The Case for Prevention-based, Host-resident Defenses in the Modern PCS Network

[Extended Abstract]

Charles Payne, Jr.
Adventium Labs, LLC
111 3rd Ave. So., Suite 100
Minneapolis, MN 55401
charles.payne@adventiumlabs.org

Richard C. O'Brien
Adventium Labs, LLC
111 3rd Ave. So., Suite 100
Minneapolis, MN 55401
richard.obrien@adventiumlabs.org

J. Thomas Haigh
Adventium Labs, LLC
111 3rd Ave. So., Suite 100
Minneapolis, MN 55401
tom.haigh@adventiumlabs.org

ABSTRACT

The process control system (PCS) owner can no longer rely on a physical air gap and custom hardware to protect her network from attack. Demand for greater visibility into PCS operations, coupled with greater use of commodity hardware, now exposes the PCS network to the same threats facing other networks. To address these threats, we argue for the deployment of prevention-based, host-resident, network layer devices, coupled with scalable, service-based management, that will not only protect PCS communications but will also support higher level reasoning about PCS trustworthiness. We explain why the modern PCS network is particularly well-suited for this approach, and we highlight where our own research supports this claim.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-Based Systems**]: Process control systems; C.2.0 [**Computer-Communications Networks**]: General—*security and protection*

General Terms

Security

Keywords

Distributed firewalls, security policy management, process control systems

CSIIRW '09, April 13-15, Oak Ridge, Tennessee, USA

1. INTRODUCTION

Adventium Labs has deep experience deploying prevention-based, host-resident, network layer devices that not only provide basic network access control and authentication but also support higher level reasoning about system trustworthiness, including control and data path integrity, insider threat protection, and data provenance. This experience has led us to develop a scalable, service-based, policy management approach called *conversations* for managing these devices more effectively. In this paper, we argue that a process control system (PCS) is well-suited for these devices and the conversations management approach. This paper highlights our research and recommends how the results could be applied for the protection of the modern PCS network.

2. PROBLEM

The PCS owner has traditionally relied on physical separation, or an *air gap*, between her network and other networks in order to block intruders. However, modern business needs require greater inspection and control of PCS operations, and that air gap is narrowing rapidly. Unfortunately, sufficiently strong protections have not emerged to restore these assurances. Defenses located at the boundary of the PCS network are unable to mediate accesses between the PCS hosts that they protect and thus do not block the intruder who gains a foothold on one of those hosts. Furthermore, modern PCS components are increasingly deployed on commodity computers that lack strong host-based security, and these computers may ship with a myriad of services and associated vulnerabilities that PCS networks have not had to contend with in the past. Thus, in addition to the loss of the air gap that once protected the PCS network from other networks, the PCS owner must now contend with the large class of threats that target the commodity computers on which the PCS network is hosted.

3. HYPOTHESES

We assert that tamper-resistant, non-bypassable, prevention-based defenses that protect, yet are isolated from, each PCS host can play a critical role in protecting the modern PCS network. *Per-host* deployment provides the appropriate granularity for protecting PCS operations behind the boundary firewall. *Strong prevention* renders obsolete those attacks that would violate the PCS-specific policy. *Non-bypassable* defenses contain the intruder who manages to gain a foothold and then wishes to propagate his attack to other PCS hosts,

and the quality of *tamper resistance* is a prerequisite for any non-bypassable defense.

In addition, these devices should be *centrally managed* to ensure consistent policies across the PCS network. We assert that such management can be both more intuitive and scalable by adopting a service-based management approach. That is, instead of writing a policy to authorize the protected host to engage in all communications necessary for its operation, we instead write policy that authorizes groups of hosts to engage in specific network services with their respective service providers. The composition of these service-based policies, called *conversations*, defines the protection strategy for the entire network. The policy for a particular device is derived from the conversations that name its protected host.

Before we present the research results that support these hypotheses, we explore the qualities that make the modern PCS network especially well-suited for these protections.

4. EXPLOITABLE QUALITIES

The modern PCS network enjoys the following qualities.

Internet Protocol (IP) based. Modern PCS networks are extending the reach of IP-based networks to remote field sites in order to leverage the richer management protocols that run over those networks. The protection devices we describe here enjoy the ready availability of components built for these types of networks.

Static configuration. The conventional wisdom that systems change too rapidly for prevention-based protections to keep pace (thereby dismissing prevention in favor of detection and response) does not hold for a PCS. The PCS may undergo an onerous certification process that demands a highly static, well understood system configuration, and that configuration can translate easily into a tight, enforceable, prevention-based security policy.

Protected paths. A PCS demands rigorously protected control and data communication paths. The PCS owner must verify that only authorized paths exist and that those paths are immune to security threats such as external attack, malicious software and malicious users. Since protecting a communication path first requires protecting that path's endpoints (the PCS hosts themselves), host-based defenses offer the appropriate granularity of protection.

Net-centricity. Controlling access to each PCS host is often sufficient to control access to a specific PCS operation, because (1) a PCS is largely net-centric and (2) each PCS host typically performs a dedicated operation. In fact, it is common to refer to a PCS host in terms of the operation it performs (HMI, Historian, etc.). This relationship between a PCS operation and its computer host means that the protection of that operation can be performed by the network — on behalf of the host — if the host itself lacks robust protections. This strategy simplifies the assurance argument for the PCS host by requiring less trust for that host, and it enables the required protections to be extended to legacy hosts that could not otherwise provide them.

5. CHALLENGES

We acknowledge two challenges to our approach. First, adopting these protections means introducing new hardware and software into an existing PCS environment, which may impose certification costs. In our research, the form factor for these devices has ranged from an embedded, network interface card (NIC) [9] to a bump-in-the-wire (BiW) appliance [3]. The NIC imposed no additional hardware footprint, but it required compatibility with the host's operating system (OS) and hardware. Installing a new NIC on existing PCS hosts could violate the vendor's configuration. The bump-in-the-wire solution avoids that concern, but extra effort is needed to keep the device reasonably transparent to PCS operations.

Second, we must ensure that the device itself does not impact PCS operations adversely, for example by introducing communication delays or by inadvertently trapping network traffic that should be passed to the host, such as broadcast traffic. We assume the deployment of these devices only on the IP-based networks, where the availability of fast processors, fast network controllers, and robust transport protocols make communication delay less of a concern. Concerns about blocking broadcast traffic can be addressed by putting the device in a bridge mode so that such traffic is passed between the host and the network without alteration. However, using bridge mode affects certain protection strategies, such as network-level encryption, so it should be used with caution.

The remainder of this paper highlights both our experience building these devices and our experience managing them.

6. BENEFITS OF THE PREVENTION-BASED ENFORCEMENT DEVICE

For the past decade, we have investigated the use of host-resident, distributed firewalls to protect critical networks. The first *distributed firewall* — a term coined by Bellovin [1] — was prototyped by researchers at Columbia University using OpenBSD [4] and later played a key role in their Strongman security architecture [5]. Around the same time, our previous employer, Secure Computing Corporation, began investigating the deployment of the distributed firewall on a NIC. This led to a collaboration with 3Com Corporation to produce the commercially available 3Com Embedded Firewall (EFW). EFW, along with its research cousin the Autonomic Distributed Firewall [7], fueled research into practical applications [6, 9], novel encryption strategies [2, 8], and the firewall's role in system survivability [10]. In 2006, Adventium deployed the EFW as part of a demonstration[1] of intelligent alerts in the modern PCS network.

We have recently developed a Linux-based, BiW appliance, called the Detection Response Embedded Device (DRED)[2], that is outfitted with multiple defenses, from the link layer to the application layer of the network stack, providing fine-grained, defense-in-depth security [3]. The DRED is deployed on a minimal hardware platform that is co-located

[1]Department of Homeland Security, `http://www.cyber.st.dhs.gov/logic.html`

[2]So named because it could be implemented on a single board computer that would be installed in the protected host

with its protected host and connected to it via a cross-over Ethernet cable. Thanks to the conversations management tool, we have policy-based control over each defense on the DRED.

The remainder of this section highlights the primary features of the DRED and their potential for protecting the modern PCS network.

6.1 Network Access Control

A foundational feature for the DRED is its ability to enforce network access control through packet filtering. The Linux packet filter `iptables` enables rich control over a packet's flow at both the link layer and the network layer. We used `iptables` to divert packets to an application layer proxy for deeper analysis, to log and drop packets that violated the conversations policy and to throttle packets that were coming too fast (potential denial of service attacks). We also added `arptables` to filter Address Resolution Protocol (ARP) requests from the protected host. Essentially, we permitted the host to make ARP requests only for the remote hosts for which it was authorized to communicate.

For the PCS network, basic network access control would limit attack propagation, but the packet filter's logging and alert features would also be useful for situational awareness. Policy violations occurring at multiple points in the network can paint a clear picture of attacker intent, which leads to more intelligent response.

6.2 Host Authentication and Communication Path Integrity

A second critical feature for the DRED is its ability to authenticate the source of filtered packets. Because of the cross-over cable, the DRED can authenticate the source of host-transmitted packets, but packets originating from the network could have their source IP address spoofed by a third party. The DRED relies on IP security (IPsec), specifically tunnel mode encryption, to authenticate all remote communications. We implemented a shared key scheme [8] to simplify key management and to ensure that only authorized, DRED-protected hosts could communicate. Because each host is protected by its own DRED, no rogue host on the network can observe or alter protected communications.

For the PCS network, packet filtering and packet encryption are necessary to guarantee the integrity of control and data paths. Such integrity is demanded, for example, when managing remote field units. The controller needs to assert positive control of the *correct* field unit and that field unit should be managed only by an authorized controller. These can be guaranteed by encrypting the control traffic and by using a packet filter at each endpoint to limit the direction of the connection. Since confidentiality-preserving encryption may not be desired due to concerns of blinding network intrusion detection systems, simply using IPsec to authenticate the sender of each packet may be sufficient.

6.3 Insider Threat Prevention

While the DRED can inspect deeply its Ethernet frames, it cannot identify with confidence *which* user or process on the protected host generated those frames. Instead, all traffic appears to come from the host itself, and thus the granularity of access control is no better than of the host. However, to address the insider threat problem, it is critical to associate a user with this traffic. So a third useful feature of the DRED was the installation of a smart card reader to prevent the user from gaining network access without first inserting a valid card. User authentication triggered an automatic, user-specific configuration of the DRED's defenses. Card withdrawal immediately revoked those protections and dramatically limited what unauthenticated users could do on the network.

In the PCS network, hosts routinely communicate without user involvement. However, certain network operations may require a user. In these cases, it may not be enough that the connection was initiated by an authorized controller host but that it was initiated by an *authorized user* on that host. In these cases, policy could require user authentication before the user can access the network to perform the operation.

6.4 Control and Data Provenance

The last DRED feature that we will highlight was again motivated by the insider threat problem, but it has relevance to PCS networks too. In order to guarantee that authorized users invoked authorized operations in the correct sequence, we configured the DRED with an application layer proxy to filter user-generated messages to and from the protected host and to log those messages to an external server for further analysis. The DRED itself was capable of detecting and preventing unauthorized messages, such as invalid origin, destination or content, but the external server was required to verify the correct operational sequence of those messages *across* all DRED-protected hosts.

For the PCS network, operations leveraging the historian may enjoy some guarantee of provenance; however, other operations may not be so well protected.

7. BENEFITS OF CONVERSATION-BASED MANAGEMENT

Traditional policy management tools focus more on the rules to be implemented by a particular policy enforcement device than on the authorizations that will be enabled by those rules. This can obscure these authorizations, especially for administrators not familiar with the enforcement device, and the practice is especially unwieldly for managing defense-in-depth [10]. To scale policy specification more effectively, Adventium developed the *conversations* policy management approach [8]. The Conversation Manager (CM), a MySQL-based application, enables the construction and maintenance of conversations, and it supports their graphical display of conversations according to the needs of the viewer.

Each *conversation* authorizes a set of service instances (that is, services and their providers) to a set of consumers. Unlike a traditional policy, the conversation does not specify all of the authorizations related to these consumers or providers but only the authorizations related to the named services. Once all required conversations are defined, the CM translates them into the permissions that must be enforced by each specific enforcement device. Each *permission* names one consumer, one provider and one service in which they

may engage. Next the device applies *service semantics* that are unique to it to determine the appropriate collection of rules (packet filtering, encryption, etc.) that should be applied to communications authorized by this permission.

Conversations separate the concerns of *what* must be authorized (the conversation statement itself), from *where* it must be enforced (the device-specific permissions), and from *how* it must be enforced (the service semantics). Since there is relatively low coupling between conversations and service semantics, the *what* may be defined by a different individual than the *where* or the *how*, and because they are service-based and composable, conversations themselves may be defined by different individuals with expertise in the named services.

8. EXPERIMENTAL RESULTS

From a policy perspective, our research confirmed the benefits of the conversations policy management approach. We observed one and a half orders of magnitude improvement, in terms of things to manage, from managing conversations versus writing the necessary enforcement rules manually. As a case in point, we added the `arptables` defense to the DRED late in the program. It did not require any change to the conversations themselves but only the addition of `arptables`-specific service semantics.

9. NEXT STEPS

We plan to conduct performance tests of the DRED in an appropriate environment; however, we expect that its overall performance is as good as its commodity component parts (e.g., Linux, iptables, IPsec, arptables).

We recently completed a study for hosting the device on a virtual machine (VM). A VM combines the benefits of the NIC and the bump-in-the-wire appliance in that it is technically embedded yet enjoys a full featured operating system. Our research reveals that emerging VMs are striving for the levels of assurance necessary to assure non-bypassability and tamper-resistance. Hardware support, in the form of processor virtualization support and input/output memory manage units (IOMMU), continues to play a key role in achieving these objectives.

10. SUMMARY

In this paper, we highlighted the benefits of host-resident, prevention-based, network layer devices for protecting PCS networks. We also highlighted the benefits of service-based authorizations to manage these devices in a scalable fashion.

Thanks to available technologies, these devices can have deep visibility into the host's network traffic, and along with supporting user authentication technologies, they can enforce per user network access. By leveraging encryption, the devices enable trustworthy and authenticated control and data paths. Attribution in the face of policy violations remains a challenge, but with these other protections in place, the field of likely candidates is significantly reduced.

With confidence that the PCS network securely allows only intended activities, PCS security analysis can now focus on deviations and aberrations within the context of those duties. In other words, strong prevention makes detection and response operationally feasible.

11. REFERENCES

[1] S. M. Bellovin. Distributed firewalls. *;login:*, 1999.

[2] M. Carney, R. Hanzlik, and T. R. Markham. Virtual private groups. In *Proceedings of the 3rd Annual IEEE Information Assurance Workshop*, 2002.

[3] J. T. Haigh, S. A. Harp, R. C. O'Brien, C. N. Payne, J. Gohde, and J. Maraist. Trapping malicious insiders in the spdr web. In *Proceedings of the Forty-Second Annual Hawaii International Conference on System Sciences (HICSS-42)*, Waikoloa, Big Island, Hawaii, January 2009. To appear.

[4] S. Ioannidis, A. D. Keromytis, S. M. Bellovin, and J. M. Smith. Implementing a distributed firewall. In *CCS '00: Proceedings of the 7th ACM conference on Computer and communications security*, pages 190–199. ACM Press, 2000.

[5] A. Keromytis, S. Ioannidis, M. Greenwald, and J. Smith. The strongman architecture. In *DARPA Information Survivability Conference and Exposition*, 2003.

[6] T. Markham and C. N. Payne. Security at the network edge: a distributed firewall architecture. In *DARPA Information Survivability Conference Exposition II*, 2001.

[7] L. M. Meredith. A summary of the autonomic distributed firewalls (adf) project. In *DARPA Information Survivability Conference and Exposition*, 2003.

[8] R. C. O'Brien and J. Charles N. Payne. Virtual private groups for protecting critical infrastructure networks. In *Cybersecurity Applications and Technology Conference for Homeland Security*, pages 118–123. IEEE Computer Society Press, 2009.

[9] C. Payne and T. Markham. Architecture and applications for a distributed embedded firewall. In *17th Annual Computer Security Applications Conference*, December 2001.

[10] P. Rubel, M. Ihde, S. Harp, and C. Payne. Generating policies for defense in depth. In *Proceedings of the 21st Annual Computer Security Applications Conference*. IEEE, 2005.