SANDIA REPORT
SAND2011-5964
Unlimited Release
Printed August, 2011

OPSAID Improvements and Capabilities Report, Release 1

Adrian R. Chavez and Ronald D. Halbgewachs

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2011-5964
Unlimited Release
Printed August, 2011

OPSAID Improvements and Capabilities Report

Adrian R. Chavez
Networked System Survivability and Assurance Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-0672

Ronald D. Halbgewachs
Effects-Based Studies Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-1248

Abstract

Process Control System (PCS) and Industrial Control System (ICS) security is critical to our national security. But there are a number of technological, economic, and educational impediments to PCS owners implementing effective security on their systems. Sandia National Laboratories has performed the research and development of the OPSAID (Open PCS Security Architecture for Interoperable Design), a project sponsored by the US Department of Energy Office of Electricity Delivery and Energy Reliability (DOE/OE), to address this issue. OPSAID is an open-source architecture for PCS/ICS security that provides a design basis for vendors to build add-on security devices for legacy systems, while providing a path forward for the development of inherently-secure PCS elements in the future. Using standardized hardware, a proof-of-concept prototype system was also developed. This report describes the improvements and capabilities that have been added to OPSAID since an initial report was released.[1] Testing and validation of this architecture has been conducted in another project, Lemnos Interoperable Security Project, sponsored by DOE/OE and managed by the National Energy Technology Laboratory (NETL).

[1] OPSAID Initial Design and Testing Report, Sandia National Laboratories, SAND2007-7552, November, 2007.

Acknowledgements

The authors acknowledge the work that produced the results presented in this paper was funded by the U.S. Department of Energy/Office of Electricity Delivery and Energy Reliability (DOE/OE) as part of the Cybersecurity for Energy Delivery Systems (CEDS) and National SCADA Test Bed (NSTB) Programs.

Executive Summary

Process Control Systems (PCS) and Industrial Control Systems (ICS) are very important for critical infrastructure and manufacturing operations, yet cyber security technology in PCS and ICS, in the past, has been generally poor. The OPSAID (Open PCS Security Architecture for Interoperable Design) Project is intended to address these security shortcomings by accelerating the availability and deployment of comprehensive security technology for PCS, both for existing PCS and inherently secure PCS in the future. Additional efforts in ICS have also seen potential application of OPSAID. As organizations within other sectors confront the issues inherent in moving from legacy to secured systems, applications of OPSAID are beginning. All activities are closely linked to industry outreach and advisory efforts.

Generally speaking, the OPSAID project is focused on providing comprehensive security functionality to PCS that communicate using IP. This is done through creating an interoperable PCS security architecture and developing a reference implementation of that architecture which has been tested extensively for performance and reliability. Three key challenges and proposed solutions identified in the Roadmap to Secure Control Systems in the Energy Sector [1] are addressed by the OPSAID architecture.

Building upon information provided in an earlier OPSAID Initial Design and Testing Report [6], this current report describes the improvements and capabilities that have been added to OPSAID since that initial report was released. This report describes the addition of Public Key Infrastructure (PKI) and Host Intrusion Detection System (HIDS). A description of the functional flow of OPSAID and the available operational configuration parameters are also provided.

Testing and validation of this architecture has been conducted in another project, Lemnos Interoperable Security Project ("Lemnos") [7], sponsored by DOE/OE and managed by the National Energy Technology Laboratory (NETL). It is through the Lemnos project that outreach to vendors and end-user utility providers has been made possible.

Through the improvements described in this report, testing directly with multiple vendors, and extensive validation testing in a variety of test scenarios within a utility company development facility, OPSAID has reached a very high Technology Readiness Level (TRL) for application within multiple sectors of the US Infrastructure. Industry outreach has been highly successful through the OPSAID and Lemnos Projects. [8]

New requirements and standards being established for the Smart Grid present future applications of the OPSAID Architecture and Design at multiple levels of security within that system. The varied areas of Smart Grid distribution systems, third-party applications, and advanced metering all present security issues that, in part, OPSAID addresses.

—This page intentionally left blank —

Table of Contents

Table of Figures

Table of Tables

1 Introduction

1.1 Background

Process Control Systems (PCS) and Industrial Control System (ICS) are very important for critical infrastructure and manufacturing operations. These systems collect and transmit information between sensors, controllers, and central management stations; concurrently they store, process, and analyze information. They have been implemented to work in a number of physical environments using a variety of hardware, software, networking protocols, and communications technologies.

The protection and security of these systems is critical to our national security. There have been a number of technological, economic, and educational impediments to PCS & ICS owners implementing effective security on their systems. Sandia National Laboratories has performed the research and development of the OPSAID (Open PCS Security Architecture for Interoperable Design), a project sponsored by the US Department of Energy's Office of Electricity Delivery and Reliability (DOE/OE), to address some of these issues.

1.2 Description

The OPSAID (Open PCS Security Architecture for Interoperable Design) Project is intended to address these security shortcomings by accelerating the availability and deployment of comprehensive security technology for PCS, both for existing PCS and inherently secure PCS in the future. Additional efforts for ICS have also seen potential application of OPSAID in industrial and manufacturing systems. As organizations within other Infrastructure Sectors confront the issues inherent in moving from legacy to secured systems, applications of OPSAID will be realized. The primary customers for this effort are PCS/ICS security technology vendors/manufactures and, ultimately, the end-user energy providers and industry.

The OPSAID project is focused on providing comprehensive security functionality to PCS that communicate using the Internet Protocol (IP). This has been done by designing and developing interoperable security architecture. A proof-of-concept implementation of that architecture has been tested extensively for performance and reliability. (Figure 1.1) These units are composed of standard off-the-shelf hardware components (Appendix B) and are installed in a standard, rack-mountable frame.

By participating in another DOE/OE supported project known as the Lemnos Interoperable Security Project [7], these OPSAID units have served as the reference implementation for vendors/developers of marketable systems. Section 3 of this report describes the testing and evaluation of both OPSAID and the vendor systems that have participated in the Lemnos Project.

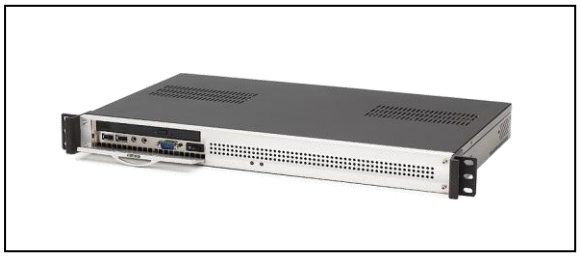

Figure 1.1 OPSAID Reference Implementation

1.3 Historical Information

There are many factors that precipitated the need for the OPSAID Project. One common thread among automation systems is that they were developed without adequate regard for security issues. Traditionally, PCS had relatively little in common with typical information technology (IT) systems. PCS communication was typically conducted over serial links and PCS assets were completely segregated from other IT assets. The PCS assets typically were purpose-built and did not incorporate commercial off-the-shelf technology (COTS) found in IT systems.

Changes in the nature of PCS assets and how they communicate have driven the need for the OPSAID Project. To reduce costs, PCS manufacturers are increasingly incorporating COTS computer hardware and software components in new devices. This has led to an increased use of PCS communication using the Internet Protocol (IP). Yet, IP-based PCS systems have typically lacked many of the basic security features ubiquitous in traditional IT systems, such as detailed logging, authentication, and firewall services.

Compounding these problems, many PCS owners need to have significant information sharing between their PCS and their traditional business systems. This potentially exposes the CS to a much wider range of cyber attack, due to the network connections between the two systems.

Three key challenges and proposed solutions identified in the Roadmap to Secure Control Systems in the Energy Sector [1] are addressed by the OPSAID architecture:

"Further the adoption of bump-in-the-wire encryption to secure communications" and "Improve performance of legacy communications to enable the application of security solutions." [3]

"Develop secure robust wireless solutions for control systems" and "Integrate cryptographic and communications modules." [4]

"Provide the research, design, and development of advanced functionality and proof-of-concept, prototype control system security devices directed toward open and interoperable security architectures." [5]

1.4 Significance

Two goals from the "*Roadmap"* document, in particular, "Develop and Integrate Protective Measures" and "Detect Intrusion and Implement Response Strategies" [3] are realizable using the components in the OPSAID architecture. The OPSAID architecture can be applied to PCS or ICS systems. Furthermore, the architecture can either be implemented as a separate PCS/ICS security appliance or can be incorporated into a PCS/ICS end device (client or server).

Security functionality is attained through a series of modules that provide encryption, authentication, secure remote management, logging, intrusion detection, and firewalls to legacy automation platforms (including PLCs, RTUs, servers, and network devices) on a per-platform basis. OPSAID provides a path forward for the development of inherently-secure PCS/ICS elements in the future.

1.5 Literature Review

Initial design and development of the OPSAID architecture has been described in detail in the report OPSAID Initial Design and Testing Report. [6] This document details the development approach and early testing results in the project. The earlier document also includes a full installation guideline. Additionally, the baseline standards used as guidelines for OPSAID are referenced.

It is intended that this current report of capabilities and improvements of the OPSAID system be used in conjunction with the Initial Report and for that reason, many of the descriptions found in the Initial Report are not reiterated in this report.

1.6 Purpose

The OPSAID project is based upon previous Sandia-led research in the area of PCS security. Research at Sandia and elsewhere has focused on how to improve security and reliability over the long-term for next-generation PCS. However, as PCS are often attractive targets for adversaries and replacement cycles generally range in decades, rather than in months or years, there clearly was a critical need to identify ways to address security shortcomings in the short- and medium-term, yet be complementary to next-generation PCS security solutions. [2]

Furthermore, many end-users identified the need for interoperability of PCS systems provided by multiple manufacturers and vendors. Too often, end-users have been locked into legacy systems for decades and these end-users would like to have the capability to select "best-of-breed" PCS components for specific parts of the control system. Secured interoperability has been the driver for the research and development of the OPSAID architecture.

A critical challenge for success of the OPSAID project was identified in the ***Roadmap*** "*Pursue* a*n aggressive outreach effort to gain industry consensus and buy-in on proposed solutions that will identify ways to cooperate among competitive entities*."[5] The Lemnos Project described earlier has served as an outreach program and more vendors now view the area of added security as a market opportunity. The approach taken by the OPSAID Project will hasten the development of new interoperable systems for PCS security.

2 OPSAID Capabilities

2.1 New Capabilities

Two primary capabilities have been added to the suite of interoperability and security capabilities since the publication of an initial report on OPSAID [6]. Those capabilities, public key infrastructure and host intrusion detection, are described in the following sections.

As revisions to open source software have been made available, those current revisions have also been implemented within OPSAID to assure that the prototype system has the latest capabilities available for testing and analysis and does not become stagnate. The current release (Release 1) software implemented is detailed in Appendix B. Note that additional software is implemented in the prototype system beyond the specific interoperability and security elements of the system. This additional software is utilized for visualization, testing, and support to the OPSAID system.

2.1.1 Public Key Infrastructure (PKI)

Internet Protocol Security (IPsec) is a suite of protocols for securing Internet Protocol (IP) communications by authenticating and/or encrypting each IP packet in a data stream. IPsec also includes protocols for cryptographic key establishment. The security of a cryptographic system depends, in large part, on the security of the methods and practices used to generate and distribute keys.

2.1.1.1 PKI Background Information

For small systems, keys can be distributed by manually installing them. There are two well-know approaches to the key-distribution problem in medium to large-scale systems: key-distribution centers (for secret-key cryptography) and certification authorities (for public-key cryptographs).

A key-distribution center (KDC) is an online automated secret-key provider. The KDC shares a secret distribution key with every party it serves, so its storage requirements are linear in the number of its clients. The KDC randomly generates a secret key for two parties to use and transmits that key information to the two parties encrypted under the distribution key it shares with the parties.

With public-key cryptograph, the challenge is distributing the public keys in a secure fashion. Confidentiality is not an issue because the public keys are not secret, but integrity protection is an issue.

A certificate should be revoked whenever the corresponding private key has been compromised or the attributes that the certificate is binding to a public key are no longer accurate. For example, a certification containing access control data must be revoked whenever access control permissions described in that certificate are changed. Implementing timely revocation of certificates requires some sort of service that is highly available, so that users can check the status of a certificate just before use.

Web browsers employ server certificates, usually issued by public certification authorities (CAs), in using the secure socket layer (SSL) protocol to establish encrypted, one-way, authenticated communications paths.

The term "public-key infrastructure" (PKI) is used in the literature, and especially in trade publications, for a collection of topics related to public-key management. Here, PKI refers to technical mechanisms, procedures, and policies that together provide a management framework for enabling public-key cryptograph deployment in a range of applications. The technical mechanisms generally include public-key digital signature and one-way hash algorithms, the syntax of public-key certificates and certificate revocation lists (CRLs), communication protocols for the issuance, reissuance, and distribution of certificates and CRLs, and algorithms for validating sequences of related certificates and associated CRLs. The procedures generally concern issuance, reissuance, and requests for revocation of certificates, and the distribution of CRLS.

When using digital certificates in practice, a Public Key Infrastructure (PKI) is necessary. A PKI defines a set of agreed upon standards, certification authorities, structures between multiple CAs, methods to discover and validate certification paths, operational protocols, management protocols, interoperable tools, and supporting legislation. A fully functional PKI requires a high level of effort and is one of the most important components of a system utilizing digital certificates. Due to the scope of the OPSAID project, only a small set of the functions of a PKI have been implemented into the prototype.

2.1.1.2 PKI Implementation in OPSAID

The first step taken into designing a PKI is the creation of the digital signatures themselves. Openssl, an open source toolkit for SSL/TLS, in conjunction with strongSwan have been chosen in order to generate, sign, and validate digital certificates. In the OPSAID prototype, a CA was also included into the PKI design which resides on the control center OPSAID. The CA holds a self-signed digital certificate which has also been propagated to each of the field OPSAID devices. When two OPSAID field devices wish to communicate, both OPSAID devices must verify the signature of the opposite endpoint by applying the CAs public key. This ensures the authenticity of both endpoints certificates assuming that the CAs private key has not been compromised. Once this validation has been completed, both endpoints can communicate with each other with confidentiality and authentication. In the OPSAID prototype, the process of generating certificates and requesting the signature of the

CA is mostly manual but can be automated fairly easily. StrongSwan is used in order to establish the IPsec tunnel once the keys of the endpoints have been verified.

Another feature utilized from the openssl tool suite is the Online Certificate Status Protocol (OCSP). OCSP is a standard which is designed to maintain and manage the status of digital certificates (revoked, good, expired, etc) and distribute that status to OCSP clients. The status of each certificate is distributed to each of the OCSP clients via the OCSP protocol. If a certificate is no longer valid, then an OCSP client communicating using the "bad" certificate may be denied based on configuration settings. StrongSwan is responsible for the configuration settings in the OPSAID prototype. The configuration chosen was to disallow the use of any "bad" certificates when attempting to establish an IPsec tunnel. This configuration setting was chosen due to the general security implications of a certificate being compromised in a production type environment, but this may vary based on the scenario. An alert or warning can be logged for an operator to view and appropriately respond if so desired.

The OPSAID prototype has utilized three of the core features in a typical PKI system. Digital certificates have been used to provide scalability and robustness. The digital signatures are validated by the use of a trusted third party, a CA. The status of each of the certificates is then managed by an OCSP which distributes, via the OCSP protocol, Certificate Revocation Lists (CRLs) to each of the OCSP clients. The CA and OCSP server reside on the control center OPSAID. Each of the field OPSAID devices act as OCSP clients and establish IPsec tunnels through the strongSwan tool based on configuration settings. Automation of such a system would be ideal in a production type system.

2.1.2 Host Intrusion Detection System (HIDS)

The possibility of malicious actions against a control system, made possible through an intrusion into that system, demands that a server or client be prepared to detect that such an intrusion is being attempted. There are two types of intrusion into a control system, the first being an attack from an external source attempting to break into a control system through a network attack. The second form of an intrusion and attack against a control system may occur through software or firmware changes to an existing system that was somehow embedded into a system upgrade or similar level of system level of modification.

2.1.2.1 HIDS Background Information

A host intrusion detection system (HIDS) is designed to help protect a system from a variety of intrusions. Contrary to a network intrusion detection system (NIDS), HIDS monitors and analyses the internal state of a system as opposed to only the systems external interfaces. The goal of HIDS is to detect malicious behavior residing on a system that may have slipped passed a NIDS. The malicious behavior can be classified in a variety of categories including, but not limited to, file modification, rootkits, and registry modification. Neglecting to

incorporate HIDS into a system can result in exploits of the system going unseen by an operator and can have a disastrous impacts.

2.1.2.2 HIDS Implementation in OPSAID

Two HIDS were integrated into the OPSAID prototype addressing the issues discussed above. Initially an in-house solution of HIDS was developed in order to meet the tight space requirements of an early version of the OPSAID prototype. On the second iteration of the OPSAID prototype, more space resources were allocated in order to support more sophisticated open source tools for the HIDS implementation. Both versions have the same goals but the open source version is much more advanced. Both versions can also be ported to many typical end devices found within a Process Control System (PCS), such as a Human Machine Interface (HMI) or a machine which provides Engineering access to the PCS.

The HIDS which was developed in house is capable of detecting file changes on a system. Initially, each file is assumed to be in a trusted state at which point a cryptographic hash is computed and the result is stored in a database. The mapping of the full path filename and the hash is used as a baseline for the initially trusted state of the system. The cryptographic hash can be any hash function provided in a configuration file (here MD5 and SHA1 are the available options). Computing hashes of every file on the entire system can be time consuming depending on the size of the file system, so a subset of the system can be configured to be monitored if desired, such as a list of directories. Once the baseline hashes are computed, every x number of seconds, where x is user configurable, a new hash is computed and compared against the baseline. If there are inconsistencies in the hashes or files are created or deleted, the HIDS can be configured to send events to a syslog server. This HIDS is very basic but is designed to be very lightweight which may suite a resource tight system. This HIDS solution can run on Windows or any Unix Operating System which has the appropriate cryptographic hash functions installed as well as the required Syslog executables to communicate with a Syslog server.

The open source HIDS tool chosen and installed on the OPSAID prototype is developed by OSSEC.[13] The OSSEC solution performs log analysis, integrity checking (similar to the in house HIDS version developed), Windows registry monitoring, rootkit detection, real-time alerting and active response. OSSEC also supports the ability to log all events to a centralized Syslog server. OPSAID utilizes the OSSEC tool by installing an OSSEC server on the Control System OPSAID and installing an OSSEC agent on each of the field OPSAID devices. Similarly, an OSSEC agent can be installed on an HMI machine or any other end device which runs on any of the Windows, Solaris, or Linux OSs. Each OSSEC agent communicates events to the OSSEC server, which in turn are then communicated to the Syslog server. The OSSEC server is responsible for managing the OSSEC agents, such as the times when the agents should update their system hashes or the rules which constitute an alarm on each of the agents. The communication between the server and the agents is encrypted using a symmetric key that is generated on the server and exported to each agent. When the system is functioning correctly, each agent will monitor and analyze their internal

state and will report any suspicious states to the server. The OSSEC server will then forward these events to the Syslog server, which an operator will act appropriately on.

The presence of HIDS in any Information Technology (IT) solution is important for several reasons. Malicious behavior residing on a system, such as a rootkit, may bypass a NIDS solution and compromise the security of the system. Without a HIDS in place, manual audits of each system would have to take place regularly which is not cost effective and error prone. HIDS automates the entire process and can be updated to detect new vulnerabilities easily on a regular basis. Ideally, a HIDS and NIDS solution work together to create better overall security architecture for the system as a whole.

2.2 Overall Capabilities

When describing the functional capability of the OPSAID architecture, it is important to consider the full list of capabilities captured in the design and what elements of this system are available to vendors considering the OPSAID architecture for development effort and the end-user of these capabilities. This list offers end-users a set of "requirements" to choose from that might be included in request for bids by vendors when defining the component elements in replacing legacy systems components.

- **Virtual Private Network - Interoperability of control system elements**
- **Use of encryption and data authentication**
- **System intrusion detection and prevention**
- **Firewalls and network filtering**
- **Authentication and logging for remote access**
- **Public Key Infrastructure – generate, sign, and validate digital signatures including a certificate authority**
- **Host intrusion detection and prevention**
- **Control system monitoring and visualization of the monitored information**
- **Data logging capture for replay and forensic analysis**

2.3 Functional Description

The capabilities of OPSAID described in the previous section are brought together in a cohesive system architecture design that can be configured specifically for the control system application. A functional flowchart of OPSAID is shown below in Figure 2.1. Two prototype variants were constructed for OPSAID demonstration and testing. The two implementations (Figure 1.1) are identical with the exception that one, called the "field unit", will detect a condition that sends a Syslog message to the second unit, called the "server unit" which contains data storage that serves as the Syslog server. This combined system allows the full implantation of a Syslog capability.

This full Syslog capability was extremely important in the testing of other vendor equipment in the Lemnos Project. By providing the OPSAID server unit, testing of vendor units required to have a Syslog capability was, in part, successful because a log of generated message traffic was available as the Syslog server found in actual systems.

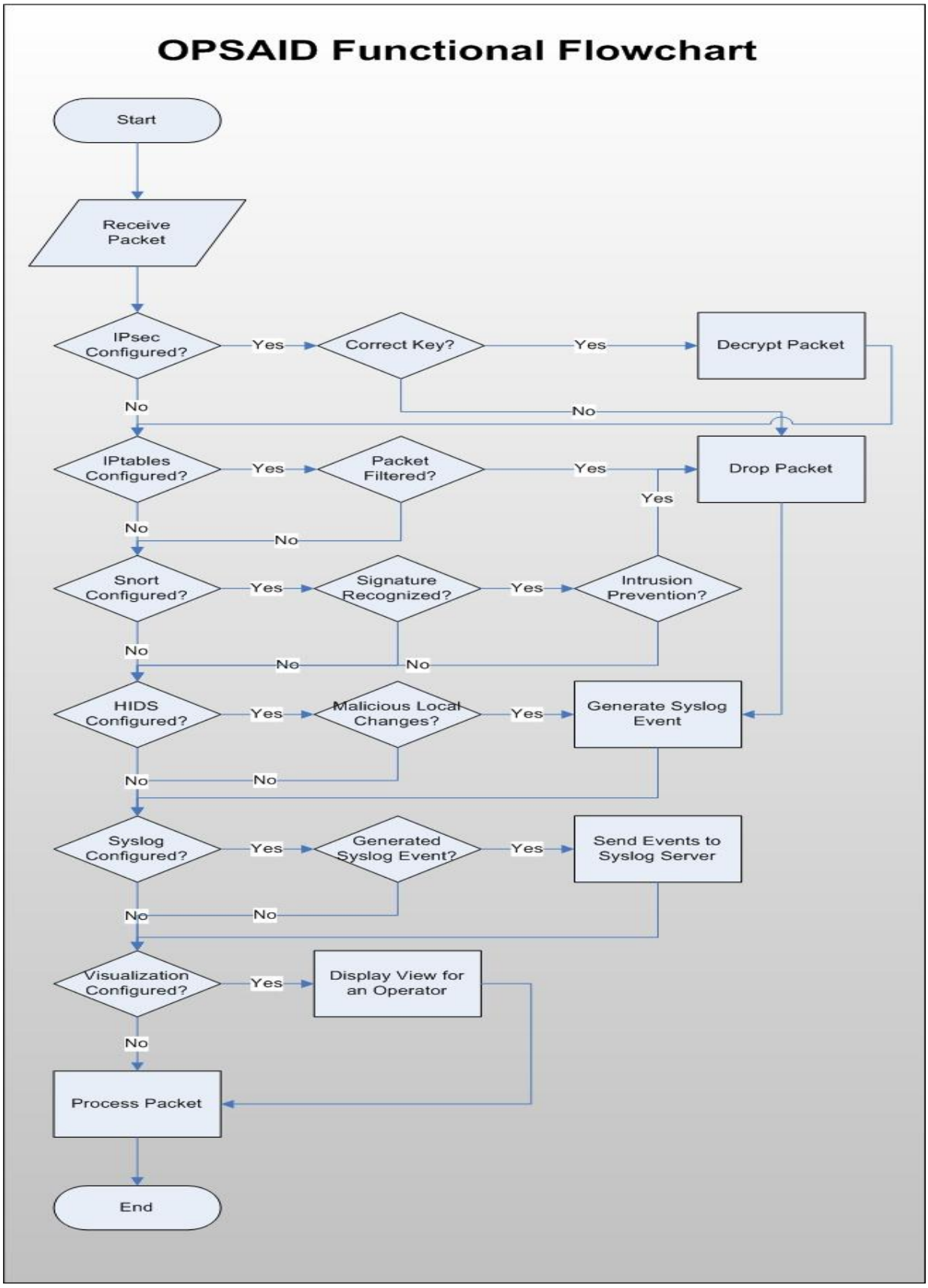

Figure 2.1 OPSAID Functional Flowchart

2.4 Configuration Parameters

Parameter	Allowed Setting
P1 Encryption	DES, 3DES, AES-128/192/256
P1 Hash	MD-5, SHA1/160, SHA2-256/512
P2 Encryption	DES, 3DES, Null, AES-128/192/256
P2 Hash	MD-5, SHA1/160, SHA2-256
PFS	yes/no
Diffie Hillman (DH) Group	1,2,5
Key Lifetimes	auto
DPD	yes & no
IKE version	1&2
Vendor Version of SW	OPSAID v.2, strongswan v.4.2.14
DES = Data Encryption Standard	
AES = Advanced Encryption Standard	
SHA = Secure Hash Algorithm Standard	
PFS = Perfect Forward Secrecy	
DPD = Dead Peer Detection	
Key times "auto"=negotiable to lowest level	
3DES=Triple DES	
IKE = Internet Key Exchange	

Table 2.1 OPSAID Configuration Parameters

3 Testing OPSAID

3.1 Industry Testing & Application to Standards

The DOE/OE NSTB project identified as the Lemnos Interoperable Security Project [7] utilizes the OPSAID architecture for the design and development of a commercial and deliverable implementation. OPSAID has continued to be used as the baseline reference system for extensive testing conducted in the Lemnos project. This testing has included interoperability and secured evaluation with multiple vendors, including evaluation testing within a producer/utility environment (Tennessee Valley Authority).

This effort has provided OPSAID with an industry-based set of requirements that have been used to continue the research necessary to meet those requirements and know this work has provided capabilities needed by industry. Testing of OPSAID within a network composed of different manufacturer/vendor equipment units provided the necessary environment for capability validation (Figure 3.1). These vendors included: Schweitzer Engineering Laboratories, Industrial Defender, Garrettcom, Phoenix Contact, and n-dimension. The description of the test units are simply identified as Units A-E, related to each of the five vendor organizations participating in the tests. In this system, the SNL-1 field unit served as both a standard control system unit and a gateway between the vendors and the Syslog file system (SNL-2) and the Industrial Defender Syslog file system (SEM). The "hosts" identified were represented by each vendor's representative laptop computer for configuration definitions and view into message traffic & diagnostics throughout the testing.

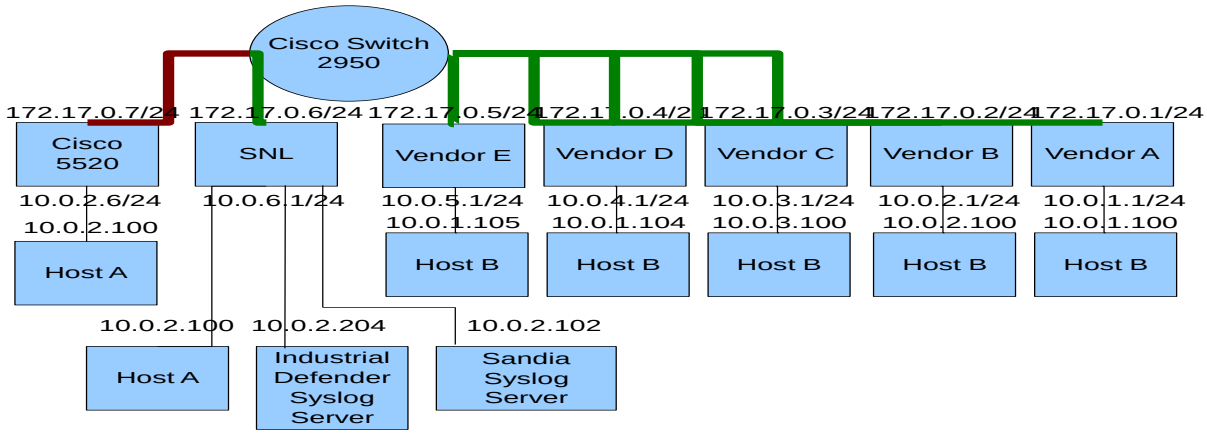

Figure 3.1 Test Diagram – Lemnos Testing

The Lemnos project has done a major amount of work to help clarify interoperable secured communications standards. The project has worked on establishing agreements among vendors & utility companies on secured interoperable communications, Internet Protocol Security (IPSec) through a Virtual Private Network (VPN) tunnel, and common identification information within the content of Syslog messages. Lemnos has produced common baselines and interoperable configuration profiles (ICPs) for the application of existing standards.

Common to all the effort by the Lemnos Project is the understanding that the beneficiaries of this work are the end-user utility companies, the government organizations that have oversight to assure the energy infrastructure is being improved and secured, and the US Critical Infrastructure Energy Sector. The Lemnos project has continually focused on addressing challenges detailed in the Roadmap to Secure Control Systems in the Energy Sector. [1]

Through the development of configuration profiles for secured communications, common baselines for vendor interoperability were established. The secure communications channel requirement was mapped to Internet Protocol Security (IPSec) described in the Internet Engineering Task Force (IETF) [9] request for comment RFC 4301. The Lemnos IPSec configuration profile defined the specific NIST Cryptographic algorithms and key strengths (Diffie-Hellman group, hashing algorithm, and authentication algorithms) that would be acceptable. In addition, the IPSec configuration profile for Lemnos established the parameters necessary for secured key exchange, i.e., number of times for key renegotiation, time until key renegotiation required, enhanced key exchange security, etc.

The Lemnos team also created the trust management baseline to establish those secure connections with the Internet Public Key Infrastructure Certificate (X.509) and the Online Certificate Status Protocol (OCSP). This profile enables a manageable widespread roll out of the secure VPN technology. These parameters are in concert with Recommended Security Controls for Federal Information Systems Special Publications (NIST SP800-53, Rev.2) [10] and Security Requirements for Cryptographic Modules (FIPS 140-2). [11]

Syslog standardization has focused on the IETF RFC 3164 and IETF RFC 5424. Through the Syslog work, a baseline has been established for common definitions of many event types and common message structure. This helps manufacturers to identify some important action that needs to be logged and what information should be in that log to assist asset owners meet requirements of the North American Electric Corporation (NERC) Critical Infrastructure Protection (CIP) CIP003, CIP005, CIP007, and CIP011. [12]

3.2 Technology Readiness

The OPSAID architecture is based upon open-source software and therefore all, or nearly all, of the components of the architecture are being used "world-wide" in many systems within real environments. That is, the components of OPSAID are at TRL 9.

The architecture and requirements of the Lemnos program are based upon a combination of the OPSAID architecture and the needs of industry. Working from Lemnos Interoperability Configuration Profiles (ICPs), testing and evaluation of other vendor systems have been tested against the OPSAID architecture. All of these systems have been, or will be, installed at many industrial facilities. The prototype, rack-mountable OPSAID units have served as the proof-of-concept units as well as the reference units for the Lemnos testing. This level of testing, demonstration, and environment with industry/vendor commercially provided equipment, places OPSAID at TRL 7.

The OPSAID architecture is being defined at "Release Levels". While building upon earlier versions of OPSAID, the architecture described in this report is identified as the Release 1 version. Improvements and added capabilities continue to be made through the NSTB OPSAID and Lemnos projects and will produce a Release 2 version. In this manner, an OPSAID architecture that has been fully tested, evaluated, and demonstrated can be distinguished from an architecture still in design and specification development.

4 Recommendations

The current OPSAID Reference Architecture employs a modular approach to providing security functionality specifications to meet the needs of PCS vendors/manufacturers and end-users. Interoperability and security have been key functionality elements designed into the architecture. Prototype implementations of OPSAID have been demonstrated utilizing standardized hardware components. The DOE/OE NSTB project identified as the Lemnos Interoperable Security Project [7] utilizes the OPSAID architecture for the design and development of a commercial and deliverable implementation. OPSAID has continued to be used as the baseline system for extensive testing conducted in the Lemnos project. This testing has included interoperability and secured evaluation with multiple vendors, including evaluation testing within a producer/utility environment (Tennessee Valley Authority).

A new development for OPSAID should be focused on producing a prototype implementation in a Virtual Machine environment. A virtual implementation of OPSAID will permit multiple instantiations of OPSAID within a simulation environment such as the Sandia-developed Virtual Control System Environment (VCSE). [14] Additionally, a virtual implementation will provide vendors and end-users with a demonstration of embedding secured interoperability within their systems beyond the "bump-in-the-wire".

Inclusion of OPSAID within the VCSE would utilize current prototype hardware and OPSAID/VM for multiple instantiations. Such a series of experiments would (1) include the validation of the OPSAID architecture through the scenario experiments, (2) identify gaps in the architecture or restrictions on how OPSAID can provide the security expected, and (3) identify ways in which the OPSAID architecture and concepts can be utilized through innovative applications within a control system. This effort would also build additional capability within the VCSE Project to develop scenarios and model representations for the scenarios.

The research into additional functionality and capabilities of OPSAID for secured interoperability should extend into secured wireless communications. This task will address enhanced architecture definitions and design for the inclusion of secured wireless communications within the OPSAID Reference Architecture. This effort is aimed to assist vendors and industry as they begin efforts to include wireless communications into their systems. This effort will research the possible inclusion of secured wireless communications to send and receive protocol packets of information that have been encrypted/decrypted through the currently defined use of IPSec within OPSAID.

Successfully demonstrating the goals of OPSAID, secured interoperability should be considered as a critical part of the Smart Grid design [8]. Research and evaluation of the application of OPSAID within the Smart Grid should be taken between utility distribution centers and third party operations for the collection and controls within the Smart Grid implementation.

Additional application of OPSAID can be realized within the Smart Grid design and other Infrastructure Sectors through the implementation of OPSAID capabilities within a smaller physical “footprint”. Such applications might be found outside the normal PCS/ICS control centers in remote areas or physical settings requiring secured communications and user authentication.

Appendix A: References

1. Roadmap to Secure Control Systems in the Energy Sector, Technology Research and Development, January, 2006.

2. Ibid, Framework for Securing Control Systems Goals, p.16.

3. Ibid, Key Challenges and Solutions, Legacy Systems, pg. A-6.

4. Ibid, Key Challenges and Solutions, Control Systems Architecture, pg. A-13.

5. Ibid, Key Challenges and Solutions, Control Systems Architecture, page A-14.

6. Hurd, Steven A., Stamp, Jason E., and Chavez, Adrian R., OPSAID Initial Design and Testing Report, Sandia National Laboratories, SAND2007-7552, November, 2007.

7. Smith, Brian, The Lemnos Interoperable Security Project, ICSJWG 2010 Spring Conference, presentation, April 21, 2010.

8. Stewart, John W., The Secure Connection, p. 24-28, Transmission & Distribution (TD) World, November, 2010.

9. Internet Engineering Task Force, Request for Comments, http://www.ietf.org/rfc.html

10. National Institute of Standards and Technology, Special Publications, http://csrc.nist.gov/publications/PubsSPs.html

11. National Institute of Standards and Technology, Federal Information Processing Standards, http://csrc.nist.gov/publications/PubsFIPS.html

12. North American Electric Reliability Corporation, http://www.nerc.com/

13. OSSEC, http://www.ossec.net

14. McDonald, Michael J., et al, Modeling and Simulation for Cyber-Physical System Security Research, Development and Applications, Sandia National Laboratories, SAND2010-0568, February, 2010.

Appendix B: OPSAID Reference System Definition

Prototype - Standardized Hardware Platform

- Mini-ITX board and fanless enclosure
- 1GHz VIA processor
- 2 Ethernet & 6 serial connections (expandable)
- PCI expandability
- 1 GB flash ROM
- 1 GB RAM

Versions of Open Software Installed

Current OPSAID Module Support:
Firewall: iptables 1.3.8
Network Metric Tool: iperf 2.0.2
VPN: strongSwan 4.1.2
OCSP Server: openssl 0.9.8g, libcurl3 7.18.0, m4 1.4.10
Logging: syslog-ng 2.0.9, mysql 5.0.51a
Visualization: php-syslog-ng 2.9.7, php5 5.2.4, php5-gd 5.2.4
NIDS: snort 2.7.0
HIDS: md5deep 1.12
Remote Login: openssh-server 1:4.7p1

Miscellaneous Tools Installed:
Traffic Replay: tcpreplay 3.2.3
Traffic Sniffer: wireshark 1.0.0, tcpdump 3.9.7
Java: JDK 1.5.0
SNMP: libsnmp 5.4.1
LDAP: libldap 2.4.7
Hex Editor: hexedit 1.2.12
Web Browser: firefox 3.0
Text Editor: emacs 22.1, nano 2.0.6
Development IDE: eclipse 3.2.2
Diff Util: diff 2.8.1
Web Server: apache2 2.2.8
Common Linux Packages: build-essential 11.3
Network Card Tool: ethtool 5
Network Mapper: nmap 4.20

Appendix C: Acronyms and Abbreviations

AES	Advanced Encryption Standard
CA	Certificate Authority
CEDS	Cybersecurity for Energy Delivery Systems
CIP	Critical Infrastructure Protection
CRL	Certificate Revocation List
DES	Data Encryption Standard
DH	Diffie Hillman
DOE	Department of Energy
DOE/OE	Department of Energy Office of Electricity Delivery and Energy Reliability
DPD	Dead Peer Detection
FIPS	Federal Information Processing Standards
HIDS	Host Intrusion Detection System
HMI	Human Machine Interface
ICS	Industrial Control System
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IP	Internet Protocol
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
NERC	North American Electric Corporation
NETL	National Energy Technology Laboratory
NIDS	Network Intrusion Detection
NIST	National Institute of Standards and Technology
NSTB	National SCADA Test Bed
OCSP	Online Certificate Status Protocol
OPSAID	Open PCS Security Architecture for Interoperable Design
PCS	Process Control System
PFS	Perfect Forward Secrecy
PKI	Public Key Infrastructure
PLC	Programmable Logic Controller
RFC	Request for Comment
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SHA	Secure Hash Algorithm Standard
SNL	Sandia National Laboratories
SQL	Structured Query Language
SSH	Secure Shell (protocol)
SSL	Secure Socket Layer
TRL	Technology Readiness Level
VCSE	Virtual Control System Environment
VM	Virtual Machine

Distribution:

(Electronic Copies)

1	DOE/OE	Carol Hawk
1	NETL	Diane Hooie
1	MS 1248	B. P. Clifford, 5623
1	MS 0671	J. M. Depoy, 5628
1	MS 0672	H. W. Lin, 5629
1	MS 0672	A. R. Chavez, 5629
1	MS 1248	R. D. Halbgewachs, 5623
1	MS 0899	Technical Library, 9536