Status Report on

Protected Domains for Cyber Infrastructure Management

Cynthia E. Irvine, Timothy E. Levin, George W. Dinolt

Center for Information Systems Security Studies and Research
Computer Science Department
Naval Postgraduate School
Monterey, California 93943

Abstract: Cyber infrastructure management is currently carried out within the same domain as both benign and malicious applications. Here we describe an architecture to provide better protection for management-level emergency response capabilities through the use of distributed, highly secure, protected domains. Instead of creating a costly physically separate domain, logical separation will be used. This work will develop an architecture and prototype demonstration in the context of an open source operating system.

Keywords: Homeland Security, Computer Security, Network Security, Information Assurance

1 Introduction

Currently, our national cyber infrastructure is vulnerable at both the node and router levels to attacks by adversaries ranging from untutored script-wielding novices to sophisticated threats from well-funded, well-organized groups and nation states. These attacks can result in the exposure of sensitive information, corruption of critical data, and the denial of system and network use by authorized entities. Although considerable effort has been devoted to the detection of attacks, little has been invested in infrastructure architectures that would permit a well-managed response to these attacks.

To exacerbate matters, many components of the nation's critical infrastructure are dependent upon the national cyber infrastructure. The latter is currently recognized as a pathway for a cascaded attack and in recognition of this situation the new Department of Homeland Security has placed "especially high priority on protecting our cyber infrastructure from terrorist attack."

In a speech presented at the Microsoft Conference Center in Redmond, Washington on 4 June 2002, Richard Clarke, Special Advisor to the President for Cyber Space Security and Chairman, President's Critical Infrastructure Protection Board, called for research to create separate protected channels for the administration of critical components of the National Information Infrastructure. Such channels would permit the management of computers and networks even

when the infrastructure was under attack and would permit the management components to allocate resources to services critical for local, state, and national response.

Current computer and network architectures do not provide separation of resource management services from those supporting run-time activities. Thus, through the corruption of payload and runtime facilities, the ability to manage the information infrastructure or provide critical emergency functions can be sabotaged. Although a physically separate resource management / emergency response channel could be constructed, its cost would be prohibitive. Logical separation of management and runtime channels provides an alternative that can be implemented in the near term and can be integrated into existing and emerging network components.

Our objective is to develop an *emergency response* capability for communication / computation facilities that will automatically become available to local authorities during a time of crisis when the standard systems become unavailable because of natural disaster or human (terrorist?) activity. The system should operate in a fashion that is analogous to the emergency lighting system in a building. When the power goes out, enough lighting comes on to ensure that a safe exit of the building is possible. When standard communication and computer facilities are disrupted, then the emergency system should automatically become available for use to provide limited, temporary support to the local authorities so that they can continue to function.

Current emergency systems obviously support some forms of emergency communications. But these communications systems are not linked to the computer networks. Much of the modern management, communications and control functions now take place, not by voice, but by computer. The National Information Infrastructure needs an *emergency response* capability.

The ultimate emergency-response system we envision will be a managed subset of the national information infrastructure using the same physical components but logically separated as an independent out-of-band domain. Key network nodes, both processing and routing, will be emergency enabled by way of this multi-domain capability. Intrusion detection and other means will provide emergency response triggers for the transition of these nodes to a "safe" mode. Once in the safe mode, the protected nodes can process emergency and management functions without interference from other system and network activities, which will be temporarily halted. After the emergency situation is resolved, the non-critical activities can be re-enabled, perhaps gradually, to bring the system back to a normal state. For protected nodes, the logical separation of the protected domain will be their most critical and highly assured security function.

To demonstrate the concepts described above, emergency protection domains will be designed and demonstrated for general-purpose processing (viz., end-system) nodes. Future work to be based on capabilities provided here will address the integration of intrusion detection and other health-status triggers, the automated intercommunication of network status among protected nodes, and the domain protection of interior (e.g., router) nodes.

2 Related Work

The related work in this area is primarily in the areas of Quality of Service (QoS) and protection against Denial of Service (DoS) attacks. QOS provides protocols that guarantee that specified levels of service will be provided under specified loads to the system. QoS models do not normally assume disaster or attack against the systems. They do assume that the underlying system "works" under the specified loads. Systems that protect against DoS provide protection against specific attacks. The mechanisms used assume that the underlying system is protected.

Much of the work in Data Integrity has been focused on providing guarantees that the structure of the information is correct and that only specified programs are permitted to access and modify information. To our knowledge, no one has used Data Integrity properties of the underlying system mechanisms to help ensure a minimal level of service.

Our work integrates Data Integrity mechanisms into the underlying systems to guarantee that the system provides a minimal (emergency) level of service in the face of hostile acts or other major disruptions.

Much of the work in supporting allocation of resources in our current computer infrastructure has centered on the Simple Network Management Protocol (SNMP). The current version is SNMPv3 and is documented in a series of "RFC's" (2570-2576). SNMP provides a framework for sharing and updating configuration information, the Management Information Base (MIB), of the components of a network. The underlying assumption is that the owners (managers) of a set of components of the Internet (an Administrative Domain) are responsible for managing (providing resources) for their part of the system. The components of the system all support the SNMP protocols. System managers use a "network management system" that communicates with the components using the SNMP protocol. RFC 2574 describes the security properties of that system. The policy and mechanisms described there do provide some level of integrity and authentication within an Administrative Domain. They explicitly do not address any provision of service guarantees or any sort of "emergency support" in the case of attack on the system. Our approaches are designed to address these issues.

Synchronization of routing information among Autonomous Systems (Administrative Domains above) is managed using the Border Gateway Protocol (BGP) defined in RFC's 1771 and 1772. This protocol explicitly does not deal with any security issues. Hence systems that implement these protocols are "on their own" when dealing with any "emergency" communications.

Much research has been done investigating how to provide guarantees of Quality of Service in the face of various demands on computer system resources. Some of the earliest work in this area was by John Nagle [Nagle84] and later Van Jacobsen in providing reliable TCP connections. More modern work is exemplified by the MONET Research Group [MONET] at the University of Illinois. The emphasis of this work is to guarantee levels of service to applications. There is no discussion of either the security or integrity implications of the services provided. NPS has developed a Quality of Security Service set of applications that are the beginnings of the kind of work that we envision.

Guarantees of service are provided for by using *rings* for protection of the underlying operating system, and by extension the infrastructures, from malicious applications and components. Ring structures were used in operating systems to provide process separation in data integrity. The most famous example of such use was Multics [Organic72]. A different approach was suggested by the work Millen [Millen92]. He advocates that one should pre-allocate resources to protect against attacks. Our model is similar to this work. It is based on integrity and triggering to reallocate resources under emergency situations rather than redesign and rebuild entire systems to guarantee resources as suggested by Millen.

3 Overview

Four interrelated elements are combined in our work.

1. Analysis and Design of Domain Architecture for Infrastructure Management

2. Implement Extended Attributes For Domain Management
3. Implement Signaling, Scheduling, and Policy Mechanisms for Emergency Response
4. Demonstrate of System Transition to and from Safe/Restricted Mode in Response to Simulated Emergency

Each element is described in detail below. The implementation is based on the OpenBSD code line. OpenBSD provides a stable development environment, and its emphasis on security and security auditing provide additional assurance, over and above that available through other commercial and open source operating systems, that trivial security errors such as buffer overflow do not occur. Many commercial entities rely upon open source platforms from the BSD family. This includes Yahoo, which runs 6000 BSD-based systems, and Hotmail, a Microsoft-owned email system.

Domain Architecture for Infrastructure Management

The protection domains provided at individual processing nodes will be based on a *ring* architecture [Organick72]. In a generalized ring mechanism, the system binds subjects and objects to specific rings, and restricts accesses of subjects to objects based on their respective ring bindings. A *ring bracket* mechanism extends rings to provide specific limitations based on the access mode (e.g., read, write, or execute). Thus, rings provide *protection domains* in which each object may be used.

In a separate project [MYSEA], we are building a mandatory access control mechanism on OpenBSD using extended attributes [Watson01], including the definition of security labels for subjects and objects and incorporation of logic to enforce mandatory security policies with respect to those labels.

To be meaningful, policies for rings or mandatory access control must be enforced globally, for all accesses, and must be enforced persistently, rather than intermittently. The usual means for ensuring global and persistent enforcement of these policies is to rely on hardware mechanisms, such as the segment descriptors available in the Intel x86 design. However, modern commercial and open source operating systems do not generally utilize this sort of hardware security support. Nevertheless, open source software implementations of mandatory security policies have proceeded, providing a limited degree of assurance for global and persistent enforcement.

A *software ring architecture* is being developed to map specific elements of the general ring mechanism to the protection structures provided by OpenBSD (including our mandatory access control extensions), to provide a limited ring mechanism. The limited rings will be of sufficient functionality to support emergency response domain separation, while allowing later extension to support a general ring bracket mechanism. The overall strategy is to leverage the assurances provided by the OpenBSD extended attribute and mandatory access control mechanisms, in support of global and persistent ring policy.

Extended Attributes For Domain Management

The extended-attribute label space provided by [MYSEA] provides a foundation that can be extended to define separate domains (rings) for critical and non-critical processing. Logic in the security management function can be used to determine whether a process is critical or non-critical. This will help to prevent corruption of information in the administrative domain. An interface to the scheduling mechanism permits it to find out which processes are critical or non-

critical. This will ensure that critical functions always have sufficient priority to execute in order to address potential emergency response requirements. This mechanism is being designed so that it is extensible to support a general ring bracket mechanism.

Signaling, Scheduling, and Policy Mechanisms for Emergency Response

External signals from intrusion detection or other sources will provide emergency response triggers for the transition of key network nodes (e.g., administrative processing and routing nodes) to a "safe" mode. Once in the safe mode, the protected nodes can process emergency and management functions without interference from other system and network activities, which will be temporarily halted. After the emergency situation is resolved, the non-critical activities can be re-enabled, perhaps gradually, to bring the system back to a normal state.

A node will be given a signal in order to know when to transition into safe mode. To accomplish this, a secure signal-receiving mechanism will be incorporated into the OpenBSD kernel, the scheduling mechanism will be modified to be able to affect transitions, and a policy definition and storage mechanism will be added to define the possible transitions.

The signaling mechanism will be based on previous work for surreptitious signaling between processing nodes [Anderson02]. In this approach, a common, well-supported communications protocol is used to tunnel information directly into the operating system. Secure cryptographic protocols will be utilized to ensure that the signal is from a protected node and cannot be spoofed. The mechanism will support local as well as remote initiation of the signal. An approach to verify the security and correctness of the protocol will be defined for later work.

A policy definition language such as in KeyNote [Blaze99] will be utilized for defining the possible security states (e.g., safe, normal) and their designators. An extension to the KeyNote framework will permit management of the policy database.

The scheduler will be modified to interface with the signal-receiving mechanism. Upon receipt of an indicator to change security state, the scheduler will interact with the policy module to determine the new state; it will query the security management function to determine which processes or process attributes qualify for the new state; and it will initiate a reschedule transition to ensure that (un)qualifying processes are (blocked)unblocked.

This mechanism is being designed to ultimately support a general ring bracket mechanism.

Demonstration

A small-scale network environment can be used to demonstrate the effectiveness of the proposed mechanisms for transitioning to and from safe/restricted mode. An "administrator" console will be utilized to simulate the signal, e.g., of an emergency. A graphic visualizer can be provided for showing current active processes and security-critical process attributes, such as rings. After a signal is received to change state, and rescheduling has occurred, the visualizer will show the difference in blocked or running processes. The demonstration will permit extension to show different "attack" scenarios, such as denial of service attacks, and the emergency response to those attacks.

4 Conclusion

We have described an approach to infrastructure protection for networks that provides a protected domain for critical management functions. This domain ensures continuity of operation for functions necessary to respond to attacks at the application level. Our architecture

is based upon the enforcement of distributed rings where administrative functions occupy a logical ring more privileged than that populated by applications. A preliminary demonstration is intended to illustrate these concepts.

5 References

[Anderson02] Anderson, Emory, A Demonstration of the Subversion Threat: Facing a Critical Responsibility in the Defense of Cyberspace, Masters Thesis, Naval Postgraduate School, March 2002

[Blaze99] Blaze, Matt, Feigenbaum, Ioannidis, Keromytis, The KeyNote Trust-Management System -- Version 2, Internet RFC2704, IETF, September 1999

[Blum99] Blumenthal, U. and Winjnen, B. "User-based Security Model (USM) for version 3 of the Simplen Network Management Protocol (SNMPv2), RFC 2574, IETF, April 1999. (http://www.ietf.org/rfc/rfc2574.txt)

[Irvine01c] Irvine, C., Levin, T., Spyropoulou, E., Allen, B. "Security as a Dimension of Quality of Security Service", Proceedings of the Active Middleware Services Workshop, August 2001, San Francisco, CA, pp 87-93

[Irvine01b] Irvine, Cynthia E., Levin, Timothy, Wilson, Jeffrey D., Shifflett, David, and Pereira, Barbara, "A Case Study in Security Requirements Engineering for a High Assurance System," Proceedings of the Symposium on Requirements Engineering for Information Security, March 2001

[Irvine01a] Irvine, Cynthia E., and Levin, Timothy, "Data Integrity Limitations in Highly Secure Systems," Proceedings of the International Systems Security Engineering Conference, Orlando, FL, February 2001

[Irvine00b] Irvine, Cynthia E., and Levin, Timothy, "Quality of Security Service," to appear in the Proceedings of the New Security Paradigms Workshop, September 2000. (Reprinted in the Proceedings of the National Information Systems Security Conference, Baltimore, MD, October 2000, CD version)

[Irvine00a] Irvine, Cynthia E. and Levin, Timothy, "Toward Quality of Security Service in A Resource Management System Benefit Function," Proceedings of the 2000 Heterogeneous Computing Workshop, Cancun, Mexico, May 2000, pp.133-139

[Irvine99] Irvine, Cynthia E., and Levin, Timothy, "Toward a Taxonomy and Costing Method for Security Services," Proceedings of the 15th Computer Security Applications Conference, December 1999, pp. 183-188, Scottsdale, AZ

[Millen92] Millen, Jonathan K. A resource allocation model for denial of service, in *1992 IEEE Symposium on Security and Privacy*, pages 137-147, Oakland, California, May 1992

[Monet] Monet Research Group, University of Illinois, http://cairo.cs.uiuc.edu/papers.html

[Multics] Multics Features, http://www.multicians.org/features.html

[Nagle84] Nagle, John, Congestion Control in IP/TCP Internetworks, *Computer Communication Review,* 14(4), October 1984

[MYSEA] MYSEA - http://cisr.nps.navy.mil/projectmysea.html, last modified 10/24/02

[Organick72] Organick, Elliot, The Multics System: An Examination of its Structure**,** MIT Press, Cambridge, MA, 1972

[Spyrop02] Spyropoulou, Evdoxia, Agar, Christopher, Levin, Timothy, and Irvine, Cynthia "IPsec Modulation for Quality of Security Service", Proceedings of the International Systems Security Engineering Conference, Orlando, FL, March 2002

[Spyrop00] Sypropoulou, Evdoxia, Levin, Timothy, and Irvine, Cynthia E., "Calculating Costs for

Quality of Security Service," to appear in Proceedings of the 16th Computer Security Applications Conference, New Orleans, LA, December 2000, pp. 334-343

[Watson01] Watson, Robert, TrustedBSD Adding Trusted Operating System Features to FreeBSD, Proceedings of the USENIX Annual Technical Conference, USENIX, Boston, Mass, Jun-01