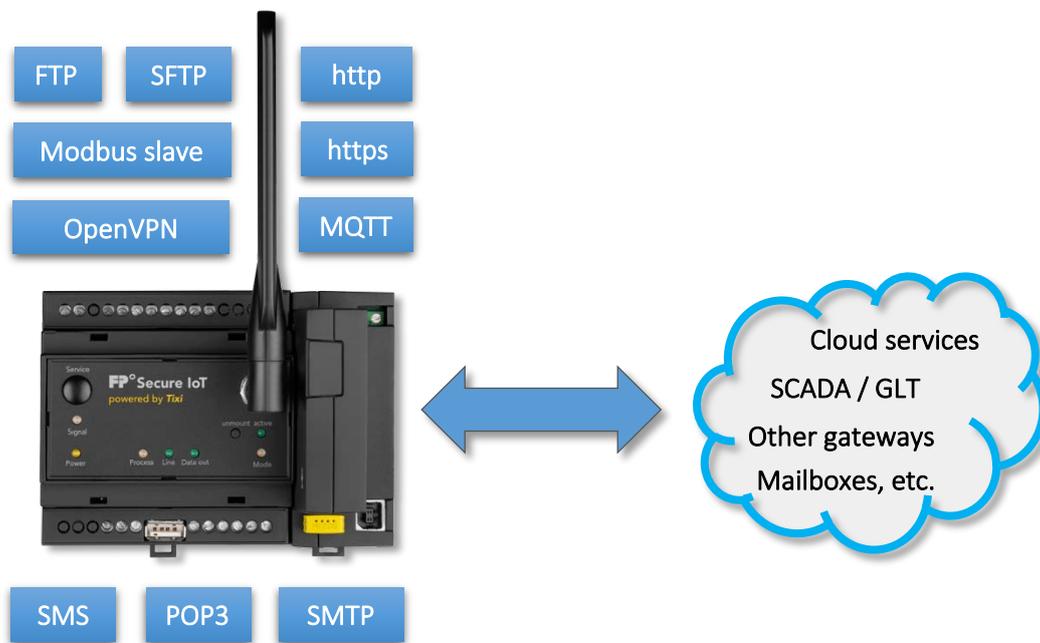


Communication Protocols for FP Gateways

Quick Guide



Version: 1.0.2

© 2018 -2021 FP InovoLabs GmbH

www.inovolabs.com

Publication date: 23/02/2021

This manual is protected by copyright. Any further dissemination is only permitted with permission from the issuer. This also applies to copies, microfilms, translations, and storing and processing in electronic systems.

Trade and brand names used in this manual are registered trademarks of the applicable companies even if they are not designated as such explicitly.

Table of contents

1	INTRODUCTION	3
1.1	Quick overview of the protocols	3
1.2	Further applicable documents	3
2	COMMUNICATION PROTOCOLS	4
2.1	http web server	4
2.2	http(s) client	4
2.3	TiXML programming interface via TCP port 8300	4
2.4	MQTT clients	4
2.5	FTP / SFTP clients	5
2.6	E-mail communication	5
2.7	SMS communication	5
2.8	OpenVPN client	5
2.9	Universal Modbus slave	5
2.10	Service routing	6
3	HISTORY	7

1 Introduction

The FP IIoT gateways support numerous communication protocols in order to include the gateways in almost any network and infrastructure.

This document is intended as a guide and refers to other documents that provide further information regarding configuring and using the communication protocols.

1.1 Quick overview of the protocols

The following communication protocols are available in FP gateways:

Protocol	Type	Short description / purpose
http	Server	Internal web server http POST / http GET
https	Client	Downloading firmware versions; cloud protocols
http	Client	http POST / http GET
FTP	Client	Uploading data to an FTP server
SFTP	Client	Uploading data to an SFTP server
OpenVPN	Client	Establishing a secure communication tunnel to OpenVPN servers
MQTT	Client	Several MQTT client variants to connect to cloud services e.g. Telekom "Cloud der Dinge" (cloud of things), Cumulocity Cloud, Juconn, AWS, etc.
Modbus slave	Server	Providing data via Modbus, which can be called by a Modbus master, e.g. in a control station (GLT, Scada or a different FP gateway)
SMTP	Client	Sending e-mails to SMTP servers
POP3	Client	Collecting e-mails from POP3 inboxes
http	Client	Transferring data to an http server via http GET
SMS		Sending and receiving SMS
TiXML control	Server	TiXML programming interface for TILA, TICO via TCP port 8300

1.2 Further applicable documents

The following documents contain comprehensive information regarding using the communication protocols:

No.	Document	Protocols
[1]	510058920601_xx_FP-SCADA-GLT_EN	http
[2]	510058920701_xx_FP-Webserver-TiXML-Manual_EN	http
[3]	510058920001_xx_FP-TiXML-Reference_EN	FTP, SFTP, SMTP, POP3, SMS
[4]	510058920801_xx_FP-Universal_MQTT_Client_EN	MQTT
[5]	510058921101_xx_FP-Juconn_MQTT_Client_EN	Juconn MQTT
[6]	510058920101_xx_FP-PLC-TiXML-Manual_EN	Modbus slave
[7]	510058921201_xx_FP-OpenVPN-Toolkit_Docu_EN	OpenVPN
[8]	510058921001_xx_FP-CoT_Client_EN	Telekom "Cloud der Dinge" (cloud of things)
[9]	510058920901_xx_FP-C8y_Client_EN	Cumulocity Cloud
[10]	510058920501_xx_FP-Remote Firmware Update QuickStart_EN	http(s) for firmware updates

xx = Revision of the document, starting with 00

For more information and to download documents, see www.inovolabs.com.

2 Communication protocols

The FP IIoT gateways have both an http web server and an http(s) client. The gateways can therefore be integrated well into existing networks.

2.1 http web server

The internal http web server can be used for the following tasks:

- Calling data points and log data
- Calling system data (system log files, system status)
- Executing commands
- Setting data points
- Displaying websites
- Configuring the system via TiXML

The web server's TCP port can be configured (default: 80). The web server supports a variety of cgi commands that can be used to execute the aforementioned functions. Special reference must be made to the "tixml.cgi" cgi for executing commands and configuring the system. The "tixml.cgi" cgi provides a TiXML programming interface via the web server port.

The web server can be provided with access protection via the AccRights database.

Detailed information regarding how the internal web server functions and how to configure it is provided in [1] and [2].

For more information regarding executing commands and configuring the TiXML, see [3].

2.2 http(s) client

The internal http(s) client can be used to download firmware versions. The client can call data from any http(s) web server. TLS encryption is supported for public web servers. Details regarding configuring firmware updates are provided in [10].

2.3 TiXML programming interface via TCP port 8300

An FP gateway can be programmed and controlled via the TiXML programming interface.

The TiXML programming interface uses TCP port 8300 (can be configured). The FP gateway provides a TiXML server service on this port.

For more information, see [3] (sections 2.4 and 3.16).

2.4 MQTT clients

The MQTT clients that are integrated into the FP gateways support the following MQTT brokers:

- Telekom Cloud der Dinge (cloud of things) -> see [8]
- Cumulocity Cloud -> see [9]
- Juconn Cloud -> see [5]
- All standard brokers (e.g. AWS Cloud, mosquitto, etc.) -> see [4]

The MQTT clients generally also support customer-specific security certificates (except for: Telekom and Cumulocity).

2.5 FTP / SFTP clients

The FP gateways support transferring data to FTP servers and SFTP servers. The SFTP client is based on the **SSH File Transfer Protocol** (also known as **Secure File Transfer Protocol**).

The TCP ports for the FTP/SFTP clients can be configured.

For more information, see [3] (section 11).

2.6 E-mail communication

The FP Gateways have SMTP and POP3 clients.

The SMTP client can be used to send e-mails (also with attachments) to any SMTP server. TLS is supported (STARTTLS and DIRECTSSL, ports can be configured).

The POP3 client can be used to collect e-mails (also with attachments) from POP3 mailboxes and to process them further. TLS is supported (STARTTLS and DIRECTSSL, ports can be configured).

For more information, see [3] (section 3.8 ff and section 9.3.4).

2.7 SMS communication

The FP gateways with mobile communications modems have the option to send and receive SMS if the mobile communications provider and the SIM card support this. Received SMS can be processed further internally.

For more information, see [3] (section 3.8 ff and section 9.3.3).

2.8 OpenVPN client

The OpenVPN client on FP gateways enable a TLS-secured VPN tunnel to be established. Customer-specific certificates can be used.

For more information, see [7].

2.9 Universal Modbus slave

The universal Modbus slave enables any process data defined in the FP gateway to be read and written from a different control system or a different FP gateway. The data points provided by the slave are used as memory cells or there is the option to assign internal bus variables, process variables or other system properties via a reference path.

The slave copies the assigned variables into its internal variable structure and provides them to the master for reading and writing purposes. For example, the Modbus TCP master can use this to read and write data points for a controller that is connected to the FP gateway (e.g. Siemens, ABB or similar PLC) or a process variable defined in the FP gateway.

For more information, see [6] (section 4.6).

2.10 Service routing

For most services on an FP gateway, the communication channel via which these services are to run can be defined.

One of the following services is designated as a service:

- Send e-mail (SMTP)
- Receive e-mail (POP3)
- Call http URL (URLSend)
- Call internet time (INetTime = DAYTIME / NTP)
- Cloud connections (CloudConn)
valid for: Universal MQTT, Juconn MQTT, Telekom CoT, Cumulocity Cloud
- FTP connections (FTPPut)
- SFTP connections (SFTPPut)
- VPN tunnel (VPN)

The communication channel for all of the aforementioned service can be defined via the ISP/ISP/OUT database. For example, you can force your VPN tunnel to always be established via the mobile communications modem.

For more information, see [3] (section 12).

3 History

Version	Date	Editor	Changes
1.0.2	23/02/2021	IVH	First English issue
1.0.1	27/10/2020	SH	New document designation
1.0.0	09/04/2020	SH	First issue