On the Use of K-NN in Intrusion Detection for Industrial Control Systems

Pedro Silva

Submitted to the Department of Information Technology,

National University of Ireland, Galway,

in partial fulfilment of the

Masters of Science in Software Design and Development.

Supervisor: Dr. Michael Schukat

Head of Department: Dr. Michael Madden

August 2014

Abstract

Intrusion detection systems (IDS) monitor network or system activities for malicious activities or policy violations. As passive and non-intrusive safeguards they are particularly useful in mission-critical networks such as industrial control systems (ICS). Such systems are particularly vulnerable to malicious attacks, which have seen a significant increase in recent years. However, IDS in ICS require different approaches to intrusion detection, which go beyond conventional blacklisting / whitelisting approaches. This thesis examines a new technique, which is based on using the K-Nearest Neighbour scoring algorithm to discover periodic patterns in ICS network traffic. Network traffic whitelisting is used to find anomalies and heuristic models detection to discard false alarms. The algorithm is experimented against datasets generated in a test bed. While results show the approach is feasible with low false positive rates, there are some implementation limitations that can be improved. Possible future work is also discussed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Industrial Control Systems (ICS) are widely deployed in large and (mission-) critical infrastructures including power grids and water / gas distribution systems. Security is definitely an important aspect of these systems as an attack can potentially cause disastrous events.

The security of Industrial Control System has been a major concern over recent years, as failures in such systems may put at risk the health and safety of entire communities [34]. In recent years we have seen a noteworthy number of incidents, including Stuxnet which is considered the most sophisticated cyber-attack targeting an Industrial Control System [25].

Existing Industrial Control Systems deployed around the world have not been designed with security in mind. These systems do not implement any secure communications and have been deployed decades ago. At that time it was not expected to have these systems connected to other networks as local office networks or directly to the Internet. However, as times evolved ICS systems are becoming accessible from the outside world but no security measures are applied which makes them highly vulnerable to attacks [30].

Another problem is the issue of cyber-fragility, e.g. the non-ability of some complex ICS to compensate for small changes in their operational environment. For example, it has been reported that small changes in communication latency times as a result of software / firmware upgrades can bring down an entire control system [31].

In this context, Intrusion Detection Systems (IDS) can play an important role to make ICS more secure. A typical IDS for networks usually does not require any change in the control system. An IDS is a passive component that captures network traffic to be analysed. The result of that analysis will determine if an intrusion has been detected.

Typically the network traffic of an ICS is highly periodic which usually leads to highly predictable and deterministic network traffic patterns. The work presented in this thesis leverages this periodicity to determine anomalies in network traffic using k-NN algorithm [33].

The aim of this thesis is to propose a new algorithm to detect anomalies in ICS networks by analysing its network traffic. A whitelist is generated by capturing what is considered as normal traffic where no anomalies are present. The anomaly detection is performed by using K-NN to discover similar packets crossing the network and with that create a time series of timestamp differences. The anomaly detection is performed by trying to find a subset of the time series in the whitelist.

The work presented in this thesis has been published in [43].

1.2 Industrial Control Systems

Industrial Control Systems (ICS) are interconnected systems used to monitor and control physical equipment in industrial environments. These environments are often critical infrastructures such as water, oil and gas distribution systems, power plants or power grid systems. Commands are sent to control devices based on data received from these. Control devices often control local operations such as opening/closing valves, breakers etc.

One major component of Industrial Networks is Programmable Logic Controllers (PLC). PLCs are specialized electronic components which are computer and solid-state based [12]. They were introduced to cut costs of deployment and maintenance of industrial networks which before were composed mainly by hard-wired relay logic circuits. PLC can easily be programmed and reprogrammed locally or remotely. PLCs are generally made of a processor, input/output modules, communication modules and a power supply. Usually these modules are interchangeable which allows developing highly flexible PLCs according to the requirements of a specific industrial network. The introduction of PLCs also permitted the creation of the interconnected industrial networks which are in use today illustrated in Figure 1.

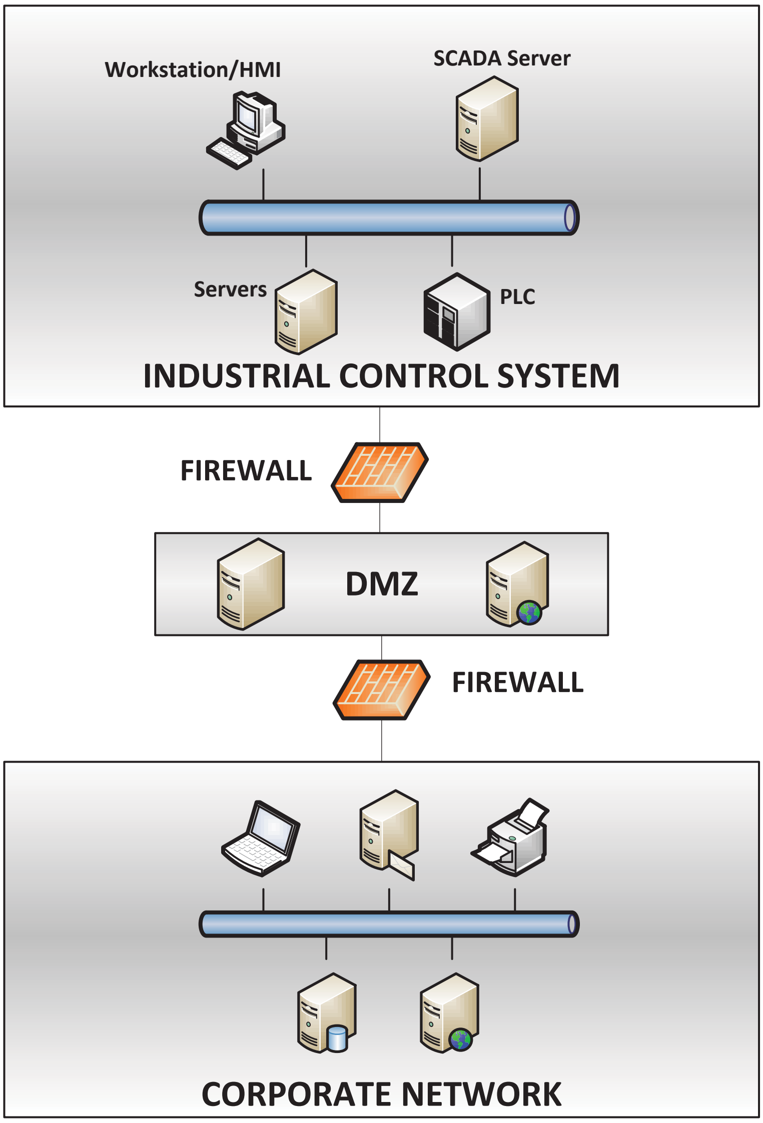

Figure 1: Common architecture of highly interconnected Industrial Control Systems (adapted from Cheung et al. [6])

There are several types of ICS. Supervisory Control and Data Acquisition (SCADA) systems are mostly used when there is a need to gather data from remote devices. The data is presented in a centralised Human Machine Interface (HMI). These systems are usually applied in large geographic areas such as electric utilities where power plants may be located thousands of kilometres away. The main component of SCADA systems is Remote Control Unit (RTU) which is a specialization of a PLC. The RTUs communicate with a Master Terminal Unit (MTU).

SCADA systems typically entail two application layers: HMI clients and server applications which may work as a MTU or an aggregator of several MTUs. The server might communicate with RTU through unreliable communication channels and, therefore, it also maintains a database of last gathered values so they can be displayed promptly to the HMI operator [12]. SCADA systems are event-driven and its main purpose is to track all state changes. A state change will trigger database updates, events, alarms or any other associated processing. These systems often maintain a list of alarms or events which can be managed and filtered by the operator.

On the other hand, Distributed Control Systems (DCS) are process oriented systems [12]. DCS are mostly used to monitor and control local industrial systems. They can be very similar to SCADA regarding to technologies and architecture as it is often to find in DCS a HMI client and a server. However, usually the server only collects specific data from the control devices and always in real-time. These systems, unlike SCADA, are interconnected with locally reliable networks which assures a steady stream of data.

1.3 Evolution

At first ICS were designed mostly as hardware based systems. At that time, ICS mostly based on analogue systems such as hard-wired relays, drum sequencers and cam timers which make the system very expensive to deploy [47]. Also, upgrading the system can be a tedious and error prone process that can be long and involve complex work since these devices have to be replaced individually. To cut costs and improve productivity, these devices started to be substituted by PLCs that are computer based. The earlier versions were programmed by very specialized programmers and most components were proprietary. At the present time, most of the PLCs can be programmed through general-purposes computers using standard technologies such as RS-232 or Ethernet.

As stated in the previous paragraph, first generation of ICS was mostly analogue with no digital connections. Security was ensured by controlling physical access to the system. When later PLC’s were firstly introduced also protocols have been created which were proprietary. However, because at that time digital security was not a big concern, they have not been designed with security in mind and therefore they do not implement, for instance, any encryption or authentication techniques. One of those protocols is Modbus which have been created by Modicon [20]. Over the years, many PLC manufacturers have adopted Modbus which has since become the *de facto* standard for PLC serial communication.

Today, Ethernet is the standard used for networking in corporate enterprises systems and with no surprise it has also become the standard for factory networks and PLCs. Modbus has been adapted to TCP/IP based communication with a new protocol named Modbus/TCP that is no more than a wrapper on top of Modbus. This move allowed PLCs to be controlled using Modbus over Ethernet which raises security concerns. As systems become more interconnected, the unsafe features of Modbus are now accessible from SCADA servers or potentially from enterprise networks. These networks consist of workstations which have access to internet, email or external devices such as USB sticks. Stuxnet which is considered the largest and most complex attacks towards ICS took advantage of infected USB sticks to exploit workstations and afterwards infect others through the network.

1.4 Comparison with Enterprise Networks

Recent and modern ICS networks are today technology wise very similar with Enterprise networks. Ethernet technology has closed a gap between them but at their core different requirements still apply. Industrial networks need to be deterministic and have real-time characteristics as they control physical systems that monitor critical infrastructures.

Different requirements come with different security approaches as well. While in Enterprise networks the main concern is data/systems security, in ICS is ensuring the availability of the system. This being said, usually Enterprise networks follow the CIA security model which stands for Confidentiality, Integrity and Availability. The former is the most critical aspect and the latter the less critical. Confidentiality is the capability of disclosing or allowing access to the people authorised to view it. Usual techniques to ensure confidentiality are the use of cryptography and encryption methods. Integrity is the ability to ensure data is a precise representation of the original source of information. Attackers might use the Man in the Middle attack to intercept data and tamper it before send it to intended recipient. Availability is the ability to ensure information is readily accessible continuously without unplanned interruptions. Attackers could make a network unavailable by performing a Denial of Service (DoS) attack to prevent access to a resource or by simply shutdown a website or defacing it.

On the other hand, ICS networks follow the AIC principle which is in exactly reverse order of CIA [9] [5]. When an ICS network is under attack the main concern regarding security is to maintain the system available since it is controlling critical infrastructure which otherwise could cause unpredictable failures and outcomes. This aspect is a key difference from Enterprise networks. When a resource of an IT network such as a webserver is under attack and confidential information might be disclosed or tampered, IT administrators will prefer to shut down this server and preserve the Confidentiality and Integrity of information instead of ensuring its Availability.

In ICS usually components last for decades. In traditional IT every 4, 5 years components are changed. This also has an impact in updates and patches. Upgrading these systems can be challenging since they are running the same software and hardware for decades. In some cases in order to update a single component of a control system, it is also needed to update the underlying operating system which might not be possible without also upgrading the hardware as well. These dependencies make any small attempt to move the system to a more secure version a tremendous challenge. These difficulties often discourage operators from updating the system or implement security measures that require newer versions. There are also other cases where operators are running proprietary software that system specification or even the source code is no longer known. In these cases is nearly impossible to update the system. There are even situations reported where updates bring down the whole control system [31].

Other key difference between these types of networks is the network traffic. Traffic of ICS networks has a tendency to be predictable and periodic. As most of the ICS processes are automated and periodic, the software control process will most likely send network commands in a periodic form as well. It is well known that connections and number of devices in an ICS network are stable and do not change considerably over time. On the other hand, Enterprise networks do not present a periodic traffic as the devices are mainly controlled by humans which do not tend to perform automated tasks. An IT network usually present more traffic during daylight hours when people are at work and before leaving the office they frequently shut down their workstation or device. This element provokes an unstable number of devices and connections that are completely different from an ICS network which runs 24/7 unchanged.

1.5 Research Hypothesis

In the previous sections, security of Industrial Control Systems has been discussed. These systems have not been designed with security in mind and upgrade them to more secure versions is not always possible. This fact disallows the implementation of intrusive security approaches. Alternatively, intrusion detection systems as passive and non-intrusive safeguards allow increasing security of ICS while not changing the existent infrastructure or cause potential disturbances that can be disastrous [31].

This thesis proposes a novel algorithm using a K-NN classifier to detect anomalies in network traffic of ICS networks. The characteristics of this approach and its suitability for ICS will be experimentally examined.

1.6 Thesis Outline

This thesis is divided in three parts. In Chapter 2 main concepts and techniques of IDS are discussed and followed up by an extended discussion of related work. At the end the novelty of the proposed approach in this thesis is discussed. In Chapter 3 rationales behind of the approach are discussed and a prototype is described implementing the algorithm. In Chapter 4 implementation details of the algorithm and prototype are discussed and in Chapter 5 the feasibility of the approach is experimented against datasets captured from a test bed which results are presented and discussed. In Chapter 6 conclusions are drawn based on the research hypothesis and results achieved. Future work is also discussed.

Chapter 6

Conclusions

Security of Industrial Control System networks is a major concern as recent attacks against such networks has proven. One example is Stuxnet which is considered the most sophisticated cyber-attack targeting an ICS. In this thesis a novel algorithm to detect anomalies in ICS networks have been proposed and experimented.

The algorithm uses whitelists which are generated by capturing network traffic that is considered as normal where no anomalies are present. The anomaly detection is performed by using K-NN classifier to discover similar packets crossing the network and with that create a time series of timestamp differences. The anomaly detection is performed by trying to find a subset of the time series in the whitelist. Model-based detection is also allowed to validate raised alarms. Models are created through a heuristic approach by manually observing network traffic of normal executions. Previous approaches proposed formal models which can be difficult to be accurately defined [6]. Previous whitelist based approaches are not as fine-grained as the one proposed in this thesis, since they do not whitelist patterns at the packet level, except for *PeriodAnalyser* from Barbosa [2].

Experiments were carried out by developing a prototype implementation of the proposed algorithm. The prototype was tested against datasets captured from a test bed. Results demonstrated the feasibility of the approach where real intrusions or anomalies are detected while maintaining a low false positive rate.

However, some limitations were also found. Current prototype might not detect all anomalies if multiple appear in a given sliding window. Also, current implementation of models allows arbitrary combinations of the time ranges which can lead to discard flagged intrusions. Furthermore, sliding window length has impact on the algorithm efficiency, but this can be solved by improving the current models implementation. Another limitation is the lack of support accurate detection when there are multiple connections between the same hosts, i.e., if a given MTU holds multiple TCP connections to an RTU on the same server port.

The work proposed in this thesis has been published in [43] and presented in a poster session at the 4th UL-NUIG Annual Research Day [29].

6.1 Future Work

In the previous sections, conclusions raised some limitations on the implementation that can be overcome.

Whitelists could be pre-processed by calculating the time series for each different packet and improve performance during detection by only calculating time series of packets in the sliding windows.

Alarms are discarded if multiple appear in a same sliding window. Current prototype can be improved by not stopping at the first detected intrusion and analyse the remaining of the time series sequence.

Current models structure and validator implementation can be improved. In this prototype, a model is a simple sequence of time range values. This can be improved to have some rules and allow unbounded time ranges. Also, other rules could be added such as blacklisting and whitelisting of time range sequences, which would prevent arbitrary combinations that cause true alarms being discarded. These rules might also resolve the issue when packets of multiple iterations are in a given sliding window. Rules could identify packets belonging to a new iteration in the same sliding window and processed them separately.

References

[1] Balducelli, C., Bologna, S., Lavalle, L., & Vicoli, G. (2007). Safeguarding information intensive critical infrastructures against novel types of emerging failures. *Reliability Engineering & System Safety*, *92*(9), 1218-1229.

[2] Barbosa, R. R. R. (2014). *Anomaly detection in SCADA systems: a network based approach*. University of Twente.

[3] Barbosa, R. R. R., Sadre, R., & Pras, A. (2012, September). Towards periodicity based anomaly detection in SCADA networks. In *Emerging Technologies & Factory Automation (ETFA), 2012 IEEE 17th Conference on*(pp. 1-4). IEEE.

[4] Bigham, J., Gamez, D., & Lu, N. (2003). Safeguarding SCADA systems with anomaly detection. In *Computer Network Security* (pp. 171-182). Springer Berlin Heidelberg.

[5] Cheminod, M., Durante, L., & Valenzano, A. (2013). Review of security issues in industrial networks. *Industrial Informatics, IEEE Transactions on*, *9*(1), 277-293.

[6] Cheung, S., Dutertre, B., Fong, M., Lindqvist, U., Skinner, K., & Valdes, A. (2007, January). Using model-based intrusion detection for SCADA networks. In *Proceedings of the SCADA security scientific symposium* (pp. 1-12).

[7] de Godoy Stênico, J. W., & Ling, L. L. (2014). 2 Network Traffic Monitoring. *The State of the Art in Intrusion Prevention and Detection*, 23.

[8] Denning, D. E. (1987). An intrusion-detection model. *Software Engineering, IEEE Transactions on*, (2), 222-232

[9] ENISA (June 2012) Report, Annex II: “Security aspects of the smart grid”.

[10] Etsion, Y., Tsafrir, D., & Feitelson, D. G. (2003, June). Effects of clock resolution on the scheduling of interactive and soft real-time processes. In *ACM SIGMETRICS Performance Evaluation Review* (Vol. 31, No. 1, pp. 172-183). ACM.

[11] Fovino, I. N., Carcano, A., De Lacheze Murel, T., Trombetta, A., & Masera, M. (2010, April). Modbus/DNP3 state-based intrusion detection system. In *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on* (pp. 729-736). IEEE.

[12] Galloway, B., & Hancke, G. P. (2013). Introduction to industrial control networks. *Communications Surveys & Tutorials, IEEE, 15*(2), 860-880.

[13] Garitano, I., Siaterlis, C., Genge, B., Uribeetxeberria, R., & Zurutuza, U. (2012, September). A method to construct network traffic models for process control systems. In *Emerging Technologies & Factory Automation (ETFA), 2012 IEEE 17th Conference on* (pp. 1-8). IEEE.

[14] Heberlein, L. T., Dias, G. V., Levitt, K. N., Mukherjee, B., Wood, J., and Wolber, D. A Network Security Model, In *Proceedings of the Symposium on Research in Security and Privacy*, Oakland, CA, pp. 296–304, 1990

[15] Hofmeyr, S. A., Forrest, S., & Somayaji, A. (1998). Intrusion detection using sequences of system calls. *Journal of computer security, 6*(3), 151-180.

[16] http://jnetpcap.com (online, accessed on July, 2014)

[17] http://man7.org/linux/man-pages/man7/time.7.html (online, accessed on July, 2014)

[18] http://www.advantech.com/products/Data_Acquisition_Modules/ADAM-6060/ mod_c3b0fabb-dce7-4a64-bcb7-7e0cbfda0364.aspx (online, accessed on 2014, July)

[19] http://www.bro.org (online, accessed on 2014, July)

[20] http://www.modbus.com/ (online, accessed on 2014, July)

[21] http://www.modbus.org/docs/Modbus_Ap plication_Protocol_V1_1b3.pdf (online, accesse d on 2014, March)

[22] http://www.modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf (on line, accessed on 2014, March)

[23] http://www.snort.org (online, accessed on 2014, July)

[24] http://www.suricata-ids.org (online, accessed on 2014, July)

[25] http://www.symantec.com/connect/blogs/stuxnet-using-three-additional-zero-day-vulnerabilities (online, accessed on 2014, March)

[26] http://www.tcpdump.org (online, accessed on July, 2014)

[27] http://www.winpcap.org/ (online, accessed on July, 2014)

[28] https://www.wireshark.org/ (online, accessed on July, 2014)

[29] http://ulnuigresearchday.wordpress.com/ (online, access on July, 2014)

[30] Knapp, E. (2011), Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems, *Syngress*, ISBN 1597496456

[31] Langner, R. (2011). Robust Control System Networks: How to Achieve Reliable Control After Stuxnet. *Momentum Press*, ISBN 1606503006.

[32] Liao, Y., & Vemuri, V. R. (2002). Use of k-nearest neighbor classifier for intrusion detection. *Computers & Security*, *21*(5), 439-448.

[33] Lopez De Mantaras, R., McSherry, D., Bridge, D., Leake, D., Smyth, B., Craw, S., ... & Watson, I. (2005). Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review*, *20*(03), 215-240.

[34] Macaulay, T, Singer, B.L. (2012), Cybersecurity for Industrial Control Systems, *Auerbach Publications*, ISBN 1439801967

[35] Oman, P., & Phillips, M. (2007). Intrusion detection and event monitoring in SCADA networks. In *Critical Infrastructure Protection* (pp. 161-173). Springer US.

[36] Paxson, V. (1999). Bro: a system for detecting network intruders in real-time.*Computer networks*, *31*(23), 2435-2463.

[37] Pleijsier, E. (2013, January). Towards anomaly detection in SCADA networks using connection patters. In *18th Twente Student Conference on IT* (pp. 1-6).

[38] Postel, J. (1981). Transmission control protocol.

[39] Pranggono, B., McLaughlin, K., Yang, Y., & Sezer, S. (2014). 5 Intrusion Detection Systems. *The State of the Art in Intrusion Prevention and Detection*, 115.

[40] Ranum, M. J. Coverage in Intrusion Detection Systems. NFRSecurity, Inc. Technical Publications, pp. 1–9, June 2001

[41] Recio-García, J. A., González-Calero, P. A., & Díaz-Agudo, B. (2014). jcolibri2: A framework for building Case-based reasoning systems. *Science of Computer Programming*, *79*, 126-145.

[42] Roesch, M. (1999, November). Snort: Lightweight Intrusion Detection for Networks. In *LISA* (Vol. 99, pp. 229-238).

[43] Silva, P., Schukat, M. (2014). On The Use of K-NN in Intrusion Detection for Industrial Control Systems, In *Proceedings of The IT&T 13th International Conference on Information Technology and Telecommunication, Dublin, Ireland,* (pp 103-106)

[44] Smaha, S. E. (1988, December). Haystack: An intrusion detection system. In *Aerospace Computer Security Applications Conference, 1988., Fourth* (pp. 37-44). IEEE.

[45] Snapp, S. R., Brentano, J., Dias, G. V., Goan, T. L., Heberlein, L. T., Ho, C. L., ... & Mansur, D. (1991, October). DIDS (distributed intrusion detection system)-motivation, architecture, and an early prototype. In *Proceedings of the 14th national computer security conference* (Vol. 1, pp. 167-176).

[46] Xiao, K., Chen, N., Ren, S., Shen, L., Sun, X., Kwiat, K., & Macalik, M. (2007, May). A workflow-based non-intrusive approach for enhancing the survivability of critical infrastructures in cyber environment. In *Proceedings of the Third International Workshop on Software Engineering for Secure Systems* (p. 4). IEEE Computer Society.

[47] Zhou, M., & Twiss, E. (1998). Design of industrial automated systems via relay ladder logic programming and Petri nets. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *28*(1), 137-150.