OAK RIDGE NATIONAL LABORATORY

MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

White Paper

12 November 2009

(Revised April 27, 2010 by F.T. Sheldon)

Title: Data Diodes in Support of a Power Grid Trustworthy Cyber Infrastructure

Topic: *Use of data diodes to maximize the trustworthiness of the power grid*

Programmatic POC:

J. Trien
Department of Energy, Oak Ridge National Laboratory (ORNL)
Computational Sciences and Engineering Division
Cyberspace Sciences and Information Intelligence Research Group
One Bethel Valley Rd, M/S 6418 Oak Ridge TN 37830
Phone: 865-241-9104 (865-576-5943 fax)
E-mail: TrienJP@ornl.gov

Technical POC:

F. T. Sheldon, L. MacIntyre, Hamed Okhravi[†] and John C. Munson^{††}
Department of Energy, Oak Ridge National Laboratory (ORNL)
Computational Sciences and Engineering Division
Cyberspace Science and Information Intelligence Research Group
Phone: 865-576-1339 (865-576-5943 fax)
[†]Currently @ The University of Illinois, Urbana Champaign ^{††}Currently at Computer Measurement Laboratory, Inc.

BUSINESS SENSITIVE INFORMATION *This document contains business sensitive and potentially patentable subject matter protectable under state and Federal law. No further dissemination is permitted without express permission of UT-Battelle, LLC. Disclosure of this document to Federal employees is made subject to 35 USC 205 and such employees are subject to 18 USC 1905 against further disclosure*

This manuscript has been authored by UT-Battelle, a contractor of the U.S. Government (USG) under Department of Energy (DOE) Contract DE-AC05-00OR22725. The USG retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for USG purposes.

Executive Summary

Industrial Control Systems (ICS) are an integral part of the industrial infrastructure providing for the national good. While sharing basic constructs with Information Technology (IT) business systems, ICSs are technically, administratively, and functionally more complex and unique than business IT systems. Critical infrastructure protection focuses on protecting and maintaining a safe and reliable supply of electric power, oil, water, gasoline, chemicals, food, etc. Cyber vulnerabilities are important if they can affect the safe, functional performance of these systems and processes. The majority of ICS exhibit vulnerable devices with unsecured physical access and/or subject to insider attack. In this whitepaper, we advocate trusted process control networks as a way to address the serious cyber security flaws which combines both white/black listing into a design philosophy that addresses information warfare scenarios, software process monitoring and an attack recognition and management architecture.

Background Industrial Control Networks

Current ICS cyber security is in the formative stage [Weiss'09]. The umbrella term “ICS” include:

- automated control systems (ACS)
- distributed control systems (DCS),
- programmable logic controllers (PLC),
- supervisory control and data acquisition (SCADA) systems,
- intelligent electronically operated field devices, such as valves, controllers,
- instrumentation
- intelligent meters and other aspects of the Smart Grid
- networked-computing systems

Figure 1. General ICS Architecture.

An ICS is actually a system of systems. A crude distinction between mainstream IT and control systems is that IT uses “physics to manipulate data” while an ICS uses “data to manipulate physics.” Compromised ICS systems can, and have, led to extensive cascading power outages, dangerous toxic chemical releases, and explosions. It is therefore important to implement an ICS

with security controls that allow for reliable, safe, and flexible performance. Data Diodes can be useful for a trusted Power Grid in some cases, but they have to be used in specific configurations and in conjunction with a more robust security infrastructure. For example, in a trusted process control network (TPCN) [Okhravi'09], data diodes can be used to enhance security, but they are by no means even a nearly complete solution.

Figure 1 shows a general but common network architecture: In the first layer the physical infrastructure is instrumented with sensors and actuators. These field devices are connected via a field area network to programmable logic controllers (PLCs) or remote terminal units (RTUs), which in turn implement local control actions (regulatory control). A control network carries real-time data between process controllers and operator workstations. The workstations are used in area supervisory control, planning the physical infrastructure set-points. The higher level is the site manufacturing operations, which is in charge of production control, optimizing the process,

Figure 2. PCN Architecture with paired firewall and data diodes.

and keeping a process history. In the following, we discuss recommendations for maximizing the trustworthiness of ICSs through an innovative the process control network architecture [Cárdenas'08].

ICS security challenges and recommendations

Consider Figure 2, which shows a paired firewall network, often used to segregate process control networks (PCNs) and enterprise networks (ENs). A firewall can be configured to block unnecessary services, protocols and ports, thereby providing a higher degree of segregation between a PCN and EN. A router may be positioned in front of the firewall to perform simple

packet filtering, leaving the firewall to perform more sophisticated tasks such as stateful filtering and acting as a proxy. Using a single firewall between a PCN and EN has a serious drawback because the firewall must allow the data historian to have a wide range of access to the PCN. Essentially, each service needs a “hole” in the firewall to operate correctly. Configuring too many holes in the firewall reduces PCN-EN segregation and opens the PCN to a slew of attacks. This problem is typically addressed by creating a “demilitarized zone” (DMZ). An architecture deploying a DMZ has three zones: an outside zone containing the EN, an inside zone containing the PCN, and a DMZ containing the data historian. Firewall rules are crafted to make the DMZ historian the sole point of contact between the EN and PCN. The historian can access PCN services that provide it data; in turn, the EN is allowed access to the historian.

Firewall rules block access to the PCN by all devices. Most attacks originating in (or passing through) the EN and targeting the historian will not affect the control systems; at worst, they would corrupt the historian’s data (a redundant copy of this data is stored elsewhere). A PCN architecture deploying paired firewalls separated by a DMZ is shown in Fig. 2. It simplifies the firewall rules and achieves a clear separation of responsibility as the PCN-side firewall can be managed by the control group and the EN-side firewall by the IT group. This architecture is highly recommended for ICSs, and best practices have been identified for configuring the firewalls. There are also mechanisms to enhance host security in a control system. An important example of such mechanisms is process-based security (PBS), which is implemented for some control devices. It shifts the access control paradigm from a user-based model to a process-based one. In the user-based model, access control rules are tied to user identities. A rogue process, however, can escalate its privilege and damage the system. In the process-based security model, on the other hand, fixed access vectors are strictly tied to process profiles, which cannot be modified in the field. PBS reduces the risk of privilege escalation as a result of an attack.

Firewall configuration errors can lead to security vulnerabilities. One problem is that firewalls often have large rule sets, which are difficult to verify. According to a study by Wool (see [Okhravi'09]), firewall rule sets may have as many as 2600 rules with 5800 objects. A significant correlation exists between rule set complexity and the number of configuration errors. Because firewalls are usually the main line of defense, configuration errors enable attackers to exploit holes and target the otherwise defenseless devices inside the network. Eighty percent of rule sets

Figure 3. Trusted process control network (TPCN) architecture.

exist behind a firewall. Firewalls can also be thwarted by tunneling attack traffic using legitimate means (e.g., via a corporate VPN) or by using encryption (firewalls do not inspect encrypted packets). A widely-reported firewall breach occurred in January 2003, when the MS SQL Server 2000 worm infected systems at the Davis-Besse nuclear power plant in Oak Harbor, Ohio (see [Okhravi'09]). An investigation revealed that a contractor established an unprotected connection to the corporate network, which bypassed the power plant firewall and provided a path for the worm.

Data diodes, in some cases could be used to simply isolate communication pathways whenever possible (assuming that the specific communications protocols that can accommodate such one way data flow limitations are in place). Judicious use of such, could mitigate the above mentioned firewall risk, but not without careful consideration of the impact to safety and functionality.

Caveat: Vulnerable devices, unsecured physical access and insider attack

Vulnerable devices are typically secured by patching their services, updating software or installing the latest version of the devices. However, manual patch/update/version management are difficult and costly tasks, especially when careless users introduce vulnerable (wireless) devices into an industrial control network that establish new entry points for attackers. The mobility of wireless devices makes it difficult for the administrator to manually manage patches by visiting the devices frequently or dedicating a specific time slot for patching the systems. Hence, patch/update/version management should be done automatically and continuously. Unsecured physical access also exposes ICSs to serious security threats. Open wireless access points and Ethernet ports on office walls enable attackers to enter ICS networks and target critical assets. Nothing in the traditional ICS architecture prevents suspect devices from connecting to the network; thus, serious threats are posed by devices whose hardware, operating systems, executables and/or configurations have been tampered with by attackers. Many ICS vulnerabilities admit malware such as worms, viruses, Trojan horses and rootkits. ICS security trends reveal that external malware attacks are becoming increasingly common. Finally, rogue users (insiders) are an ever-present threat to ICSs.

Trusted PCNs

A trusted network (TN) architecture uses information about the hardware and software states of devices in admission and access control decisions. When a device first joins the network, its

allow “any” service on inbound traffic and insecure access to firewalls. The analysis of real configuration data shows that corporate firewalls are often enforcing rule sets that violate well-established security guidelines. Numerous studies have empirically demonstrated that firewall configuration errors pose a real threat to ICS security. Even properly configured firewalls can be bypassed. This occurs, for example, when a vendor creates a direct (e.g., dial-up) connection to a device for maintenance, or when unsecured wireless access points

hardware and software are checked; based on these checks, the appropriate access control rules are applied dynamically to the user, device and traffic. The same principle can be applied to process control architectures. A TN architecture uses existing standards, protocols, and hardware devices to extend the concept of “trust” to the network architecture. TNs provide important security services such as user authentication, comprehensive network device admission control, end-device health check, policy-based access control and traffic filtering, automated re- mediation of non-compliant devices, and auditing¹. A TN has the following components:

- Client device: Every client device must be evaluated prior to admission to a TN.
- Network Access Device (NAD): All connectivity to a TN is implemented via a network access device (NAD), which enforces policy. NAD functionality may exist in devices such as switches, routers, VPN concentrators and wireless access points.
- Authentication, Authorization, and Access Control (AAA) Server: maintains the policy and provides rules to NADs based on the results of authentication and posture validation.
- Posture Validation Servers (PVSs): evaluate the compliance of a client before it can join a TN. A PVS is typically a specialization for one client attribute (e.g., operating system version and patch or virus signature release).
- Posture Remediation Servers: provide remediation options to a client device in the case of non-compliance. For example, a server may maintain the latest virus signatures and require a non-compliant client device to load the signatures before joining a TN.
- Directory Server: authenticates client devices based on their identities or roles.
- Other Servers: These include trusted versions of Audit, DNS, DHCP and VPN servers.

A trusted process control network (TPCN) architecture is presented in Fig. 3. A client device intending to join the network communicates its request to the NAD. The NAD establishes the client device’s identity using EAP over the 802.1x protocol and sends the results to the AAA server using the RADIUS protocol. The AAA server returns a list of posture validation requirements and the addresses of the appropriate PVSs. The client then validates its posture with each of the PVSs. If the client is in compliance, the results are sent to the AAA server using the HCAP protocol. On the other hand, if the client lacks one or more requirements, the appropriate posture remediation servers suggest remediation actions to the client.

The directory server determines the client’s group or role. Given all the results from the PVSs and the directory server, the AAA server determines the set of rules that apply to the client’s access and traffic and sends them to the NAD for enforcement. From this point on, the client is permitted to communicate via the NAD and all its activities are monitored for policy compliance. The policy held by the AAA server is in the form of an authentication requirement and a list of posture validation requirements. For example, token-based authentication may be required and postures must be validated with the anti-virus server, patch management server and driver validation server. When a client device joins the network, an NAD communicates with an AAA server on behalf of the device. The AAA server authenticates the device and provides rules based on the device’s security postures to the NAD. From this point on, the NAD enforces the policy on all ingress and egress traffic to/from the device. For example, an RTU with valid firmware is allowed to communicate with the historian; all other traffic is blocked. [Okhravi’09] provides two examples to further clarify the workings of a TPCN.

Recommendation

The TPCN approach can be described as a black list approach whose goal is to keep out the bad guys using all the trusted appliances (e.g., authentication, compliance enforcement, etc.). If any

¹ The Trusted Computing Group (TCG) has promulgated industry standards for TNs. Several commercial TN technologies have been developed, including Cisco TrustSec, Cisco CleanAccess (formerly known as Cisco Network Admission Control (NAC)), and Microsoft Network Access Protection (NAP). Cisco NAC is interoperable with Microsoft NAP; details about their interoperation can be found in. TN component vendors use a variety of names to describe their products. We use generic terms with a bias towards those adopted by Cisco CleanAccess.

of those guarding principles fails, the event/perpetrator is black-listed (prevented access/entry). On the other hand, in the white list approach, only the known good behavior will be allowed access/entry to the system. Data diodes can facilitate both black list and white list approaches. By design, data diodes are placed at control points where we know that one-way data flow prevents perpetrators from accessing the system (black-list) while allowing the information to flow out (white-list). In the case of Figure 2, the data historian and the PCN can be instrumented with data diodes so that the supervisory data can be sent out, but no data (potentially malicious) can enter the PCN. In cases where two-way data communication is necessary, we can monitor and permit only known good traffic.

Power Grid, Open Loop and Uncontrolled Access:

The lack of basic principles of engineering design and discipline has led to some very serious problems in the computer security arena. At the heart of modern power systems are Supervisory Control and Data Acquisition (SCADA) software systems that provide real time control. It is interesting to note that every aspect of a hardware system that is under control of the SCADA architecture is fully instrumented and the operation of the monitored system is totally under management of the control system. However, the software that controls the entire power grid is essentially open loop and uncontrolled. It would be unthinkable to build a complex hardware system without embedding monitors for process control throughout the whole architecture. Ironically, the backbone of modern control systems, SCADA for example, is almost entirely dependent on a software system that is running out of control. The lack of monitoring of complex software systems opens the door for problems in both security and reliability. From a security perspective, the software may be compromised and made to perform the bidding of a hostile agent. From a reliability perspective, the software may begin to execute software components that are not thoroughly tested. The end result of either of these two scenarios is the same. The software system will be compromised. What is worse, because the processes that constitute the software system are not being monitored, the integrity of the hardware component is most certainly compromised as well.

Design Philosophy: The foundation of our current work is the design and development of software control systems. These systems are built to provide engineering integrity for executing software systems. It is our philosophy that software systems must be subject to the same rigorous scrutiny during their operation as would any mission critical or safety critical hardware system.

Information Warfare Scenario: A most pressing problem looms in the immediate future. In an information warfare scenario, the enemy will deploy an attack that has never been seen before. This attack will completely debilitate both defensive and offensive software capabilities. Unfortunately, the state space of possible attacks is at once large and unknown for a complex software system. If a system is monitored for nominal operation, then it is possible to recognize instantly any abnormal activity. Once the abnormality is recognized, it is then possible to restore the system to a known certified state. Our project's major innovation is a comprehensive approach for tolerating security violations in mission/safety critical software systems by closing the software feedback control loop. To the best of our knowledge, this project is the first project that uniformly applies the real-time measurement paradigm to the reliability and security of mission critical software.

Software Process Monitoring: In our control theoretic approach to software process monitoring, we are interested in measuring and monitoring the activity of the software system. We have designed a security coprocessor that monitors the running software system and measure its activity while it is running. The central issue here is that once we understand the notion that a program exhibits activity that can be measured, we can begin to assert control on the program execution. We can certify certain activities as nominal and reject activities that are outside of this

range. We can do this in real time because we are monitoring the activity of the system in real time. Process control should be a major component of computer security. It is also a totally neglected component of computer security. It has long been accepted that data control in the form of encryption is a necessity to preserve the integrity of information flowing from one agency to another. Controlling access to system resources has also shown great value for imposing a security regime.

Attack Recognition and Management Architecture

In response to the need for a software process control system, we have developed the CML Attack Recognition and Management Architecture, CARMA, under the aegis of a contract from the Office of the Secretary of Defense. The primary objective of the CARMA project is to create the infrastructure for an autonomic kernel protection system and then productize this infrastructure. There are essentially two types of attacks that can be made on an operating software system: those that have been deployed against existing systems and those that have never been seen before. Current approaches to the kernel defense rely on the knowledge of the attack signature. That is, there are recognizable symptoms of the effect of the attack on the kernel. Clearly, this approach renders most systems vulnerable to the very type of attack that is most likely in an information warfare scenario.

CARMA is a dual of the traditional security paradigm. We model the normal activity of a system and detect departures from that normal model. This methodology permits each implementation of a software system to be calibrated for operation in a particular environment. Once a departure from the nominal model of software activity has been detected, the offending activity may be arrested before it has compromised the system. The problem that we have solved is the mathematical representation of normal system activity. The state space of normal activity for a large number of mission critical systems is very small and well defined. Alternatively, the state space of abnormal activity is unbounded. This technology enhances and hardens the kernel software and other executing software systems from malicious alteration and assaults from the communication network. Monitoring functionality is implemented out of band in a hardware coprocessor and it is simply not possible for an external program activity to access the monitoring system because the monitoring function is an external activity, the knowledge of its activity and design does not provide access to an attacker.

The most threatening attacks to the grid and its network infrastructure are those that we haven't seen yet. This, we believe, is the most pressing problem confronting computing security in our nation. These attacks will certainly come as a surprise. Our technology was not designed to identify or classify particular attacks. It was designed to recognize normal system activity. This makes it possible to identify abnormal activity whether it is novel or well known. Therefore, in this approach, our security thesis is that, software vulnerabilities are not the fundamental issue. It is very expensive and impractical to try to build a vulnerability free system. No civil engineer would ever expect to build a perfect bridge. An engineer would, instead, expect to build a bridge within practical tolerances and design in compensation for environmental variation. The problem in software, then, is not the vulnerabilities. The problem is that the vulnerabilities get expressed because no one is watching and controlling. Trust nothing. Monitor and control everything.

References

- [Cárdenas'08] Cárdenas, A. A., Amin, S., and Sastry, S. Research challenges for the security of control systems. In /Proceedings of the 3rd Conference on Hot Topics in Security/ (San Jose, CA), 2008 (http://www.usenix.org/events/hotsec08/tech/full_papers/cardenas/cardenas_html/)
- [Okhravi'09] Hamed Okhravi and David Nicol, Application of Trusted Network Technology to Industrial Control Networks, Elsevier International Journal of Critical Infrastructure Protection (IJCIP), Elsevier, Vol. 2, No. 3, pp. 84-94, 2009

[Weiss'09] Control Systems Cyber Security—The Current Status of Cyber Security of Critical Infrastructures <http://cimic.rutgers.edu/WS-CFP/Joe-Weiss.pdf>