# Passive Security Monitoring for IEC-60870-5-104 based SCADA Systems

Reddi Hareesh, Rajesh Kalluri, Lagineni Mahendra, R. K. Senthil Kumar, B. S. Bindhumadhava
*Center for Development of Advance Computing (C-DAC)*
*C-DAC Knowledge park, Opp. HAL Aeroengine division, India*

## Abstract

*Rapid changes in Supervisory-Control-and-Data-Acquisition (SCADA) systems used in power systems from traditional proprietary serial-based communication protocols to Internet-protocol (TCP/IP) based standard communication protocols such as IEC-60870-5-104 have made the smart grids susceptible to malicious cyber threats and attacks. Current hierarchical SCADA systems are vulnerable to cyber threats as their communication protocols are originally designed without any built-in security mechanisms, and they are well-documented protocols that help attackers exploit these vulnerabilities to sabotage the SCADA systems. It is necessary to develop security solutions tailored to power sector SCADA systems to sustain the reliability and availability of the power systems. This paper proposes white-list rules and a passive-monitoring based anomaly detector called security monitoring unit (SMU) to detect anomalous communication in the SCADA system. The proposed anomaly detector uses Deep Packet Inspection (DPI) based white-list rules as detection rules that are modelled specifically for IEC-60870-5-104 based SCADA systems. Along with the white-listed rule sets, the solution also includes data correlation, where the field data (sensor value) is mapped against data-in-transit from RTU to the controlling station to perform in-line message validation. The proposed rule-based solution can effectively detect known and as yet unknown zero-day attacks on the IEC-60870-5-104 based SCADA systems.*

*Keywords: SCADA, Deep packet inspection (DPI), white listing, passive monitoring, security monitoring, remote terminal unit (RTU), master terminal unit (MTU)*

## 1. Introduction

The power system is sometimes called the world's largest interconnected machine. Safety, security and reliability are always important issues [1] in the design and operation of power systems. And these days importance of cyber security is increasing as the power system relies heavily on information infrastructure. SCADA plays a vital role in controlling dispersed assets for the functioning of the power system. An increase in interconnections in the SCADA systems makes power systems highly automated by leveraging information technology fully and becomes more capable in managing energy. And at the same time, it potentially widens the prospect of intrusions, malicious attacks, and other threats to the power system. The communication protocols are some of the critical parts in the functioning of the SCADA system, and they were initially designed without any security considerations. This is luring the attackers nowadays and could lead to power system compromise to malicious attackers, disgruntled employees via unauthorized access at vulnerable points [2][3][4]. Such attacks can result in a widespread failure of power system operation, safety and stability. Therefore, protection of SCADA systems from cyber attacks, equipment malfunctions, communication equipment failures, etc., is at most necessary [5] and the immediate requirement to modern Power systems.

Several open international standards exist for SCADA in electrical engineering and power system automation. Some of them are IEC-60870 part 5, IEC-61850, Modbus, and DNP3. IEC-60870-5-104 protocol [6] provides network access for IEC-60870-5-101 protocol [7] using standard transport profiles of TCP/IP protocol. However, IEC-60870-5-104 is a plain-text protocol, i.e., it transmits data in clear text form with no authentication mechanism [2][3] over TCP/IP. And TCP/IP itself is an entry point for several malicious attacks, makes IEC-60870-5-104 much more vulnerable to cyber attacks [4], and hence proves the absolute requirement for cyber security measures.

To provide a complete end-to-end security model for power systems communication on IEC-60870-5 protocols (TC57), the International Electrotechnical Commission (IEC) developed IEC-62351 standard series, which provides a list of guidelines and framework to secure the communication between the control station and the controlled station through end-to-end encryption. Although the IEC-62351 ensures authenticated access to sensitive power systems that are operating on IEC-60870-5-104 protocol, it is difficult to quickly upgrade the legacy SCADA systems [8] due to their limited computing resources, lack of consideration for security mechanisms, and the implementation challenges and risks. Unless IEC-62351 protocol is implemented on the actual SCADA communication devices (such as RTU/PLC), the protocol as a bump-in-the-wire [2] solution will not provide adequate end-to-end security. The IEC-TR-62351-90-2:2018 also addresses the need for DPI based monitoring though the communication is encrypted. The detection of attack is equally important as prevention, hence the system in need of a monitoring solution. This paper proposes a DPI and passive monitoring-based anomaly detection solution called SMU.

The active monitoring solutions may introduce overhead to



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

the SCADA network, which is sensitive to unexpected network traffic, SMU works on passive monitoring mode and uses DPI based white-list rules and signatures to effectively detect security anomalies and incidents on the SCADA networks. White-listing [5] is to grant access to known good instead of denying access to the known bad. According to NIST guidelines on SCADA security, the white-listing approach is more effective than the black-listing for SCADA systems. And the SCADA systems with a stable structure, their static properties, predictable traffic and lack of past attack signatures make white-listing more practical for SCADA systems.

The rest of the paper is organized as follows. Section 2 gives an overview of IEC-60870-5-104 protocol application layer, Section 3 describes cyber vulnerabilities of IEC-60870-5-104 protocol, Section 4 describes the architecture of the proposed SMU, Section 5 gives in-depth details of the proposed white-list rules with examples, Section 6 discusses the proposed field-data correlation-based detection mechanism, and finally, Section 7 describes attack simulation on the SCADA system, and detection using SMU.

## 2. IEC-60870-5-104 Application layer

Figure 1 illustrates the application layer frame of IEC-60870-5-104 protocol, called the application protocol data unit (APDU) [6]. The APDU is divided into two parts, application service data unit (ASDU) and application protocol control information (APCI). IEC-60870-5-104 provides TCP/IP network access to IEC-60870-5-101 serial based protocol on a standard specified TCP port 2404. APCI control information is added to the frame to demark the start and the end of the APDUs and provide protection against loss and duplication of messages. The APDU is a maximum 255 octets' sized frame.

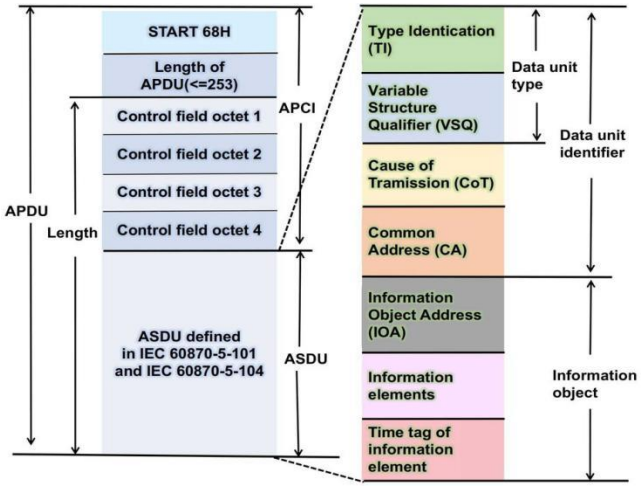


Figure 1. IEC-60870-5-104 frame structure

The ASDU [6] as illustrated in Figure 1 carries actual data in the SCADA system. It carries sensor data in monitor direction and supervisory control commands and data requests in the control direction. Each device (sensor and actuator) in the field is uniquely identified by the information-object-address (IOA) in the ASDU. The type of data (measured/indication/control) carried by the ASDU is identified by the type-identification TI (also called ASDU-type) field. The cause-of-transmission (CoT) field identifies which application task to handle the ASDU once received. And APCI carries control information for transport connection supervision. Three different formats of the APCI are used to represent three functionalists, I-frame (information) to perform information transfer, S-frame (supervisory) for supervisory functions such acknowledgment, and U-frame (unnumbered) for control functions.

## 3. IEC-60870-5-104 Vulnerabilities

IEC-60870-5-104 is a widely used standard communication protocol in current SCADA systems of power sector utilities. This protocol is lacking in the application layer and the data link layer security [2][4] as a result of its plain-text transmission property. At the application layer, the protocol is vulnerable to attacks such as spoofing and non-repudiation and at the data link layer, it is vulnerable to sniffing, data modification and replay attacks. The protocol also lacks authentication [3] to judge identity, lacks data integrity against modification, lacks confidentiality towards critical information such as device addresses (IOA), RTU addresses, etc., and lacks authorization and restriction against malicious function code running on RTU. These vulnerabilities allow the attacker to gain unauthorized access/control to SCADA critical infrastructure system to launch attacks such as MITM [9][10], DoS [11], replay, packet injection, data modification, identity spoofing attack, etc. . This may lead to severe damage to the power systems operation, reliability and safety.

Consider a scenario: The SCADA systems operations are largely [2] dependent on the data received from the RTU/PLC, based on which the control actions will be performed. An attacker performs a MITM attack to block or modify the data in transit from RTU to the master terminal unit (MTU) to force the MTU or the operator to make inappropriate decisions. The attacker can also learn critical device (sensor) address by sniffing, and launch an identity spoofed attack by exploiting the lack-of-authenticity vulnerability of the protocol. With the field device address known, the attacker issues an identity spoofed remote control command on a critical field device (actuator) to disrupt the process, and also blocks the message from RTU to MTU to evade the detection.

## 4. Security Monitoring Unit (SMU)

SMU listens to the real-time network traffic using special devices such as port-mirroring enabled Ethernet switch or network-tap, and packet capture libraries such as PF_RING [12] to capture the SCADA network traffic. Along with the captured packets the current status/values of the field-devices, (sensors) read from their redundant ports are another input to the SMU. SMU performs DPI on captured packets for in-depth analysis through rule matching to detect and report the attacks. The real-time SCADA network traffic is analyzed and mapped against proposed white-list rules (explained in further sections), and the data in transit from RTU/PLC to MTU is



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

validated through real-time field-data-correlation to detect anomalies including zero-day attacks and other malware exploits without introducing overhead to the SCADA network.

Figure 2 shows the functional blocks of the SMU, where it

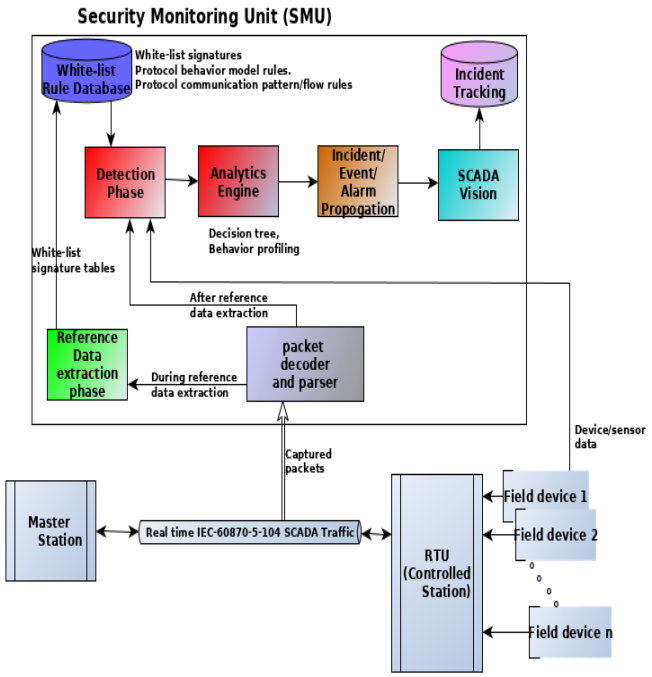


Figure 2. Internal architecture of SMU

takes two inputs from the SCADA network that is being monitored; a copy of all the packets leaving or entering the RTU/PLC using a mirror port or a network tap, and a copy of the field device data (sensor present status/data) directly from the device (through a redundant port). SMU functions in two different phases of operations, the reference-data-extraction phase, and the detection phase.

The Packet-decoder-and-parser captures the network packets at wire speed using network taps or SPAN ports and special software libraries such as pf-ring [12], logs into capture files and filters them through a lightweight network-stack (which implements IP and TCP header-checksum calculation, handling duplicate and re-transmitted TCP packets, etc.) to accurately identify the packets that are entering and leaving the application layer of the RTU/PLC being monitored. Because the packets such as re-transmitted and duplicate have no significance and will be dropped on actual SCADA devices. These captured packets are the inputs to both Reference-data-extraction and detection phases as shown in Figure 2.

SMU uses the captured and filtered packets in the data extraction phase to extract the reference data for the base-lining, and the base-lined data is stored as a part of white-list signatures in the rule-database. Along with the white-list signatures, the rule-database also includes the pre-configured protocol-behavior-model rules and the communication-pattern-based white-list anomaly rules as shown in Figure 2. At the end of the data extraction phase, SMU changes its mode of operation to the detection phase. The analytic engine implements decision trees of proposed white-list rules from the rule-database with each rule as a decision rule. A decision tree is implemented to analyze the packets, and identify significant characteristics in the network packets that indicate malicious activities. Each captured packet is mapped against all the decision rules to classify it as either normal or malicious. Any packet that fails to pass a rule will identify it as malicious and triggers an appropriate alert, or a set of alerts with a risk level associated with that rule. Based on the risk level, alerts are categorized into incidents, events and alarms, where incidents represent high-risk alerts, alarms represent low-risk alerts, and events represent medium-risk alerts.

SMU also performs behavior profiling on the set alerts generated by the decision trees to perform behavior based detection. And the SCADA vision logs the alerts for future use and visualizes the alerts to the concerned authorities on a graphical UI in real-time, where incident tracking is also possible.

## 5. White-list rules

White-listing is a policy that denies or suspects the packet by default and allows or treats it as normal by exception [5]. According to NIST guidelines on SCADA security, the white-listing approach is more effective than the black-listing for SCADA systems. And the SCADA systems with a stable structure, predictable traffic, and lack of past attack signatures make the white-listing more practical. The SMU introduces two types of white-list rules; white-list signatures and protocol anomaly rules.

### 5.1. White-list signatures

Typical signature-based detection techniques [13] are knowledge-based, i.e. they detect attacks by looking for specific patterns (black-listed attack signatures) used by malware in the network traffic. With these black-listed signatures, one can effectively detect known attacks, but it is impossible to detect attacks for which no signatures are available. The signatures used by the proposed solution are white-listed, i.e. absence of these signatures in any packet will consider the packet as malicious. These signatures can effectively detect attacks such as unauthorized and unauthenticated access, unauthorized port access, identity spoofed attacks, port scanning, etc. The proposed signature-based detection mechanism works in two phases, the reference data extraction phase, and the detection phase.

**5.1.1. Reference data extraction phase.** The NIST guidelines on SCADA security [5] suggest that having an access restriction based on white-listed IP addresses, IP-Port pairs, TCP/UDP Ports, and other static information that is specific to the SCADA network is more practical in securing the system. In this phase of operation, the SMU extracts all the static information specific to the SCADA system, such as valid IP addresses, IP-Port pairs, IP-MAC pairs, TCP/UDP Ports, IEC-60870-5-104 ASDU types (TI), information element addresses (IOA) used and their types (measured or indication), common



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

address (CA) of the RTU/PLC, average packet size, average packet rate, RTU/PLC average response time, etc. This extracted reference data along with operator supplied identifiers is stored in the rule-database as a list of white-list signatures to be used in the detection phase. This phase ends when SMU extracts an adequate amount of information, i.e. when SMU captures an IEC-60870-5-104 general interrogation (C_IC_NA) sequence [6] which provides all the data of interest about the network being monitored, or when it acquires the threshold amount of information based on operator supplied inputs (for example, count of MTUs, count of data points associated, etc.). In this phase, it is assumed that during the period of reference-data extraction, the network being monitored is ideal and attack-free. The SMU operates in this phase only once in the beginning and always operates in the detection phase. But the operator can re-run the SMU in this phase for updating of signatures.

**5.1.2. Detection phase.** In this phase, SMU performs DPI to map each packet against the white-listed signatures from the rule-database to detect any violation of rules. For example, a TCP connection request to port 2404 of RTU from a client whose IP address is not white-listed, will trigger an unauthorized-access alert, and a client IP address that is authorized to communicate on port 2404 of RTU, sends a connection request on port 80 of RTU (opened for RTU configuration) that violates valid IP-Port pair rule. These signatures effectively identify suspicious communication, unauthorized access, malicious behavior of RTU/PLC, identity spoofed attacks, brute force attacks to find field device addresses, and their types (indication or measured) to control them with a spoofed identity, and policy violations in the SCADA network.

Along with white-list signatures mapping, SMU uses the protocol anomaly rules in this phase for threat detection and reporting.

## 5.2. Protocol anomaly rules

The protocol anomaly rules use the suspect-by-default approach as a key security principle. These rules are derived from IEC-60870-5-104 protocol definition [6] to validate the network packet by looking for any deviation in the normal-use models of the protocol. These models are derived with high accuracy.

**5.2.1. Protocol behavior-based model rules.** The protocol behavior models are used to verify SCADA traffic adherence [5] to IEC-60870-5-104 protocol definition and specifications at the application layer to find any protocol anomalies. These models are formed by considering the known fact from the protocol definition that, the application layer frame (APDU) of IEC-60870-5-104 protocol has a limited and fixed number of frame formats (I-frame, S-frame, and U-frame) and is divided into a fixed number of known length fields (example: 68H, Len-of-APDU, control fields, TI, VSQ, CoT, CA, IOA) with a known range of values that each field can take (based on the number of bits reserved per field) and their predictable behavior patterns. The following examples describe how models can be formed based on these facts.

Example-1: A particular value in a particular field of any IEC-60870-5-104 frame will be restricted to one particular direction that is either RTU to MTU or MTU to RTU.

Example-2: A particular value in a particular field restricts the length of the frame to one particular known value (for example, if the TI field value is 100, then the Length_of_APDU field value will be restricted to 14 in decimal).

The definition of the IEC-60870-5-104 SCADA protocol with in-depth analysis enables us to write a list of sophisticated protocol models. Any violation of these models in the network traffic will lead to suspect malicious anomalous activity in the network. These rules are capable of detecting unknown attacks on the SCADA network. The following are some of the proposed protocol behavior model rules.

***Model 1- ASDU-Type (TI) models***. The ASDU-Type or TI field in the ASDU is an 8-bit length field that represents the type of ASDU that the packet carries. With an 8-bit size, it can take any value between 0 to 255, which means it can represent 256 different types of ASDUs. But according to protocol definition TI=0 is unused [7], and TI=128 to 255 are undefined, (TI=136 to 255 may be defined independent of each user of this standard, but for 100% interoperability, only TI=1 to 127 should be used). Hence the range of values the TI field can take is limited to 1 to 127. And among these valid ASDU Types (TI=1 to 127), some have direction restrictions, i.e. packet with TI=1 to 40 can be sent only from RTU to MTU in monitor direction, and TI=102 should be sent only from MTU to RTU in control direction (remaining ASDUs can be sent in any direction).

Based on the above-mentioned fact all possible ASDU type models are formed as shown below that restricts the value of TI field of any ASDU to:

1) *For I - frame in control direction, TI should be: TI = {45 - 51, 58 - 64, 100 - 103, 105, 107, 110 - 113}*
2) *For I - frame in monitor direction, TI should be: TI = {1, 3, 5, 7, 11, 13, 15, 20, 21, 30 - 40, 45 - 51, 58 - 64, 70, 100, 101, 103, 105, 107, 110 - 113, 120-126}*

***Model 2- Cause of Transmission (CoT) models.*** According to the APDU definition, as illustrated in Figure 1, the CoT field in the ASDU is a 1 or 2 octet (user defined) length field, used to direct the ASDU [7] to a specific application task for processing. Irrespective of the size of the CoT field (either 1 octet or 2 octets), the first 6 bits of the first octet are used to represent the transmission-cause. With 6 bits, the field can take any values between 0 and 63 but according to IEC-60870-5-104 definition CoT=0 is undefined, CoT=14 to 19, CoT=42, 43 are undefined, and CoT=48 to 63 are reserved for future use [7]. Therefore, a valid range of values for CoT is CoT=1-13,



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

20-41 and 44-47. Based on this fact the CoT models are formed, that restricts the value of the CoT field of any ASDU to:

1) *For any I-frame in any direction, CoT should be:*
   *CoT = {1 - 13, 20 - 41, 44 - 47}*

Figure 3 shows how both TI and CoT model rules are applied against each I-frame on the network. And violation of these rules generates associated alerts.

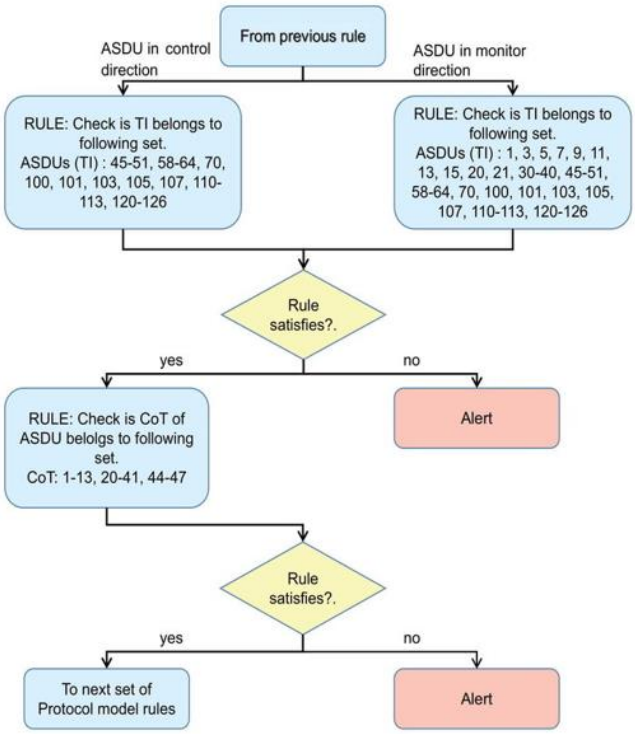


Figure 3. ASDU-Type/Type-Id and CoT models

***Model 3- Length_of_APDU models.*** The length_of_APDU field in the IEC-60870-5-104 frame is a 1 octet length field, used to represent the total length of APDU in bytes, i.e. the size APDU can be any size between 0 and 255 bytes, but it is restricted to the known range of values for any frame. Based on the above fact following length field model is formed.

1) *For I/U/S - frame in any direction, length should be:*
   *Length_of_APDU={ 4, >12, <254 }*

***Model 4- TI-CoT-Direction Model (CoT field dependency on TI field and the packet direction).*** The values of IEC-60870-5-104 frame fields such as TI, CoT, VSQ, length, etc, have a relation with values of other fields in the same frame, i.e., a particular value in a field has a unique relationship with a particular value in another field in the same frame. The following are some of the several dependent models.

As illustrated in Figure 4, the value at the CoT field changes based on the value of the TI field and the direction of the packet. Consider a scenario; A packet with TI=105 in monitor direction can have CoT= 7 or 44-47 only, and for the same TI=105 in control direction, the CoT is restricted to CoT= 6 only. Based on this protocol definition some of the TI-CoT-Direction Models formed as below.

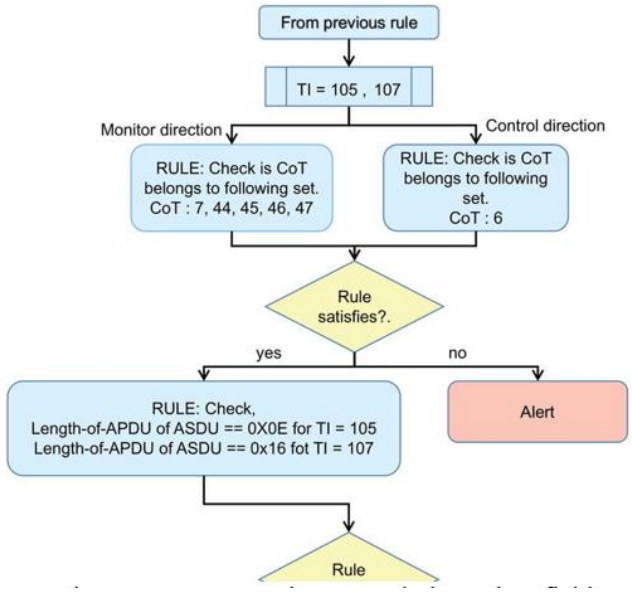


Figure 4. TI-CoT and TI-Length dependent field models

*1) I-frame with TI=105 in Monitor direction:*
*CoT= { 7, 44-47 }*

*2) I-frame with TI=105 in Control direction:*
*CoT= { 6 }*

*3) I-frame with TI=100 in Monitor direction:*
*CoT= { 7, 9, 10, 44-47 }*

*4) I-frame with TI=100 in Control direction:*
*CoT= { 6, 8 }*

***Model 5- ASDU_Length-TI Model (ASDU_Length field dependency on TI field).*** The length (Length-of-APDU) field of IEC-60870-5-104 frame varies based on TI field value in the same frame. Consider a scenario: for a packet with TI=105 in any direction, the length (Length-of-APDU) field should be Length=0x0E (14 in Decimal). Below are some of the Length-TI Models.

1) *I-frame with TI=105 in any direction, length should be:*
   *Length_of_APDU = {14}*
2) *I-frame with TI=103 in any direction, length should be:*
   *Length_of_APDU = {2}*

***Model 6- IOA-TI Model (IOA field dependency on TI field).*** The IOA (sensor address) field of IEC-60870-5-104



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

frame should be 0x000000 for a set of known TI values in the same frame. Consider a scenario: for a packet with TI=100 in any direction, the information object address (IOA) field should be IOA=0x000000 (0 in Decimal). Below are some of the IOA-TI Models.

1) *I-frame with TI=100, 101, 103, 105, 107 in any direction, IOA should be:*
   *IOA = {0}*

**5.2.2. Protocol communication pattern/flow based rules.** SCADA protocols are created with known definitions and specifications for their proper usage and communication. Every connection-oriented protocol such as IEC-60870-5-104 will have states that tell what event should take place at a particular time [14][15] (examples for states in IEC-60870-5-104 protocol are: STARTDT_Act, STARTDT_Act_Con, STOPDT_Act). Hence for any communication protocol, a time and communication state-based state-machine can be drawn, where each state represents a part of communication. And each transition between the states represents a predefined and expected change between the states. Any undefined transition between states will lead to suspect a protocol anomaly.

The proposed communication pattern rules are formed based on the state transition [6] of aggregated IEC-60870-5-104 communication flow on every pair of source and destination IP addresses (each MTU and RTU connection) communicating on port 2404 of RTU with the aggregated information such as packet arrival time and sequence, send and receive sequence number, etc. These white-listed rules identify any anomalous packets that violate defined protocol behavior. As illustrated in Figure 5 all the communication states of the protocol are represented as different states of a finite state machine that changes its present state based on two types of events, the packet event pkt_eve (on the arrival of a packet), and the time event tim_eve (on timer timeout, see Table 2). The arrow between two states indicates the valid state transition from one state to another state based on the event (packet or timeout) that arrived, and any invalid event in any state will generate an alert associated with that particular state. The packet captured is given to the state machine as a packet event where the packet is validated against the present state of communication. Table 1 [6] lists all the different possible states of IEC-60870-5-104 communication, Table 3 lists all the possible timer events supported by the protocol, and Table 4 lists all the possible packet events the protocol can take.

Malicious packet such as an injected packet (with spoofed identity), validates through the proposed protocol model rules and white-listed signatures, but it can be effectively identified based on the state of actual communication at which the injected packet arrives. Any event (packet event) that deviates from the expected state-transition, will be suspected as a malicious packet. Every state in the state machine will have a list of valid events that can change the present state of the state machine to another appropriate valid state. Any event that is not a part of the list of valid events will lead to suspect it as a malicious packet event.

Consider a scenario (refer Figure 5, and Table 1,3,4), if the Present_state=*ST_STOPDT_Pending (ST12)* i.e. the last valid event was *EV_STOPDT_ACT (pkt_eve4)* request from MTU, then the valid list of events that can arrive in this present state of communication are;

1) *EV_STOPDT_CON (pkt_eve5).* From RTU as a confirmation for the *STOPDT*_ACT sent by MTU. Upon this event, the Present_state will change to *ST_Connection-established* state.
2) *EV_S-Frame-Monitor (pkt_eve8).* From RTU, as an acknowledgment for any unacknowledged I-Frames sent by MTU (before sending the *STOPDT_CON).* Upon this event, the present state will not change its state although the event is valid.
3) *EV_t1_timeout (tim_eve2).* When timer t1 expires without a *STOPDT_CON* confirmation event from RTU for the *STOPDT_ACT* sent by MTU. Upon this timer event the present state of the state machine changes to *ST_T1_Timeout* state *(ST13)*, (wherein *ST_T1_Timeout* state, the only valid event is *EV_Connection_Close***,** i.e. *pkt_eve12*).

If the next event arrived is not among the three events mentioned above, for example, EV_I-Frame-Control (pkt_eve11) or EV_I-Frame-Monitor (pkt_eve10) is invalid at this present state, and this will lead to classify the packet as malicious, suspect a malicious behavior in the network and generate an appropriate alert.

Table 2 lists the default definition of timeouts [6] given in the IEC-60870-5-104 protocol, but the protocol allows the SCADA operators to choose custom timeout definitions (within the maximum limit set by the protocol) based on the requirement. The custom definitions of timeouts will be supplied to SMU, otherwise it works with the default definitions.

Table 1. IEC-60870-5-104 Communication States

| Communication State | Description |
|---|---|
| ST_Default | No connection state |
| ST_Connection-established | Network connection established |
| ST_STARTDT_Pending | Data transmission activation request received |
| ST_STARTDT | Confirmation for STARTDT_ACT received |
| ST_I-Frame- Control | I-frame from MTU received |
| ST_I-Frame-Monitor | I-frame from RTU received |
| ST_S-Frame-Control | MTU S-frame |
| ST_S-Frame-Monitor | S-frame received in monitor direction |
| ST_TESTFR_Pending | Periodic connection test request received |
| ST_TESTFR | Confirmation for Connection test received |
| ST_STOPDT_Pending | Stop-data-transmission request received from Master |
| ST_T3_Timeout | Network is idle for T3 seconds |
| ST_T1_Timeout | No acknowledgment in t1 seconds |



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

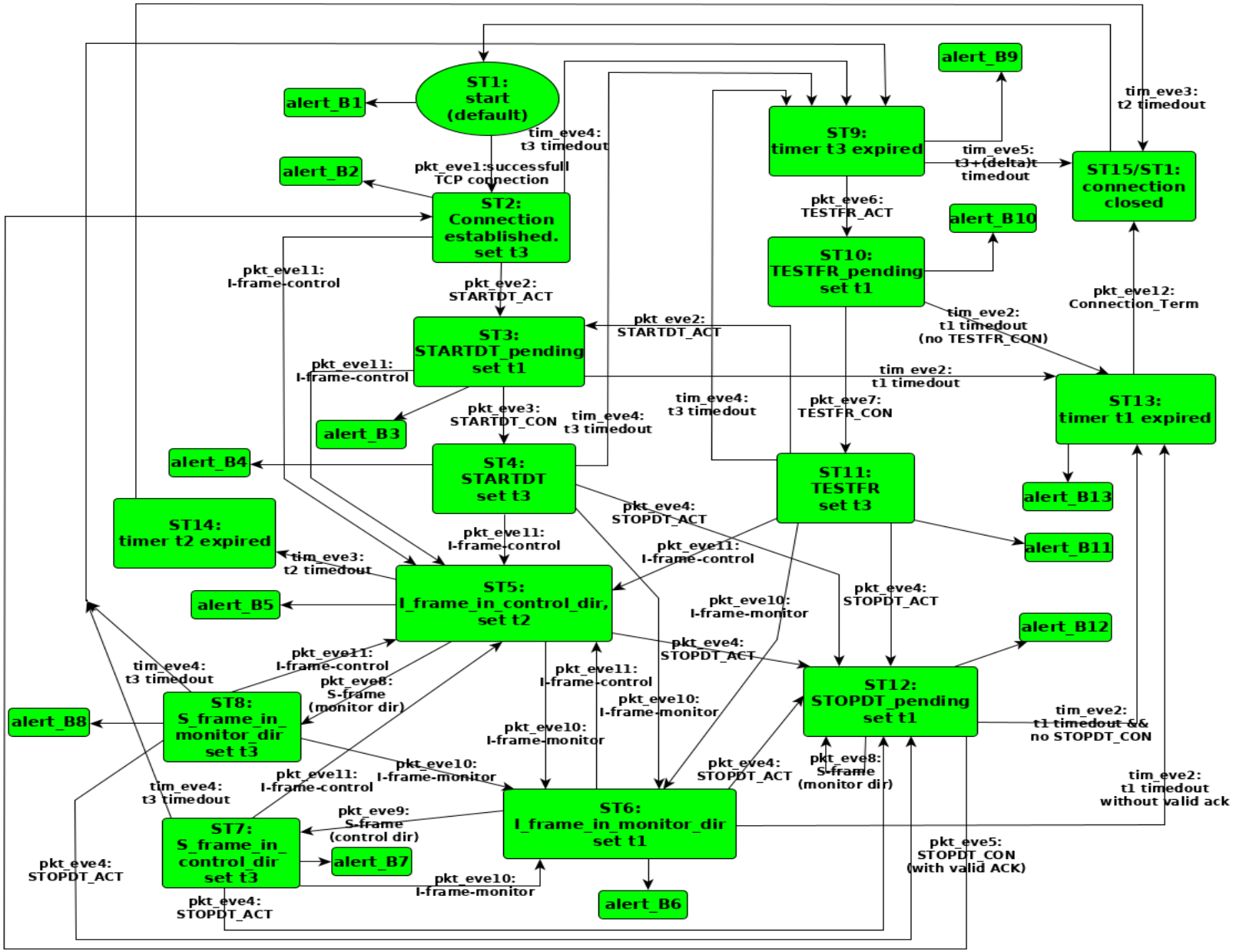


Figure 5. IEC-60870-5-104 Communication flow based Sate machine



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

Table 2. Default time-out definition

| Para meter | Default value | Remarks |
|---|---|---|
| t0 | 30 s | Time-out of connection establishment |
| t1 | 15 s | Time-out of send or test APDUs |
| t2 | 10 s | Time-out for acknowledges in case of no data messages t 2 < t 1 |
| t3 | 20 s | Time-out for sending test frames in case of a long idle state |

Table 3. Timer events

| Time Event | Description |
|---|---|
| EV_t1_timeout | Event triggered upon Timer t1 expire |
| EV_t2_timeout | Event triggered upon Timer t2 expire |
| EV_t3_timeout | Event triggered upon Timer t3 expire |

Table 4. Packet events

| Packet Event | Description |
|---|---|
| EV_TCP_Connection | Successful TCP Connection Sequence |
| EV_STARTDT_ACT | Arrival of STARTDT Activation request |
| EV_STARTDT_CON | Arrival of STARTDT_Act Confirmation |
| EV_I-Frame-Monitor | Arrival of any I-frame in Monitor direction |
| EV_I-Frame-Control | Arrival of any I-frame in Control direction |
| EV_STOPDT_ACT | Arrival of STARTDT Activation request |
| EV_STOPDT_CON | Arrival of STARTDT_Act Confirmation |
| EV_S-Frame-Monitor | Arrival of S-frame in Monitor direction |
| EV_S-Frame-Control | Arrival of S-frame in Control direction |
| EV_TESTFR_ACT | Arrival of STARTDT Activation request |
| EV_TESTFR_CON | Arrival of STARTDT_Act Confirmation |
| EV_Connection_Term | TCP Connection Termination Sequence |

## 6. Field data correlation

Along with the white-list signatures and protocol anomaly based detection, SMU also performs data correlation to detect security events on RTU such as malicious RTU behavior, that may the result of a vendor implanted time bombs or logic bombs, malicious RTU firmware, malicious RTU configuration files, etc. This is done by validating the data sent by the RTU to the controlling station through real-time correlation with the actual data received from the field device. Any mismatch in data will lead to suspect an abnormality. The correlation will be done only for the spontaneous (CoT=3) data sent by the RTU. The actual field device data is acquired using a redundant port.

Consider a scenario: An RTU reads an abnormal change in the line frequency from a frequency sensor, and the information has to be sent immediately to MTU for corrective action. But in the place of sending the correct value, a malicious code running on RTU that intercepts the data processing logic and sends a spontaneous ASDU with a frequency value within the normal range. This misleads the operator and makes him not take any action on the abnormality in the field. This can severely damage the power system. In such cases, the proposed data correlation can detect the abnormal behavior of RTU and report it to the operator.

## 7. Attack simulation and detection

CDAC's IEC-60870-5-104 SCADA testbed has been extensively used to generate the live SCADA traffic. As illustrated in Figure 6 SMU is introduced to the SCADA testbed using a mirrored port of a centralized Ethernet switch such that SMU can capture all the traffic leaving and entering the RTU/PLC. SMU captures all the traffic on its network interface in promiscuous mode for real-time analysis using proposed detection techniques. SMU is initialized in the reference-data-extraction phase to extract the white-list signatures specific to the network. Upon capturing a general interrogation (C_IC_NA_1) sequence, MTU changed its phase of operation to the detection phase. And for the data correlation, SMU given the permission to read the real-time data directly from redundant ports of the field devices.

An attacker system [16] is introduced to the SCADA network as shown in Figure 6. Tools such as Ettercap, Hping3, Nmap, etc., on Kali-Linux, are used to perform attacks such as MITM, replay, command-injection, flooding, network scanning, etc. between the RTU and MTU.

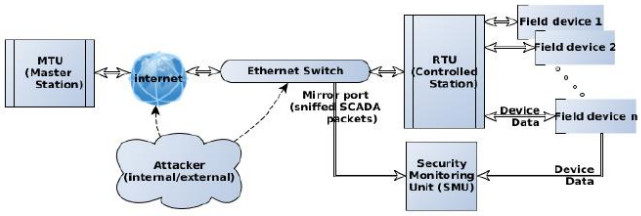


Figure 6. Attack simulation and detection on the SCADA test-bed

A MITM tool developed by CDAC is also extensively used to intercept the communication between the legitimate MTU and RTU, and perform a MITM attack on the IEC-60870-5-104 application layer. The tool is capable of framing and injecting all the valid IEC-60870-5-104 application layer PDUs to insert them into the SCADA network to have maximum impact.



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

The following are some of the several significant attacks [17] simulated (and detected) on CDAC's IEC-60870-5-104 SCADA testbed [18] as they are capable of creating a severe impact on SCADA systems. All the simulated attacks are successfully detected, and appropriate alerts are generated by the SMU. And each alert with its associated risk level, visualized on the graphical UI based SCADA Vision dashboard.

### 7.1. Unauthorized Control command on RTU

Simulated Attack: The attacker uses the CDAC's MITM tool to introduce himself between the MTU and RTU. Pretending to be a legitimate MTU and sends a digital control command (C_SC_NA_1) to switch on/off the circuit breaker connected at an unknown device address by brute forcing.
Detection: The attacker is detected even though the attacker uses a legitimate IP address of a valid MTU. The attacker spoofs the IP address but the attacker's MAC address is not part of the white-listed MAC addresses. And if the attacker spoofs his identity to a legitimate MTU with a valid IP and MAC, then injected digital command is detected by the violation of the communication pattern rules as the packet is insignificant in the present state of communication or the mismatch in send and receive sequence numbers. And as the attacker does not know the device (example: actuator) addresses, the brute force attack triggered an alert based on the violation of a proposed signature rule that restricts device addresses (IOA) to those assigned to the actuators. Each rule violation resulted in the generation of alerts with an associated impact level. Behavior profiling on these sets of alerts is done to precisely find the type of attack, and hence unauthorized control command attack is successfully detected and visualized by SMU.

### 7.2. Data modification attack

Simulated Attack: An attacker performs an ARP-Poisoning MITM attack between and MTU and RTU using the Ettercap tool and intercepts a legitimate control command (C_SC_NA_1) sent by the MTU, modifies the contents of the packet by injecting some extra bytes into the frame before sending it to the RTU to disrupt the normal operation of the system through buffer overflow attack.
Detection: The injected bytes violate the Length-TI model of the behavior-model rules along with the violation of IP-MAC pair signatures which results in the generation of alerts. And a behavior profiling on these sets of alerts is performed, and hence the data modification attack is successfully detected and visualized.

### 7.3. Malicious RTU behavior

Simulated Attack: An attacker injects a malicious RTU configuration file to modify the field values in the ASDU, which carries data from RTU to MTU. The attack is intended to mislead the operator with wrong information about the process being monitored. The malicious code is written to intercept M_ME_NA_1 (which carries measured data from analog sensors) and modify the measured line frequency value at frequency sensor with IOA address <X> to a safe value, although the actual value is unsafe.
Detection: The spontaneous ASDU leaving the RTU is captured at the SMU, then the values sent in the message for the particular device address <X> are correlated with the actual field data from the same device (obtained through the redundant port of the device). As the value sent by the RTU is deviating from the actual value, an alert is successfully triggered to indicate abnormal behavior of the RTU.

### 7.4. Flooding attack on RTU

Simulated Attack: An attacker spoofs his identity to an authorized MTU and uses the hping3 tool on Kali Linux to perform a flooding attack on RTU. The attack overwhelms the network bandwidth and the RTU resource to cause a denial of service on the RTU. This will disrupt the power system operation by making it unavailable for legitimate monitoring and control [9].
Detection: The flooding attack is successfully detected based on the deviation in average packet rate between the particular MTU and RTU, along with the deviation in RTU response time. The attack is detected though the identity (IP, MAC) of the attacker looks authenticated.

## 8. Conclusion

IEC-60870-5-104 lacks a security mechanism both at the application layer and the data link layer and it also lacks signatures of past attacks. As suggested by the NIST guidelines on ICS security, the proposed solution uses a white-listing approach in passive monitoring mode to add a layer of cyber security for IEC-60870-5-104 based SCADA systems, without adding overhead to the sensitive network. With the proposed data-correlation, white-listed Signatures, protocol behavior models and Communication pattern/flow based anomaly rules; SMU is capable of detecting many known malicious attacks and also several unknown zero-day attacks on the SCADA systems.

## 9. References

[1] IEC Power Systems Management and Associated Information Exchange—Data and Communications Security—Part 5: "Security for IEC 60870-5 and Derivatives", IEC Standard 62351, 2009.

[2] Bindhumadhava BS, Senthil Kumar RK, Kalluri R, Pidikiti DS(2013),"SCADA communication protocols: vulnerabilities, attacks and possible mitigations". In: 2013.

[3] Yikai Xu, Yi Yang, Tianran Li, Jiaqi Ju, Qi Wang, "Review on cyber vulnerabilities of communication protocols in industrial control systems". Published in: 2017 IEEE Conference on Energy Internet and Energy System Integration (EI2).



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010

[4] Maynard P, et al. “Towards Understanding Man-In-The-Middle Attacks on IEC 60870-5-104 SCADA Networks”, International Symposium for ICS & Scada Cyber Security Research. 2014:30-42.

[5] Stouffer, Keith, Joe Falco, and Karen Scarfone. "Guide to industrial control systems (ICS) security." NIST special publication 800, no. 82 (2011): 16-16.

[6] International Electrotechnical Commission. "IEC 60870–5-104 Telecontrol equipment and systems-Part 5-104: Transmission protocols–Network, access for IEC 60870-5-101 using standard transport profiles." IEC Standard Document (2006): 60870-5.

[7] Equipment, IEC Telecontrol. "Systems—Part 5-101: Transmission Protocols—Companion Standard for Basic Telecontrol Tasks." IEC Standard 60870 (2003).

[8] Y. Yang, K. McLaughlin, T. Littler, S. Sezer, B. Pranggono, and H. F.Wang, "Intrusion Detection System for IEC 60870-5-104 based SCADA networks," in Proc. 2013 IEEE Power and Energy Society General Meeting, pp. 1-5.

[9] Amaraneni, Abhiram, Mahendra Lagineni, Rajesh Kalluri, R. K. Senthilkumar, and GL Ganga Prasad. "Transient analysis of cyber-attacks on power SCADA using RTDS." Power Research 11, no. 1 (2015): 79-92.

[10] Kalluri Rajesh, Lagineni Mahendra, R. K. Senthil Kumar, G. L. Ganga Prasad, and B. S. Bindhumadhava. "Analysis of Communication Channel Attacks on Control Systems—SCADA in Power Sector." In ISGW 2017: Compendium of Technical Papers, pp. 115-131. Springer, Singapore, 2018.

[11] Kalluri, Rajesh, Lagineni Mahendra, RK Senthil Kumar, and GL Ganga Prasad. "Simulation and impact analysis of denial-of-service attacks on power SCADA." In 2016 national power systems conference (NPSC), pp. 1-5. IEEE, 2016.

[12] Pf-ring, “High speed packet capture, filtering and analysis”Ntop, https://www.ntop.org/products/packet-capture /pf_ring/ (accessed March 5, 2021).

[13] vol. 7, pp. 179-186, May. 2011. Ali A. Ghorbani, Wei Lu, and Mahbod Tavallaee, “Network Intrusion Detection and Prevention: concepts and techniques”. London: Springer, 2010, pp. 27-49.

[14] Y. Yang, K. McLaughlin, S. Sezer, Y.B Yuan, W. Huang, "Stateful Intrusion Detection for IEC 60870-5-104 SCADA Security," 2014 IEEE PES General Meeting | Conference & Exposition.

[15] Igor Nai Fovino, Andrea Carcano, "Modbus/DNP3 State-Based Intrusion Detection System", Published in: 2010 24th IEEE International Conference on Advanced Information Networking and Applications.

[16] Payam Mahmoudi Nasr, Ali Yazdian Varjani "Alarm based anomaly detection of insider attacks in SCADA system", in 2014 Smart Grid Conference (SGC).

[17] Steven Cheung, Bruno Dutertre "Using Model-based Intrusion Detection for SCADA Networks", In Proc. 2007 the SCADA Security Scientific Symposium, pp. 127–134.

[18] Mahendra, Lagineni, Rajesh Kalluri, R. K. Senthil Kumar, B. S. Bindhumadhava, and G. L. Ganga Prasad. "SCADA Research Lab Kit for Educational Institutes." IETE Journal of Education (2019): 1-11.



 | DOI: 10.20533/ijicss.9781.9083.20346.2020.0010