DOI 10.5162/iCCC2026/P35

Step by Step from Modbus to secure IoT

Reinhardt Karnapke
Perinet GmbH
Siemens-Halske-Ring 2
03046 Cottbus
reinhardt.karnapke@perinet.io

Karsten Walther
Perinet GmbH
Rudower Chaussee 29
12489 Berlin
karsten.walther@perinet.io

Abstract

Many industrial deployments have been realized using Modbus technology in the last 46 years. However, new challenges like the Data Act of the European Union, security issues in deployments as well as a general need for more Data availability up in the Enterprise System lead to demands that traditional Modbus deployments simply cannot fulfill.
Fulfilling these new requirements seems to mandate a complete rebuild of legacy deployments. However, no one wants to stop production and incur large downtimes for entire sections of factories at once.
In this paper we introduce a step-by-step approach for switching from classic Modbus or Modbus TCP to a secure IoT based operation with minimal to no downtime.

1 Introduction

Despite having been designed almost 50 years ago with PLCs in mind, Modbus1 remains very relevant today. Many factory installations are built using field busses, with Modbus being most prominent among them.
However, even though the installations still work, there are some additional challenges that have arisen over the years, which cannot be addressed by legacy Modbus installations very well. These include, among others, regulatory issues, accessibility from the enterprise system and security concerns.

Having the ability to access data from the enterprise system would in itself be rewarding. Whether the rewards are high enough to warrant the costs depended on the factory owner, until recently.
However, with the Data Act of the European Union[1] that came into effect on January 11, 2024, this changed.
Since September 12th, 2025, this Data Act grants all customers the right to demand access to their product related Data. By September 12th, 2026 all new products placed on the European market will have the have included Data accessibility already during the draft phase.

Security has long been realized with physical access to factories in mind, not with access to Data. Therefore, classic Modbus deployments transmit Data unencrypted. However, nowadays a lot of operations are outsourced, meaning that engineers from different companies, in some deployments even customers, can get close to the machines and the Modbus cables, usually RS-485. As this data is transmitted unencrypted, anyone with physical access can just attach a Data sniffer and collect Data.
There is the theoretical option of using TLS over the existing Modbus. However, this would decrease the Bandwidth available to the already running applications. Even if this could be tolerated, the even bigger drawback is that the Firmware of all participating machines would have to be updated, and simultaneously.

Even though Data availability and security are big concerns, it is simply not practicable to remove whole installations and replace them with other, secure approaches as this would result in huge downtimes of the deployments. Therefore, a way to gradually introduce Data collection and security into existing Modbus deployments without interrupting day to day operation, or at least with minimal interference, is needed.

In this paper we describe our approach for retrofitting existing industrial Modbus deployments with security and connections to the Enterprise System without shutting down the operation of the whole deployment. Instead, we focus on a step by step approach, adding additional communication capabilities and replacing elements only when they fail or are not in use, leading to undisrupted operation.

2 Step by Step from Modbus to secure IoT

Figure 1 shows a typical Modbus deployment as can be found in any number of factories today.

Local controllers are connected to Modbus TCP/RTU gateways, which are in turn connected with RS-485 cables on which the Modbus devices reside. Currently, there is no connection to the Enterprise System.

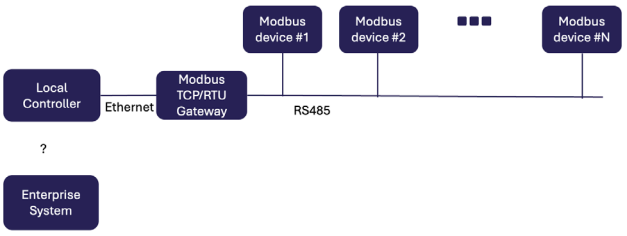

Figure 1 Current Situation in many Industrial Deployments

DOI 10.5162/iCCC2026/P35

2.1 Minimal approach: Insert Secure Data Sniffer

For the minimal approach we focus on gathering Data from the Modbus and making it available to the Enterprise System and, therefore, the customer, when requested.
For this, we can make use of the currently still insecure Data transmission on the Modbus by inserting a Modbus Sniffer.
Please note that for this step the existing installation is not changed at all, and all devices continue to run their legacy code and use their legacy communication.
For the Enterprise System to be able to understand the Data that is made available, digital twins of the Modbus deployment should be realized.

Please also note that the Modbus sniffer should not just be a dumb device, as this would introduce additional security risks. Instead, we propose an intelligent device that can publish named Data, e.g. using MQTT[2], and that also encrypts all Data it is transmitting using state of the art encryption technologies like TLS[3]. In our case, this was realized with a so-called periNODE[4]. The cabling be realized using Ethernet , in our case Single Pair Ethernet(SPE)[5] cables. In our opinion, this is the minimal approach that is necessary to become compliant with European law.

The major advantage of this minimal approach is that the Modbus installation can remain unchanged, the Modbus applications on the devices can remain unchanged and still IoT Data becomes available. Using this IoT Data we can now respond to the customers' requests for product relevant Data in accordance to the Data Act of the European Union.

Figure 2 visualizes the minimal approach.

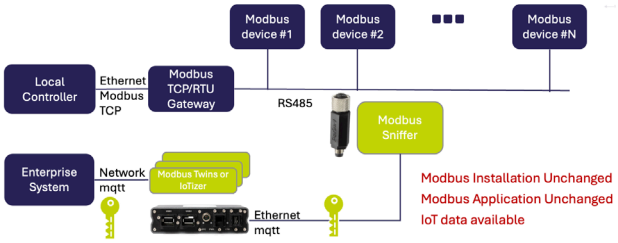

Figure 2 IoT Add On Data Sniffer

2.2 Introducing Secure Field Communication

While the minimal approach suffices to be compliant with the Data Act of the European Union, the Modbus itself remains insecure. Therefore, in order to remove the security risks mentioned above, (parts of) the RS-485 cabling shall be replaced with SPE. To enable legacy devices to use SPE, we once more use a periNODE, similar to the way we previously used a sniffer on the Modbus. However, instead of having only one in the whole system, we now connect one of them to each Modbus device. This way, the Modbus devices remain unchanged and the applications running on the Modbus devices also remain unchanged.

On the security side, however, we now have a secure communication between all the devices, as all of the periNODEs realize Modbus TCP over SPE by transmitting TLS encrypted messages among each other.

Therefore, even if anyone from outside would gain physical access to the cables and install a sniffer, they would no longer be able to interpret the gathered Data.

The big advantage of this step is that devices and applications remain unchanged, and while we still use Modbus communication, it is now secure as it is tunneled over SPE.

Figure 3 visualizes the new, secure communication.

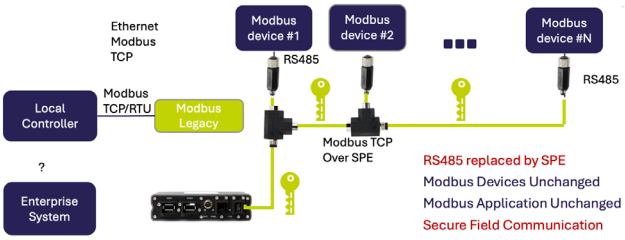

Figure 3 Switching from RS485 to SPE

2.3 Connecting the Enterprise Systems using Linux Containers

In the next step we want to connect all of our newly realized secure communication over SPE with the enterprise system. Of course, we do not want to re-cable our enterprise system with SPE. Therefore, we use our small industry PC to run the digital twin of our Modbus system, and also to connect to conventional Ethernet. This way, when using the enterprise system, we only need to connect to the digital twin and can retrieve all data we are interested in. This is shown in Figure 4.

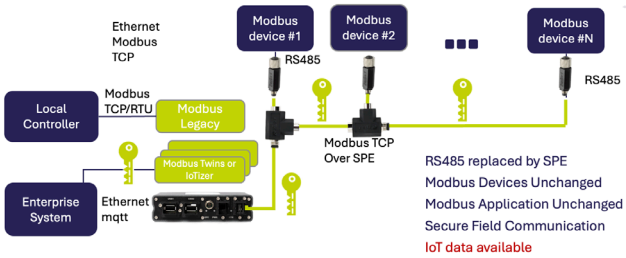

Figure 4 Adding IoT Connection

2.4 Updating Field Devices to SPE

Now that we have full access from the enterprise system, we can safely start replacing some of our devices. Of course, we do not want to replace all of them, especially not at once as stated before. However, there might be some devices that have broken down partially or deliver only reduced functionality for whatever reason. If these

devices are quite old, there often are no replacement parts necessary for repair available anymore. In this case, replacing to old device with a new Modbus TCP device can increase productivity. On the other hand there might be devices which still work, or for which there are simply no new versions available. The big advantage of this step by step approach is that we can have a mixed system – some old devices with added secure communications that still run the old code working together with newly installed devices. And due to the added bandwidth and scalability that result from the use of SPE instead of RS-485, it is even possible to keep all of the old devices and just expand the system.
Figure 5 shows our running example, now with one of the old devices replaced by a new Modbus TCP device.

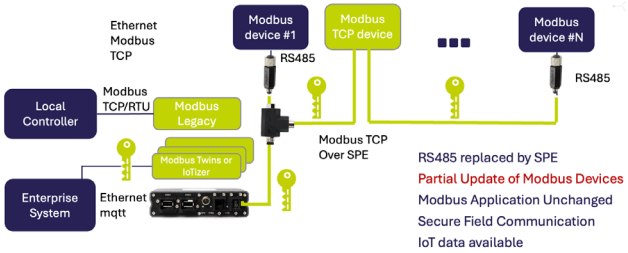

Figure 5 Updating Field Devices to SPE

2.5 Changing Communication to IoT communication

Currently, we are having powerful SPE cabling but only using it to transmit Modbus TCP data. In this step, we want to take things one step further and introduce IoT communication. The typical protocol used in new deployments nowadays is MQTT as it has numerous advantages over Modbus. Therefore, in this step, we switch to MQTT/HTTPS over SPE. As we still have our application running as Modbus client software, we introduce a virtual Modbus between it and our SPE, which runs on the industrial PC, in our case a periMICA[6].

However, even though we update our communication to MQTT, legacy devices can still be used with the adapters, as these can translate between Modbus and MQTT directly at these devices.

One huge advantage that is introduced here is that new devices that are added to our deployment now can either be Modbus devices or new smart devices that are incompatible with Modbus but can communicate using MQTT, resulting in IoTized Plug'n'Play devices.

Another advantage in this that there is no need to change any hardware – the whole step is only a software update.

Figure 6 shows our running example with a new deployment layout. The virtual Modbus and a smart device have been added.

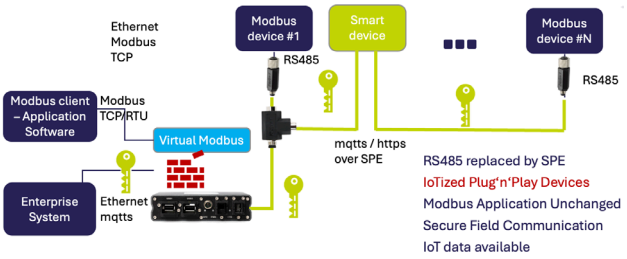

Figure 6 Changing to IoT Communication

2.6 Final Step: Introducing new application Software

In this last step we gradually replace the old application Software with a new one that uses secure IoT Protocols and the additional features those Protocols provide.

Figure 7 shows our running example again, now with a completely IoTized application software and another smart device added.

However, the legacy device that still can be seen will remain, as it is still working according to specification and fulfilling all current requirements. For this to work, we also keep one of the adapters in the system, which will continue to translate between MQTT/HTTPS and Modbus for that legacy device until the legacy device breaks or becomes obsolete.

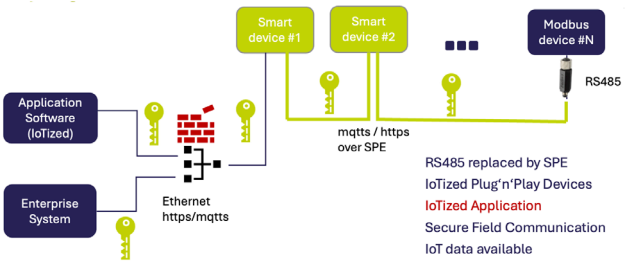

Figure 7 Transition from Modbus to secure IoT finished

3 Conclusion

In this paper we have shown a step-by-step approach for replacing Modbus installments with secure IoT installations as is required in order to confirm with upcoming EU regulations. This can also be seen as a chance to introduce additional security measures.
The big advantage of this step-by-step approach is that the deployment can continue to operate during the changes, we introduce minimal to zero downtime. Also, legacy devices that cannot be repaired anymore can now be replaced with modern smart devices. On the other hand, legacy devices which are still operational do not have to be changed as a smart adapter was introduced that can connect to their Modbus in-/output and translate between Modbus and an IoT protocol like MQTT.

DOI 10.5162/iCCC2026/P35

4 Bibliography

[1] Data Act of the European Union https://digital-strategy.ec.europa.eu/en/policies/data-act last accessed 2025-10-24

[2] MQTT https://mqtt.org/ last accessed 2025-10-24

[3] TLS https://datatracker.ietf.org/doc/html/rfc8446

[4] periNODE Modbus https://www.perinet.io/de/produkte/smart-components/perinode-modbus last accessed 2025-10-24

[5] Single Pair Ethernet according to IEC 63171-6:2021 https://www.vde-verlag.de/iec-standards/250428/iec-63171-6-2021.html

[6] periMICA https://www.perinet.io/de/produkte/edge-computer/perimica last accessed 2025-10-24