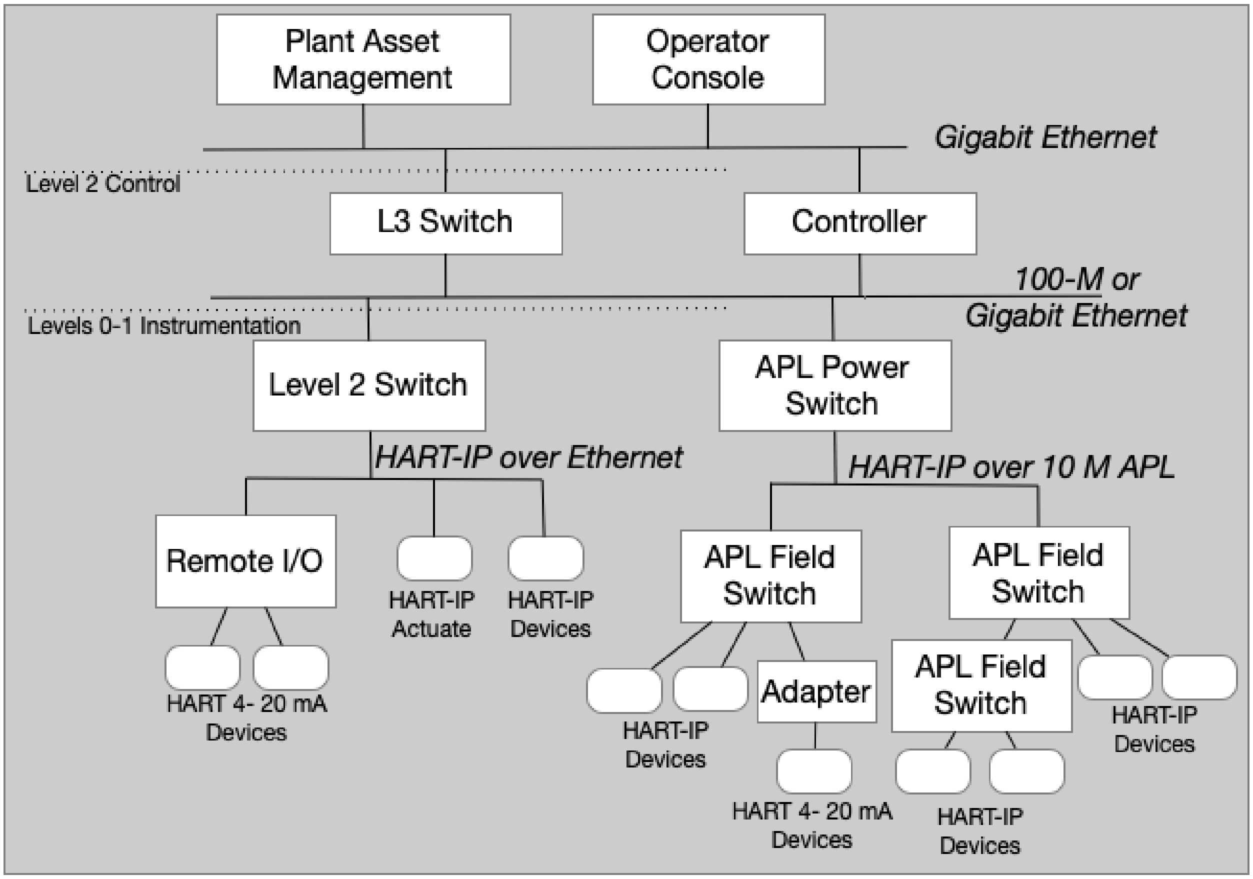
HART-IP® Security Technical Paper



Document Distribution / Maintenance Control / Document Approval

To obtain information concerning document distribution control, maintenance control, and document approval please contact FieldComm Group at the address shown below.

Trademark Information

FieldComm Group™, FDI™, FOUNDATION™ Fieldbus and PA-DIM ™ are trademarks, and HART®, HART-IP® and *Wireless*HART® are registered trademarks of FieldComm Group, Austin, Texas, USA. Any use of these terms hereafter in this document, or in any document referenced by this document, implies the trademark/registered trademark. All other trademarks used in this or referenced documents are trademarks of their respective companies. For more information, contact FieldComm Group at the address below.

FIELDCOMM GROUP™
Connecting the World of Process Automation

FieldComm Group
Attention: President and CEO
9430 Research Boulevard
Suite 1-120
Austin, TX 78759, USA
Voice: (512) 792-2300
FAX: (512) 792-2310

http://www.fieldcommgroup.org

Intellectual Property Rights

The FieldComm Group (the Group) does not knowingly use or incorporate any information or data into the HART, FOUNDATION Fieldbus and FDI protocol standards, which the Group does not own or have lawful rights to use. Should the Group receive any notification regarding the existence of any conflicting private IPR, the Group will review the disclosure and either (A) determine there is no conflict; (B) resolve the conflict with the IPR owner; or (C) modify the standard to remove the conflicting requirement. In no case does the Group encourage implementers to infringe on any individual's or organization's IPR.

Table of Contents

1. Introduction 5
2. An Introduction to HART-IP and HART-IP Devices 5
  2.1 Overview 5
  2.2 Network Topologies 6
    2.2.1 Control Topology 6
    2.2.2 Cloud-based Topology 7
3. Security Threat and Risk Models 9
  3.1 Overview 9
4. How HART-IP Implements Security 10
  4.1 Overview 10
  4.2 HART-IP Security 10
    4.2.1 Communication Security 10
    4.2.2 Audit Logs 11
    4.2.3 Syslogs 11
  4.3 Comparing HART, WirelessHART, HART-IP 11
5. Implementing a HART-IP Device 12
  5.1 Overview 12
  5.2 Root of Trust 13
  5.3 Secure Boot 13
  5.4 Endpoint Identity 14
  5.5 Secure Storage 14
  5.6 Cryptographic Capabilities 14
  5.7 Secure Communications 14
  5.8 Device Provisioning 15
  5.9 Security Audit Functions (Syslog) 15
6. Device Lifecycle 15
  6.1 Overview 15
  6.2 Device Procurement 16
  6.3 Device Provisioning 16
  6.4 Device Installation and Configuration 16

6.5 Device in Operation 17
6.6 Device in Maintenance 17
6.7 Device Replacement 17
6.8 Device Decommissioning and Disposal 17
7. Conclusions 19
Annex A. Keeping UP TO DATE 20
A.1.1. Sources of Information 20
Annex B. STRIDE 21
Annex C. A Short Introduction to TLS 23
C.1.1. What is TLS 23
C.1.2. TLS Handshake 23
Annex D. References 26

1. INTRODUCTION

In the Process Automation industry field device networking today is largely dominated by HART 4-20, Foundation Fieldbus, Profibus, Modbus, and several other protocols. Emerging alongside these protocols are Ethernet-based protocols such as Ethernet/IP and Modbus/TCP To fully support ethernet in process automation what has been needed is the capability to both provide power to devices and to be safe for use in hazardous locations. To meet these requirements in a standardized way, an IEEE working group was formed. The output of this working group is the Advanced Physical Layer (APL) standard which supports the connection of field devices in process operations, including hazardous locations. APL, or IEEE P802.3cg, specifies enhancements to the existing IEEE 802.3 ethernet standard (IEEE 802.3) for ethernet using twisted-pair wiring (10BASE-T1L).

Ethernet support within the FieldComm group is not new. HART-IP, which can be used with ethernet, was first specified as part of HART 7 over ten years ago. This initial specification was largely intended for communications between hosts and gateways. In the latest release of HART-IP, clarity around support for HART-IP devices is now included. As part of these enhancements to HART-IP, security is explicitly specified. This technical paper describes these security enhancements.

This document begins with a brief introduction to HART-IP. It then discusses threats to HART-IP devices and the HART-IP device networks. The document then explains security mitigation strategies and concludes by summarizing how HART and HART-IP implement security.

2. AN INTRODUCTION TO HART-IP AND HART-IP DEVICES

2.1 Overview

HART-IP combines the Internet Protocol (IP) network infrastructure and the HART network and application layers. HART-IP can be used over any communication medium (Physical Layer) that supports the Internet Protocol. It enables HART data to work across several physical networks including Ethernet (IEEE 802.3), Wi-Fi (IEEE 802.11b/g), RS232 using Point-to-Point Protocol (PPP), 4G and 4G-LTE enabled cellular networks. HART-IP is illustrated in Figure 1.

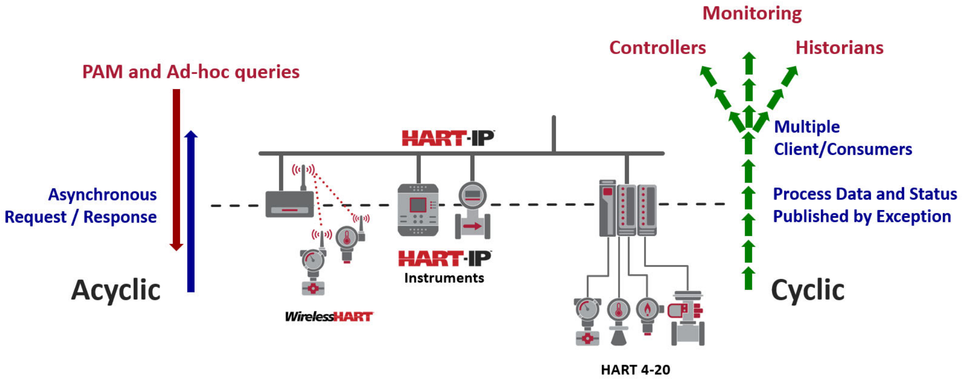


Figure 1 – HART-IP Devices and Communications

HART-IP enables an IP-based connection between HART-enabled data producers (i.e., field instrumentation) and HART data consumers. HART-IP consists of HART-IP Servers and HART-IP Clients. HART-IP servers would be embedded within devices and infrastructure such as, Ethernet-APL Field Devices (like flow computers, process analyzers); Remote HART 4-20mA I/O; and WirelessHART Gateways. HART-IP clients are incorporated into Hosts such as PLCs, DCS controllers, Plant Asset Management, and Edge Gateways.

HART-IP devices and networks must implement security. HART-IP security mechanisms are the topic of this paper. Discussions on security follow the introduction to HART-IP devices, the HART-IP protocol, and network topologies.

2.2 Network Topologies

Many topologies can be supported with HART-IP networks. In this paper two topologies are described: the first being a traditional control and asset management arrangement and the second being connections to an edge device or a cloud-based application hosted on-premise or an off-premise using a provider such as Microsoft Azure, AWS, etc.

2.2.1 Control Topology

The control topology mirrors how instrument and control are typically setup in plants. Devices are installed in the plant with the process equipment and connected to controllers over IO networks. In most process plants equipment is organized into functional groupings known as units. For example, a refinery may have several crude units, an FCC unit, a hydrocracking unit, and many other units. Each unit covers many acres with the refinery itself often covering over 100 acres. Instrumentation networks, such as HART-IP networks, connect devices back to controllers associated with each of the units. Devices are known by their tag and location. The control topology is shown in Figure 2.

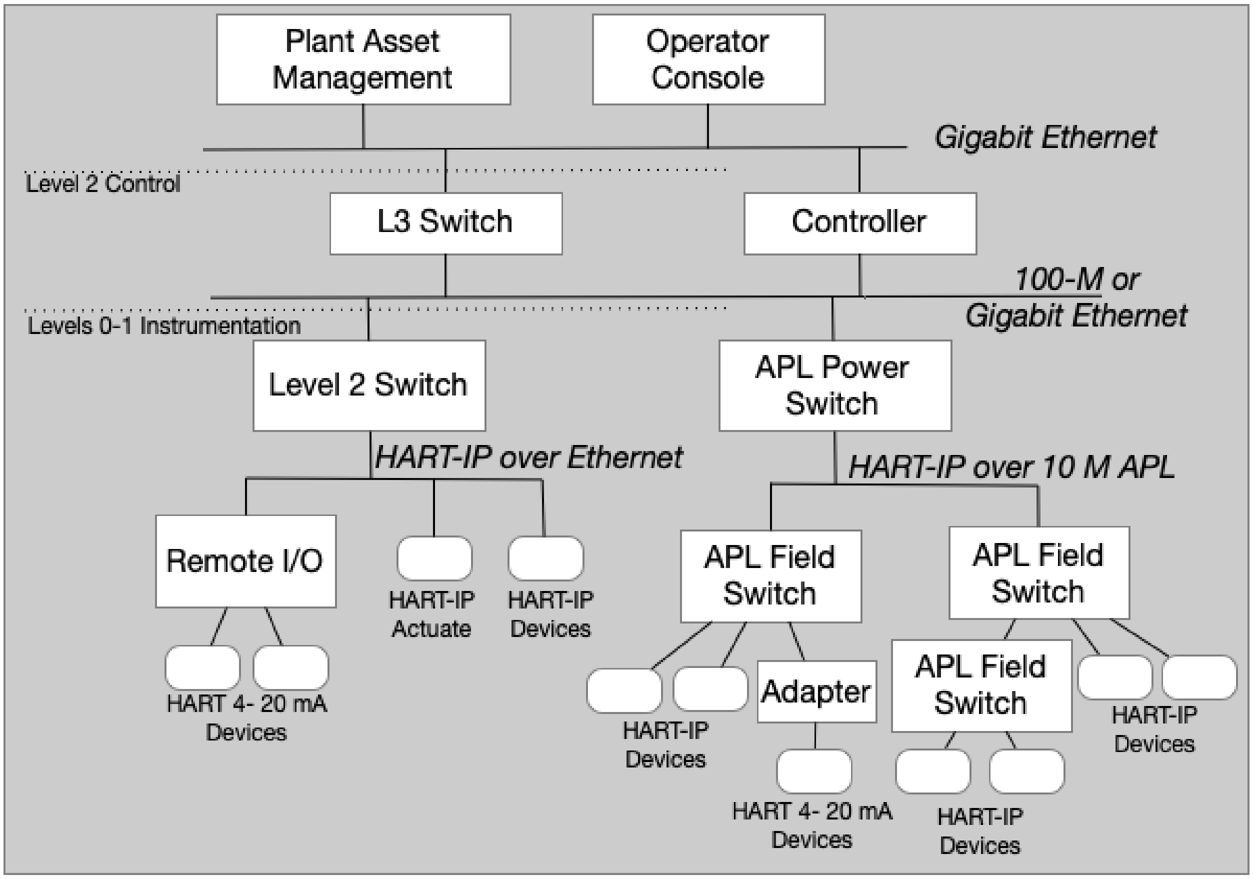


Figure 2 – Network Control Topology

As shown in Figure 2, HART-IP devices may be connected to traditional Ethernet or to an APL network. Remote I/O devices, acting as both client and server, are used to connect to HART 4-20 mA devices. Adapters, shown on the APL side of the figure, may also be used to connect to HART 4-20 mA devices.

The Advanced Physical Layer, APL, is a ruggedized, two-wire, twisted pair, loop-powered Ethernet physical layer that uses 10BASE-T1L plus extensions for installation within operating conditions and hazardous areas of process plants. APL supports field switches and power switches that may be arranged in various combinations as shown in Figure 2. Field switches can be nested up to five deep.

Most plants include the unit as part of the plant. For example, a flow on the #10 crude unit could be tagged as 10FI0001, with the last four digits a unique identifier for the instrument. For a more complete discussion of instrument tagging refer to ANSI/ISA-5.1-2009.

2.2.2 Cloud-based Topology

The cloud-based topology is an emerging setup in process plants that often runs in parallel to process instrumentation. These cloud-based topologies are often used for condition-based monitoring which are used for energy monitoring, equipment and device monitoring, and process monitoring. The monitoring systems stream data from the plant, perform their analysis, and provide recommendation back to users and in some cases the control systems themselves. The cloud-based topology is shown in Figure 3.

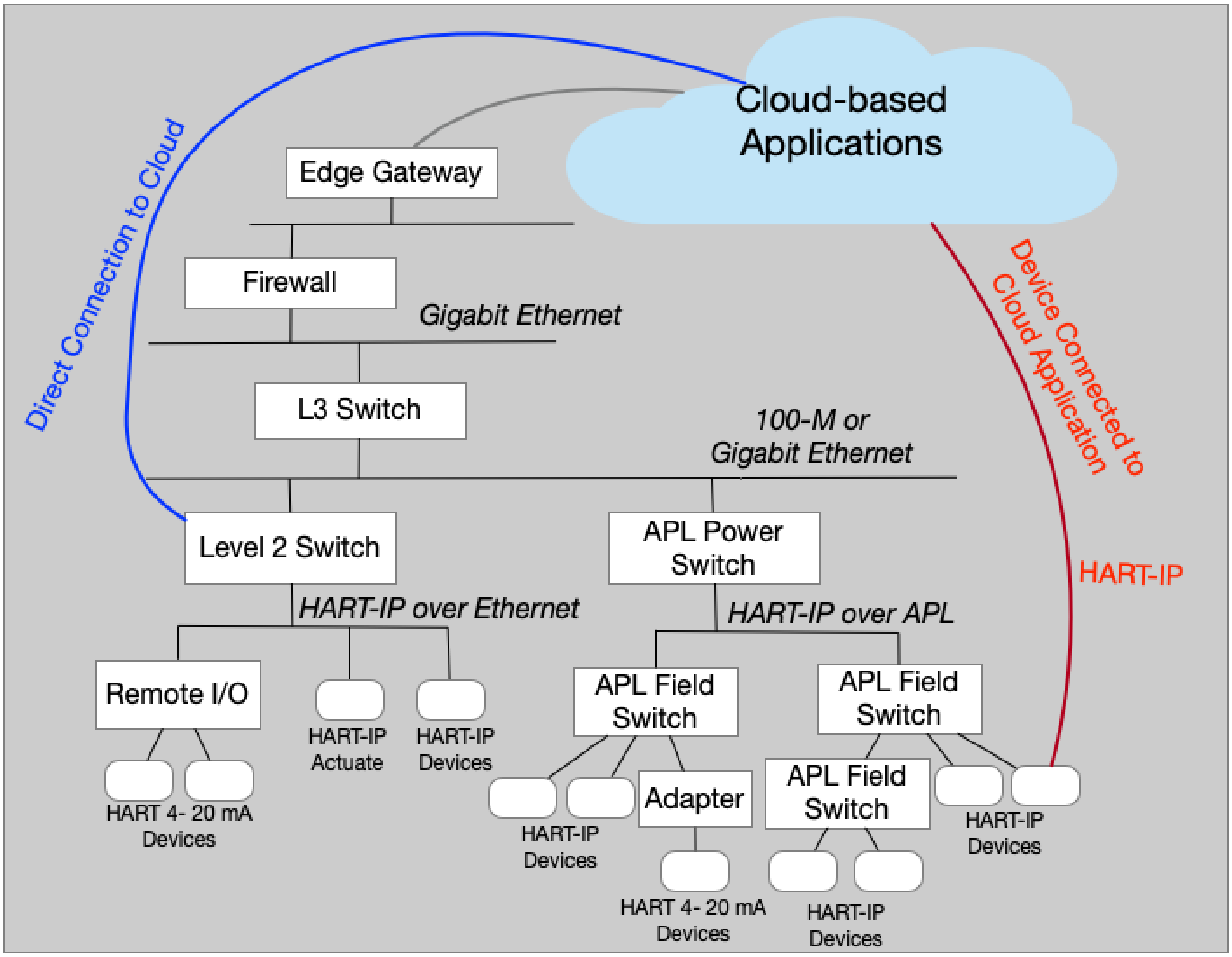


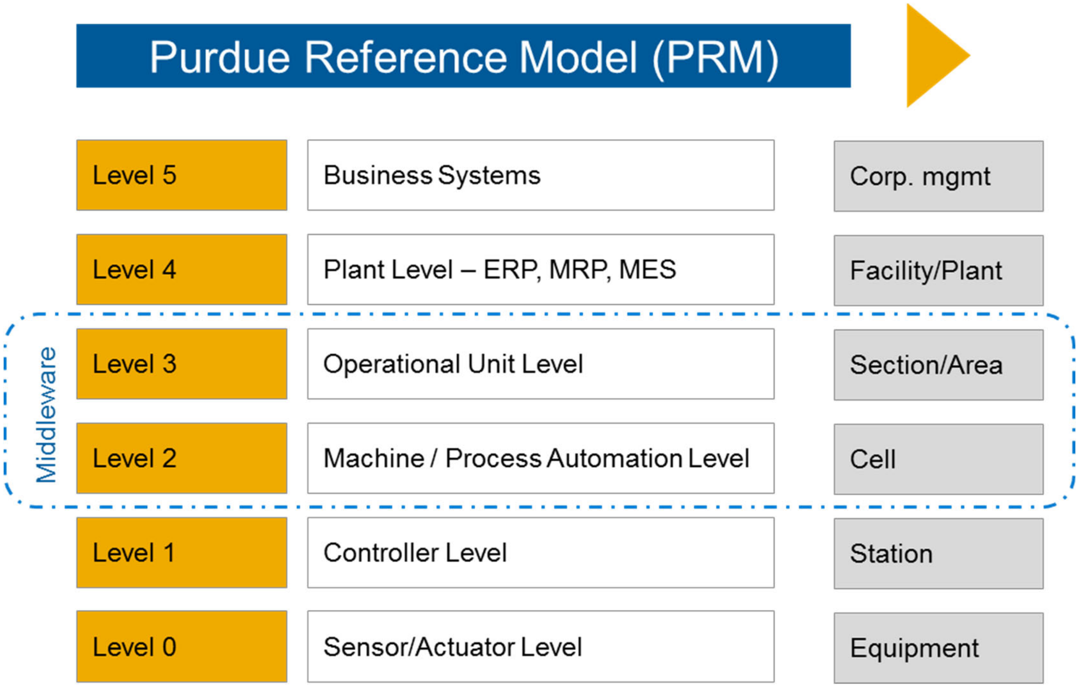
Figure 3 – Network Cloud Topology

Three different setups for cloud-based topologies are illustrated in Figure 3. The first is through an edge gateway which collects data from the process plant and passes the data on to higher level applications such as condition monitoring. The second topology shows a direct connection from an L2 switch to the cloud. The third shows a direct connection to cloud based applications, a scenario for this may be an instrument mounted in a remote location that is used to report readings back to a centralized monitoring system. Cloud monitoring applications use a separate session from the control and asset management sessions to connect to the device. Each of these setups also introduces additional security requirements and threats. More information on these is included in the threat discussions.

3. SECURITY THREAT AND RISK MODELS

3.1 Overview

Control system cyber security and control system design and operation, which often includes instrumentation and instrumentation networks, makes the assumption that instruments and instrumentation networks are secure because they are physically isolated and protected by the control system networks. In support of this claim, control system implementations often include considerable use of firewalls, network monitoring, and other defensive mechanisms positioned between layers as defined by the Purdue Reference Model, shown in Figure 4.



Level 5	Business Systems	Corp. mgmt
Level 4	Plant Level – ERP, MRP, MES	Facility/Plant
Level 3	Operational Unit Level	Section/Area
Level 2	Machine / Process Automation Level	Cell
Level 1	Controller Level	Station
Level 0	Sensor/Actuator Level	Equipment

Figure 4 – Purdue Reference Model

In reality, field devices and field device networks themselves are often not monitored which makes anomaly detection difficult. They are also not protected which makes attacking them a target. This is often further worsened when the protocols are converted to other protocols, such as MODBUS, where semantics and context are lost (e.g., unit codes). When clients receive the measured value, such as a HART 4-20 mA value converted to a MODBUS value, the client will not be aware that values have been altered. What is needed is a way to securely tunnel the original HART value, including its original status and unit code, from the HART device to the control applications and other clients. All of this is addressed with HART-IP.

4. HOW HART-IP IMPLEMENTS SECURITY

4.1 Overview

HART-IP includes several provisions that are designed to protect the device and the device network. These provisions include the use of a secure transport TLS, audit logging, and an interface to Syslog to persist audit log. HART-IP products implement the security features specified in the HART-IP network management specification [1]. These security features include:

- Communication security based on Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)
- HART-IP Server local Audit Logs
- Publishing syslog messages
- Secure operation of the products themselves

Security best-practices for IP-based products continue to evolve. Consequently, all products must support field upgrades allowing the addition of improved security mechanisms in the future. The following section covers HART-IP security details.

4.2 HART-IP Security

4.2.1 Communication Security

HART-IP devices (servers and clients) support the following communication security protocols:

- TLS Version 1.2 and 1.3
- DTLS Version 1.2 and 1.3

For both TLS / DTLS, secure sessions are initiated using either Pre-Shared Key (PSK) or Secure Remote Password (SRP) key exchange algorithms.

The AES-128 symmetric cipher are used as the TLS encryption algorithm. Devices expected to support the following TLS Cipher Suites in the listed preference order:

- TLS_PSK_WITH_AES_128_GCM_SHA256 (0x00, 0xA8)
- TLS_PSK_WITH_AES_128_CBC_SHA256 (0x00, 0xAE)
- TLS_PSK_WITH_AES_128_CCM (0xC0,0xA4)
- TLS_SRP_SHA_WITH_AES_128_CBC_SHA (0xC0, 0x1D)

Devices support both a default PSK and Password. These defaults are used to create an initial session that is used to write the server's PSK and/or Password. Servers decline all subsequent Session Initiate requests if the PSK or Password (at least one of them) is not written during that initial session. Once written, the PSK and/or SRP keys are used by clients to establish secure sessions with the server. The server can support multiple simultaneous sessions. If the initial session fails to write the server's PSK and/or Password, factory reset must be used to reset the device so that the provisioning can be tried again. The factory reset may also be used to recover a device that has been quarantined. A short introduction to TLS is provided in Annex C.

4.2.2 Audit Logs

HART-IP servers support audit logs. These servers record the following in local, volatile Audit Logs:

- Last time/date the server was powered up
- The last time/date the security credentials were modified
- Server Status
- Circular 128 entry Session Summary list

Each Session Summary record consist of:

- Client identity including IP addresses (IPv4, IPv6) and Port Numbers
- Connect/disconnect time/date for that client
- Starting / ending Configuration Change Counter
- Session status
- Communications counters indicating the number of publish (burst), request, and response PDUs.

Audit logs include an 8-byte unsigned integer timestamp.

4.2.3 Syslogs

Devices also support / push "syslog" messages. Syslog messages are pushed to Security Information and Event Management (SIEM) systems and are critical to detecting security attacks and unauthorized manipulation of HART-IP field devices. Detection enables improvements to plant security policies and procedures to minimize vulnerabilities.

4.3 Comparing HART, WirelessHART, HART-IP

HART 4-20mA has little cybersecurity support because it is not possible to access a device without physical connection via a handheld or host system. Many of the security features in WirelessHART have equivalent features in HART-IP. Table **1** compares these features.

Table 1 – Comparing HART, WirelessHART and HART-IP

Security Feature	HART 4-20 mA	WirelessHART	HART-IP
Join Key		X	X
Join Key Provisioning		X	X
Network Keys		X	X
Encryption (mandatory)		X	X
Secure Sessions		X	X
Security Manager		X	
Network Management		X	
Message Authentication Codes (or Hashes)		X	X
Audit Logs			X
Quarantining device (or bricked device)		X	X
Whitelists and Blacklists		X	
Limited Sessions	X	X	X
Throttling – limit number of outstanding transport messages	X	X	X
Burst messages contain timestamps	X	X	X
Use of sequence numbers in messages	X	X	X
Factory reset	X	X	X
Synchronized time	X	X	X

5. IMPLEMENTING A HART-IP DEVICE

5.1 Overview

Implementing HART-IP devices requires supporting the secure features described in the HART Network Management Specification [1]. To complement these security features, device manufactures should also implement best design practices described in this section. These design practices are illustrated in Figure 5.

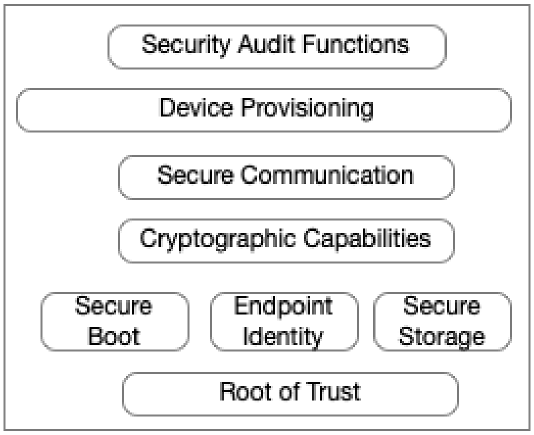


Figure 5 – Security Reference Model

5.2 Root of Trust

Each HART-IP device should contain a Root of Trust that forms the basis for the devices' security. Trust in embedded security refers to an expectation that a HART-IP device is operating as designed. Software trusts that hardware is operating as it should be. Applications trust that the operating system is not corrupting device configuration. Remote systems trust in the device's identity to which it's connected. This process of establishing trust is called attestation. A device's root-of-trust is the point where attestation and authentication start and then extends through each layer. A root of trust is an important building block to secure HART-IP devices.

A root of trust can be established by a variety of methods. The simplest mechanism is to run start-up code directly from a non-writable location in the processor's memory map. Alternatively, to allow updates and patches, the code can be loaded from a protected memory region into a protected memory store of some sort set aside for firmware execution. The important aspect for a root of trust is to be sure that the initial code is what the manufacturer intended, before execution. When it starts, the root of trust derives its internal keys from supplied device identity inputs and executes self-tests and code validation for itself.

5.3 Secure Boot

A secure boot process prevents the execution of unauthorized code at the time of device power up and prevents the exposure of embedded boot code and software. A secure boot process can be accomplished in many different ways, including using digitally signed binaries, secure and trusted boot loaders, boot file encryption, and security microprocessors.

While secure boot center around digitally signed boot files, unless those signatures are verifiable using some sort of an immutable root of trust, however, it is not secure. To validate boot files a digital key will need to be installed when the device is manufactured or after manufacturing using a trusted application. Considerations include:

- Protecting IP – protects software IP such as proprietary algorithms.
- Trusted remediation – The ability to safely remediate in case of device failure or compromise is a critical ability that relies on having a secure boot process that checks the validity of the firmware image being booted with a root of trust.
- Secure firmware update – Validating incoming payloads intended to replace existing firmware images is critical to maintaining device integrity throughout the system lifecycle. The source of the payload and the payload itself must be validated prior to being applied, and with a properly implemented secure boot process, failure to validate results in a safe rollback to a known verified image.
- Secure connectivity to cloud resources – A secure boot process ensures that the device is authenticated with the cloud each time it attempts a connection through the use of embedded keys and certificates.

5.4 Endpoint Identity

Endpoint identity is a fundamental building block essential for most other security measures. Piece of identity to uniquely identify devices. In HART-IP the device identity is tied the device's Unique ID.

5.5 Secure Storage

Program storage is likely to be in off-chip Flash memory, whose contents are copied into SRAM or external DDR memory to run once the device is booted. The hardware root of trust protects the device's unique key, and the keys and facilities that derive from it. It provides secure storage for the rest of the chip, and acts as an identification or authentication device.

5.6 Cryptographic Capabilities

HART-IP security requires proper use and implementation of cryptography across transport protocols (data-in-motion), storage (data-at-rest), and applications (data-in-use). Although this paper makes recommendations for storage and applications, these are ultimately the responsibility of the device suppliers. Cryptographic services include:

- Standards-based symmetric cipher suites (PSK and SRP), hashing functions, and random number generators of appropriate strength
- NIST/FIPS standards-based validated cryptographic algorithm implementations
- Interoperability of cryptographic keys

5.7 Secure Communications

A secure end-to-end communications protocol stack is required. HART-IP specifies the following:

- Secure transport: TLS-SRP or TLS-PSK
- Hash functions: SHA-2 (hash function for generating MAC codes)
- Encryption: AES-128

5.8 Device Provisioning

Device provisioning is used to write the following into the device:

- Device tag
- Pre-shared key
- Secure remote password
- Syslog server hostname and port
- DNS server name
- (optional) change the default port number that devices listen on
- (optional) provide a static IP address and mask

5.9 Security Audit Functions (Syslog)

For incident response and audits, event logs are required as a valuable input that can be used to measure and assess risks continually and mitigate threats. This requires the ability to:

- capture system and security events (it may not be a security event)
- transfer events to a host using Syslog
- an application to review the system events

6. DEVICE LIFECYCLE

6.1 Overview

HART-IP devices are installed with a long service life expectancy. For example, some devices will be operational for 15 to 30 years. Over the device's lifespan, plant overhauls, process modifications, and business integration activities can have security implications. Consequently, it is necessary to manage security through the entire life cycle. The phases of the lifecycle are:

1. Procurement
2. Provisioning
3. Installation and Configuration

4. Operation

5. Maintenance

6. Replacement

7. Decommissioning & Disposal

The phases in the device lifecycle are illustrated in Figure 6.

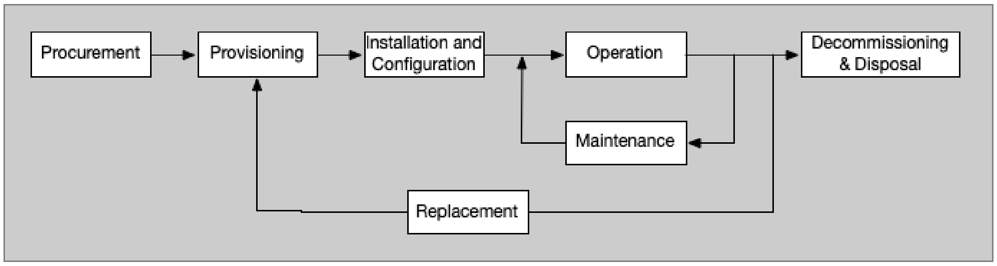


Figure 6 – HART-IP Device Lifecycle

Each of the phases in the lifecycle is discussed in the following sections.

6.2 Device Procurement

The first stage in the device lifecycle is procurement. Procurement is outside of the scope of this paper.

6.3 Device Provisioning

Devices should be provisioned before they are installed in the plant. This provisioning is often referred to as onboarding. During the provisioning phase the devices should have the following set:

- The device name and the unit name it will be installed
- The TLS PSK and SRP that will be used to establish a session
- The hostname and port used to connect to the Syslog Server

All of this information should be setup in a secure location. The information will also have to be made available to the hosts that will open sessions with the devices.

6.4 Device Installation and Configuration

Devices are configured using FDI. From a life cycle point-of-view here are some considerations:

1. Configuration management

   a. Define standard configuration templates that cover the security settings mentioned in the provisioning section. If deviations from the template are necessary to accommodate the installation of the device, or the environment it is being installed into, these changes should be properly managed.

2. Create a baseline snapshot of the configuration of a HART-IP device

    a. Having a baseline will help with troubleshooting issues and investigating security incidents.

    b. Before a device is placed in the production environment and/or shortly after, a configuration baseline snapshot should be established.

6.5 Device in Operation

Once the device is installed and in production, it will need to be monitored as part of regular operations. To assist with this, HART-IP supports audit logging. These audit logs are summarized in a Syslog server.

6.6 Device in Maintenance

Device maintenance is a regular part of the plant maintenance. Process sites have operating and maintenance procedures for checking devices and performing maintenance on them. An additional step that may need to be is to support firmware upgrades to HART-IP devices. Firmware upgrades are required for all IP-based devices in order to incorporate security changes in addition to feature improvement.

Periodically, a snapshot of the running configuration of the devices should be taken and compared to the baseline configuration snapshots to uncover any risk or unexpected deviations in configuration. These snapshots become important when devices must be replaced.

6.7 Device Replacement

Most sites have a device replacement procedure. As long as a device is being replaced by a device of the same revision from the same manufacture the procedure should go smoothly. If a different device version is being used, then additional testing will need to be performed to validate that the devices work the same. If a device from a different manufacture is being used, then additional testing will need to be performed to validate that the measurements are consistent and the integration with the control strategy works the same. The device setup for burst or publishing will also have to be validated.

6.8 Device Decommissioning and Disposal

When it comes time to retire a device, some care should be taken with the way it is disposed of. This includes:

1. Efficient data sanitization of the device storage which may include memory cards and flash memory.
    a. NIST has guidelines (need to reference them if we keep any of this)
2. Decide which devices can be repurposed, reused, or recycled:
    a. Some devices could be reused in different areas of the process.
    b. Some devices will work well for training, test, and development environments.
3. Efficient disposal procedures:
    a. Are you simply throwing the devices in the garbage can?
    b. Should some equipment be physically destroyed in order to minimize the chance of data leakage?

7. CONCLUSIONS

The major thing to consider is that HART-IP devices are still HART devices, only in this case running over an IP-based network. Many process installations have hazardous locations. For these sites HART-IP over the Advanced Physical Layer (APL) may be used to support both installation in these hazardous locations and device power. Since APL is a 10 M connection, it will be possible to send considerably more measurement and device information from devices to the hosts.

Ethernet support within the FieldComm group is not new. HART-IP, which can be used with ethernet, was first specified as part of HART 7 over ten years ago. This initial specification was largely intended for communications between hosts and gateways. In the latest release of HART-IP, support for HART-IP devices is now included. As part of these enhancements to HART-IP, security is now explicitly specified.

ANNEX A. KEEPING UP TO DATE

There are a number of ways to keep up to date on threats and attacks. One way is following the latest treads and alerts, and another is to follow research that is being done.

A.1.1. Sources of Information

There are many sources of information on threats and attacks. Some of the keys are summarized below:

- Automated vulnerability management from NIST: The National Vulnerability Database (https://nvd.nist.gov/)
- General cybersecurity alerts: The US Computer Emergency Readiness Team (US-CERT) (https://www.us-cert.gov/ncas)
- Industrial control system threat information: The Industrial Control System Cyber Emergency Response Team (ICS-CERT) (https://ics-cert.us-cert.gov)
- Medical device and health information cybersecurity sharing: The National Health Information and Analysis Center (NH-ISAC) (http://www.nhisac.org)

ANNEX B. STRIDE

STRIDE stands for **S**poofing, **T**ampering, **R**epudiation, **I**nformation Disclosure, **D**enial of Service (DoS), and **E**levation of Privilege. STRIDE was initially proposed for threat modeling but is now used more broadly. The Microsoft-developed threat tool is used in the evaluation and in this document to evaluate security threats. A brief description of each of these is:

> **Spoofing** is pretending to be something or someone you're not.
>
> **Tampering** is modifying something you're not supposed to modify. It can include packets on the wire (or wireless), bits on disk, or the bits in memory.
>
> **Repudiation** means claiming you didn't do something (regardless of whether you did or not).
>
> **Information Disclosure** is about exposing information to people who are not authorized to see it.
>
> **Denial of Service** (DoS) are attacks designed to prevent a system from providing service, including by crashing it, making it unusably slow, or filling all its storage.
>
> **Elevation of Privilege** is when a program or user is technically able to do things that they're not supposed to do.

Table 2 provides a summary of each of the threat types and the security property it is associated with.

Table 2 – STRIDE Threat Types and Security Properties

Threat Category	Threat Description	Security Property
Spoofing identity	An example of identity spoofing is illegally accessing and then using another devices authentication information, such as its unique device ID.	Authentication
Tampering with data	Tampering threats come in several flavors, including tampering with bits on disk, bits on a network, and bits in memory. There are three main ways to address tampering threats: relying on system defenses such as permissions, use of cryptographic mechanisms, and use of logging technology and audit activities as a deterrent. Permissions are not used in field devices. They are used in hosts such as PAM, Controllers, IO Servers, Edge Gateways, and so on.	Integrity

Threat Category	Threat Description	Security Property
Repudiation	Repudiation threats are associated with users or devices who deny performing an action without other parties having any way to prove otherwise. Nonrepudiation refers to the ability of a system to counter repudiation threats. For example, a digital signature may have to be validated before a packet is accepted. Logging may be used as evidence that the packet was accepted. In HART-IP devices and networks, statistics will be kept on the number of packets and commands processed.	Non-repudiation
Information disclosure	Information disclosure can happen with information at rest (in storage) or in motion (over a network). The information disclosed can range from the content of communication to the existence of an entity with which someone is communicating.	Confidentiality
Denial of service	Denial-of-service (DoS) attacks work by disrupting access to some resource. DoS often cause the targeted system to either run out of resources or crash a service or block access to it. Traditionally, those resources are CPU, memory (both RAM and hard drive space can be exhausted), and bandwidth. Denial-of-service attacks can also exhaust user access to the system and devices.	Availability
Elevation of privilege	Devices do not implement user authorization and roles. Applications connecting to devices is where privileges and roles are implemented. That said, there are a number of mitigation that devices support. These include: 1. Limiting the attack surface by only supporting a limited set of well-defined functions 2. Providing a configuration lock 3. Tracking configuration changes 4. Utilizing an audit log	Authorization

ANNEX C. A SHORT INTRODUCTION TO TLS

C.1.1. What is TLS

Transport Layer Security, or TLS, is a widely adopted transport-layer protocol designed to facilitate privacy and data security for communications over the Internet. A primary use case of TLS is encrypting the communication between web applications and servers, such as web browsers loading a website. In these web-based scenarios the TLS handshake uses certificates and a Public Key Infrastructure (PKI) for mutual authentication and key exchange. However, in deployments such as such device level networks, a public-key based handshake is not necessary. In such cases a more lightweight approach is to use a secret that is shared in advance between the TLS client and TLS server via out-of-band means, for example when the device is onboarded or commissioned at the end-user site. In these cases, a set of pre-shared key (PSK) based cipher suites for TLS or a set of secure remote passwords (SRP) cipher suites for TLS may be used. An example cipher suite for PSK is shown below:

TLS_PSK_WITH_AES_128_CBC_SHA256

- TLS defines the protocol that this cipher suite is for; for HART-IP it is TLS 1.2 or 1.3.
- PSK indicates the key exchange algorithm being used. The key exchange algorithm is used to determine if and how the client and server will authenticate during the handshake.
- AES_128_CBC indicates the block cipher being used to encrypt the message stream, together with the block cipher mode of operation.
- SHA256 indicates the message authentication algorithm (hash function) which is used to authenticate a message.

A pre-shared key or secure remote password is simply a secret that both the client and the server know, and that can be used to derive symmetric keys for the session. The next section provides more details on the TLS-PSK handshake.

C.1.2. TLS Handshake

At a high level, TLS communications are split into two phases:

- A handshake phase where a secure communication is negotiated and created between two participants.
- A post-handshake phase where communications are encrypted between the two participants.

These handshakes are illustrated in Figure 7.

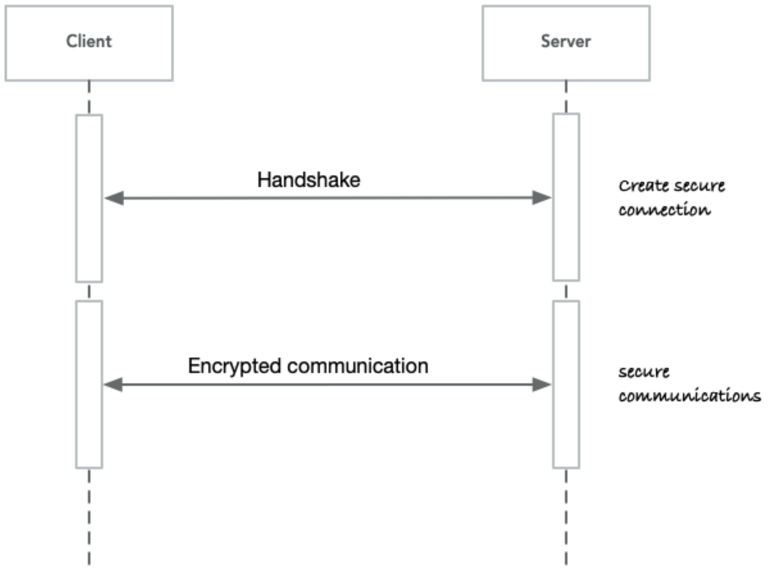


Figure 7 – High Level View of TLS Handshake

The handshake for TLS-PSK 1.2 works by having the client advertise in its Client Hello message the list of cipersuites that it supports. The server selects one of the ciphers and responds. The TLS-PSK 1.2 handshake is illustrated in Figure 8.

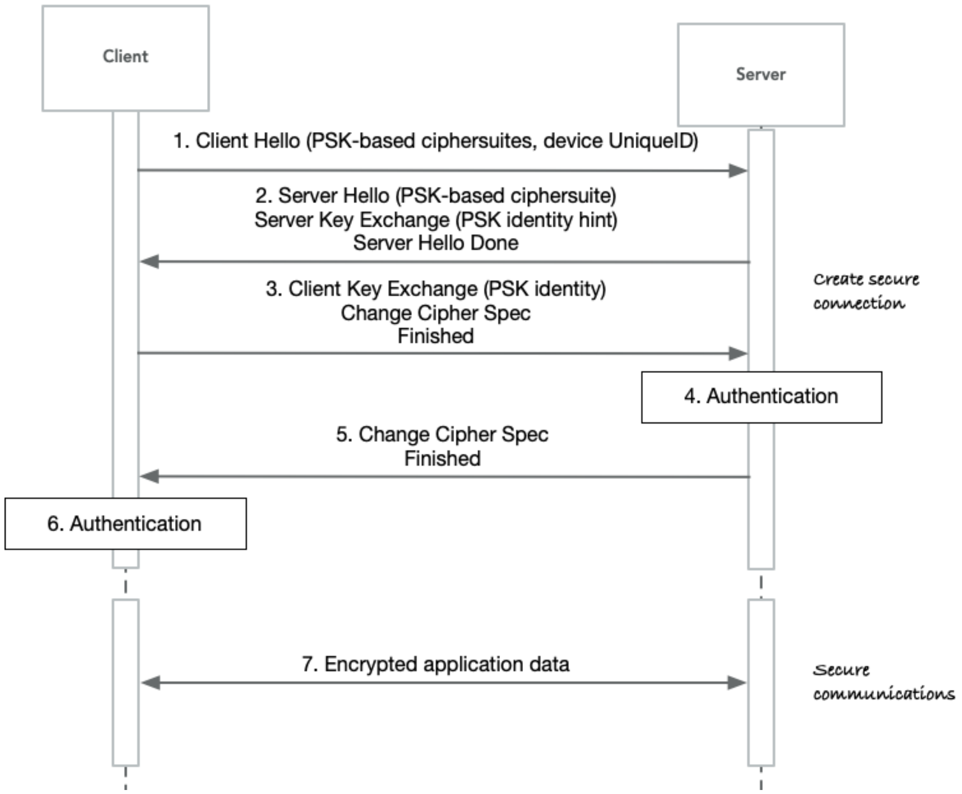

Client
Server
1. Client Hello (PSK-based ciphersuites, device UniqueID)
2. Server Hello (PSK-based ciphersuite)
Server Key Exchange (PSK identity hint)
Server Hello Done
3. Client Key Exchange (PSK identity)
Change Cipher Spec
Finished
Create secure connection
4. Authentication
5. Change Cipher Spec
Finished
6. Authentication
7. Encrypted application data
Secure communications

Figure 8 –TLS-PSK 1.2 Handshake

The steps in the handshakes as described below:

1. Client sends Client Hello message with:
    a. one or more PSK-based ciphersuites
    b. device Unique ID
2. Server responds with Server Hello message:
    a. select one of the PSK-based ciphersuites that were included in the Client Hello message and send this to the client in the Server Hello message
    b. include a string “some string” to be used as a PSK identity hint
    c. indicate that the Server Hello done
3. The Client sends the following:
    a. Client Key Exchange (with PSK identifier)
    b. Change Cipher Spec
    c. Finished
4. The Server does the following:
    a. Authenticates
5. The Server does the following:
    a. Change Cipher Spec
    b. Finished
6. The Client does the following:
    a. Authenticates
7. Client and Server exchange encrypted application data

The client should indicate which key to use by transmitting a “PSK identity” to the server because both clients and servers may have pre-shared keys with several different parties. As shown in Figure 8, the “PSK identity” is included in the Client Key Exchange message and transmitted to the server. After the negotiation for “PSK identity” is done, the client and the server can generate their pre-master secrets with the pre-shared key. As normal TLS process, the Finished message is computed for the authentication of the TLS handshake. Since the PSK key exchange mechanism uses only symmetric key algorithms, there is no burden caused by public key operations. Hence, the handshake time of PSK should be very fast.

TLS Handshake Protocol negotiates a session, which is identified by a Session ID.

ANNEX D. REFERENCES

[1] FieldComm Group, https://fieldcommgroup.org

[2] Shostack, A., “Threat Modeling: Designing for Security”, Wiley, 2014.

[3] Vacca, J., “Computer and Information Security Handbook, 3rd Edition”, Morgan Kaufmann, 2017.

[4] HART Specifications

[5] The Transport Layer Security (TLS) Protocol Version 1.3, https://tools.ietf.org/html/rfc8446

AUTHORS

FieldComm Group would like to thank the authors and contributors to this technical paper:

Mark Nixon