



Securing EtherNet/IP Using DPI Firewall Technology

Technical Track

www.odva.org



Erik Schweigert

- ▶ Leads device firmware development at Tofino Security
- ▶ BSc in Computer Science from VIU



Michael Thomas

- ▶ Leads management/configuration software development at Tofino Security
- ▶ BSc in Computer Science from VIU

Introduction

- ▶ Industry has embraced technology like Ethernet and TCP/IP
- ▶ Increases efficiency, but also increases exposure to security threats
- ▶ A simple, robust security solution is needed
- ▶ In this talk we will look at using Deep Packet Inspection (DPI) technology for securing EtherNet/IP and CIP Protocols

- I. The Common Industrial Protocol (CIP) Overview
- II. Security Threats to EtherNet/IP Deployments
- III. Possible Methods to Secure EtherNet/IP
- IV. Analysis of EtherNet/IP for Deep Packet Inspection (DPI)
- V. Implementing DPI for EtherNet/IP
- VI. Field Usability of DPI for EtherNet/IP
- VII. Case History



The Common Industrial Protocol

General Overview

The Common Industrial Protocol

- ▶ CIP can be used above varying network protocols including DeviceNet, ControlNet, EtherNet/IP, and CompoNet
- ▶ EtherNet/IP uses the standard Ethernet layer of TCP/IP
- ▶ CIP is based on abstract object modeling
 - Object classes
 - Services

The Common Industrial Protocol

- ▶ Two types of Communication:
 - Class 1 implicit messaging – low jitter and latency (typically UDP)
 - Class 3 explicit messaging – reliable connection based (typically TCP)
- ▶ The CIP layer is complex, but follows a consistent format
- ▶ It can map itself to various functions, physical entities, and products



Security Threats to EtherNet/IP Implementations

US ICS-CERT Security Advisories

Prior to Stuxnet:

- ▶ 5 security advisories
- ▶ 3 vendors involved

2011:

- ▶ 215 disclosed vulnerabilities
- ▶ 104 security advisories
- ▶ 39 vendors involved

2012

- ▶ 248 disclosed vulnerabilities

2013

- ▶ ~ 400 disclosed vulnerabilities

**40% of disclosed
vulnerabilities included
working attack code**

Security Threats to EtherNet/IP Deployments

- ▶ Linking corporate and control systems together increases exposures to threats
- ▶ Protocols like EtherNet/IP not secure by design
- ▶ The lack of authentication presents a vector for attack
- ▶ Threats can also stem from implementation issues

Lack of Command Granularity

- ▶ July 2008 DHS warning to energy companies:
“A vulnerability has been identified and verified within the firmware upgrade process used in control systems deployed in Critical Infrastructure and Key Resources (CIKR)... development of a mitigation plan is required to protect the installed customer base and the CIKR of the nation.
Firmware Vulnerability Mitigation Steps [include]
***blocking network firmware upgrades with appropriate firewall rules.*”**
- ▶ BUT traditional firewalls only provide complete protocol allow or deny



Methods to Secure EtherNet/IP

Methods to Secure EtherNet/IP

- ▶ Security issues that are part of the current EtherNet/IP standards and implementations are likely to remain with us for at least the next decade
- ▶ Industry must explore methods to secure these existing protocols and systems, independent of future improvements to the specifications
- ▶ There are two primary technologies used in the IT world to secure on-the-wire messages
 - Encryption, signing and cryptographic techniques
 - Packet filtering

Encryption of Data and Virtual Private Networks

- ▶ One common method of securing communications is to use cryptographic techniques, such as encryption-based tunneling
- ▶ Usually referred to as Virtual Private Network (VPN)
- ▶ VPN technologies create a secure 'tunnel' between two end points over an untrusted network

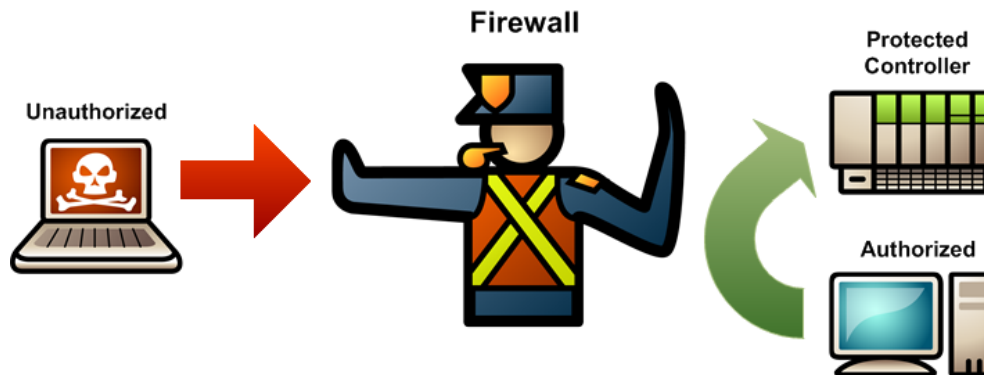


Encryption of Data and Virtual Private Networks

- ▶ VPNs provide three key capabilities:
 - Privacy
 - Authentication
 - Integrity
- ▶ Limitations:
 - Overhead of encryption – negative impact on time sensitive communications
 - No data validation – Bad data in means bad data out
 - Reliable key or certificate management is challenging

Packet Filtering via Firewalls

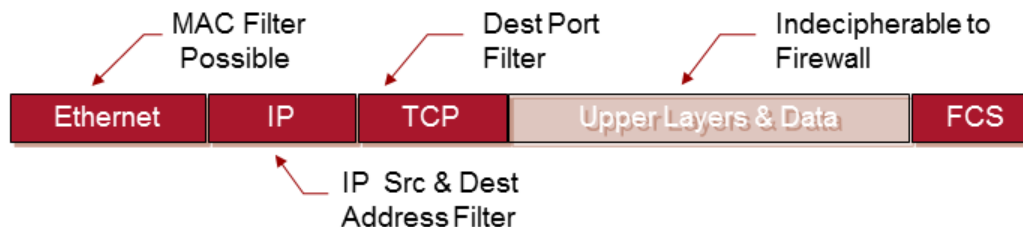
- ▶ A firewall is a device that monitors and controls traffic flowing in or between networks
- ▶ Traditional IT firewalls apply filters at the TCP and IP layers of a message, using Access Control Lists (ACLs) to check three primary fields in a message:
 - Address of the computer sending the message
 - Address of the computer receiving the message
 - Application layer protocol contained by the IP message



Limitations of Traditional Firewalls

- ▶ IT firewalls do not allow for fine grained control
- ▶ This is an issue – as discussed, SCADA/ICS protocols have no granularity
 - A data read message looks EXACTLY like a firmware update message to IT firewall
 - If you allow data read messages, you are also allowing programming or firmware upgrade messages to pass through - This is a serious security risk

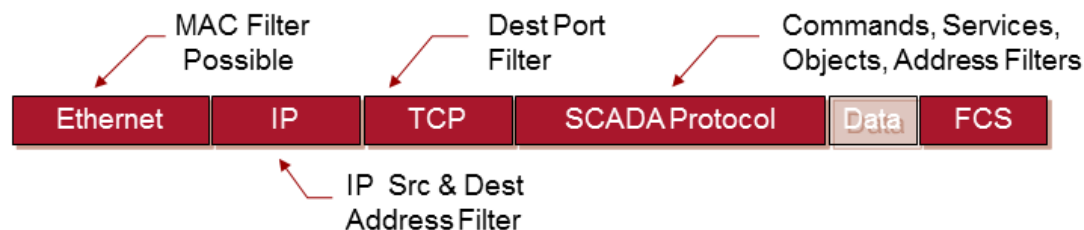
What a traditional IT firewall can understand and filter on



Firewalls and Deep Packet Inspection

- ▶ Deep Packet Inspection (DPI) is an extension to traditional firewall technology that can provide the fine grained management of EtherNet/IP traffic
- ▶ DPI allows the firewall to understand what tasks the protocol is being used for (e.g. read, write, etc)
- ▶ It can interpret specific ICS protocols and apply filters on fields and values that matter to control systems
- ▶ It can also check the validity of the messages – do they conform to the specifications?

What a SCADA DPI firewall can understand and filter on



What Deep Packet Inspection Is and Is Not

- ▶ DPI is:
 - The deterministic and repeatable filtering of packets based on parsing at all layers
 - Think of it as Wireshark in real-time
- ▶ DPI is not:
 - A signature based intrusion detection system (IDS) like SNORT
 - An authentication mechanism



Analysis of EtherNet/IP for Deep Packet Inspection (DPI)

Analysis of EtherNet/IP for Deep Packet Inspection (DPI)

- ▶ Must understand the structure of the protocol in its most basic form; that is bits and bytes
- ▶ For example, consider the field “Length” in an EtherNet/IP message that is executing the Get Attribute All service on an Analog Input object
 - What exactly does “Length” indicate?
 - Is there a minimum or maximum that, if violated, might indicate an attempted attack?
 - Would these values change depending on the CIP object or service involved?

EtherNet/IP Header Structure

- ▶ The EtherNet/IP header is a 24 byte fixed length header with a trailing optional data portion
- ▶ The “Command Specific Data” field encapsulates CIP

Table 2-3.1 Encapsulation Packet

Structure	Field Name	Data Type	Field Value
Encapsulation header	Command	UINT	Encapsulation command
	Length	UINT	Length, in bytes, of the command specific data portion of the message, i.e., the number of bytes following the header
	Session handle	UDINT	Session identification (application dependent)
	Status	UDINT	Status code
	Sender Context	ARRAY of octet	Information pertinent only to the sender of an encapsulation command. Length of 8.
	Options	UDINT	Options flags
Command specific data	Encapsulated data	ARRAY of 0 to 65511 octet	The encapsulation data portion of the message is required only for certain commands



CIP Messaging Structure

- ▶ The CIP header has multiple static fields and a request path that contains a set of CIP segments
 - This includes port segments, logical segments, network segments, symbolic segments, data segments, and key segments
 - Not every packet contains every type of segment.
- ▶ There is always a CIP service executing some action upon the logical class segment

CIP Messaging Structure

- ▶ For DPI design, the logical object class is a main field of interest, with the logical segment encoding being the most important
- ▶ The “logical segment” is an 8 bit field with
 - The 3 highest bits denote the “logical segment type”
 - The following 3 bits denote the “logical type”
 - The last 2 bits denote the “logical format”
 - Used to identify the size of the “logical value” that follows the “logical segment” field

	Logical Type		
Class ID	0	0	0
Instance ID	0	0	1
Member ID	0	1	0
Connection Point	0	1	1
Attribute ID	1	0	0
Special	1	0	1
Service ID	1	1	0
Extended Logical	1	1	1
	Logical Format		
8-bit logical value	0	0	
16-bit logical value	0	1	
32-bit logical value	1	0	
Reserved for future	1	1	

CIP Messaging Structure

- [-] Common Industrial Protocol
 - [-] Service: Unknown Service (0x54) (Request)
 - 0... = Request/Response: Request (0x00)
 - .101 0100 = Service: Unknown (0x54)
 - Request Path Size: 2 (words)
 - [-] Request Path: Connection Manager, Instance: 0x01
 - [-] Path Segment: 0x20 (8-Bit Class Segment)
 - 001. = Path Segment Type: Logical segment (1)
 - ...0 00.. = Logical segment Type: Class ID (0)
 -00 = Logical segment Format: 8-bit Logical segment (0)
 - + 8-Bit Class Segment
 - [-] Path Segment: 0x24 (8-Bit Instance Segment)
 - 001. = Path Segment Type: Logical segment (1)
 - ...0 01.. = Logical segment Type: Instance ID (1)
 -00 = Logical segment Format: 8-bit Logical segment (0)
 - + 8-Bit Instance Segment

- [-] Transport type/trigger: 0x05
 - Connection Path Size: 4 (words)
 - [-] Connection Path: Port: 1, Address: 0, Message Router, Instance: 0x01, Connection Point: 0x01
 - [-] Path Segment: 0x01 (Port Segment)
 - [-] Path Segment: 0x20 (8-Bit Class Segment)
 - 001. = Path Segment Type: Logical segment (1)
 - ...0 00.. = Logical segment Type: Class ID (0)
 -00 = Logical segment Format: 8-bit Logical segment (0)
 - [-] 8-Bit Class Segment
 - Class: Message Router (0x02)

Analysis of EtherNet/IP Summary

- ▶ The architecture of the EtherNet/IP and CIP layers are not trivial and contain many moving parts
- ▶ Session establishment utilizes a set of commands with varying results and dynamic fields
- ▶ Session establishment spread over 3 layers, so DPI engine must communicate between layers
- ▶ CIP in itself carries complexity based on the type of connection, whether it is using a message router or Unconnected Send affects the packet structure



Implementing DPI for EtherNet/IP

What's Important?

- ▶ How can you interpret this complexity and design a usable filter mechanism to ensure credible network traffic in an efficient manner?
- ▶ Two main tasks for DPI:
 - Sanity Check – Does this message conform to spec?
 - Identify critical fields or actions - What makes sense to filter for security?
- ▶ Pick carefully - You don't want to sanity check or filter absolutely every field or else:
 - Latency will be excessive
 - The end user will be confused with too many options
 - Configuration mistakes will occur

Sanity Checking EtherNet/IP Messages

- ▶ There is no shortage of ways in which an attacker could harm a PLC by sending malformed frames
- ▶ Sanity checking needs to protect against the most dangerous field violations
- ▶ From the user's perspective, sanity checking is executed 'behind the scenes'
- ▶ Must be a selectable option because in some cases a vendor may fall short when attempting to adhere to the specification

Sanity Checking EtherNet/IP Messages



Managing Sessions When Packets are Dropped

- ▶ If a packet fails a sanity check, then it is dropped
- ▶ Any message that is dropped is part of a TCP stream
- ▶ If one message within a TCP stream is just silently dropped, then the TCP sequence count in the frame received by the target wouldn't align
- ▶ For many HMIs this will cause a lockup until the TCP timer resets the session (a long time)
- ▶ To mitigate this, a TCP reset must be sent to both parties to properly close and re-establish the session

Identifying Critical CIP Fields for Filtering

- ▶ Since CIP is an object based system with various CIP services acting upon these objects, it makes sense to filter on the fields that denote what object and service is being invoked
- ▶ Users can then specify object and service pairs that are safe for the firewall to allow (and block all others)
- ▶ The ODVA specification outlines a set of common services and optional object specific services that an object may adhere to. It also allows for vendor specific services
- ▶ Each object supports a specific set of CIP services



Field Usability of DPI for EtherNet/IP

Field Usability of DPI for EtherNet/IP

- ▶ Customers rarely understand what is “on-the-wire”
- ▶ 99% of the options are never used
- ▶ Must be easy to use, or customers will make mistakes

Field Usability of DPI for EtherNet/IP

- ▶ Start with the traditional IT Firewall as a platform
- ▶ Keeps all rules in one ACL style list
- ▶ Add DPI filtering to extend firewall filtering capabilities

!	Asset	Interface	Direction	Asset	Interface	Protocol	Permission	Log
<input checked="" type="checkbox"/>	Any	Supervisory Network	↔	Any	Control Network	🔌 ARP	✅ Allow	<input type="checkbox"/>
<input checked="" type="checkbox"/>	🖥️ HMI 2	Supervisory Network	➡	🏭 PLC 1	Control Network	🌐 HTTP	✅ Allow	<input type="checkbox"/>
<input checked="" type="checkbox"/>	🖥️ HMI 2	Supervisory Network	➡	🏭 PLC 1	Control Network	🌐 FTP	✅ Allow	<input type="checkbox"/>
<input checked="" type="checkbox"/>	🖥️ HMI 2	Supervisory Network	➡	🏭 PLC 1	Control Network	🏭 EtherNet/IP (CIP) Explicit Msg	🛡️ Enforcer	<input type="checkbox"/>
<input checked="" type="checkbox"/>	🖥️ HMI 2	Supervisory Network	↩	🏭 PLC 1	Control Network	🌐 NTP	✅ Allow	<input type="checkbox"/>

Rule Details

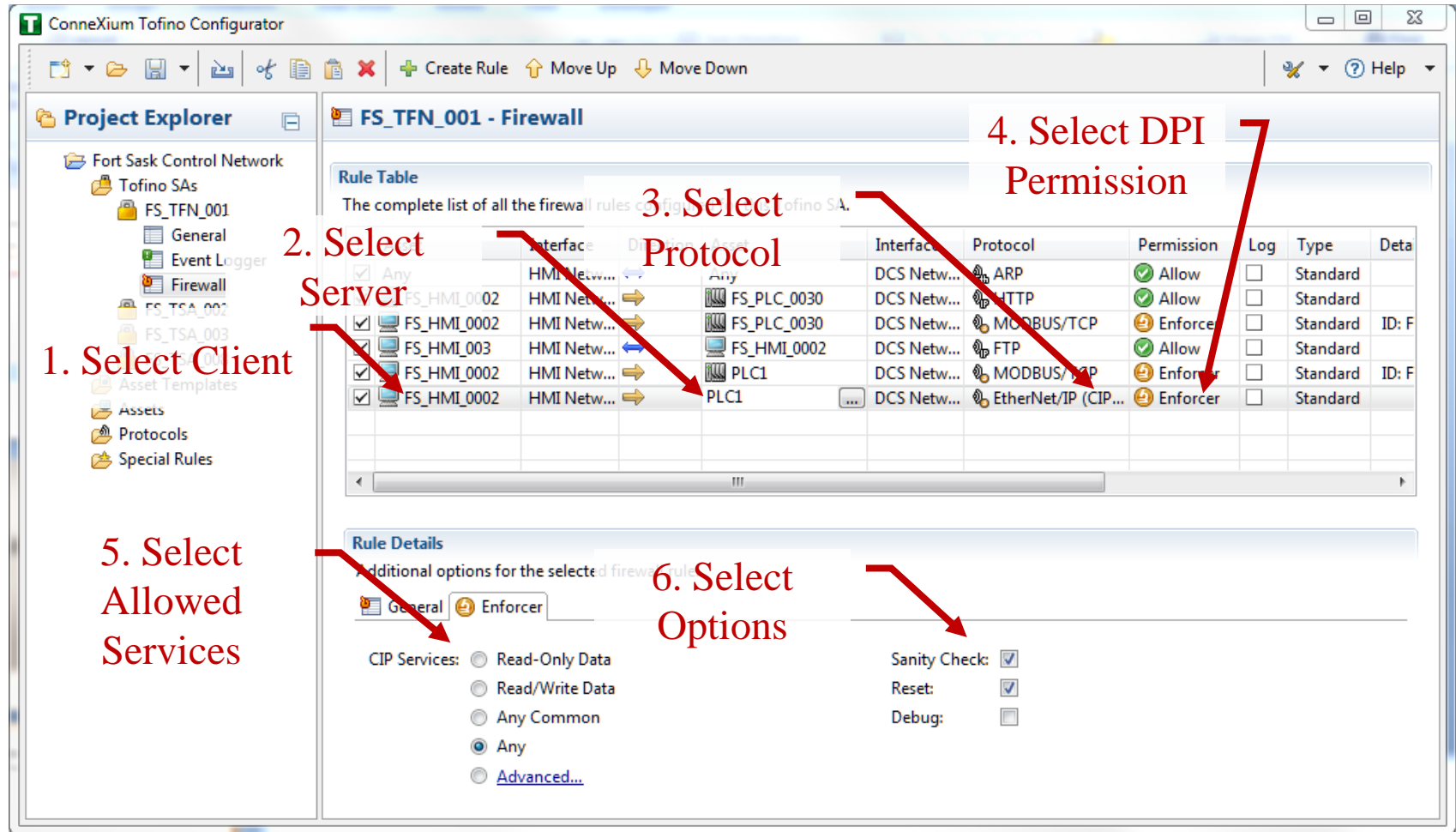
Additional options for the selected firewall rule

📄 General **🛡️ Enforcer**

CIP Services: Read-Only Data
 Read/Write Data
 Any
 [Advanced...](#)

Sanity Check:
Reset:
Debug:

DPI Security Made Easy



1. Select Client (points to FS_HMI_002 in the Rule Table)

2. Select Server (points to FS_HMI_002 in the Rule Table)

3. Select Protocol (points to FS_PLC_0030 in the Rule Table)

4. Select DPI Permission (points to Enforcer in the Rule Table)

5. Select Allowed Services (points to Any in the Rule Details CIP Services section)

6. Select Options (points to Sanity Check checkbox in the Rule Details section)

Interface	Destination Asset	Interface	Protocol	Permission	Log	Type	Details
Any	Any	DCS Netw...	ARP	Allow	<input type="checkbox"/>	Standard	
FS_HMI_0002	FS_HMI_0002	DCS Netw...	HTTP	Allow	<input type="checkbox"/>	Standard	
FS_HMI_0002	FS_HMI_0002	DCS Netw...	MODBUS/TCP	Enforce	<input type="checkbox"/>	Standard	ID: F
FS_HMI_0002	FS_HMI_0002	DCS Netw...	FTP	Allow	<input type="checkbox"/>	Standard	
FS_HMI_0002	PLC1	DCS Netw...	MODBUS/TCP	Enforce	<input type="checkbox"/>	Standard	ID: F
FS_HMI_0002	PLC1	DCS Netw...	EtherNet/IP (CIP...)	Enforcer	<input type="checkbox"/>	Standard	

Rule Details

Additional options for the selected firewall rule:

General Enforcer

CIP Services:

- Read-Only Data
- Read/Write Data
- Any Common
- Any
- Advanced...

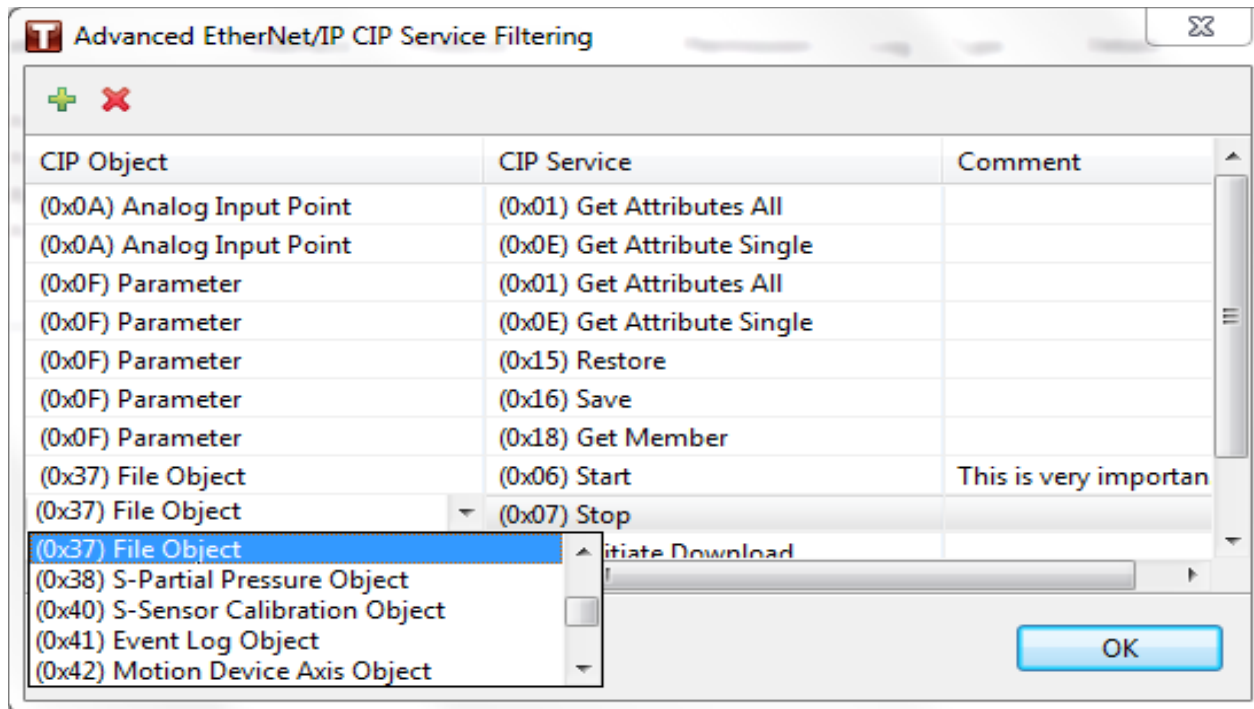
Sanity Check:

Reset:

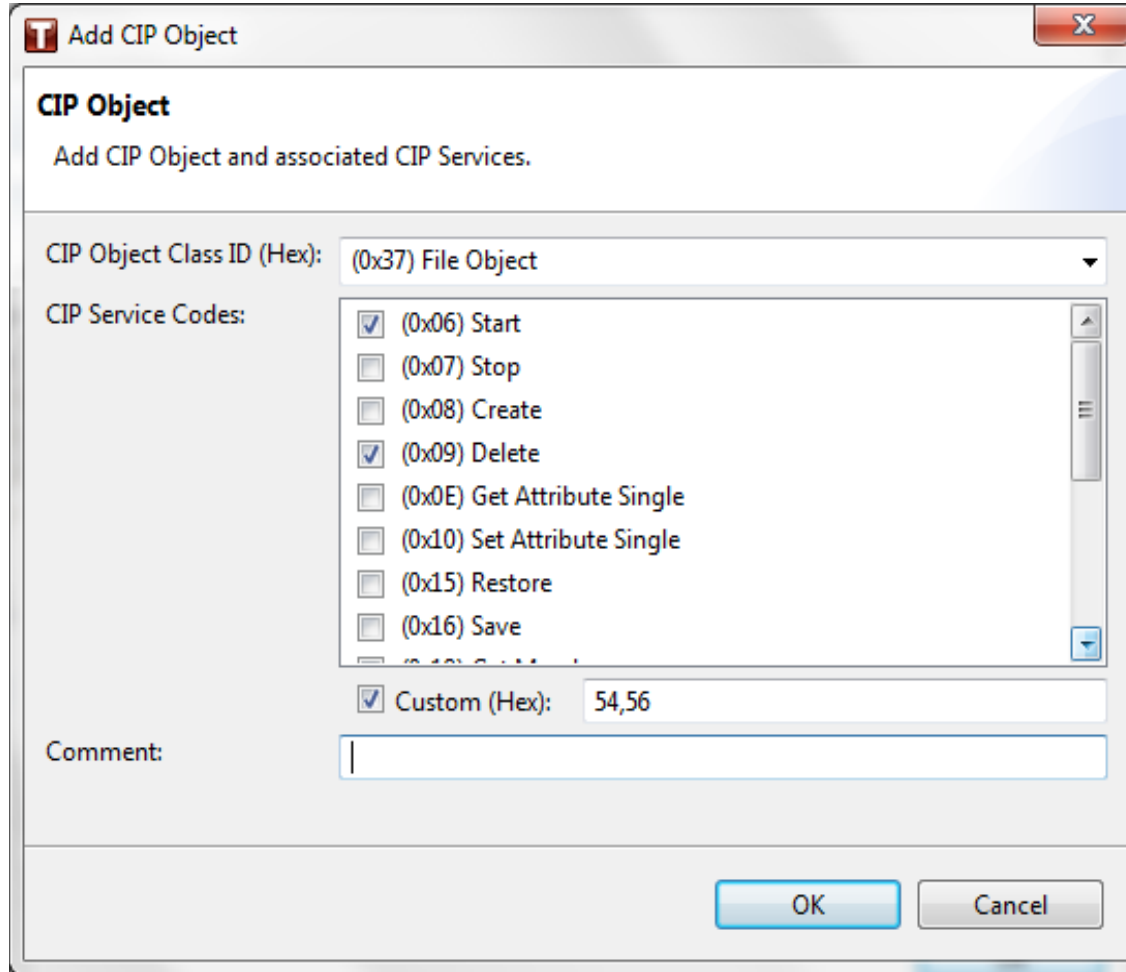
Debug:

Advanced DPI Configuration for EtherNet/IP

- ▶ Provide advanced CIP filtering for skilled customers



Allow User Defined Objects and Services



The screenshot shows a Windows-style dialog box titled "Add CIP Object". The dialog has a close button in the top right corner. The main content area is titled "CIP Object" and contains the instruction "Add CIP Object and associated CIP Services." Below this, there is a dropdown menu for "CIP Object Class ID (Hex):" with the value "(0x37) File Object" selected. Underneath is a list of "CIP Service Codes" with checkboxes: (0x06) Start (checked), (0x07) Stop, (0x08) Create, (0x09) Delete (checked), (0x0E) Get Attribute Single, (0x10) Set Attribute Single, (0x15) Restore, (0x16) Save, and Custom (Hex): 54,56 (checked). A "Comment:" text box is located at the bottom left. At the bottom right, there are "OK" and "Cancel" buttons.

Add CIP Object

CIP Object
Add CIP Object and associated CIP Services.

CIP Object Class ID (Hex): (0x37) File Object

CIP Service Codes:

- (0x06) Start
- (0x07) Stop
- (0x08) Create
- (0x09) Delete
- (0x0E) Get Attribute Single
- (0x10) Set Attribute Single
- (0x15) Restore
- (0x16) Save
- Custom (Hex): 54,56

Comment:

OK Cancel

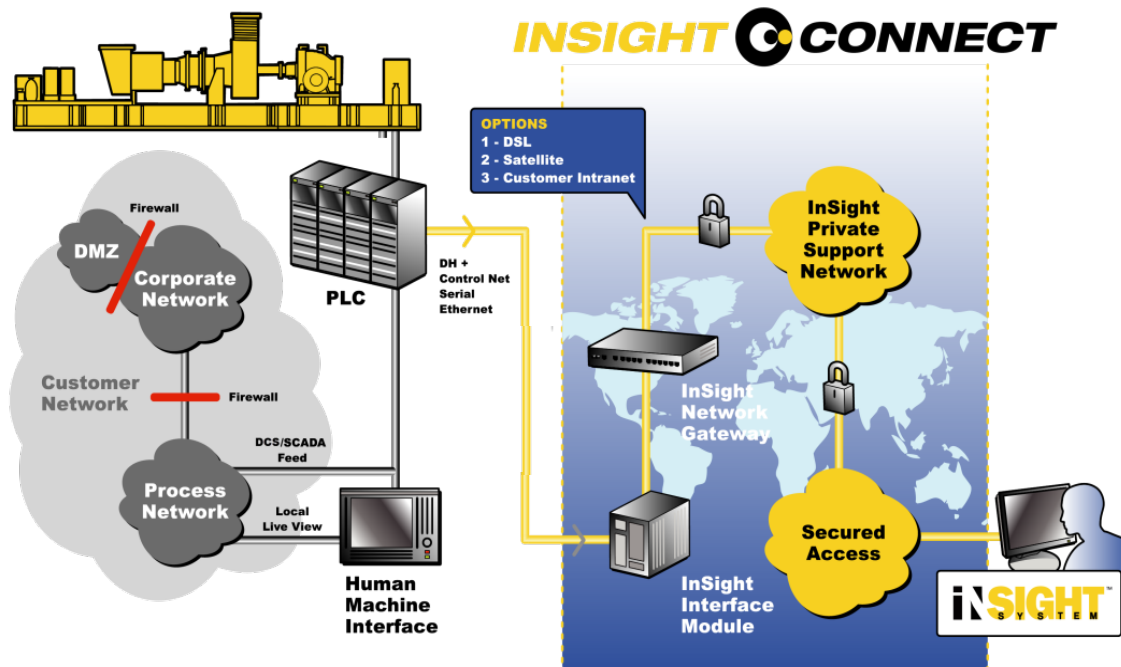


Case History

Securing Pipeline Compressor Controls

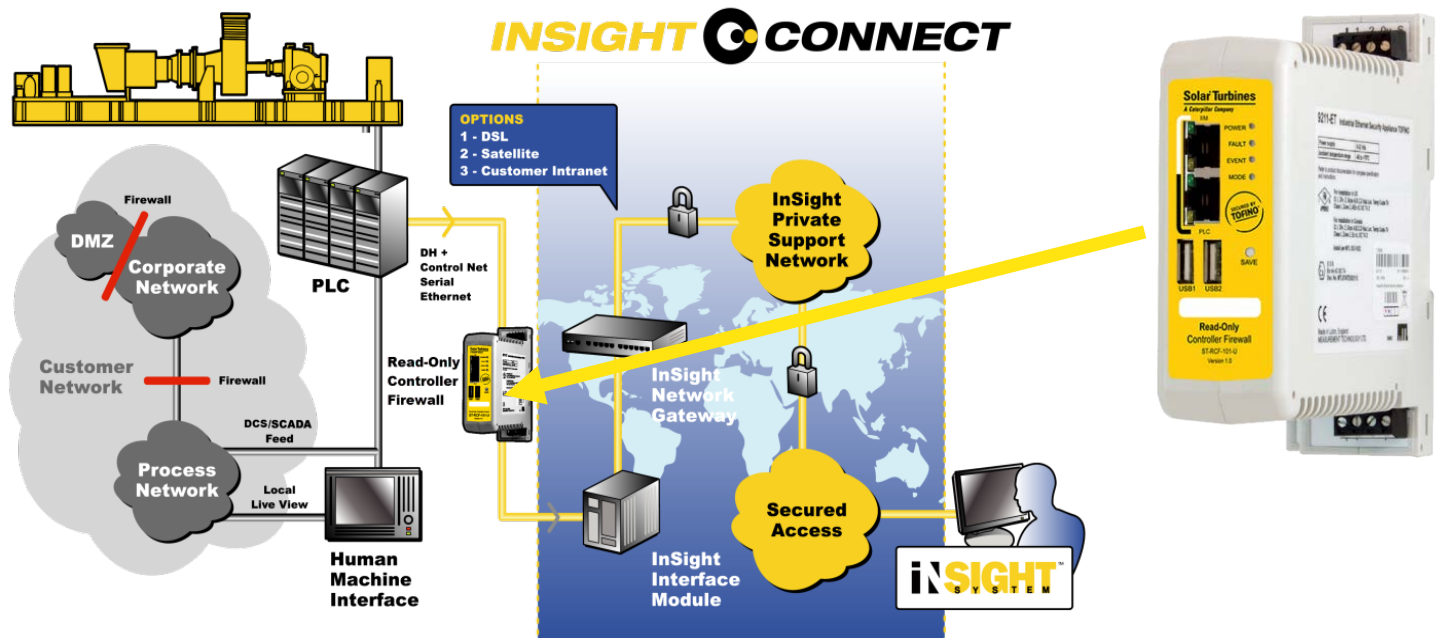
Securing Pipeline Compressor Controls

- ▶ Compressor packages are complex, high risk systems
- ▶ Monitoring of platforms is critical in remote locations
- ▶ Risk of rogue insiders and unpatched vulnerabilities (even with VPN)



Read-only Controller Firewall

- ▶ Firewall inspects each EtherNet/IP message to ensure only data-read commands are allowed to the PLCs
- ▶ Requires no configuration, no network changes and no disruption to monitoring or turbine operations



Conclusions

- ▶ There is a need for an effective security solution for IDS/SCADA systems
- ▶ Deep Packet Inspection of EtherNet/IP significantly increases reliability and security
- ▶ There is a way to make use of EtherNet/IP DPI without adding overwhelming complexity
- ▶ EtherNet/IP DPI is applicable to real world situations



Questions?