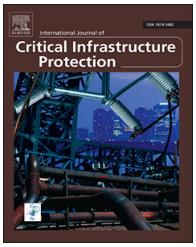
ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

www.elsevier.com/locate/ijcip



Exploiting traffic periodicity in industrial control networks

Rafael Ramos Regis Barbosa[a,c,*], Ramin Sadre[a,c], Aiko Pras[b,c]

[a]3D Hubs, Herengracht 182, 1016BR Amsterdam, The Netherlands
[b]INGI Computer Science Department, Catholic University of Louvain, Place Sainte Barbe 2, B-1348 Louvain-la-Neuve, Belgium
[c]Design and Analysis of Communication Systems, Faculty of Electrical Engineering, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 8 June 2015
Received in revised form
16 January 2016
Accepted 17 February 2016
Available online 2 March 2016

Keywords:
Industrial control networks
Traffic periodicity
Traffic models
Intrusion detection
Device performance evaluation

ABSTRACT

Industrial control systems play a major role in the operation of critical infrastructure assets. Due to the polling mechanisms typically used to retrieve data from field devices, industrial control network traffic exhibits strong periodic patterns. This paper presents a novel approach that uses message repetition and timing information to automatically learn traffic models that capture the periodic patterns. The feasibility of the approach is demonstrated using three traffic traces collected from real-world industrial networks. Two practical applications for the learned models are presented. The first is their use in intrusion detection systems; the learned models represent whitelists of valid commands and the frequencies at which they are sent; thus, the models may be used to detect data injection and denial-of-service attacks. The second application is to generate synthetic traffic traces, which can be used to test intrusion detection systems and evaluate the performance of industrial control devices.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

Industrial control system (ICS) networks are commonly deployed to support the operation of critical infrastructure assets. Historically, industrial control networks have incorporated special-purpose embedded devices that communicate using proprietary protocols. During the past decade, however, the trend has been to adopt the TCP/IP protocol stack and use commercial off-the-shelf devices.

Despite the use of TCP/IP, communications in industrial control networks differ significantly from communications in the Internet or office local-area networks. In an industrial control network, data is continuously retrieved from field devices such as programmable logic controllers (PLCs), so that a real-time view of the supported industrial processes can be established. Typically, data is retrieved via automated, periodic polling processes run by industrial control devices (i.e., master devices). Manual interventions, such as sending commands to field devices, are rare. As a consequence, traffic regularities arise – the set of communicating nodes is stable and network traffic exhibits strong periodic patterns [3]. In fact, well-known characteristics of Internet traffic [12], such as self-similarity and heavy-tailedness, are generally absent [4].

This paper presents a novel approach for learning models of industrial control network traffic; the approach is

[*]Corresponding author at: 3D Hubs, Herengracht 182, 1016BR Amsterdam, The Netherlands.
E-mail address: rafael@3dhubs.com (R.R.R. Barbosa).

http://dx.doi.org/10.1016/j.ijcip.2016.02.004
1874-5482/© 2016 Elsevier B.V. All rights reserved.

automated in a tool named PeriodAnalyser. Traffic models of industrial control networks are important in several application areas. For example, the trend toward TCP/IP-based protocols has made industrial control networks more vulnerable to the same threats that plague traditional information technology networks [11].

Due to the presence of large numbers of zero-day attacks, anomaly-based intrusion detection in industrial control networks is of particular interest [8]. In information technology networks, anomaly-based approaches typically exhibit high false-positive error rates due to the enormous variability of network traffic [22]. In contrast, the regularities present in industrial control network traffic make anomaly detection very promising. The models learned by PeriodAnalyser can be viewed as whitelists of valid commands and the frequencies at which they are sent. These whitelists provide protection against a number of attacks, including data injection and denial-of-service attacks.

Traffic models are also needed to generate synthetic traffic traces. Such traces, combined with attack traffic, can be used as the ground truth for testing and evaluating intrusion detection systems. Alternatively, the synthetic traces could be used to evaluate the performance of industrial control devices.

An extensive search of the literature reveals that the automated approach presented in this paper is the first to *directly* exploit periodicity in industrial control network traffic. Message repetition and timing information are used to detect periodic cycles in the traffic; this addresses the limitations of existing approaches. The approach is validated using traffic traces collected from operational industrial control networks – two water treatment plant networks and one electric-gas utility network.

2. Related work

The problem of periodic traffic pattern identification is different from the classical problem of cycle detection, which is solved by Floyd's algorithm [19]. The classical cycle detection problem involves (efficiently) detecting that a (large) sequence has become periodic under the assumption that the sequence is perfectly periodic. In contrast, the problem considered in this paper is similar to the problem of periodicity detection in temporal data studied by the data mining community [20]. In the data mining context, the problem involves finding repeating patterns in a time series of symbols from a certain alphabet, typically represented as a string (e.g., `abcdabyzabfg`). The approach proposed in [16] allows for imperfect patterns (i.e., patterns that do not reoccur in every cycle) and partial patterns (i.e., only a subset of the patterns are periodic). For example, the substring `ab` is considered to be periodic in the string `abcdabyzabfg`.

The difficulty in applying such methods to network traffic is to define a proper time bin size (i.e., interval between two symbols) in order to create the time series. If the chosen time bin size is too large, unrelated network messages end up in the same time bin and, hence, are represented by only one symbol; this obfuscates the periodic pattern. A very small time bin size could be used to reduce the extent of this problem, but not completely, because TCP can merge multiple application protocol data units (PDUs) into a single segment, causing the protocol data units to be observed at identical times. In addition, this solution is not efficient because it produces many empty time bins and long symbol strings.

Spectral analysis is commonly used to uncover periodicity in network traffic (see, e.g., [1,6,14]). Van Splunder [24] proposes an approach based on the variance of packet inter-arrival times to detect network traffic periodicity. In previous work [5], the authors of this paper investigated the idea of detecting anomalies in the periodic behavior of industrial control network traffic using tools such as discrete Fourier transforms and autocorrelation functions. The main limitation of these methods is the so-called "semantic gap" due to the fact that they operate on information based on the observed network packets (e.g., number of packets or bytes sent per time interval). While it is relatively easy to detect periodic activities using this information, little insight is provided into which packets caused the periodic behavior [16].

The work by Goldenberg and Wool [13] is closely related to the research presented in this paper. Based on the observation that traffic exchanged between a human–machine interface (HMI) and a programmable logic controller consists of requests for the same values being sent periodically, Goldenberg and Wool attempted to model Modbus traffic using a deterministic finite automaton (DFA). The automaton captures the order in which requests and their respective responses are normally exchanged and triggers alarms when an unexpected transition (i.e., unexpected sequence of two messages) is observed. The approach is able to automatically learn an automaton from a training set. However, because only a single automaton is used per connection, a connection carrying requests sent with different periods results in a large model. Moreover, small fluctuations that change the relative order of messages are not captured by the model. Goldenberg and Wool propose a two-level approach that reduces (but does not eliminate) the problem. A second limitation is that the model of Goldenberg and Wool captures the order of messages but not their inter-arrival times. For instance, the model cannot distinguish when a sequence of requests is sent every ten minutes and when the same requests are sent every ten milliseconds.

More recently, Caselli et al. [7] proposed a general approach to detect sequence attacks – sequences of "valid" events (e.g., network messages, log entries and variable values) that have an adverse impact on a system. Their sequence-aware intrusion detection system models normal behavior using discrete-time Markov chains. Anomalies are detected when unknown states are reached or unlikely or unknown transitions occur. Although periodic behavior can be indirectly captured in terms of sequences of states, complex periodic patterns (e.g., multiple periods with possible drift because of timing variations), which are addressed in the present work, would either result in a large model or a compact model that only provides a fuzzy description of the real process.

Download English Version:

https://daneshyari.com/en/article/275039

Download Persian Version:

https://daneshyari.com/article/275039

Daneshyari.com