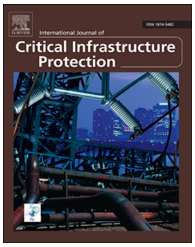
INTERNATIONAL JOURNAL OF CRITICAL INFRASTRUCTURE PROTECTION 6 (2013) 63–75

Available online at www.sciencedirect.com

SciVerse ScienceDirect

www.elsevier.com/locate/ijcip



Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems

*Niv Goldenberg, Avishai Wool**

School of Electrical Engineering, Tel Aviv University, Ramat Aviv 69978, Israel

ARTICLE INFO

Article history:
Received 9 January 2013
Accepted 24 April 2013
Available online 4 May 2013

Keywords:
SCADA systems
Modbus/TCP
Network intrusion detection system

ABSTRACT

The Modbus/TCP protocol is commonly used in SCADA systems for communications between a human–machine interface (HMI) and programmable logic controllers (PLCs). This paper presents a model-based intrusion detection system designed specifically for Modbus/TCP networks. The approach is based on the key observation that Modbus traffic to and from a specific PLC is highly periodic; as a result, each HMI-PLC channel can be modeled using its own unique deterministic finite automaton (DFA). An algorithm is presented that can automatically construct the DFA associated with an HMI-PLC channel based on about 100 captured messages. The resulting DFA-based intrusion detection system looks deep into Modbus/TCP packets and produces a very detailed traffic model. This approach is very sensitive and is able to flag anomalies such as a message appearing out of its position in the normal sequence or a message referring to a single unexpected bit. The intrusion detection approach is tested on a production Modbus system. Despite its high sensitivity, the system has a very low false positive rate—perfect matches of the model to the traffic were observed for five of the seven PLCs tested without a single false alarm over 111 h of operation. Furthermore, the intrusion detection system successfully flagged real anomalies that were caused by technicians who were troubleshooting the HMI system. The system also helped identify a PLC that was configured incorrectly.

1. Introduction

Supervisory control and data acquisition (SCADA) systems are used for monitoring and controlling numerous industrial and infrastructure processes. In particular, SCADA systems are used in critical infrastructure assets such as chemical plants, electric power generation, transmission and distribution systems, water distribution networks and wastewater treatment facilities. SCADA systems have a strategic significance due to the potentially serious consequences of a fault or malfunction.

SCADA systems typically incorporate sensors and actuators that are controlled by programmable logic controllers (PLCs), which are themselves managed using a human–machine interface (HMI). SCADA systems were originally designed for serial communications and were built on the premise that all the operating entities would be legitimate, properly installed, perform the intended logic and follow the protocol. Thus, many SCADA systems have almost no measures for defending against deliberate attacks. Specifically, SCADA network components do not verify the identity and permissions of other components with which they interact (i.e., no authentication and authorization mechanisms); they do not verify message content and legitimacy (i.e., no data integrity checks); and all the data sent over the network is in plaintext (i.e., no encryption to preserve confidentiality).

*Corresponding author.
E-mail address: yash@acm.org (A. Wool).

1874-5482/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ijcip.2013.05.001

Meanwhile, technological and economic trends have driven SCADA systems away from proprietary components and serial communications to off-the-shelf commodity components and IP-based communications protocols. The Modbus/TCP protocol is commonly used in SCADA networks for HMI-PLC communications. An attacker who injects malicious Modbus messages a SCADA network could cause significant damage. Therefore, deploying an intrusion detection system in a Modbus network is an important defensive measure.

This paper describes a model-based intrusion detection system designed specifically for Modbus/TCP networks. The detection approach is based on the fact that Modbus traffic to and from a specific PLC is highly periodic, with the same messages being sent repeatedly according to a fixed pattern. As a result, it is possible to model each HMI-PLC channel using its own unique deterministic finite automaton (DFA).

An algorithm is presented for automatically constructing a DFA associated with a HMI-PLC channel based on about 100 captured messages. The resulting DFA-based intrusion detection system looks deep into Modbus/TCP packets and produces a traffic model that captures detailed packet characteristics—not just function codes, but also the specific registers and coils referred to by messages. Based on the packet characterization, the model captures the precise periodic traffic pattern between an HMI and a PLC. Thus, the intrusion detection approach is very sensitive and is able to flag anomalies such as a message appearing out of position in the normal sequence or a message referring to a single unexpected bit.

The intrusion detection approach was tested on a production Modbus system that controls electric power supply at Tel Aviv University. The testing used more than 120 h of live traffic collected in two sessions several months apart. Despite its high sensitivity, the intrusion detection system has a very low false positive rate—five of the seven PLCs tested yielded perfect matches of the model to traffic, without a single false alarm over 111 h of operation. The system successfully flagged real anomalies produced when technicians were troubleshooting the HMI system. Moreover, the system helped identify a PLC that was configured incorrectly.

2. Related work

Media coverage of cyber attacks such as Stuxnet [3] has emphasized the need for strong and reliable security mechanisms for SCADA systems. Several researchers have focused on intrusion detection approaches for SCADA systems. Yang et al. [23] employed an auto associative kernel regression model coupled with a statistical probability ratio test to match patterns in simulated SCADA systems. Their model uses predetermined features, representing network traffic and hardware operating statistics, for intrusion detection.

Tsang and Kwong [19] proposed a detection approach based on an unsupervised anomaly-learning model. They developed an ant colony clustering model based multi-agent decentralized intrusion detection system. Their approach has been shown to reduce data dimensionality while preserving model accuracy.

Naess et al. [15] have proposed the use of interval-based sensors, procedural-based sensors and misuse-based detectors. Interval-based sensors identify if parameter values and method invocation frequencies fall within their predefined ranges. Procedural-based sensors are embedded at the entry and exit points of applications to monitor their execution patterns. Misuse-based detectors are positioned within application code at locations where vulnerabilities are known to exist.

Gao et al. [8] have presented a neural network based intrusion detection system that monitors the physical behavior of control systems to detect artifacts of command and response injection denial-of-service attacks.

Digital Bond [7] has specified a set of Modbus/TCP Snort rules for intrusion detection. The set includes fourteen rules that are broadly divided into three groups: (i) unauthorized Modbus protocol use; (ii) Modbus protocol errors; and (iii) scanning. Our method successfully detects all the anomalies encoded in the Digital Bond Snort rules. However, during an evaluation using a production Modbus/TCP system, our method flagged real anomalies that the Snort rules were unable to catch.

Nai Fovino et al. [16] have presented a state-based intrusion detection system. Their approach uses explicit knowledge of a SCADA system to generate a system virtual image. The virtual image represents the PLCs and remote terminal units (RTUs) of a monitored system, with all their memory registers, coils, inputs and outputs. The virtual image is updated using a periodic active synchronization procedure and via a feed generated by the intrusion detection system (i.e., known intrusion signatures).

The approach closest to our method was proposed by Cheung et al. [4]. They designed a multi-algorithm intrusion detection appliance for Modbus/TCP with pattern anomaly recognition, Bayesian analysis of TCP headers and stateful protocol monitoring complemented with customized Snort rules [17]. Three model-based techniques characterize expected/acceptable system behavior according to the Modbus/TCP specification: (i) a protocol-level technique that verifies the Modbus/TCP specifications for individual fields and groups of dependent fields in Modbus/TCP messages; (ii) a communication pattern modeling technique based on Snort rules; and (iii) a learning model that describes the expected trends in the availability of servers and services. The appliance was integrated into a control system testbed at Sandia National Laboratories and tested under a multi-step attack scenario. Our approach is also model-based, but it goes much deeper into the Modbus/TCP specifications and captures inter-packet relationships. Thus, it is able to perform all the tests of the first two levels of the system of Cheung and colleagues, but with higher sensitivity and with minimal training.

In subsequent work, Valdes and Cheung [20,21] incorporated adaptive statistical learning methods in two anomaly detection techniques—pattern-based detection for communication patterns among hosts and flow-based detection for traffic patterns in individual flows. In addition, they developed a visualization tool that assists human analysts. More recently, Briesemeister et al. [2] integrated these intrusion detection technologies into the EMERALD event correlation framework [18].