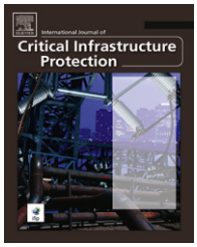
available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/ijcip



Attack taxonomies for the Modbus protocols

*Peter Huitsing, Rodrigo Chandia, Mauricio Papa, Sujeet Shenoi**

Department of Computer Science, University of Tulsa, 800 S. Tucker Drive, Tulsa, OK 74104, USA

ARTICLE INFO

Article history:
Received 1 May 2008
Accepted 6 August 2008

Keywords:
Modbus serial
Modbus TCP
Attacks
Attack taxonomies

ABSTRACT

The Modbus protocol and its variants are widely used in industrial control applications, especially for pipeline operations in the oil and gas sector. This paper describes the principal attacks on the Modbus Serial and Modbus TCP protocols and presents the corresponding attack taxonomies. The attacks are summarized according to their threat categories, targets and impact on control system assets. The attack taxonomies facilitate formal risk analysis efforts by clarifying the nature and scope of the security threats on Modbus control systems and networks. Also, they provide insights into potential mitigation strategies and the relative costs and benefits of implementing these strategies.

1. Introduction

The Modbus protocol defines the message structure and communication rules used by process control systems to exchange supervisory control and data acquisition (SCADA) information for operating and controlling industrial processes [1]. Modbus' open protocol specifications and TCP extension have contributed to its popularity, especially in the oil and gas sector, where it is the predominant control protocol for pipeline operations.

The Modbus protocol has two principal variants, Modbus Serial [6] and Modbus TCP [5]. In the Modbus Serial protocol, messages are transmitted between a master and slaves (field devices) over serial lines using the ASCII or RTU transmission modes. The newer Modbus TCP protocol provides connectivity within a Modbus network (master and its slaves) as well as for IP-interconnected Modbus networks (multiple masters, each communicating with possibly overlapping sets of slaves). The TCP variant enables a master to have multiple outstanding transactions and permits a slave to engage in concurrent communications with multiple masters.

Attacks on Modbus systems and networks can produce effects ranging from sporadic disruptions of field devices (sensors and actuators) to large-scale outages or even loss of control in the case of a spoofed master. The attacks can be grouped into three categories. The first category includes attacks that exploit the Modbus protocol specifications. The second category comprises attacks that exploit vendor implementations of the Modbus protocols. Attacks in the third category target the support infrastructure, which includes information technology, networking and telecommunications assets.

This paper considers attacks in the first category, i.e., attacks that are common to all Modbus implementations that conform to the protocol specifications [4–6]. The analysis focuses on the Modbus Serial and TCP protocols and presents the corresponding attack taxonomies. The primary targets include the master, field devices, serial communication links (Modbus Serial) or network communication paths (Modbus TCP). Four threats are considered: interception, interruption, modification and fabrication. Attack preconditions include the availability of a Modbus sniffer and/or packet injector. Avenues for attack include the master, field devices and serial communication links or network communication paths.

Our comprehensive analysis of Modbus has identified 20 and 28 attacks for the serial and TCP protocols, respectively.

* *Corresponding author.*
E-mail address: sujeet@utulsa.edu (S. Shenoi).

1874-5482/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.ijcip.2008.08.003

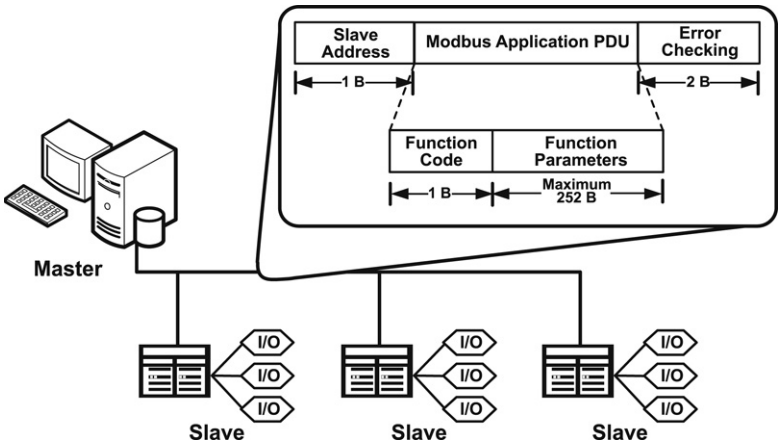


Fig. 1 – Modbus serial architecture.

These attacks can be used to target Modbus assets in 59 and 113 distinct ways for the serial and TCP protocols, respectively. For reasons of space and sensitivity, it is not possible to discuss all the attacks in detail. However, representative attacks are discussed and the corresponding attack taxonomies are presented. The attack taxonomies provide insights into the nature and scope of security threats as well as strategies for securing Modbus systems and networks.

2. Modbus protocol

Originally formulated in 1979, the Modbus protocol is one of the oldest, but most widely used, industrial control protocols [4–6]. Modbus engages a simple request/reply communication mechanism between a control center and field devices. For example, a control center (master unit) might send a "read" message to a sensor (slave device) to obtain the value of a process parameter (e.g., pressure). Alternatively, it might send a "write" message to an actuator (slave device) to perform a control action (e.g., open a valve).

A unicast transaction involving a master and an addressed slave involves two messages, a request message (e.g., to measure pressure or open a valve) and the corresponding response message (e.g., the pressure measurement or an acknowledgment that the valve was opened, or an error message indicating that the operation could not be performed). A broadcast transaction involves the master sending a message to all the slaves; the slaves do not send response messages. An example broadcast transaction is a "write" message that resets all the sensors and actuators.

Modbus communications occur over serial lines or, more recently, using TCP/IP as a transport mechanism. The following sections describe the Modbus Serial and TCP protocols in more detail.

2.1. *Modbus serial protocol*

Modbus Serial protocol messages are transmitted between a master and slave devices over serial lines using the ASCII or RTU transmission modes (Fig. 1). The messages have three components: (i) slave address, (ii) Modbus application protocol data unit (PDU), and (iii) an error checking field. The slave address in a request message identifies the recipient; the corresponding address in a response message identifies the responding slave. A unicast message has an address in the [1, 247] range that identifies an individual slave. A broadcast message uses a slave address of zero. Values in the [248, 255] range are reserved addresses.

The Modbus PDU has two fields, a one-byte function code and function parameters (maximum 252 bytes). The function code field in a request message specifies the operation requested by the master; the corresponding field in the response message is used to convey status information to the master (e.g., error information when an exception occurs in the slave device). The function parameters field contains data pertaining to function invocation (request messages) or function results (response messages).

Modbus function codes specify read and write operations on slaves, diagnostic functions and error conditions. Modbus has three types of function codes: public codes, user-defined codes and reserved codes. Public codes correspond to functions whose semantics are completely defined in the Modbus standard. Valid public codes fall in the non-contiguous ranges: [1, 64], [73, 99] and [111, 127]. User-defined codes in the [65, 72] and [100, 110] ranges are not considered in the Modbus standard; their implementations are left to vendors. Reserved function codes are public codes that may be used to ensure compatibility with legacy systems. Function code values in the unused range [128, 255] indicate error conditions in response messages.

Response messages have the same structure as request messages. The Modbus specification defines positive and negative responses to request messages. A positive response informs the master that the slave has successfully performed the requested action; in this case, the function code of the request message is included in the response message. A negative or exception response notifies the master that the transaction could not be performed by the addressed slave. The function code for a negative response is computed by adding 128 to the function code of the request message;

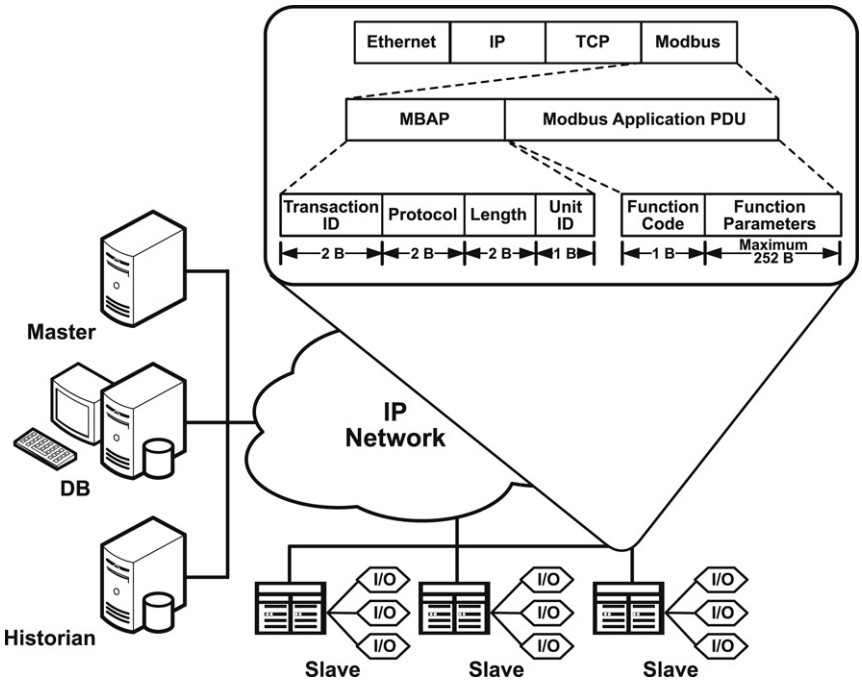


Fig. 2 – Modbus TCP architecture.

thus, function codes in the [128, 255] range denote error conditions. A negative response also includes an exception code as a function parameter, which provides information about the cause of the error. The Modbus specification defines nine exception responses whose format and content depend on the issuing entity and the type of event producing the exception.

2.2. *Modbus TCP protocol*

The Modbus TCP protocol provides connectivity within a LAN-based Modbus network (a master and its slaves) as well as for IP-interconnected Modbus networks (multiple masters, each with multiple slaves). Modbus TCP extends its serial counterpart by enabling a master to have multiple outstanding transactions, and a slave to engage in concurrent communications with multiple masters.

Fig. 2 shows a master connected to multiple slaves via an IP network. Note that the master may also be connected to other resources in the control center such as databases and historians.

Modbus slaves listen for incoming TCP connections on port 502 (IANA assigned port) or optionally on additional ports. In a Modbus TCP transaction, the slave is designated as the "server" because it performs the passive open operation on TCP; the Modbus master, which performs the active open operation on TCP, is designated as the "client". Once a TCP communication channel is established, Modbus roles cannot be changed on that channel; however, multiple outstanding transactions can exist on the channel. A new communication channel must be opened to permit a Modbus device to assume a different role.

Modbus TCP transactions are similar to their serial counterparts—devices exchange PDUs, except that the transactions are encapsulated in TCP messages. Consequently, a Modbus TCP PDU includes the Modbus application protocol (MBAP) in addition to the Modbus application PDU used in the serial protocol.

The MBAP header has four fields: (i) transaction identifier, (ii) protocol identifier, (iii) length, and (iv) unit identifier (Fig. 2). The transaction identifier permits devices to pair matching requests and replies on a communication channel. The protocol identifier indicates the application protocol encapsulated by the MBAP header (zero for Modbus). The length field indicates the length in bytes of the remaining fields (unit identifier and PDU). The unit identifier indicates the slave associated with the transaction (this is used only in legacy implementations).

The Modbus TCP specification requires that only one application PDU be transported in the payload of a TCP packet. Since application PDUs have a maximum size of 253 bytes and the length of the MBAP is fixed at seven bytes, the maximum size of a Modbus TCP data unit is 260 bytes.

3. Attack identification methodology

In general, attacks on Modbus systems and networks can be grouped into three categories: (i) attacks that exploit the Modbus protocol specifications, (ii) attacks that exploit vendor implementations of the Modbus protocols, and (iii) attacks that target the support infrastructure, which includes information technology, networking and telecommunications assets.

This paper considers attacks in the first category, i.e., attacks that are common to all Modbus systems and

Table 1 – Modbus serial attacks (DigitalBond attacks are underlined)

20 distinct attacks (59 instances)	Master	Field device	Comm. link	Message
Interception		S5-1 B6-1 B7-1 B8-1 B9-1 B12-1 B14-1	B6-2 B9-2 B12-2 B14-2	B14-3
Interruption	B2-1 B10-1 B14-4	S2-1 S3-1 S4-1 B1-1 B2-2 B4-1 B5-1 B10-2 B11-1 B14-5 B15-1	B14-6	B2-3 B11-2 B14-7
Modification	B2-4 B3-1 B10-3 B11-3 B13-1 B14-8 B15-2	S1-1 S2-2 S3-2 S4-2 B1-2 B2-5 B3-2 B4-2 B5-2 B10-4 B11-4 B13-2 B14-9 B15-3	B14-10	B14-11
Fabrication	B4-3 B14-12	B2-6 B14-13	B14-14	B14-15

networks that conform to the protocol specifications. Note that some control system vendors may not implement certain Modbus function codes and/or sub-function codes (especially diagnostic codes) listed in the Modbus specifications. Obviously, attacks that exploit these unimplemented codes would not work.

Our attack identification methodology involved the comprehensive analysis of each protocol. Four threat categories, interception, interruption, modification and fabrication [7], were analyzed for each of the primary targets. In the case of the Modbus Serial protocol, the primary targets are the master, field devices, serial communication links and messages. The corresponding targets for Modbus TCP are the master, field devices, network communication paths and messages.

Attacks were theorized based on the availability of a Modbus sniffer and a packet injector with the ability to block, modify and fabricate arbitrary Modbus messages and sequences of messages. The principal entry points for the attacks included the master, field devices and serial communication links in the case of Modbus Serial; and the master, field devices and network communication paths in the case of Modbus TCP. Instances of each attack were identified by examining the various manifestations of the attack on the targeted assets. For example, Direct Slave Control, an attack that involves locking out a master and controlling one or more field devices, can be used to interrupt and modify a field device, as well as to fabricate a master.

The attack instances were used to create attack taxonomies for the Modbus Serial and TCP protocols (Tables 1 and 2, respectively). The rows of the tables list the threat categories (interception, interruption, modification and fabrication) while the columns list the targeted assets. The impact of each attack instance on each of the targeted assets was also evaluated; the results are summarized in Tables 3 and 4. The results of the attacks include accessing message or field device data, denying service to the master unit, field devices or communication links or network paths, providing bad data or rewriting key data in the master unit, field devices or messages, and seizing control of the master unit or field devices.

4. Modbus attacks

This section discusses attacks on the Modbus Serial and TCP protocols. To simplify the presentation, the attacks are divided into three groups: (i) attacks unique to the Modbus Serial protocol, (ii) attacks common to the Modbus Serial and TCP protocols, and (iii) attacks unique to the Modbus TCP protocol.

Tables 1 and 2 present the attack taxonomies for the Modbus Serial and TCP protocols, respectively. Modbus Serial attacks are designated by S*x*–*y*, where *x* identifies the attack and *y* denotes the instance of the attack (i.e., the exploitation of a Modbus asset). Attacks common to the Modbus Serial and TCP protocols, and attacks unique to the Modbus TCP protocol are designated as B*x*–*y* and T*x*–*y*, respectively.

Our analysis of Modbus Serial has identified 20 distinct attacks and 59 attack instances. Modbus TCP is a more complex protocol; consequently, the numbers of attacks and attack instances are higher—28 and 113, respectively.

To our knowledge, DigitalBond [2] is the only entity to have published information about attacks on the Modbus protocol. In 2004, DigitalBond produced several Modbus attack signatures [3] under a DHS HSARPA project. These signatures were eventually implemented as a set of twelve Snort rules [8]. In April 2007, DigitalBond published two additional Modbus attacks [3]. Tables 1 and 2 identify the DigitalBond attacks as underlined instances (e.g., S5-1, B6-1 and T1-1).

4.1. *Serial only attacks*

Five distinct attacks (designated as S1 through S5) have been identified for the Modbus Serial protocol. All five attacks require the use of a Modbus protocol sniffer and message generator, along with connectivity to the master device or a serial communication link. They involve sending one or more fabricated Modbus messages with special function code and/or sub-function code (parameter) values.

Table 2 – Modbus TCP attacks (DigitalBond attacks are underlined)

28 distinct attacks (113 instances)	Master	Field device	Network path	Message
Interception		T2-1 T4-1 B6-1 B7-1 B8-1 B9-1 B12-1 B14-1	T2-2 B6-2 B9-2 B12-2 B14-2	T2-3 T4-2 B14-3
Interruption	T1-1 T2-4 T3-1 T4-3 T5-1 T6-1 T7-1 T8-1 T9-1 T10-1 T11-1 T12-1 T13-1 B2-1 B10-1 B14-4	T1-2 T2-5 T3-2 T4-4 T5-2 T6-2 T7-2 T8-2 T9-2 T10-2 T11-2 T12-2 T13-2 B1-1 B2-2 B4-1 B5-1 B10-2 B11-1 B14-5 B15-1	T2-6 T4-5 T8-3 T9-3 T10-3 T11-3 B14-6	T2-7 T4-6 T7-3 T8-4 T9-4 T10-4 T11-4 T12-3 T13-3 B2-3 B11-2 B14-7
Modification	T3-3 B2-4 B3-1 B10-3 B11-3 B13-1 B14-8 B15-2	T1-3 T2-8 T3-4 T4-7 T13-4 B1-2 B2-5 B3-2 B4-2 B5-2 B10-4 B11-4 B13-2 B14-9 B15-3	T2-9 T4-8 B14-10	T2-10 B14-11
Fabrication	T1-4 T2-11 B4-3 B14-12	T2-12 B2-6 B14-13	T2-13 T4-9 B14-14	T1-5 T2-14 B14-15

The Modbus Serial attacks impact control system assets in various ways. Attacks on confidentiality involve reading Modbus messages or obtaining configuration data from slave devices. Attacks on integrity involve inserting erroneous data or reconfiguring slave devices. Attacks on availability cause slave devices to lose key functionality (e.g., the ability to read or produce Modbus messages), or to reboot or crash. We discuss three Modbus Serial attacks in more detail.

- Diagnostic Register Reset (S1): This attack sends a Modbus message with function code 08 and sub-function code 0A, which clears all the counters and the diagnostic register of the addressed field device. The attack changes the configuration of the field device and impacts diagnostic operations, but does not affect control and communications functionality. Table 1 shows the one instance of this attack: modification (S1-1) of a field device.
- Remote Restart (S4): This attack sends a Modbus message with function code 08 and sub-function code 01, which causes the addressed field device to restart and execute its power-up test. The addressed field device is rendered inoperable when asked to restart repeatedly. Two instances of this attack are shown in Table 1: interruption (S4-1) and modification (S4-2) of a field device.
- Slave Reconnaissance (S5): This attack on confidentiality sends a Modbus message with function code 17, which causes the addressed field device to return status information. Table 1 lists the one instance of this attack: interception (S5-1) of field device information.

4.2. *Serial and TCP attacks*

A total of fifteen attacks (designated as B1 through B15) exploit both the Modbus Serial and TCP protocols. The attacks require the use of a Modbus protocol sniffer and message generator, along with access to the master device or serial communication link (Modbus Serial), or the master device, network communication path or field device (Modbus TCP).

Attacks on confidentiality involve reading Modbus messages, obtaining network information, or obtaining configuration and other data from slave devices. Attacks on integrity involve inserting erroneous data in Modbus messages or network traffic, modifying communication paths, providing bad data to the master or reconfiguring slave devices. These attacks may also spoof the master unit, field devices, network paths or messages. Attacks on availability result in denial of service of the master or slave devices; in particular, slave devices may lose key functionality, reboot or crash. It is also possible to block Modbus messages and disable communication links or network paths. The most serious attacks are those that disable or bypass the master unit and seize control of field devices. We discuss seven attacks common to the Modbus Serial and TCP protocols in more detail.

- Broadcast Message Spoofing (B1): This attack involves sending fake broadcast messages to slave devices. The attack is difficult to detect because no response messages are returned from the slaves to the master device. Tables 1 and 2 list two instances of this attack: interruption (B1-1) and modification (B1-2) of a field device.
- Baseline Response Replay (B2): This attack involves recording genuine traffic between a master and a field

device, and replaying some of the recorded messages back to the master. Tables 1 and 2 list six instances of this attack: interruption of a master (B2-1), field device (B2-2) and message (B2-3), modification of a master (B2-4) and field device (B2-5), and fabrication of a field device (B2-6).

- Direct Slave Control (B4): This attack involves locking out a master and controlling one or more field devices. Tables 1 and 2 list three instances of this attack: interruption (B4-1) and modification (B4-2) of a field device, and fabrication of a master (B4-3), which is the most serious of all the attacks.
- Modbus Network Scanning (B6): This attack involves sending benign messages to all possible addresses on a Modbus network to obtain information about field devices. Tables 1 and 2 list two instances of this attack: interception of field device data (B6-1) and interception of network topology information (B6-2).
- Passive Reconnaissance (B9): This attack involves passively reading Modbus messages or network traffic. Tables 1 and 2 list two instances of this attack: interception of field device data (B9-1) and interception of network topology information (B9-2).
- Response Delay (B11): This attack involves delaying response messages so that the master receives out-of-date information from slave devices. Tables 1 and 2 list four instances of this attack: interruption of a field device (B11-1) and message (B11-2), modification of a master (B11-3) and modification of a field device (B11-4).
- Rogue Interloper (B14): This attack involves attaching a computer with the appropriate (serial or Ethernet) adaptors to an unprotected communication link. This "man-in-the-middle" device can read, modify and fabricate Modbus messages and/or network traffic at will. Tables 1 and 2 list fifteen instances of this most serious attack: interception of field device (B14-1), network (B14-2) and message (B14-3) data; interruption of a master (B14-4), field device (B14-5), network (B14-6) and message (B14-7); modification of a master (B14-8), field device (B14-9), network (B14-10) and message (B14-11); and fabrication of a master (B14-12), field device (B14-13), network (B14-14) and message (B14-15).

4.3. *TCP only attacks*

Thirteen attacks (designated as T1 through T13) are unique to Modbus TCP. These attacks require the use of a Modbus protocol sniffer and message generator, along with access to the master device, network path or field device.

Attacks on confidentiality involve reading field device data, network traffic or messages. Attacks on integrity involve inserting erroneous data in Modbus messages or network traffic, or reconfiguring master or field devices. It is also possible to spoof the master or field devices. Attacks on availability can take down TCP/IP network connections, or cause the master or field devices to lose key functionality, reboot or crash. The most serious attacks may seize control of the master and/or field devices. We discuss four Modbus TCP attacks in more detail.

- Irregular TCP Framing (T5): Multiple Modbus messages cannot be placed in a single TCP frame. This attack injects improperly framed messages or modifies legitimate messages to create improperly framed messages, which may cause a master unit or field device to close a connection. Table 2 lists two instances of this attack: interruption of a master (T5-1) and interruption of a field device (T5-2).
- TCP FIN Flood (T9): This attack launches a spoofed TCP packet with the FIN flag set after a legitimate Modbus message to a Modbus client (master) or server (field device) to close the TCP connection. Table 2 lists four instances of this attack: interruption (T9-1) of a master, interruption of a field device (T9-2), interruption of a network connection (T9-3), and interruption of a message (T9-4).
- TCP Pool Exhaustion (T10): The Modbus TCP specification describes two classes of connection pools: priority connection pools and non-priority connection pools. Exhausting the connections in these pools prevents a Modbus device from accepting new connections. The attack opens large numbers of TCP connections with a device using marked IP addresses corresponding to priority connections and unmarked IP addresses corresponding to non-priority connections. Note that network activity must be maintained in all these connections to implement a denial-of-service attack. Table 2 lists four instances of this attack: interruption (T10-1) of a master, interruption of a field device (T10-2), interruption of a network path (T10-3), and interruption of a message (T10-4).
- TCP RST Flood (T11): This attack launches a spoofed TCP packet with the RST flag set after a legitimate Modbus message to a Modbus client (master) or server (field device) to close the TCP connection. Table 2 lists four instances of this attack: interruption (T11-1) of a master, interruption of a field device (T11-2), interruption of a network connection (T11-3), and interruption of a message (T11-4).

5. Attack impact

Table 3 summarizes the impact of the twenty Modbus Serial attacks (59 attack instances). A total of twelve attack instances affect confidentiality—seven enable an attacker to obtain information about field devices, four impact communication links, and one affects messages.

Of greater concern are the eighteen attack instances that interrupt the master unit (3 instances), field devices (11), link operation (1) and message passing (3). Equally serious are the 23 attack instances that modify Modbus assets—seven alter the data received by the master, eleven result in field devices obtaining bad data, and one each affect communication links and messages; three other modification attack instances result in improper control of field devices.

Six fabrication attack instances target Modbus Serial assets—two each affect the master and field devices and one each impact communication links and messages. The two fabrication attack instances on the master are most serious as they lock out the master unit and seize control of the slave devices.

Table 3 – Impact of Modbus serial attacks on target assets

20 distinct attacks (59 instances)	Master	Field device	Comm. link	Message
Interception		7 obtain field device data	4 obtain comm. link data	1 read message
Interruption	3 DoS master	11 DoS field device	1 DoS comm. link	3 block message
Modification	7 bad data in master	11 bad data in field device 3 control field device	1 bad traffic	1 bad data in message
Fabrication	2 control process	2 fabricated field device	1 fabricated comm. link	1 fabricated message

Table 4 – Impact of Modbus TCP attacks on target assets

28 distinct attacks (113 instances)	Master	Field device	Network path	Message
Interception		8 obtain field device data	5 obtain network data	3 read message
Interruption	16 DoS master	21 DoS field device	7 DoS network path	12 block message
Modification	8 bad data in master	12 bad data in field device 3 control field device	3 bad traffic	2 bad data in message
Fabrication	4 control process	3 fabricated field device	3 fabricated network path	3 fabricated message

Table 5 – Impact of attacks on control objectives

	Serial attacks	TCP attacks	Common attacks
Loss of confidentiality	12 (7)	16 (8)	11 (6)
Loss of awareness	27 (13)	67 (23)	24 (10)
Loss of control	20 (14)	30 (23)	16 (10)

Table 4 summarizes the impact of the 28 Modbus TCP attacks (113 attack instances). Sixteen instances correspond to attacks on confidentiality—eight target field devices, five target network paths and three target messages exchanged between the master and field devices.

Much more serious is the fact that 56 attack instances (nearly one-half of all attack instances) impact the availability of Modbus assets—sixteen instances result in denial-of-service of the master unit, 21 affect field devices and seven disrupt network paths, while twelve attack instances block individual messages. A total of 28 attack instances modify Modbus assets—eight alter the data received by the master, twelve result in field devices obtaining bad data, three impact network paths and two affect Modbus TCP messages; three other modification attack instances result in improper control of field devices.

Thirteen fabrication attack instances target Modbus TCP assets—four affect the master, and three each affect field devices, network paths and messages. Most worrisome are the four master fabrication instances that spoof the master and seize control of the process.

To clarify the impact of Modbus attacks and facilitate formal risk analysis efforts, it is useful to categorize the attack instances based on their impact on three high-level control system objectives: confidentiality of process data, awareness of the process and the ability to control the process (Table 5).

Loss of confidentiality occurs when an attack reveals information about field devices, network topology or messages. Loss of awareness occurs when operators are unable to obtain accurate and timely information about a process either due to denial of service or data modification; this row lists attack instances and attacks that interrupt field devices, network connectivity or messages, as well as those that modify the master or involve the fabrication of field devices. The worst category, loss of control, occurs when an attacker spoofs the master and/or seizes control of the process; this row includes attack instances and attacks that modify field devices, network paths or messages as well as those that result in the fabrication of the master, network paths or messages.

Table 5 summarizes the impact of attack instances and distinct attacks on the three control system objectives. Note that the numbers of distinct attacks in each cell of the table are listed in parentheses (e.g., the seven distinct Modbus Serial attacks that result in a loss of confidentiality).

6. Conclusions

Our detailed analysis of the Modbus Serial and TCP protocol specifications with respect to threats, control system targets and attack entry points has facilitated the identification of attacks and their categorization within attack taxonomies. The analysis of the protocols, while thorough, is certainly not comprehensive. Indeed, we believe that many attacks are yet to be theorized. Nevertheless, the numbers of attacks and attack instances discovered are much higher than expected. Even more surprising is the large proportion of high-impact attacks, especially those involving the interruption, modification and fabrication of control system assets.

We hope that our work will inspire renewed efforts at characterizing attacks on Modbus and other SCADA protocols. The results will help clarify the nature and scope of the threats facing critical infrastructure assets. Also, they will support formal risk analysis and risk mitigation strategies as well as the design and deployment of next generation SCADA protocols that are secure, reliable and resilient.

Acknowledgements

This work was partially supported by the Institute for Information Infrastructure Protection (I3P) at Dartmouth College, Hanover, New Hampshire, under Award 2003-TK-TX-0003 and Award 2006-CS-001-000001 from the US Department of Homeland Security.

REFERENCES

[1] S. Boyer, SCADA: Supervisory Control and Data Acquisition, Instrumentation, Systems and Automation Society, Research Triangle Park, North Carolina, 2004.

[2] DigitalBond. www.digitalbond.com.

[3] DigitalBond, Modbus TCP IDS signatures. www.digitalbond.com/wiki/index.php/Modbus_TCP_IDS_Signatures.

[4] Modbus IDA, MODBUS Application Protocol Specification v1.1a, North Grafton, Massachusetts. www.modbus.org/specs.php, 2004.

[5] Modbus IDA, MODBUS Messaging on TCP/IP Implementation Guide v1.0a, North Grafton, Massachusetts. www.modbus.org/specs.php, 2004.

[6] Modbus.org, MODBUS over Serial Line Specification and Implementation Guide v1.0, North Grafton, Massachusetts. www.modbus.org/specs.php, 2002.

[7] C. Pfleeger, S. Lawrence Pfleeger, Security in Computing, Prentice Hall, Upper Saddle River, New Jersey, 2007.

[8] Snort.org, Snort – The *de facto* standard for intrusion detection and prevention. www.snort.org.