to Kaspersky IoT Secure Gateway 100.
- Kaspersky IoT Secure Gateway 100 does not have internal administration resources. The software portion of Kaspersky IoT Secure Gateway 100 and configuration files are stored on an extractable SD card that can be accessed only by the administrator.
- While Kaspersky IoT Secure Gateway 100 is running, the settings, certificates and encryption keys stored on the SD card are read-only.
- A medium level of threat (basic elevated) from the external network is assumed.
- A low level of threat (basic) from the internal network is assumed.

For more detailed information on assessing the information security threat level, please refer to the website of the relevant government agency with jurisdiction over technical and export regulations.

Kaspersky IoT Secure Gateway 100 cannot guarantee the integrity of data transmitted within the internal network from equipment to Kaspersky IoT Secure Gateway 100.

Kaspersky IoT Secure Gateway 100 cannot ensure that devices connected to Kaspersky IoT Secure Gateway 100 will be protected against attacks launched from within the internal network.

Threats associated with a vulnerability of the hardware platform are not considered.

The following threats associated with breached availability of the infrastructure are not considered:

- Communication channels between the sides of network interaction
- Siemens MindSphere cloud platform

Licensing

The terms of use of the application are set forth in the End User License Agreement or in a similar document regulating usage of the application.

Data provision

Kaspersky IoT Secure Gateway 100 does not collect, use, or process personal user data.

Administration

This section provides instructions on configuring the Kaspersky IoT Secure Gateway 100 settings for transmitting data received from industrial facilities in the internal enterprise network to the Siemens MindSphere cloud platform.

To perform this configuration, you must extract the SD card from the SD slot of the Siemens SIMATIC IoT2040 device, insert the SD card into the workstation, perform the actions described in this section, then insert the SD card back into its original slot. Please note that the TGW-HW-BOOT section on the SD card is formatted with the FAT32 file system. All other sections are formatted with the ext3 file system.

To correctly transmit data from enterprise equipment to the Siemens MindSphere cloud platform via Kaspersky IoT Secure Gateway 100, the following is required:

1. Generate certificates for OPC UA protocol clients in case an encrypted connection and/or user authorization is used.

2. At the industrial enterprise facility (equipment), configure the nodes used to transmit data over the OPC UA protocol.

   You can read about the OPC UA protocol specification on the developer's website. The current version of Kaspersky IoT Secure Gateway 100 supports OPC UA protocol version 1.04.

3. Obtain access to the Siemens MindSphere cloud platform and configure the data points for receiving data from the MindConnect Lib agent.

   For detailed information on the ways to obtain access to the Siemens MindSphere platform and configure data points, please refer to the Siemens MindSphere documentation.

4. Configure the Kaspersky IoT Secure Gateway 100 settings for transmitting data received from industrial facilities in the internal enterprise network to the Siemens MindSphere cloud platform.

Configuring network settings

To connect a device to an external and internal network, you must establish a physical connection (using an RJ45 cable) to the network and configure the external and internal network settings if necessary. By default, Kaspersky IoT Secure Gateway 100 is delivered with a dynamic configuration for the internal and external network.

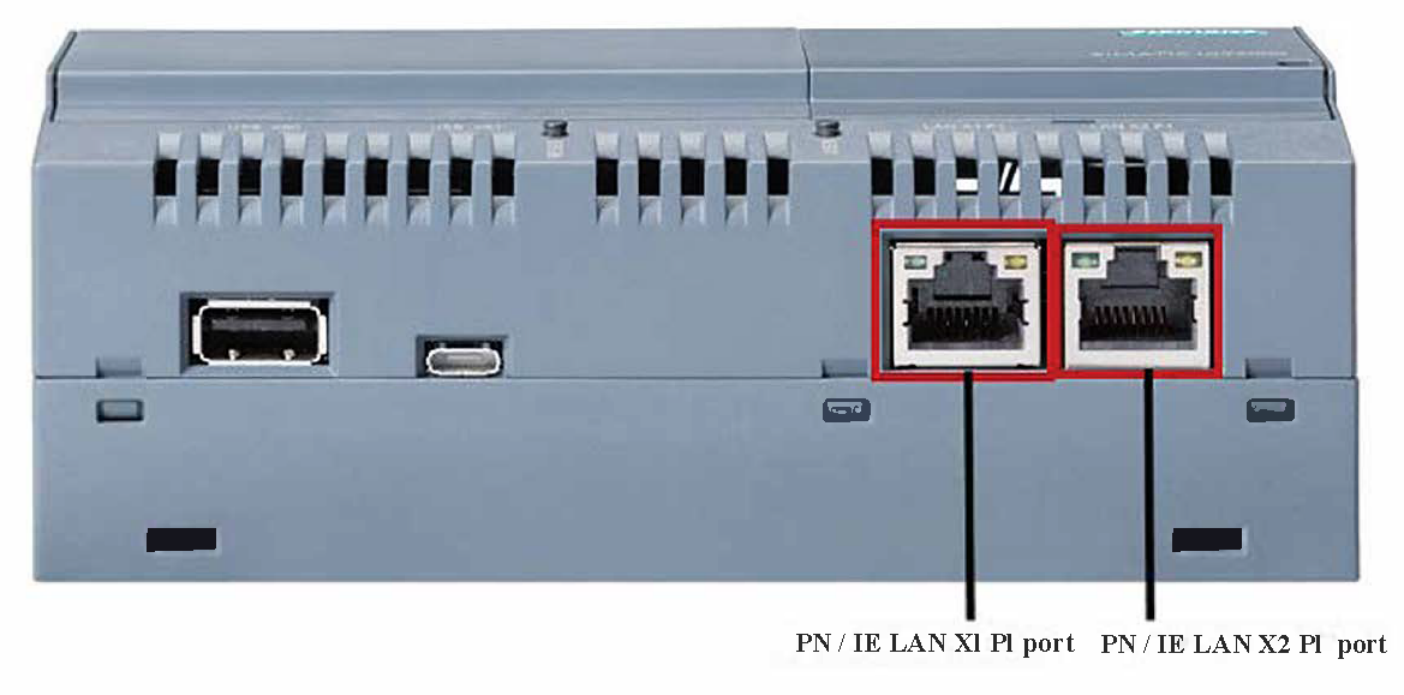
> Dynamic configuration of network settings requires the availability of a configured DHCP server residing in the same network.

To connect Kaspersky IoT Secure Gateway 100 to a data transfer network, you can use the following Ethernet ports on the Siemens SIMATIC IoT2040 device:

- PN/IE LAN X1 P1 port for access to the *external* network to transfer data received from industrial facilities to the Siemens MindSphere cloud platform.

- PN/IE LAN X2 P1 port for access to the *internal* enterprise network to receive data from industrial facilities over the OPC UA protocol.

The layout of Ethernet ports on the Siemens SIMATIC IoT2040 device is presented in the figure below.



Layout of Ethernet ports on the Siemens SIMATIC IoT2040 device

External network settings are configured in the dhcpcd.conf configuration file, which is located in the folder /etc in the TGW-HW-**ENW** section of the SD card.

Internal network settings are configured in the dhcpcd.conf configuration file, which is located in the folder /etc in the TGW-HW-**INW** section of the SD card.

The SD card is included in the distribution kit.

To configure the settings of the external or internal network, use the dhcpcd.conf file to specify the necessary settings according to the documentation and save your changes.

The settings defined in the dhcpcd.conf file will be applied the next time Kaspersky IoT Secure Gateway 100 is started.

Configuring data acquisition using the OPC UA protocol

Kaspersky IoT Secure Gateway 100 receives data from equipment residing within the internal enterprise network over the OPC UA protocol, which is described by the OPC Unified Architecture specification. You can read about the OPC UA protocol specification on the developer's website. Kaspersky IoT Secure Gateway 100 supports OPC UA protocol version 1.04.

The `None` security profile in the Kaspersky IoT Secure Gateway 100 settings is the security profile that is most compatible with various types of industrial equipment for OPC UA connections.

When generating certificates for a connection between a client (Kaspersky IoT Secure Gateway 100) and the OPC UA server, make sure that the certificates comply with the following settings:

- The settings of keys and certificates are compliant with the selected security policy.
- DER or PEM format is used for certificates and client keys.

- For the client certificate, the Subject Alt Name field contains the value URI:urn:aprotech:KISG100:OpcUaClient.

Kaspersky IoT Secure Gateway 100 uses the following folders to store certificates and keys for a connection with an OPC UA server:

- /app/Core/pki/private/transfer/opc_ua/client/ – folder in the TGW-HW-IDS section of the SD card for storing OPC UA client keys.
- /app/Core/pki/certs/transfer/opc_ua/client/ – folder in the TGW-HW-IDS section of the SD card for storing certificates of the OPC UA client and OPC UA server.

You can configure the settings for receiving data from monitored objects over the OPC UA protocol in the OpcUaClientSettings-0.json configuration file.

To configure data acquisition using the OPC UA protocol:

1. Create an OpcUaClientSettings-0.json configuration file and put it in the folder /app/Core/config/transfer/opc_ua/client in the TGW-HW-IDS section of the SD card.

   All of the actions described next are performed within the OpcUaClientSettings-0.json file.

2. To correctly route data from industrial equipment to the MindSphere repository, specify the ID and name of the OPC UA client:

   a. In the mandatory `id` parameter, define the ID of the OPC UA client that will receive data from the OPC UA server (industrial facility). For example, `"id": 0`.

   b. In the mandatory `name` parameter, define the name of the OPC UA client that will receive data from the OPC UA server (industrial facility). For example, `"name": "Kaspersky IoT Secure Gateway 100 OPC UA Client"`.

3. To make it easier to read the configuration, you can use the optional `description` parameter to enter a description of the OPC UA client that will receive data from the OPC UA server (industrial facility). For example, `"description": "Collect data from CNC by Kaspersky IoT Secure Gateway 100"`.

4. To connect an OPC UA client to equipment, specify the OPC UA server address and port in the mandatory `url` parameter. For example, `"url": "opc.tcp://192.168.177.7:4840"`.

5. Specify a time interval (in seconds) in the `readingCycle` parameter to define how frequently the gateway will read data. For example, `"readingCycle": 1`.

6. Configure the security settings in the mandatory `security` settings block:

   a. In the `mode` parameter, indicate the security management mode for the connection of the client application that is being used on your OPC UA server. The following security management modes are available:

      - `Sign` means that the connection requires a digital signature for data.
      - `SignAndEncrypt` means that the connection requires both a digital signature and data encryption.
      - `None` means that the connection does not require a digital signature or data encryption. It is not recommended to use this mode because it does not ensure a secure connection between the OPC UA client and the OPC UA server.

- `Any` means that the connection will use any of the listed modes that are supported by the server: `Sign`, `SignAndEncrypt`, `None`.

b. In the `policy` field, specify the name of the security profile that is being used on your OPC UA server. The following security profile options are available:

- `Basic128Rsa15`.
- `Basic256`.
- `Basic256Sha256`.
- `None`.
- `Any` means that any of the listed policies can be used (if supported by the server): `Basic128Rsa15`, `Basic256`, `Basic256Sha256`, `None`.

c. For secure communication over OPC UA, you must create a private key and certificate and put them into the client and server configuration. To set up a secure connection over OPC UA, define the following settings in the `clientPkiData` settings block:

- In the `certificate` field, specify the name of the certificate file for the OPC UA client. For example, `"certificate": "client.crt"`.
- In the `privateKey` field, specify the name of the private key file for the OPC UA client certificate. For example, `"privateKey": "client.key"`.

> The `clientPkiData` settings block must be completed even if the `None` value is set for the `mode` and `policy` fields.

d. In the `trustList` field, specify the array that contains the names of trusted certificate files. For example, `"trustList": ["server.crt"]`. If the OPC UA server configuration prescribes the use of a custom trusted list, add the client certificate to the list of trusted certificates of the server. If certificate verification is not required, indicate the `AllowAll` value for this parameter.

> If you do not need to complete the `mode`, `policy` and `clientPkiData` settings blocks, define the `null` value for the `security` settings block. The security mode will be set to `None` in this case.

7. For OPC UA client authentication on the OPC UA server, provide the user account credentials for the connection in the mandatory `userCredentials` settings block:

a. In the `username` field, enter the name of the user account for authorization on the OPC UA server.

b. In the `password` field, enter the password of the user account for authorization on the OPC UA server.

> If you want to allow anonymous connection of the OPC UA client to the OPC UA server, define the `null` value in the `userCredentials` block. In this case, you do not need to fill in the `username` and `password` fields.

8. If you want to configure periodic forwarding of the health signal (heartbeat) of Kaspersky IoT Secure Gateway 100 to the Siemens MindSphere cloud platform, do the following:

a. Make sure that the Siemens MindSphere cloud platform has a data point configured for implementing the heartbeat transmission function.

b. In the optional `heartbeat` settings block, define the following settings:

- Enter the data node ID in the `id` field. For example, `"id": 0`.
- Specify the data node name in the `name` field. For example, `"name": "Heartbeat"`.
- In the `timeout` field, specify the time period (in seconds) between the generation of heartbeat signals. For example, `"timeout": 60`. This field is optional. The default value for the time period between generated signals is 30 seconds.

If you skip configuration of periodic heartbeat signal transmission or define `"heartbeat": null`, no heartbeat signals will be transmitted.

9. In the mandatory `nodes` settings block, specify the following parameters for each data node:

a. Enter the data node ID in the `id` field.

b. Specify the data node name in the `name` field.

c. In the `nodeId` settings block, provide the following data:

1. ID of the OPC UA server namespace in the `ns` (namespace index) field
2. ID of the data node in the OPC UA server namespace The following options are available:

- `s` (string identifier) – string value for the data node ID. For example, `"nodeId": "ns=1;s=Variable temperature"`.
- `i` (numeric) – numerical value for the data node ID. For example, `"nodeId": "ns=2;i=2045"`.

10. Save the changes in the OpcUaClientSettings-0.json file.

The settings defined in the OpcUaClientSettings-0.json file will be applied the next time Kaspersky IoT Secure Gateway 100 is started.

Kaspersky IoT Secure Gateway 100 will receive data from industrial facilities within the internal enterprise network via the protocol that is described by the OPC Unified Architecture specification.

Special considerations for the configuration of data acquisition using the OPC UA protocol

Kaspersky IoT Secure Gateway 100 does not establish a connection in the following cases:

- The server does not have a certificate, and an unsafe connection is not allowed.
- The `trustList` parameter lacks a server certificate, and the `AllowAll` value is not set.
- The client certificate, server certificate or keys do not comply with the settings of the selected security policy.

The OPC UA server and client establish an unsafe connection in the following cases:

- The `null` value is set for the `security` and `userCredentials` settings blocks (and the server supports this type of connection).
- The `Any` value is set for the `mode` and `policy` fields (and the server offers the choice for an unsafe connection).

Any weakening of the security settings reduces the security of the connection. For example, the following settings reduce the security of a connection over the OPC UA protocol:

- Use of the `null` value for the `security` settings block will result in the use of a connection without encryption and without a signature.
- Use of the `AllowAll` value for the `trustList` field disables server certificate verification.
- Use of the `null` value for the `userCredentials` settings block disables the capability to connect to a server by using a login and password.
- Use of the `Basic128Rsa15` or `Basic256` values for the `policy` field of the OPC UA v.1.4 protocol specification is considered to be obsolete because the SHA-1 hashing algorithm is no longer considered to be secure.
- Use of the `None` value for the `policy` or `mode` fields will result in the following:
  - Use of a connection without encryption and without a data signature
  - Transmission of a plaintext password to the server.
- Use of the `Any` value for the `policy` or `mode` fields may result in the use of an unencrypted connection without a signature if this option is offered by the server as the priority option.

Configuring transmission of data from Kaspersky IoT Secure Gateway 100 to Siemens MindSphere

Kaspersky IoT Secure Gateway 100 receives data from industrial facilities residing within the internal enterprise network and forwards that data to the Siemens MindSphere cloud platform.

Siemens MindSphere is a cloud platform that was developed by the company Siemens to receive and analyze industrial data from the Internet of Things (IoT). The Siemens MindSphere cloud platform saves and analyzes all types of industrial data received from industrial facilities. You can use this information to optimize industrial processes.

The Siemens MindSphere component known as MindConnect Lib agent is used to transfer data from Kaspersky IoT Secure Gateway 100 to the Siemens MindSphere cloud platform.

Kaspersky IoT Secure Gateway 100 uses the following folders to store the information necessary for transmitting data to the Siemens MindSphere cloud platform:

- /app/Core/pki/certs/transfer/mind_sphere/agent/ – folder in the TGW-HW-IDS section of the SD card for storing the certificate from the Siemens MindSphere Certificate Authority. The certificate file in this folder must be named `mindsphere.io`.

- /app/Core/config/transfer/mind_sphere/agent/ – folder in the TGW-HW-IDS section of the SD card for storing files containing the settings for connecting to the Siemens MindSphere cloud platform.

- /app/Core/data/transfer/mind_sphere/credentials/ – folder in the TGW-HW-EDS section of the SD card for storing files containing registration data for gaining access to the Siemens MindSphere cloud platform.

You can use the MindSphereAgentSettings-0.json configuration file to configure the settings for transmitting data from Kaspersky IoT Secure Gateway 100 to the Siemens MindSphere cloud platform.

Prior to configuring these settings, you must obtain the Siemens MindSphere registration data by using the tools of the MindConnect LIB plugin. For detailed information on receiving registration data using the tools of the MindConnect LIB plugin, please refer to the MindSphere documentation. Below is an example of MindConnect LIB registration data.

Example of MindConnect LIB registration data:


```
{
 "content": {
 "baseUrl": "https://southgate.eu1.mindsphere.io",
 "iat":
"eyJraWQiOiJrZXktaWQtMSIsInR5cCI6IkpXVCIsImFsZyI6IlJTMjU2In0.eyJpc3MiOiJTQ0kiLCJzdWIiOiI
C781RRabMRrNpPcOZxucv4n9jIpIZjUx9owGNXT0g-zYb8HYjB13HSvOBZW2_wmPLthxYEFlHU1dqi8ThPtcNE0C
 "clientCredentialProfile": [
 "SHARED_SECRET"
 ],
 "clientId": "3a990ec12fd94a8e81f5f11df8a634d9",
 "tenant": "aprotech"
 },
 "expiration": "2019-08-12T13:00:15.000Z"
 }
```


To configure the settings for transmitting data from Kaspersky IoT Secure Gateway 100 to Siemens MindSphere:

1. Create a MindSphereAgentSettings-0.json configuration file and put it in the folder /app/Core/config/transfer/mind_sphere/agent in the TGW-HW-IDS section of the SD card.

   All of the actions described next are performed within the MindSphereAgentSettings-0.json file.

2. To correctly route data from industrial equipment to the MindSphere repository, specify the ID and name of the MindSphere agent that will be used to transmit data to the cloud:

   a. In the mandatory `id` parameter, specify the ID of the MindConnect LIB agent. For example, `"id": 0`.

   b. In the mandatory `name` parameter, specify the name of the MindConnect LIB agent. For example, `"name": "Kaspersky IoT Secure Gateway 100 MindSphere Agent"`.

3. If necessary, you can provide a description of this client in the optional `description` parameter for convenient accounting and to make it easier to read the configuration. For example, `"description": "Transfer data to MindSphere by Kaspersky IoT Secure Gateway 100"`.

4. In the mandatory `boardingConfiguration` settings block, enter the registration data that was received using the tools of the MindConnect LIB plugin. An example of MindConnect LIB registration data is provided below.

5. If you need to use a proxy server to transmit data from the MindSphere agent to the cloud, enter the following data in the optional `proxySettings` settings block:

   a. In the `type` field, specify the connection type as HTTP: `"type": "HTTP"`.

b. In the `host` field, specify the IP address of the proxy server that will be used for the connection. For example, `"host": "192.168.188.1"`.

c. In the `port` field, specify the port of the proxy server that will be used for the connection. For example, `"port": 3128`.

> The `type`, `host` and `port` fields must be completed if the `proxySettings` block is not empty.

6. To configure custom settings for grouping by timestamp, provide the following data in the optional `limits` settings block:

   a. In the `maxStorageSize` field, specify the maximum number of items that will be stored in the ring buffer of Kaspersky IoT Secure Gateway 100. For example, `"maxStorageSize": 90000`. The default value is 90000, and the minimum value is 1.

   b. In the `itemGroupTimeout` field, specify the timeout (in seconds) of data items with the same timestamp. For example, `"itemGroupTimeout": 5`. The default value is 5, and the minimum value is 0.

   c. In the `maxTimeseriesSize` field, specify the maximum number of data items in one time series. For example, `"maxTimeseriesSize": 64`. The default value is 64, and the minimum value is 1.

   d. In the `maxHttpPayloadSize` field, specify the maximum size of an HTTP request (in bytes) sent to MindConnect Lib. For example, `"maxHttpPayloadSize": 16384`. The default value is 16384 Kb, the minimum value is 400, and the maximum value is 10485760.

> If you skip configuration of grouping by timestamp, the default values will be set for the fields in the `limits` settings block.

7. In the mandatory `dataPoints` settings block, enter the following data for each data point created in the MindSphere cloud platform:

   a. Enter the data point ID in the `id` field. For example, `"id": 0`.

   b. Specify the data point name in the `name` field. For example, `"name": "Heartbeat"`.

   The name of the data point in the MindSphere cloud platform must match the name of the date node of the OPC UA server that you indicated in the `nodes` settings block in the OpcUaClientSettings-0.json configuration file.

   c. In the `dataPointId` field, enter the ID of the data point defined in MindSphere. For example, `"dataPointId": "1625019234863"`.

   You can use the tools of the MindConnect LIB plugin to obtain the data point ID for this data point in MindSphere. For detailed information on obtaining a data point ID using the tools of the MindConnect LIB plugin, please refer to the MindSphere documentation.

8. Save the changes in the MindSphereAgentSettings-0.json file.

The settings defined in the MindSphereAgentSettings-0.json file will be applied the next time Kaspersky IoT Secure Gateway 100 is started.

Kaspersky IoT Secure Gateway 100 will receive data from monitored objects within the internal network of your organization and forward that data to the Siemens MindSphere cloud platform.

Configuring data routing from Kaspersky IoT Secure Gateway 100 to Siemens MindSphere

To correctly transmit data received from industrial equipment over the OPC UA protocol from Kaspersky IoT Secure Gateway 100 to the Siemens MindSphere cloud platform, you need to map the MindConnect data points to their corresponding OPC UA server data nodes.

You can configure mapping of MindConnect data points to OPC UA server data nodes in the GuideSettings-0.json configuration file.

To configure mapping of MindConnect data points to OPC UA server data nodes:

1. Create a GuideSettings-0.json configuration file and put it in the folder /app/Core/config/transfer/navigation in the TGW-HW-IDS section of the SD card.

   All of the actions described next are performed within the GuideSettings-0.json file.

2. Enter the data route ID in the mandatory `id` parameter. For example, `"id": 0`.

3. In the mandatory `receivingHubId` parameter, enter the ID of the OPC UA client that you indicated in the OpcUaClientSettings-0.json configuration file (`id` parameter). For example, `"receivingHubId": 0`.

4. In the mandatory `sendingHubId` parameter, enter the ID of the MindSphere agent that you indicated in the MindSphereAgentSettings-0.json configuration file (`id` parameter). For example, `"sendingHubId": 0`.

5. In the mandatory `roadmap` settings block, provide the following data for each connection:

   a. In the `sourcePortId` field, enter the ID of the OPC UA server data node that you indicated in the `nodes` settings block in the OpcUaClientSettings-0.json configuration file. For example, `"sourcePortId": 0`.

   b. In the `targetPortId` field, enter the ID of the MindConnect Lib data point that you indicated in the `dataPoints` settings block in the MindSphereAgentSettings.json-0 configuration file. For example, `"targetPortId": 0`.

   The `roadmap` settings block may be empty.

6. Save the changes in the GuideSettings-0.json file.

The settings defined in the GuideSettings-0.json file will be applied the next time Kaspersky IoT Secure Gateway 100 is started. Any data received from equipment over the OPC UA protocol will be forwarded from Kaspersky IoT Secure Gateway 100 to the Siemens MindSphere cloud platform.

Managing the Kaspersky IoT Secure Gateway 100 health log

Kaspersky IoT Secure Gateway 100 lets you configure the settings of the system health log. Health log settings are configured in the .log configuration file, which is located in the TGW-HW-LOG section of the SD card. The SD card is included in the distribution kit. System health logs are stored in the folder /logs.

To configure the Kaspersky IoT Secure Gateway 100 health log settings, do the following in the .log configuration file:

1. In the `LogFileSizeLimit` parameter, enter the maximum size (in bytes) of one system health log file. For example, `LogFileSizeLimit=100000000`.
2. In the `DirectorySizeLimit` parameter, enter the maximum size (in bytes) of the folder containing all system health log files. For example, `DirectorySizeLimit=1500000000`.

   Kaspersky IoT Secure Gateway 100 does not process full disk space warning events by default. For this reason, make sure that the configured size settings for the log storage folder are appropriate for the available disk space of the device.
3. Save the changes in the .log file.

If the .log file is absent, the system log will be deemed disabled and the log will not be written to the /logs folder.

The settings defined in the .log file will be applied the next time Kaspersky IoT Secure Gateway 100 is started.

Rules for writing and rotating log files

Log files are written and rotated according to the following rules:

- If the size limit of one log file is exceeded during data write operations, a new file is created and the data is written to the new file.
- If a data string could not be completely written due to the size limit of one log file, a new file is created, the data string is written to this new file, and subsequent data will be written to the new file.
- If the size limits of one log file and all log files are exceeded when writing a data string, the old log file is deleted, a new file is created, and subsequent data will be written to this new file.
- If the limit for all log files is exceeded when writing a data string, the old file is deleted and the data string is written to the current file.
- Old log files are deleted in such a way to ensure that there is sufficient memory for writing a data string.
- A new file for log writing is created at the beginning of each work session (gateway loading/restart).

Logging data

The following information is logged:

- Initialization status of hardware components
- OS load status
- Initialization status of system components
- Initialization and operating status:
  - Network services

    - Data repository
    - System logging service
    - Application component:
        - OPC UA client and manager
        - MindSphere agent and manager
        - Data transfer component
- Transmitted data elements
- Initialization and loading errors

Appendices

This section describes the location of the main configuration files on the SD card, and provides examples of the structure of each of these files.

Example dhcpcd.conf configuration file

The dhcpcd.conf configuration file is located in the folder /etc in the TGW-HW-ENW and TGW-HW-INW sections of the SD card. The SD card is included in the distribution kit.

Below is an example of a dhcpcd.conf configuration file in which you need to define the settings for the external or internal network.

Example dhcpcd.conf configuration file:

```
static ip_address=192.168.1.177/23
static routers=192.168.1.1
```

Example OpcUaClientSettings-0.json configuration file

The OpcUaClientSettings-0.json configuration file is located in the folder /app/Core/config/transfer/opc_ua/client in the TGW-HW-IDS section of the SD card. The SD card is included in the distribution kit.

Below is an example of an OpcUaClientSettings-0.json configuration file in which you need to define the settings for receiving data from a monitored object over the OPC UA protocol.

Example OpcUaClientSettings-0.json configuration file:

```
{
 "id": 0,
 "name": "Kaspersky IoT Secure Gateway 100 OPC UA Client",
 "description": "Collects data from CNC by Kaspersky IoT Secure Gateway 100",
 "url": "opc.tcp://192.168.177.7:4840",
 "readingCycle": 1,
 "security": {
 "mode": "SignAndEncrypt",
 "policy": "Basic256Sha256",
 "clientPkiData": {
 "certificate": "client.crt",
 "privateKey": "client.key"
 },
 "trustList": ["server.crt"]
 },
 "userCredentials":
 {
 "username": "KISG100",
 "password": "0R20jN#yZd~zaLKe?2J#@~|YC"
 },
 "heartbeat": {
 "id": 0,
 "name": "Heartbeat",
 "timeout": 60
 },
```

```
 "nodes": [
 {
 "id": 1,
 "name": "Temperature",
 "nodeId": "ns=1;s=VariableTemperature"
 },
 {
 "id": 2,
 "name": "Speed",
 "nodeId": "ns=2;i=2045"
 }
 ]
}
```

Example MindSphereAgentSettings-0.json configuration file

The MindSphereAgentSettings-0.json configuration file is located in the folder /app/Core/config/transfer/mind_shere/agent in the TGW-HW-IDS section of the SD card. The SD card is included in the distribution kit.

Below is an example of a MindSphereAgentSettings-0.json configuration file.

Example MindSphereAgentSettings-0.json configuration file:

```
{
 "id": 0,
 "name": "Kaspersky IoT Secure Gateway 100 MindSphere Agent",
 "description": "Transfers data to MindSphere by Kaspersky IoT Secure Gateway 100",
 "boardingConfiguration": {
 "content": {
 "baseUrl": "https://southgate.eu1.mindsphere.io",
 "iat":
"eyJraWQiOiJrZXktaWQtMSIsInR5cCI6IkpXVCIsImFsZyI6IlJTMjU2In0.eyJpc3MiOiJTQ0kiLCJzdWIiOiI
55bKT5DHR3JEYEChbUxRx1xz-TlX9elZPokstLCb58l7pjlNnkg8YhW43Od0vixNOHWGKjVnLhwwJ0yyB9z4S54W
 "clientCredentialProfile": [
 "SHARED_SECRET"
 ],
 "clientId": "4e0ab70efc724445a5f483f344a22f1c",
 "tenant": "aprotech"
 },
 "expiration": "2021-03-24T18:53:51.000Z"
 },
 "configurationId": "1606380355815",
 "proxySettings": {
 "type": "HTTP",
 "host": "192.168.188.1",
 "port": 3128
 },
 "limits": {
 "maxStorageSize": 90000,
 "itemGroupTimeout": 5,
 "maxTimeseriesSize": 64,
 "maxHttpPayloadSize": 16384
 },
 "dataPoints": [
 {
 "id": 0,
```

```
 "name": "Heartbeat",
 "dataPointId": "1625019234863"
 },
 {
 "id": 1,
 "name": "Temperature",
 "dataPointId": "1616007325504"
 },
 {
 "id": 2,
 "name": "Speed",
 "dataPointId": "1616007338184"
 },
 ]
}
```

Example GuideSettings-0.json configuration file

The GuideSettings-0.json configuration file is located in the folder /app/Core/config/transfer/navigation in the TGW-HW-IDS section of the SD card. The SD card is included in the distribution kit.

Below is an example of a GuideSettings-0.json configuration file in which you need to define the settings for configuring data transfer between the OPC UA data source node and data points in MindSphere.

Example GuideSettings-0.json configuration file

```
{
 "id": 0,
 "receivingHubId": 0,
 "sendingHubId": 0,
 "roadmap": [
 {
 "sourcePortId": 0,
 "targetPortId": 0
 },
 {
 "sourcePortId": 1,
 "targetPortId": 1
 },
 {
 "sourcePortId": 2,
 "targetPortId": 2
 }
 ]
}
```

Limitations

Kaspersky IoT Secure Gateway 100 1.2 has the following limitations:

- Kaspersky IoT Secure Gateway 100 must be restarted before it can apply new security settings for the OPC UA server after reconnection.
- The OPC UA client certificate does not undergo verification.
- The OPC UA client verifies the server certificate in the list of trusted certificates by checking for a matching server certificate and verifying its validity period.
- If it is configured to trust all certificates (`"trustList": "AllowAll"`), the client does not verify the server certificate.
- When the `None` security policy and mode are indicated in the Kaspersky IoT Secure Gateway 100 settings, the OPC UA client certificate and key must also be provided.
- Only the following data types described in the OPC UA specification are supported:
  - Boolean
  - SByte
  - Byte
  - Int16
  - UInt16
  - Int32
  - UInt32
  - Int64
  - UInt64
  - Float
  - Double
  - String
  - DateTime
  - XmlElement
  - NodeId (only numeric and string)
  - ExpandedNodeId (equivalent to NodeId)
  - StatusCode
  - QualifiedName

    - LocalizedText (partially)
    - Variant
- Data of the Double and Float data types received over the OPC UA protocol is rounded to the nearest six significant digits.
- When collecting data over OPC UA, the server must support a publisher-subscriber communication model.
- Only one OPC UA client connection to one OPC UA server is available.
- During the early initialization stages of the MindSphere agent, information from the agent is not written to the health log of Kaspersky IoT Secure Gateway 100. This is due to the restrictions imposed by KasperskyOS security policies.
- The log does not record the warning that is displayed if the name of the MindSphere agent configuration file is invalid.
- Kaspersky IoT Secure Gateway 100 will not apply new network settings received from the DHCP server until the lease time of the IP address expires.
- If powered off, Kaspersky IoT Secure Gateway 100 does not retain any unsent data because it does not store this data in non-volatile memory.
- Kaspersky IoT Secure Gateway 100 does not process full disk space warning events when maintaining the health log.

Information about third-party code

Third party code information is contained in the file named legal_notices.txt, which is provided on the SD card. The SD card is included in the distribution kit.

Trademark notices