SANDIA REPORT
SAND2007-3345
Unlimited Release
Printed June 2007

Secure ICCP Integration Considerations and Recommendations

John T. Michalski, Andrew Lanzone, Jason Trent, and Sammy Smith

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

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

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

SAND3345
Unlimited Release

Secure ICCP Integration Considerations and Recommendations

John Michalski, Jason Trent, and Sammy Smith
Critical Infrastructure Systems

Andrew Lanzone
Cryptography and Information Systems Surety

Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-0672

Abstract

The goal of this report is to identify the operation and implementation issues associated with the introduction of the secure form of the Inter-control Center Communications Protocol, or ICCP, formally referred to as IEC 60870-6-TASE.2, into the utility infrastructure. The report provides considerations and recommendations to assist a utility owner to advance the security of the utility's data exchange operations. The report starts with a description of information assurance, and then discusses end node authentication and Public Key Infrastructures (PKI) using Certificate Authority (CA) certificates. Network infrastructures and protocols associated with ICCP are reviewed, assessed, and modeled to identify the impact of these structures and protocols to the efficient delivery of ICCP data. The report highlights certificate management and implementation issues and discusses some of the transitional issues and strategies to overcome security limitations during the introduction phase of Secure ICCP. Finally the report provides some performance measurement data of the configuration impacts of using security layers to provide Secure ICCP implementations.

Acknowledgements

The author would like to acknowledge that 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 National SCADA Test Bed (NSTB) Program.

Executive Summary

The Inter-control Center Communications Protocol (ICCP) was developed to enable data exchange over Wide Area Networks between utility control centers, Independent System Operators (ISOs), Regional Transmission Operators (RTOs), and other Generators.

This document describes the intent, operation, and behavior of the ICCP and technological means by which ICCP transmission can be secured, discussing both the built-in protection of Secure ICCP (a version of ICCP that has some built-in security elements) and several independent technologies that can be added to ICCP, such as Internet Protocol Security (IPSec). Recommendations for using the ICCP are provided throughout, especially regarding effective use of its secure form.

This document also describes the impact of Wide Area Network (WAN) design on the transport of ICCP data streams. The importance of using appropriate quality of service (QoS) on the supporting WAN links is demonstrated by including the results of the modeling and simulation of WAN link congestion. Overall, using Secure ICCP and other secure protocols has minimal effect on end-to-end performance, although certain situations with respect to traffic congestion described within the report can cause exceptional delays and should be avoided. Also management complexity increases with each layer of protection added to the ICCP environment.

The primary objectives of the research activity described in this report were to provide insight into the security enhancements of the new ICCP protocol and to identify the integration impact of this emerging standard when implemented within the utility industry infrastructure control system.

These were accomplished by investigating and interpreting documentation of ICCP, Secure ICCP, and related technology including relevant standards, implementation guidelines, and descriptive material; and by implementing and performance testing a Secure ICCP testbed. Section 4 of this report provides the observations and conclusions of the investigation and Section 5 contains a highlighted summary of recommendations found throughout this report associated with four primary areas of analysis; Secure ICCP Certificate Management, Network System Design, Transition Strategy, and Performance.

Table of Contents

1 Introduction 9
1.1 Background 9
1.1.1 Description 9
1.1.2 Historical Information 9
1.1.3 Significance 9
1.1.4 Literature Review 10
1.2 Purpose 10
1.2.1 Reason for Investigation 10
1.2.2 Roadmap Challenges 10
1.2.3 Audience 11
1.2.4 Desired Response 11
1.3 Scope 11
1.3.1 Extent and Limits of Investigation 11
1.3.2 Goals 11
1.3.3 Objectives 12
1.3.4 Organization 12
2 Approach 13
2.1 Methods 13
2.2 Assumptions 13
2.3 Procedures 13
3 Results and Discussion 15
3.1 Impact of the Wide Area Network on ICCP Operations 15
3.1.1 Overview of WAN Impact on ICCP Operations 15
3.1.2 Typical Configuration 16
3.1.3 Infrastructure Design and Protection 17
3.1.4 SCADA Wide Area Networks 20
3.1.5 IP Congestion and QoS management 23
3.1.6 Frame Relay Switched Network 26
3.1.7 Network Impact Summary and Recommendations 31
3.2 Secure ICCP and Information Assurance 32
3.2.1 Overcoming physical layer availability disruptions 32
3.2.2 Overcoming logical layer availability disruptions 33
3.3 ICCP Use of Public Key Cryptography 35
3.4 ICCP Use of Public Key Infrastructure Certificates 35
3.4.1 PKI Certificate Hierarchy Recommendations for ICCP Networks 35
3.5 Secure ICCP Certificate Management Issues 36
3.5.1 Number of Certificates per ICCP Node 36
3.5.2 ICCP Security Policy and Certificates 37
3.5.3 Permanent ICCP SSL Sessions 38
3.5.4 ICCP Internet Certificate Authorities 39
3.5.5 CA and CRL Domains 40
3.5.6 Secure ICCP Stale Certificate Detection 48
3.6 Strategy for the transition from ICCP to Secure ICCP 48
3.6.1 Layer 3 Link Protection 49

3.6.2 Layer 2 Link Protection 56
3.7 Security Configurations and Performance 60
3.7.1 Introduction 60
3.7.2 Security Layer overview 60
3.7.3 ICCP Network-Based Performance Testing 63
3.7.4 ICCP Software-Based Performance Testing 67
3.7.5 Overall ICCP Performance Testing Summary 73
3.7.6 Frame Relay Performance Discussion 76

4 Conclusions 78
4.1 Conclusions relating to Overall ICCP Network System Design 78
4.1.1 Design Components 78
4.1.2 Conclusions Relating to Quality of Service and Service Level Agreements 78
4.2 Conclusions Relating to Secure ICCP Certificate Management 78
4.2.1 PKI domain design Conclusions 78
4.2.2 Inter-Domain Communication Conclusions 79
4.2.3 Secure ICCP application issues 79
4.3 Conclusions Relating to Transition from ICCP to Secure ICCP 79
4.4 Performance of Networks Incorporating Secure ICCP 79

5 Recommendations 80

Appendix A: References 82

Appendix B: Security Technology 84

1 Public Key Cryptography 84

2 Public Key Infrastructure 84
2.1 Registration Authority 84
2.2 Registration Authority (RA) 85
2.3 Certification Authority (CA) 85
2.4 Certificate Repository 85
2.5 Certificate Revocation Mechanism 85

3 Certificates in the Secure Sockets Layer (SSL) 86

4 Certification Hierarchy Schemes 87
4.1 Flat Hierarchy 87
4.2 Tiered Hierarchy 89

5 Certificate Management 92
5.1 Certificate Issuance 92
5.2 Certificate Expiration and Renewal 92
5.3 Certificate Revocation 93
5.3.1 Certificate Revocation Lists 93
5.3.2 Online Certificate Status Protocol 94
5.3.3 Impetus for Certificate Revocation 94

Appendix C: Acronyms 96

Appendix D: Glossary 97

Appendix E: For More Information 98

Table of Figures

Figure 1. Monolithic Master Controller Configuration 15
Figure 2. LAN-based Master Controller Configuration 16
Figure 3. Control Center to Control Center Communications 17
Figure 4. Simulated IP Routed Network 21
Figure 5. Data Flow Statistics on IP Routed Network 22
Figure 6. Data flow on IP Network with QoS Statistics 24
Figure 7. Typical Frame Relay Network Interface 27
Figure 8. Simulated Frame Relay Network 29
Figure 9. Data Flow Statistics on Frame Relay Network 29
Figure 10. Data Flow on Frame Relay Network with QoS Statistics 30
Figure 11. Main Interconnections of the US Electric Grid and Ten NERC Regions 42
Figure 12. Peer-to-Peer Cross-Certification 44
Figure 13. Hierarchical Cross-Certification 45
Figure 14. Central Authority Chain-of-Trust Structure 47
Figure 15. IPSec modes of operation 50
Figure 16. IPSec Gateway Center to Center Communications 51
Figure 17. IPSec Port Filter Implementation 54
Figure 18. Point-to-Point WAN Connection 57
Figure 19. Frame Relay WAN Connection 58
Figure 20. ICCP Network Layout 64
Figure 21. End-to-End Measurement Configuration 65
Figure 22. End-to-Midpoint Measurement Configuration 65
Figure 23. End-to-End Non-Secure Measurements 66
Figure 24. End-to-Midpoint Non-Secure Measurements 66
Figure 25. End-to-End Secure Measurements 66
Figure 26. End-to-Midpoint Secure Measurements 66
Figure 27. Example Use of Timing Analysis Tool 68
Figure 28. Observed Send Preparation Times for Unencrypted Tunnel 70
Figure 29. Observed Send Preparation Times for Shared-key Encrypted Tunnel 70
Figure 30. Observed Send Preparation Times for Public-key Encrypted Tunnel 71
Figure 31. Observed Receive Processing Times for Unencrypted Tunnel 72
Figure 32. Observed Receive Processing Times for Shared-key Encrypted Tunnel 72
Figure 33. Observed Receive Processing Times for Public-key Encrypted Tunnel 73
Figure 34. Non-Secure versus Secure End-to-End Latency with Cisco 3600 VPN 74
Figure 35. Non-Secure versus Secure End-to-Midpoint Latency with Cisco 3600 VPN 74
Figure 36. Average Observed Send Preparation Time by Configuration 75
Figure 37. Average Observed Receive Processing Time by Configuration 76
Figure 38. The server (S) sends its certificate to the client (C) 86
Figure 39. Flat certification authority hierarchy 87
Figure 40. Certificate exchange in a Flat PKI 88
Figure 41. Tiered certification authority hierarchy 89
Figure 42. Incorrect certificate exchange in a tiered PKI 90
Figure 43. Correct certificate exchange in a tiered PKI 91

1 Introduction

1.1 Background

1.1.1 Description

The Inter-control Center Communication Protocol (ICCP) was developed to enable data exchange over Wide Area Networks between utility control centers, Independent System Operators (ISOs), Regional Transmission Operators (RTOs), and other Generators. The security enhancements to ICCP were developed and specified by the Technical Committee 57 (TC57) Working Group 07 (WG07) of the International Electrotechnical Commission (IEC).

Real-time data exchange has become critical to the operation of interconnected systems within the electric power utility industry. The ability to exchange power system data with boundary control areas and beyond provides visibility for disturbance detection and reconstruction, improved modeling capability, and enhanced operation of future control centers.

1.1.2 Historical Information

The *ICCP User Guide* [1] states: "ICCP began as an effort by power utilities, several major data exchange protocol support groups (WEICG, IDEC and ELCOM), EPRI, consultants and a number of SCADA/EMS vendors to develop a comprehensive, international standard for real-time data exchange within the electric power utilities industry."

The vulnerability of unprotected ICCP communication led to inclusion of the application-layer encryption protocol called Secure Sockets Layer (SSL) and its similar successor, Transport Layer Security (TLS) [2]. This resulted in an ICCP whose communication could be encrypted and authenticated, AKA *Secure ICCP*.[1]

This work documented in this report is follow-on to the quick-look review conducted by National SCADA Test Bed (NSTB) on the IEC ICCP-IEC60870-6-TASE.2 draft, which defined the security enhancements to ICCP.

1.1.3 Significance

Secure ICCP is an extension of the existing standard ICCP. Essentially, Transport Layer Security[2] (TLS) is inserted into the appropriate layer of the standard communications profile. TLS [3] is a certificate-based cryptographic protocol that provides encryption and authentication. Secure ICCP provides application layer authentication and message encryption between ICCP servers. This alone is significant because it provides a standard communication protocol for critical infrastructure control systems that is not only widely accepted but also has important built-in security elements.

[1] In this report, "Secure ICCP" indicates this exact form of ICCP. The capitalization of "security" differentiates this protocol from other ICCP installations that may have similar but independently added security features.
[2] TSL is a modernized version of Secure Sockets Layer (SSL). There are differences between TLS and SSL, but the protocols are substantially the same.

1.1.4 Literature Review

A comprehensive report on the design and operational issues associated with introducing Secure ICCP into the Utility community networks does not exists. However, starting within the topic of SCADA architectures, there is work in designing EMS system architectures that are open to future changes and upgrades [4]. ICCP implemented within a Distributed Control System is described in [5]. Experimental work is being conducted to integrate the real-time transport protocol (RTP) with ICCP to determine and validate the effectiveness of this integration and the efficiency of link communication [6]. There is also work on coupling ICCP more closely into EMS applications [7]. Regarding SCADA network reliability, there is a report that discusses the general functional problems of SCADA systems in relation to ICCP and similar protocols [8]. There is also a survey that provides a current status of information security technology needs that relate to transmissions and distribution systems [9]. EPRI addresses the use of security domains and what constitutes a security policy and ranking of assets in the final report of the *Integrated Energy and Communication System Architecture* project [10]. Authentication across borders and the difficulties of sharing information, such as ICCP data, across independent domains are discussed in [11]. A report describing the results of testing on a representative set of SCADA protocols to determine whether identified vulnerabilities could be exploited is of particular interest [12].

The *Request for Comment* (RFC) collection is a series of memoranda encompassing research, innovations, and methodologies applicable to Internet technologies. The serialized RFCs comprise a continuous historical record of the evolution of Internet standards and are cited throughout this report. For more details about RFCs and the RFC process, see RFC 2026, *The Internet Standards Process, Rev. 3* [13]. References to individual RFCs appear explicitly in *Appendix A*, this report's reference section. The RFCs themselves can be accessed at http://www.faqs.org/rfcs/.

1.2 Purpose

Secure ICCP does not provide total security for control system data environments. In addition, certain choices consistent with Secure ICCP but not specified in the ICCP model can weaken or disable security or reduce performance under some conditions or along some pathways. This report advises the practitioner making the transition to Secure ICCP.

1.2.1 Reason for Investigation

Industry is using the Internet more and more to communicate among control centers and is moving towards ICCP and Secure ICCP. The work described in this document is based on lessons learned and an understanding of security requirements and best practice [14]. The general intent of the work and the report is to discover and warn against difficulties and pitfalls. Asset owners and technology providers can use this document to achieve a given level of operational security sooner than by going down the blind alleys and wrong turns themselves. This will result in reduced total infrastructure risk over any given period.

1.2.2 Roadmap Challenges

The *Roadmap to Secure Control Systems in the Energy Sector* [15] says it's important to "Identify best practices for … cyber security of substations and control centers." These

practices "should address extending the fleet of existing legacy systems to new functionality, incorporating advanced components, and migrating to fully advanced systems." The current document describes strategy and tactics for using ICCP (Secure ICCP, in particular) to ensure that communication between substations and control centers is secure.

1.2.3 Audience

The recommendations and best-practice guidance found in this document are intended to inform asset owners and technology providers who will either provide data surety for standard ICCP for communication between control centers or transition to Secure ICCP.

1.2.4 Desired Response

Asset owners and technology providers should follow the recommendations in this document to understand the issues associated with the introduction of Secure ICCP and what is required to reduce the amount of time needed to achieve secure communication using ICCP-centered technology.

1.3 Scope

This document covers ICCP, Secure ICCP, the degree and type of security that can be achieved using Secure ICCP, consideration of security elements not provided by Secure ICCP, consequences of decisions that need to be made in order to use Secure ICCP, and infrastructure needed to support Secure ICCP.

1.3.1 Extent and Limits of Investigation

The investigation that resulted in the content of this document covered:

- Control system requirements, architecture, implementation, and practice;
- General requirements and practice for secure communication;
- Internet security;
- Distributed control system architecture and operation;
- Industry experience with ICCP and Secure ICCP;
- Construction and operation of an ICCP test environment;
- ICCP and Secure ICCP structure, implementation and practice;
- Transition from ICCP to Secure ICCP;
- Measurement of communication system throughput with and without various security features in place, in particular Secure ICCP.

1.3.2 Goals

The goals of this project are to shorten the time needed to implement a secure infrastructure control system based on Secure ICCP by discussing the operational impact of its introduction, how to avoid certain known near-term implementation and operational problems using Secure ICCP, and how to address vulnerabilities not covered by ICCP and/or Secure ICCP.

1.3.3 Objectives

The primary objective of this report is to investigate and provide insight to the security enhancements of the new ICCP protocol and to identify the integration impact of this emerging standard when implemented within the utility industry.

1.3.4 Organization

The report is organized as follows: Section 1 discusses background and motivation. Section 2 describes how the research was structured and performed. Section 3.1 describes the infrastructure design and its impact on the reliable and secure delivery of ICCP data. Network infrastructures and protocols associated with ICCP are assessed to identify the impact of these structures and protocols on the efficient delivery of ICCP data. The report also describes the important components that are essential in maintaining reliable and secure communications and provides modeling and simulation results of data traffic congestion on ICCP data transported over a WAN. Section 3.2 highlights the tenets of information assurance and how Secure ICCP can satisfy some of these tenets. Section 3.3 and 3.4 describe how ICCP uses Public Key Cryptography (3.3) and Certificates (3.4). Section 3.5 describes several certificate management issues and how they can be addressed. Section 3.6 discusses issues in transition from ICCP to Secure ICCP issues and strategies to overcome security limitations during the transitional phase. Section 3.7 provides some performance measurement data of the configuration and operational impacts of using security layers to provide Secure ICCP implementations. Section 4 contains the overall conclusions and section 5 is a summary of the report's recommendations. Appendix B describes the infrastructure and operation of several important information protection technologies, including public key crypography and authentication certificates

2 Approach

2.1 Methods

Two methods were used to obtain the content of this report:

1. Integrate ICCP reference material and security protocol requirements to present a comprehensive picture of the tasks and technologies involved in using Secure ICCP;
2. Construct, operate, and measure the performance of a Secure ICCP implementation.

The report presents the reader with findings, observations, and recommendations within each section. This allows the reader to easily associate shortcomings and benefits with the subject matter it applies to. Each larger section is accompanied by a summary section that provides a synopsis of the issues and recommendations that appear in its subsections.

2.2 Assumptions

This report assumes that the reader is either already running or is planning to run standard ICCP, Secure ICCP, or some combination of standard ICCP and Secure ICCP, and wishes to achieve data surety regardless.

2.3 Procedures

The premise of the research was to determine the impact of moving towards the new Secure ICCP standard on the Utilities network architecture and operations. The research was conducted to answer the following questions:

1. Implementation issues associated with architecture: What are some of the network configuration issues that need to be identified when deploying Secure ICCP?
2. What role does Quality of Service play when deploying ICCP?
3. What are the performance issues surrounding the new secure implementation?
4. What are the transition issues that need to be addressed when moving from a non-secure to a secure form of ICCP?
5. What vulnerability issues remain “after” the deployment of Secure ICCP?
6. What are some alternatives to Secure ICCP?
7. What information assurance areas need to be addressed to provide a comprehensive approach to security?

This investigation was accomplished using the following procedures:

- A test network was configured and maintained between the three participating labs Sandia National Laboratories (SNL), Idaho National Laboratory (INL), and Pacific Northwest National laboratory (PNNL) to provide the test bed needed to characterize some of the performance issues associated with the configuration and use of both the secure and non-secure forms of ICCP.
- A bench top configuration was created to capture highly granular measurement characteristics associated with the software implementation of a TLS-like process
- Industry partners were included to help identify current and near future issues associated with the introduction of Secure ICCP into the utility network computing backbone.

— This page intentionally left blank —

3 Results and Discussion

3.1 Impact of the Wide Area Network on ICCP Operations

It is important to note that an upper layer protocol such as ICCP is subject to, and dependent on, many elements out of its control. One of these elements is the network environment that it will be operating within. This section describes some general network architectures that the ICCP protocol will be deployed within and some of the operational concerns that must be addressed.

3.1.1 Overview of WAN Impact on ICCP Operations

As with many other forms of technology, the SCADA control systems continue to change as more efficient and capable technologies and protocols are defined [16]. Early SCADA systems were built around monolithic computing platforms. Each SCADA system was a standalone structure with no connections to other systems. A master controller communicated through a serial WAN interface to each subsystem via a direct connection, with each subsystem consisting of proprietary vendor environments. This serial WAN interface allowed the transfer of field data and control information to and from the Master Controller and distant Remote Telemetry Units (RTUs). This WAN interface comprised many different technologies including dial-up modem, leased line modem, radio, cable, point-to-point microwave, and satellite. The master controller comprised a single computer, normally a mainframe, that provided the system's man-machine interface and processed the information received from the RTU sites. Figure 1 depicts this architecture.

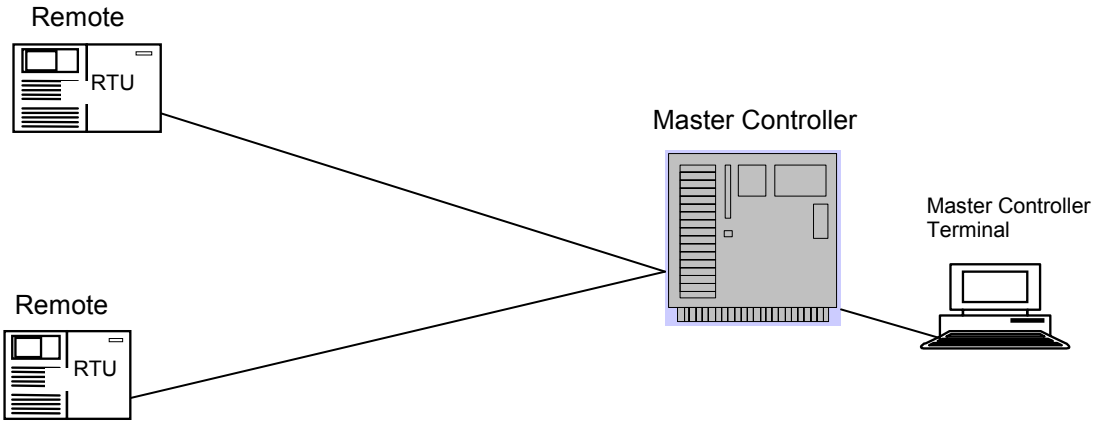

Figure 1. Monolithic Master Controller Configuration

With the introduction of Local Area Networking (LAN) technology, the SCADA environment moved from a central monolithic structure to a more distributed design. SCADA control and processing tasks were distributed across multiple processing systems. Multiple workstations, each with a specific function, were connected to a LAN and shared information with each other in real time.

Some of these distributed workstations provide communication, primarily with field devices such as RTUs. Others serve as operator interfaces, providing the human-machine interface

(HMI) for system operators. Still others serve as state or calculation processors and database servers. The distribution of individual SCADA system functions across the LAN made it easier and cheaper to add processing power than the previous single processor design.

Many SCADA systems are now using or moving toward open system architecture. Vendor-controlled proprietary systems and protocols are now being replaced with open standards and protocols allowing the distribution of SCADA functionality across LAN and the WAN. Figure 2 displays this configuration.

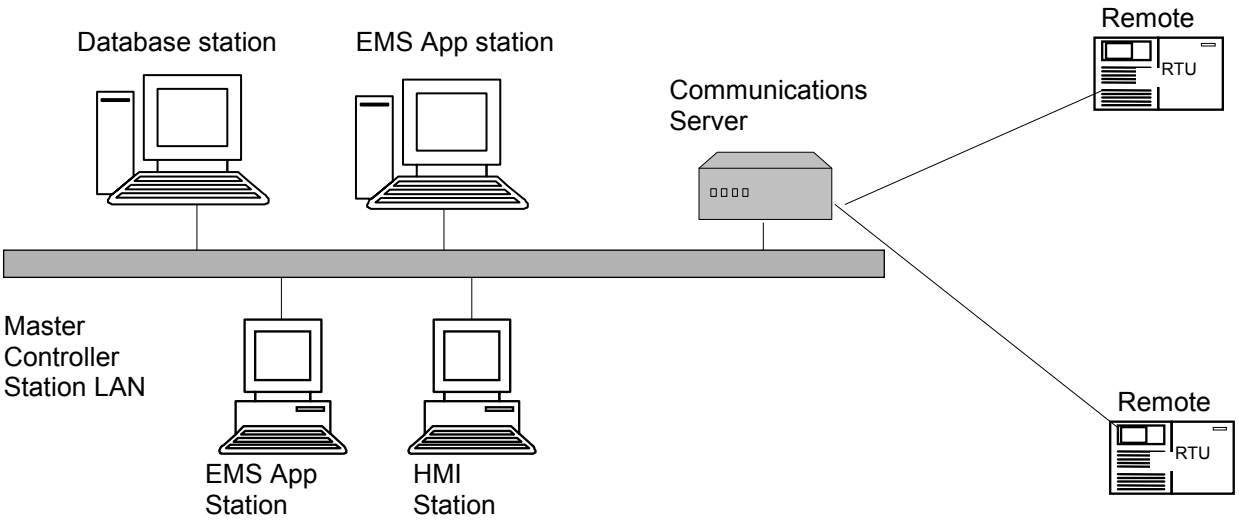

Figure 2. LAN-based Master Controller Configuration

3.1.2 Typical Configuration

ICCP allows the exchange of real-time and historical power system monitoring and control data. This includes measured values, scheduling data, energy accounting data, and operator messages.

Although, in some configurations, ICCP has been seen as a protocol interface to a substation gateway, for the most part it will be used to facilitate Control Center-to-Control Center communications to provide inter-utility data exchange between connected systems of the utility industry.

To initiate the sharing of Control Center information, a network must be in place to enable application protocols, such as ICCP, to intercommunicate. This sharing of information between Control Centers provides a means of organizing, planning, and portioning of Grid power. This is important because it allows analysis of exchange power system data between boundary control areas and enables enhanced operation between independent utility system operators. It is important that the network be able to sustain near real time communications even during times of congestion and network node failures. The design of the network is paramount to achieving and maintaining consistent and reliable communications during all hours of operations.

3.1.3 Infrastructure Design and Protection

Along with performance issues, the sharing of grid information needs to be protected from manipulation and unauthorized access.

The network architecture that will support the ICCP applications will primarily comprise an Ethernet Local Area Network (LAN) across a Wide Area Network (WAN). Because ICCP allows the sharing of control and status information between Control Centers, the actual database that will be used to exchange near real-time information should be in the form of a proxy. This will prevent direct access to the Master Controller LAN from outside users. This configuration will allow network administrators to apply security profiles to the access and extraction of internal SCADA information from the Master Controller LAN to a Control Center LAN segment. This segregation provides an additional layer of protection from external users accessing local ICCP data. Along with an internal proxy an external firewall should be maintained at the edge of the WAN to provide a filter. Figure 3 depicts this configuration.

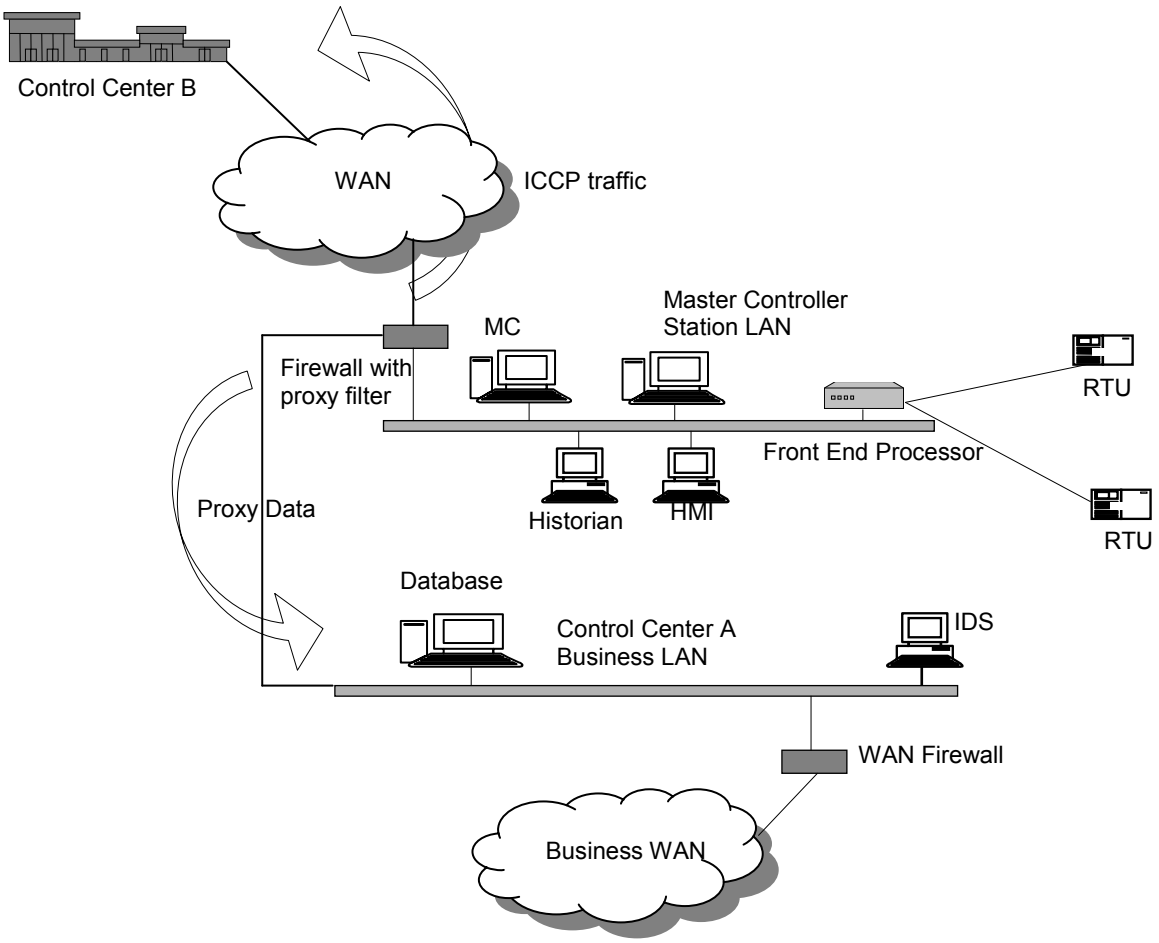

Figure 3. Control Center to Control Center Communications

3.1.3.1 Proxy Implementation Description

The purpose of a proxy server is to prevent direct access to information hosted on a critical asset within a control systems operations network. This standard approach to security has been implemented on information technology (IT) network systems for some time. There are novel approaches to providing this sort of obfuscation (see [17], e.g.), but typical company-hosted services such as Web service or e-mail service, which are advertised to the external world, use a proxy approach to prevent direct internal access to these services. The segment that supports the external advertised services is referred to as the demilitarized zone (DMZ). This terminology stresses a military approach to sharing a parcel of land where neither side fully trusts the other. In the case of an ICCP connection, the lack of trust may be associated with the connection to the WAN rather than the participating end-node utilities; the WAN must be treated as distrusted because it provides an avenue of access for many users.

3.1.3.2 Control Center Firewall Description

Another means of providing security for a network is the insertion of a firewall. A firewall is a feature setup between demarcation points of a network. It is a line of defense that allows a WAN or LAN administrator to implement a utilities security policy that associates users or end nodes with allowable access. The firewall can also be used to provide a termination point for virtual private networks (VPN) technologies that add protection mechanisms to the transported data. This access filtering can take the form of address identification, port identification or filter more deeply into aspects of the application.

3.1.3.3 Control Center IDS Description

Another important aspect of providing security for a network is the insertion of an intrusion detection system (IDS). An IDS and its associated sensors provide a means of identifying the types of data and protocols that are transported on the network. An IDS can be implemented in two primary modes, host based or network based.

A host-based IDS configuration allows for an IDS software utility to be installed on any host of choice. It is used to monitor all user interactions with the host, including user permission profiles, file manipulation, and all data received and transmitted from the host. It also monitors processes within the operating systems, including OS process calls and memory manipulation. All profile violations can be logged and reviewed.

A network-based IDS is configured and inserted onto the network. It can monitor all transactions between communicating nodes on the network. It can also monitor protocols, communication patterns and usage load to provide the network administrator a better understanding of activity on the network. The IDS can also provide signature-based identification of known virus and exploitation patterns to determine in many cases if the network is being scanned or attacked by an adversary.

Another form of network monitoring is associated with an intrusion prevention system (IPS), which, as the name implies, are designed to prevent attacks before they occur. Their technique is based upon knowing the standard usage pattern of the network and triggering defensive mechanisms that “prevent” the onset of an attack. Because an IPS must be accurately tuned to a baseline usage pattern, it can be more prone to false attack indications.

3.1.3.4 Access Control & Auditing

Another important aspect of secure communications is the need to provide a means of enforcing access level or need-to-know (NTK) authority. Access control can be implemented on individual workstations and servers or as a network level implementation such as a role-based access control (RBAC) service, which can provide a system level means of translating a "user's role" to application permission. If there is a need for remote access to the control network, then there are some common applications available that can be used to provide a means of enforcing a remote access policy.

Two popular applications are the Terminal Access Controller Access Control System (TACACS+) [18], a Cisco base product, and Remote Authentication Dial-in User Service (RADIUS), described in RFC 2865 [19] and subsequent updates. Both of these applications supply authentication, authorization, and accounting protocols for protecting access to services on the hosted network.

TACACS+ is a proprietary implementation of Cisco, Inc. and is a client/server protocol where the client takes the form of a network access server (NAS) that sends requests to and receives responses from the server. The server or servers supply the authentication, authorization and accounting services.

RADIUS is another form of access control that can be enforced for remote access security and provide authentication and authorization for who is allowed to gain access to the LAN. Simple authorization methods use a database of usernames and passwords on the terminal server or access server. More advanced authorization systems use methods such as a centralized Token card systems and Kerberos.

3.1.3.5 Virtual Network Segregation

Along with role based access control, which is administered at the application level, there is another form of NTK separation that can be implemented at the network device level. Network devices, primarily Ethernet switches, can be configured to separate user traffic by the administration of Virtual Local Area Networks (VLANs).

VLANs are defined by a switch in an internal database. After a VLAN has been created within the database, then end ports are assigned. These end ports map to end user devices or a server. A VLAN is assigned a unique number or name that is distributed by the VLAN Trucking Protocol (VTP). VTP provides the means of distributing and updating the VLAN database. If a VLAN is not known to a switch, the switch (normally an Ethernet device) cannot transfer data across any of its ports. This provides the network administrator the ability to segment users or services on a common LAN, such as one that is hosting an ICCP server, into separate VLANs. This provides a virtual separation of users that need access to sensitive information from the rest of the general users on the LAN, regardless of their physical location. This can prevent an inside user who has no need to participate in ICCP transaction from monitoring the ICCP traffic.

3.1.3.6 Server Process Lock Down

Another important aspect of securing ICCP transactions, which lies outside any direct association with the application or connection setup processes, is the disabling of

unnecessary services or ports on the ICCP server, sometimes referred to as “hardening”. Hardening makes a server more secure, and should be used along with other good security practices such as file permissions and password policies.

Every service running on a server increases the size of the attack surface for an adversary. Reducing the number of unnecessary services decreases the vulnerability of the server. The first step in hardening the server is to determine all the essential services. Services not considered essential can often be disabled without any negative effect on the operation of the server. There may also be services on a system that support media protocols and participate in remote access services that are not needed in an ICCP environment. Which services you can disable will depend on what applications and functions the server must support.

Before turning off apparently irrelevant ports and services—which is generally good security practice—note that primary services may depend on subsidiary services that seem independent but without which the primary service will not run. Some operating system companies, in particular Microsoft, have posted guidelines for determining which services are considered vital for the operating environment and which can be disabled without impacting operations. This may also help the administrator identify related service dependencies. When disabling services, disable one service at a time, review the resulting operation, and record any unexpected events.

3.1.4 SCADA Wide Area Networks

To be able to connect distant Control Centers together, multiple WAN access protocols may be deployed to facilitate IP to IP connections between the Control Centers. At the data link layer, this may take the form of a Point-to-Point Protocol (PPP) over copper or fiber optics or Frame Relay over copper or fiber-optic-based T1 or sub-T1 interfaces. The major difference between IP and PPP is that the former is normally deployed to access the Public backbone (Internet) and serviced by Independent Service Providers (ISPs). Frame Relay is used to connect to a semi-private switched network portioned manually by the telecommunications companies.

3.1.4.1 IP Routed Network

As a cost cutting measure, some smaller utilities are starting to use IP—Internet Protocol—networks for communication between participating end nodes. The IP enables source information to reach its destination by routing data packets through a network of computers and data links.

Dynamic routing protocols maintain “reachability” information for all participating end nodes so that they can be located on the network. These routing protocols are responsible for finding the most efficient route between any two end points.

3.1.4.2 IP Congestion

One primary weakness of a routing approach is that the most efficient and highly available routes will, over time, become congested. Without a means of effectively handling this congestion, communication between participating end nodes—for example, SCADA control centers—can be severely delayed or lost altogether.

ICCP uses the ISO Association Control Service Element (ACSE) to establish logical associations. Multiple associations may be established from a client to multiple, different control center servers. Although ICCP may be operated over a point-to-point link, it is envisioned that most ICCP installations will operate over a WAN that's either Frame-Relay switched or based on IP routers. ICCP is independent of the underlying transport network, so the WAN may comprise any combination of sub networks, including the LANs within a site.

Multiple associations may also be established to the *same* control center for the purpose of providing associations with different Quality of Service (QOS) attributes. An ICCP client then uses the association with the appropriate QOS for the operation to be performed. For example, to ensure real time data messages are not delayed by non-real-time data transfers, both a High and Low priority association may be established, with a separate message queue for each. ICCP will service messages on the High priority message queue before servicing the Low priority message queue. This permits a common data link to be shared for the transfer of both high-priority SCADA data and lower-priority information message transfers. This ICCP priority queuing scheme is applied only at the ICCP client and server and the QoS parameters impact only local queue processing.

The ICCP protocol, in other words, cannot compensate for network congestion. To show how network congestion might impact ICCP applications, a modeling and simulation scenario has been created to show the impact of a congested link carrying representative SCADA ICCP traffic. A communications modeling and simulation software package from Opnet, Inc. is utilized to model the IP communications stack and the routing protocol found on each router. Figure 4 shows a simulation scenario that represents some important features of an IP-routed network used to transport SCADA control center information across the IP WAN.

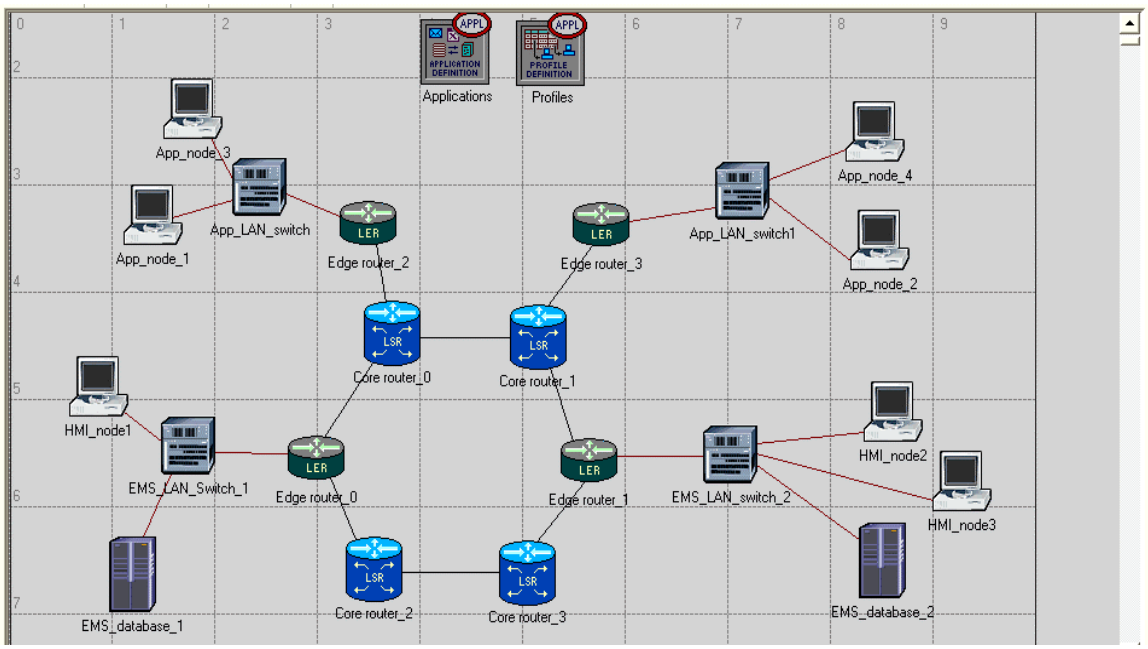

Figure 4. Simulated IP Routed Network

The scenario consists of two primary ICCP client workstations labeled HMI_node2 and HMI_node3 that extract database information from the SCADA database server labeled EMS_database_1. Representing other nodes competing with the SCADA data flow traffic across the WAN are a set of workstations labeled App_node_1, App_node_2, App_node_3, and App_node_4 created to participate in video teleconferencing sessions. At a predetermined time, workstation App_node_1 will set up a video teleconference with workstation App_node_2, while workstation App_node_3 will create a videoconference with workstation App_node_4.

The SCADA ICCP traffic is initiated with both client workstations extracting database information from the distant server database. Open Shortest Path First (OSPF), a standard link-state routing protocol is being used in this case. OPSF finds the most efficient route for the client data query and response to be through a set of core routers labeled Core_router_0 and Core_router_1. This route is chosen because the link between these core routers is a T1 line with a nominal capacity of 1.5 Mbit/sec. The other two core routers, Core_router_2 and Core_router_3 in the figure, have a DS0 interconnected link with throughput of only 64 kbit/sec. The IP routing protocol selects the most efficient routes, normally those with larger link capacity and/or minimal node hops between the communicating nodes. The routing protocol is unaware of any congestion that may be occurring in the network.

Although the interconnected core link rates do not represent the actual link rates of larger core networks, it accurately represents the route selection of data flows and the impact of data flow aggregation that can result in network congestion. As seen in the generated modeling statistics in Figure 5, the HMI workstation queries to the EMS database are not initially hindered by any network congestion. Their flows, approximately 150 kbit/sec, go through the primary core link unabated. Then the video conferencing applications are brought on at different intervals. Since the primary link between the two core routers is a T1 with a bandwidth of 1.5 Mbit/sec, the SCADA applications are unaffected as long as the aggregation of all data flows doesn't exceed the total amount of available link bandwidth. As soon as the second video-conferencing application is brought on-line, about 18 minutes into the simulation, the SCADA applications (light blue and yellow) are severely hampered, as shown in Figure 5.

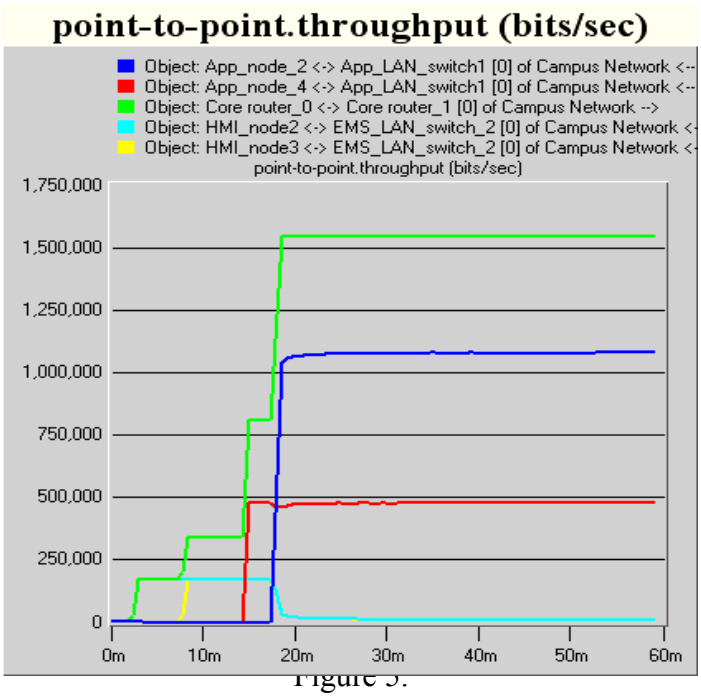

Figure 5. Data Flow Statistics on IP Routed Network

The ICCP SCADA applications are represented by yellow and light blue. It can be seen that the first node, labeled HMI_node2 workstation, comes on-line at approximately 3 minutes into the simulation. The second node, labeled HMI_node3 workstation, comes online at approximately 7 minutes into the network simulation. The total aggregate bandwidth at this time (represented in green) shows the combined throughput of the two HMI workstations to be approximately 200 kbit/sec. The first videoconference application, in red, comes online at 15 minutes into the simulation. The total aggregate rate at this time is approximately 800 kbit/sec, well within the sustainable link rate of 1.5 Mbit/seconds. As the simulation progresses, the second teleconferencing application, in dark blue, comes on-line 18 minutes into the simulation. At this point the total aggregate bandwidth exceeds that of the supporting T1 rate. Both of the client workstation data flows are now reduced to approximately zero throughput. This represents a dire situation if any near real-time SCADA information must be transferred. The reason the client data flows are subjected to such reduction is because of the difference in the transport protocol that is being used to transport the higher layer applications. The SCADA applications are using the Transport Control Protocol (TCP) which has a built in congestion avoidance mechanism that reduces the amount of data that is sent out when it senses loss of IP packets due to congestion. This mechanism prevents the TCP flows from competing with the video conferencing applications, which are using the User Datagram Protocol (UDP). The User Datagram Protocol does not implement any congestion avoidance mechanism and thus continues to grab all the available link bandwidth.

3.1.5 IP Congestion and QoS management

To be able to guarantee the service level of near real-time ICCP applications, it is important to implement a means of assuring the quality of service, usually abbreviated QoS. QoS assurance is important to prevent the denial of service that can be caused by competing network traffic, as shown above in Figure 5 (in section 3.1.4.2, *IP Congestion*).

To keep operating expenditures down, many business-critical applications using Layer 2 services (e.g., Frame Relay and ATM) are migrating to the IP network infrastructure. This eliminates the need to maintain several physical networks, but presents a challenge, in that both new and legacy services usually require strict QoS guarantees.

QoS guarantees are usually implemented in the form of *Service Level Agreements* (SLAs). An SLA defines, in terms of jitter, latency, bandwidth guarantees, and redundant route selection, the required service quality for traffic transiting the network. SLA requirements specify traffic scheduling, queuing, and drop behavior based on the application type and bandwidth guarantees on a per-application basis. See [20] for a thorough discussion of QoS.

Differentiated services (DiffServ), defined in RFC 2474 [21] and subsequent updates, and Multi-Protocol Label Switching (MPLS), described in RFC 2702 [22], are two separate standards that can help address the IP quality of service (QoS) problem. Diffserv uses the IP Type Of Service (TOS) field to carry information about IP packet service requirements. It operates at Layer 3 only and does not deal with lower layers. Diffserv relies on traffic conditioners sitting at the edge of the network to indicate each packet's requirements. MPLS creates Label Switch Paths (LSPs) that request and then reserve necessary bandwidth. The network is made capable of supporting this session setup and reservation by deployment of Label Switched Routers (LSRs).

3.1.5.1 Differentiated Services

DiffServ approaches the problem of QoS by assigning traffic flows to a small number of classes and allocating network resources on a per-class basis. The class is identified by providing a mark directly on the packet in the 6-bit DiffServ Code Point (DSCP) field. The DSCP field is part of the original type of service (ToS) field in the IP header

The DSCP determines the QoS behavior of a packet on each node in the network. This is called the per-hop behavior (PHB) and is expressed in terms of the scheduling and drop preference for each marked packet. The PHB is defined by a packet queue used for forwarding. The packet queue defines the drop probability of a packet flow when the queue exceeds a threshold limit, the resources (buffers and bandwidth) allocated to each queue, and the frequency at which a queue is accessed.

To show how a QoS scheme such as differentiated services can provide some bandwidth guarantees for ICCP applications, an IP network was recreated using a Weighted Fair Queue (WFQ) implementation. Two flow identifications were created, one was used to implement a Best Effort (BE) queue, which provided no packet guarantees, and the second was using an Expedited Forwarding (EF) queuing scheme, which had bandwidth assignment guarantees. The BE queue was assigned to the video conferencing applications and the EF queue was assigned to the ICCP applications. Figure 6 shows the result of the QoS bandwidth allocation for the ICCP application flow when the core link became congested.

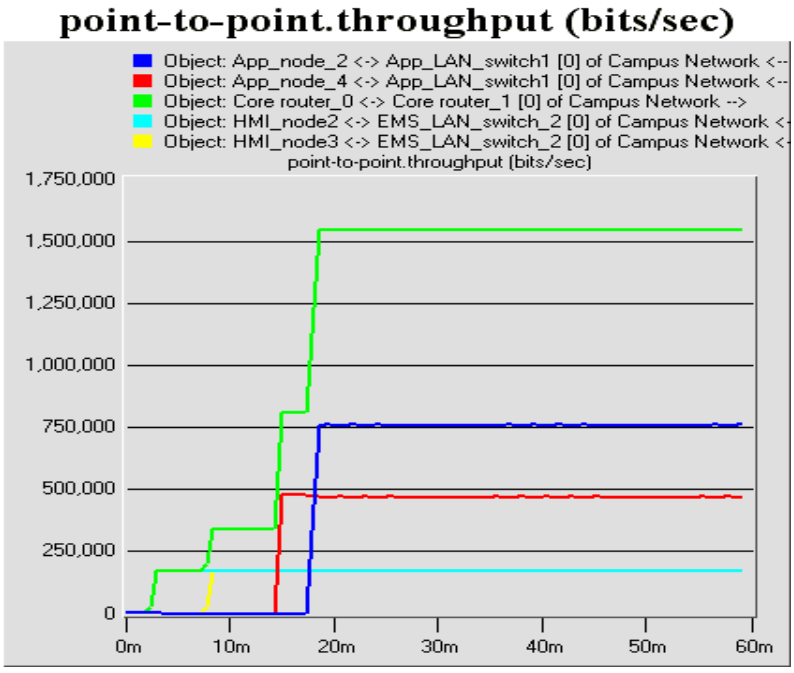

Figure 6. Data flow on IP Network with QoS Statistics

As can be seen in Figure 6, the ICCP applications start at approximately 3 minutes and 7 minutes into the simulation. They each consume about 100 kbit/sec bandwidth during their database querying routines. At approximately 15 minutes into the simulation, the first video conferencing application comes on-line. Because there is still plenty of link bandwidth

available, both applications co-exist without hindering each other. Then at approximately 17 minutes into the simulation, the second video conferencing application comes on-line. This second application flow originally consumed over 1 Mbit/sec of bandwidth, grabbing all the allocation from each ICCP application, as was shown in the previous simulation results (Figure 5). Because a QoS scheme has been implemented on the network, this video conferencing application has been reduced to 750 kbit/sec allowing the ICCP apps to maintain their required bandwidth.

The differentiated services QoS scheme provides differential forwarding treatment to traffic, thus enforcing QoS for different traffic flows. It is a scalable solution that does not require per flow signaling and state maintenance in the core. However, it cannot guarantee QoS if the path followed by the traffic does not have adequate resources to meet the QoS requirements. Another QoS scheme that provides a means of requesting and reserving bandwidth can be implemented. This approach is called Multiple Protocol Label Switching (MPLS).

3.1.5.2 Multiple Protocol Label Switching

MPLS specifies ways that Layer 3 traffic can be mapped to connection-oriented Layer 2 transports like PPP, ATM and Frame Relay. MPLS adds a label to the header of the Layer 2 transport protocols that represents specific routing information used to forward each IP packet and allows routers to assign explicit paths to various classes of traffic. It also offers traffic engineering that can improve IP routing efficiency.

MPLS traffic engineering (TE) enables resource reservation, fault-tolerance, and optimization of transmission resources. MPLS DiffServ-TE combines the advantages of both DiffServ and TE. The result is the ability to give strict QoS guarantees while optimizing use of network resources. The QoS delivered by MPLS DiffServ-TE allows network operators to provide services that require strict real-time and near real-time performance guarantees and to consolidate IP and ATM/FR networks into a common core.

Traffic engineering is used to achieve optimization of network resources by identifying and directing flow direction of traffic on particular links within the network. MPLS accomplishes this by computing a path from source to destination that is constrained by a set of requirements and forwarding traffic along this path called a label switched path (LSP). Traditionally IP networks do not use Layer 2 forwarding techniques to forward traffic along such a path. An IP forwarding decision is made independently at each hop by a route look-up and is based solely on the packet's IP destination address. The explicit routing capabilities of MPLS allow the originator of the LSP to do the path computation, establish MPLS forwarding state along the path, and map packets into that LSP. Once a packet is mapped onto an LSP, forwarding is done based on the label, and none of the intermediate hops make any independent forwarding decisions based on the packet's IP destination.

MPLS-TE introduces the concept of LSP priorities. The purpose of priorities is to mark some LSPs as more important than others and to allow them to preempt resources from less important LSPs. If high-priority LSPs do not exist along a path, resources may be reserved by less important LSPs. High-priority LSPs are established along the most optimal path regardless of any existing reservations of lower priority LSPs. And during times of link failures, when LSPs need to reroute, high-priority LSPs have a better chance of finding an

alternate path. MPLS-TE defines eight priority levels that are used for LSP assignments and path calculations. To perform path calculations, relevant link properties have to be advertised throughout the network. This is achieved by adding TE-specific extensions to the link-state protocols IS-IS and OSPF, extensions that allow them to advertise not just the availability state (up/down) of the links but also the link's near real time attributes such as available bandwidth and packet latency. This mechanism allows each node to obtain knowledge of the current properties of all the links in the network.

A simulation to show the QoS advantages of using MPLS was not conducted because of the lack of a licensed MPLS module for the OPNET simulator. Information presented about MPLS provides the SCADA network implementer additional information on QoS alternatives when pursuing choices for implementing a QoS scheme for inter-utility SCADA applications.

3.1.6 Frame Relay Switched Network

Frame Relay is a popular Wide Area Network (WAN) protocol that is used by some utilities to enable communications between network end-nodes. Telco carriers build and partition frame relay networks using frame relay switches that form frame relay switched networks. The interior network which can be built on high-speed technologies such as T3, Sonet and/or ATM, is hidden from the customer who normally is required only to furnish the access interface device called a Frame Relay Access Device (FRAD), which typically has a built-in Customer Service Unit/Data Service Unit CSU/DSU to interface directly to the carrier network.

Frame relay network allocation is built upon permanent virtual circuits (PVCs). These circuits are established by developing a Service Level Agreement (SLA) contract with the carrier and typically are built on a flat-rate basis with port speed being the most costly parameter. Each access point onto the frame relay network is assigned a Data Link Connection Identifier (DLCI), which allows the frame relay switches to forward each frame to it proper destination.

The following parameters can be assigned for each PVC:

Access Rate	The rate at which the customer access nodes join the frame relay network. These are typically 56 kbit/second or fractional T1 which is a multiple of 56 kbit/second or 64 kbit/second.
Committed information rate (CIR)	The amount of data per unit time that the network will receive from the access circuit.
Committed Burst Size	The maximum amount of data that the network will transfer in a burst defined over a short interval.
Excess Burst Size	The amount of data above the committed burst size that the network will try to deliver. Frames delivered at this level may be marked as "discard eligible" (DE) and will be dropped if there is not enough bandwidth capacity on a link.
Oversubscription	An instance where the sum of CIRs exceeds the capacity of the port or access channel rate.

3.1.6.1 Frame Relay Network

A Frame Relay interface is used to multiplex traffic onto a carrier's backbone. It's important to note that the Telco's backbone is shared by many users and possibly multiple services. To prevent customers from sending more data than the network can hold, frames sent above a contracted rate can be marked at the ingress of the provider's network as Discard Eligible (DE). DE-marked-frames received from the carrier network indicate that data being sent at the current rate from the user in the future may get dropped. This provides an indication that there is congestion in the network and frames originated above the Committed Information Rate (CIR) will be discarded. DE frames being received by user interface equipment may be an early indicator of poor traffic rate planning in the design of the frame relay WAN [23].

Access from a local site is provided through a Local Exchange Carrier (LEC). Each LEC has interface access to a Local Access Transport Area (LATA) which provides access to the frame relay backbone. Links between LATAs are provided by an inter-exchange carrier. In some cases the inter-exchange carrier is a different company than the LEC. It is possible that a point-to-point connection between two different utilities may involve two or three different vendors. Figure 7 shows a typical point-to-point interface to a Frame Relay network.

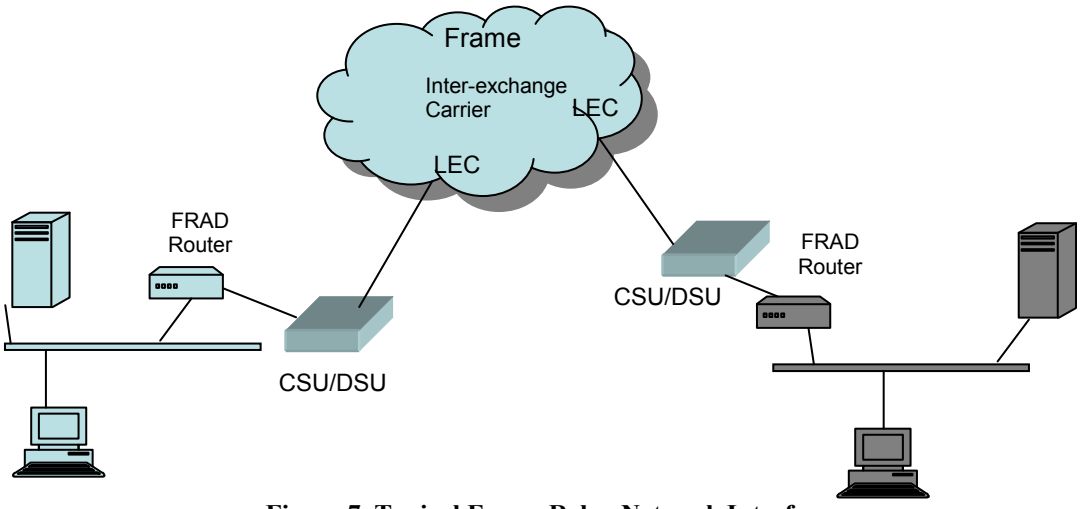

Figure 7. Typical Frame Relay Network Interface

Frame Relay networks can consist of a mesh or partial mesh design. This allows for graceful failover during link failures associated with the carriers interconnected network. But it is important to note, unless FRADs and carrier backbone switches have a coupled prioritization scheme interacting fully to share information about traffic priorities, the ability to prioritize traffic to ensure transmission of time-critical data such as ICCP cannot be maintained.

Most FRADs align with RFC 2427 [24], which supports multiple traffic types over one integrated network. This allows network managers to take advantage of frame relay's convergence technique to lower costs and provide efficient bandwidth usage. But simply supporting the Frame Relay standard doesn't guarantee traffic management or quality of service guarantees. The main purpose of RFC 2427 is multi-vendor compatibility with a frame relay encapsulation method.

Each Inter-Exchange Carrier has its own service options that cover aspects of network design and management, such as route diversity, network management, and installation support, to manage equipment such as FRADs and Channel Service Unit/Data Service Units (CSU)/DSU and to facilitate disaster recovery.

Reliability is central to the utility company. It is important to minimize the impact of service disruptions when accessing and using a Frame Relay network. Recovery options and access protection need to be associated with any key ICCP nodes. Although some carriers support automatic recovery architectures, they may be reliant on external inter-exchange carriers to provide some of the back-haul recovery circuits. This is normally the case when a carrier does not have enough switches in its networks capable of supporting multiple recovery paths, inherently an unreliable situation.

3.1.6.2 Frame Relay Congestion

Frame Relay equipment notices congestion when it sees frames marked with the Forward Explicit Congestion Notification (FECN) and Backward Explicit Congestion (BECN) bits. These merely indicate an overload within the carrier network, and are of value only in monitoring the carrier's health. Frame Relay equipment *does not notify* end stations to stop sending data to keep additional frames from being discarded or causing additional congestion on the network. Higher layer congestion-sensitive protocols, such as TCP/IP, are expected to react implicitly to the packet loss.

The ICCP protocol which relies on TCP for its transport protocol will not be able to maintain data flow rates during times of congestion on a Frame Relay network. To show how Frame Relay network congestion might impact ICCP applications a modeling and simulation scenario has been created to show the impact of a congested link that contains some represented SCADA ICCP traffic. A communications Modeling and Simulation software package from Opnet, Inc. is utilized to model the communications and the PVCs on each Frame Relay access device. Figure 8 shows a simulation scenario that represents some important features of a Frame Relay network that is used to transport SCADA control center information across a Frame Relay WAN.

The scenario consists of two HMI workstations, labeled HMI_node2 and HMI_node3, that extract database information from the SCADA database server labeled EMS_database_1. To represent other nodes competing with the SCADA data flows, a set of workstations participating in some video teleconferencing has been created. At a predetermined time, workstation App_node_1 will set up a video teleconference with workstation App_node_2, while workstation App_node_3 will create a video conference with workstation App_node_4.

The SCADA ICCP traffic is initiated first with both HMI workstations extracting database information from the distant EMS database, and then both pair of video conferencing applications are brought on line. To provide the end-node to end-node connectivity, two Frame Relay PVCs have been created, the first originates from access point DSU_CUS_node_1 and terminates at DSU_CSU_node_3. The second originates at access point DSU_CSU_node_2 and terminates at DSU_CSU_node_4. A common core link is represented by the three interconnected Frame Relay core switches FR_node_0 and

FR_node_3, and FR_node_4. The link between the cores is a T1 which represents a 1.5 Mbit/sec data rate. Although the interconnected core link rates do not represent actual link rates that may be found in larger core networks, it still accurately represents the route selection of data flows and the impact of data flow aggregation that may result from network congestion.

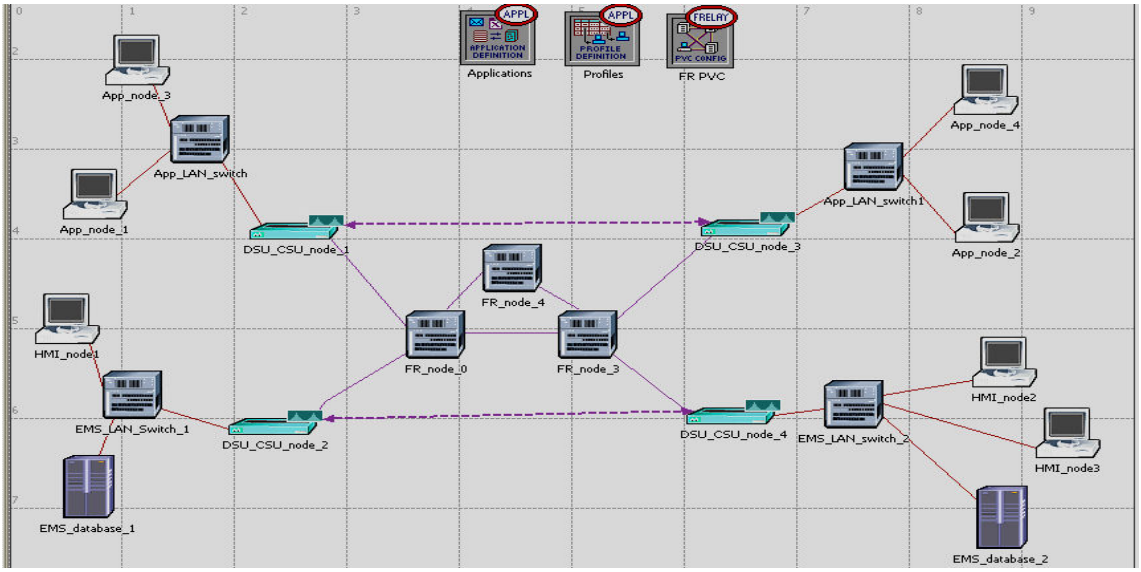

Figure 8. Simulated Frame Relay Network

As seen in the generated modeling statistics shown in Figure 9, the HMI workstation queries to the EMS database are not initially hindered by any network congestion. Their flows at approximately 100 kbit/sec go through the primary core link unabated. Then the first video conferencing application is brought on line. Since the primary link between the two core routers represents a T1 that has 1.5 Mbit/sec available bandwidth, the SCADA applications are unaffected as long as the aggregation of all data flows doesn't exceed the total amount of available link bandwidth. As soon as the second video conferencing application is brought on line about 15 minutes into the simulation, the T1 aggregate rate is exceeded and the SCADA ICCP applications (red and green) are severely hampered.

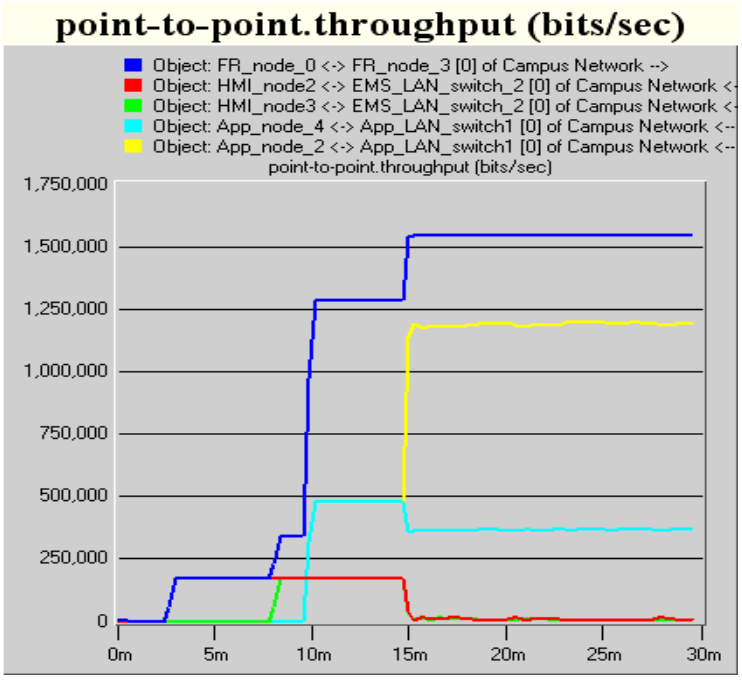

Figure 9. Data Flow Statistics on Frame Relay Network

3.1.6.3 Frame Relay Congestion and QoS Management

As mentioned in the previous section 3.1.5, *IP Congestion and QoS management*, Service Level Agreements define the quality of service experienced by traffic transiting the network and are expressed in terms of jitter, latency, bandwidth guarantees, and redundant route selection. The SLA requirements use traffic scheduling, queuing, drop behavior based on the application type; and bandwidth guarantees on a per-application basis.

To demonstrate the importance of proper QoS management, the two PVCs created in the Frame Relay scenario were assigned different operational characteristics based on the priority of the traffic flows. The video conferencing applications were assigned to the PVC with the lower priority SLA contract parameters, while the ICCP HMI applications were assigned to the PVC with a higher priority SLA contract parameters. The primary parameters included the committed information rate, committed burst size, and excess burst size. The simulation results are seen in Figure 10.

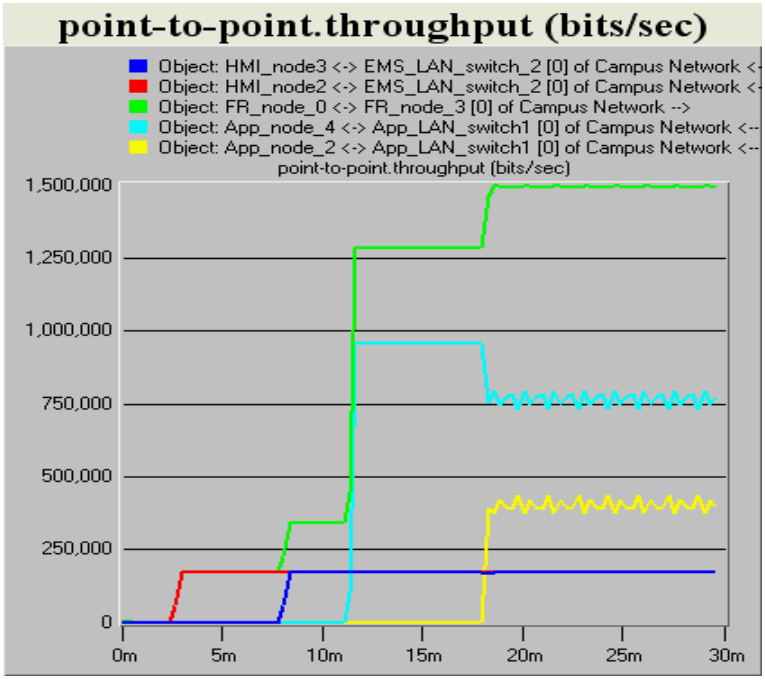

Figure 10. Data Flow on Frame Relay Network with QoS Statistics

The ICCP applications start at approximately 3 minutes into the simulation for HMI_node2 (red) and 7 minutes for HMI_node3 (blue). Each consumes about 150 kbit/sec bandwidth during its database-querying routine as shown in the aggregate rate FR_node_0, FR_node_3 (green). At approximately 12 minutes into the simulation, the first video conferencing application comes on line, App_node_4 (light blue). Because there is still plenty of link bandwidth available, both applications co-exist without hindering each other. Then at approximately 18 minutes into the simulation the second video conferencing application comes on line, App_node_2 (yellow). This second application flow originally consumed over 1 Mbit/sec of bandwidth, grabbing all the allocation from each ICCP application, as was shown in the previous simulation results (Figure 5). Because a QoS scheme has been implemented on the network, this video conferencing application has been reduced to 450 kbit/sec thereby allowing the ICCP applications to maintain their required bandwidth.

3.1.7 Network Impact Summary and Recommendations

The following bulleted statements are important observations and recommendations based on information contained in section 3.1, *Impact*.

Observation summary for Section 3.1.3, *Infrastructure Design and Protection*

- SCADA architectures are migrating from standalone monolithic proprietary architectures to distributed open system LAN and WAN structures.
- Security becomes paramount when isolated, proprietary systems are transformed into more open architectures using standard communication protocols

Recommendation summary:

A structured line of protection needs to be implemented to allow the sharing of information without a compromise of a system. To accomplish this goal, each participating ICCP data exchange node should implement the following structure:

- Master control LAN to business LAN segregation
- Business LAN to WAN filtering
- End node to End node authentication
- Configuration Management

Observation summary for Section 3.1.4, *SCADA Wide Area Networks*, and 3.1.5, *IP Congestion and QoS management*:

- Although ICCP has configurable priority QoS for identified data streams, it is administered only locally and only at the application level and is not an end-to-end system implementation.
- IP route selection algorithms can, over time, create bottlenecks within IP networks.
- QoS guarantees can be administered across multiple layers of the protocol stack including Layer 2 (data-link layer) and Layer 3 (network layer).

Recommendation summary:

This recommendation highlights Layer 3 for the IP network. If the WAN connection is comprised of an IP routed service, verify a system approach to QoS will be implemented. This will prevent a Denial of Service situation during peak usage times on the WAN.

A systems approach in this case includes multiple IP solutions to QoS management. The two most popular approaches are differentiated services and Multiple Protocol Label Switching (MPLS). The primary differences in these approaches are the granularity that can be applied to data streams and the efficiency of transport. For a coarse granularity when multiple flows can be grouped together and share the same QoS assignment, a differentiated service is sufficient. When individual flows need to be isolated for more specific QoS handling, use MPLS. As for efficiency, MPLS implementations are designed to be switched at Layer two reducing the need for route information look-up, thus increasing the efficiency of the hop-by-hop transport mechanism although with modern networking equipment design, where each router port is assigned a separate processor for processing, the efficiency difference becomes less a factor.

- Review the lower layers in the communication protocol stack to determine which QoS is most useful and appropriate.
- Review the WAN carriers approach to guarantee service and testing.
- Identify the ICCP streams that need an end-to-end guaranteed responsive service.

Observation summary for section 3.1.6, *Frame Relay Switched Network*:

- Frame Relay is a popular WAN protocol used by the Utilities to connect LAN segments.
- Mesh and partial mesh designs allow for graceful failover but do not guarantee available bandwidth
- QoS guarantees can be administered across multiple layers of the protocol stack. Layer 2 (data-link layer) or Layer 3 (network layer). *The following highlights the data-link (Frame Relay) layer.*

Recommendation summary:

If Frame Relay is the choice for WAN protocol, implement a system approach to QoS. This will prevent a Denial of Service during peak usage times on the WAN. A system approach in this case includes fail-over protection that does not rely on back haul lines portioned by multiple vendors. It also includes a QoS scheme that includes ingress-to-egress PVC identification to provide end-to-end prioritization information to ensure the ability to prioritize and maintain time-critical circuits. Development of a Service Level Agreement with the WAN provider to identify the specifics of the guarantee is paramount.

3.2 Secure ICCP and Information Assurance

The term *Information Assurance* (IA) refers to the ability of a system to protect the availability, confidentiality, integrity, reliability, and authenticity of the data. Using Secure ICCP affords some IA. The following describes the aspects of IA that are realized by Secure ICCP and what areas are not addressed. For areas not addressed by the ICCP, implementation guidance and design are presented.

3.2.1 Overcoming physical layer availability disruptions

Data is *available* if it is accessible to an authorized user. If an authorized user cannot access data specified to be available, the data is *unavailable*. Data unavailability can be induced on the physical or logical plane. Physical plane unavailability can be caused by any of several physical means that can be used to prevent timely delivery of data, such as the failure of critical network components, power disruptions, physical plant disruptions, either malicious or natural. Physical plane unavailability is not associated with ICCP protocol.

Although the ICCP protocol itself affords no means of protecting against physical failures, the network architecture and the computer systems that support ICCP can provide some measures to protect against physical failures. For example, the network equipment that makes up the infrastructure can take advantage of dual power supply setups that allow a secondary system to overcome the failures of a primary. Also, at critical network nodes, redundant configurations associated with routing and switching can create primary and secondary devices that can auto-sense when there has been a power disruption or an operating system

malfunction and provide automatic failover. Emergency and contingency planning is critical to provide operational guidelines for continuity of operations. Collectively, plans of this sort allow owners and operators to review all phases of contingent operations and identify dependencies that need to be addressed.

3.2.2 Overcoming logical layer availability disruptions

The ICCP protocol alone cannot protect itself against logical layer disruptions that impact the availability of ICCP data. The ICCP protocol resides primarily in the upper layers of the communication stack. The logical layer refers to all of the communication software processes that reside on computers and intermediate network nodes that are responsible for the end-to-end delivery of data. The primary logical disruption that can impact the flow of ICCP data is Denial of Service (DoS). A denial of service can be created either maliciously by an adversary that launches an active attack against an ICCP server by for example, requesting a large number of TCP connection requests that exceed the ability of the server to process each request. Or it may naturally occur on the network due to excessive amounts of competing network traffic that causes network node buffers to become clogged and ICCP packets to be discarded.

Because ICCP implements Secure Socket Layer (SSL) protection for its in-transit data, it must service each TCP-generated connection request prior to validating the source. This is because SSL and TLS are both forms of transport layer encryption and are processed by the transport layer prior to its invocation. Since this is the case, each TCP connection request is serviced prior to validating its source and thus is vulnerable to TCP connection request DoS. One means of overcoming this type of attack is to implement a network layer Virtual Private Network (VPN) between the edge of the network that contains the server and the distant connecting client or clients. This will prevent unauthorized connections to the ICCP server. Another approach is to configure a firewall between the ICCP server and the external network and allow only connections based on attributes of the source connection, such as its IP address, to be allowed to pass through the firewall. These methods can substantially reduce the effects of external adversarial DoS attack against the ICCP server.

There is another type of DoS that is not the result of an active adversary but can be just as effective in preventing timely communications with the ICCP server. This DoS is evident when the network carrying the ICCP data becomes congested. Network congestion can have as debilitating an effect on end to end message traffic as any deliberate adversary attack. Identifying priority ICCP connections along with proper Quality of Service (QoS) portioning can help alleviate this vulnerability. See sections 3.1.5, *IP Congestion and QoS management* , and 3.1.6, *Frame Relay Switched Network*, for a discussion on this topic.

Integrity of information refers to the ability of a system or mechanism to detect changes or modifications to a message. Modern techniques implement integrity across a header and/or data field of an IP packet by creating a hash across the contents of the packet. This hash is based on a one-way function and enables detection of virtually any modification[3] to the

[3] The hash function is applied to the message prior to transmission and the resulting hash is sent along with the message. If the same hash function produces a different hash when applied to the message after transmission, the received message and the transmitted message cannot be the same. It is possible for different messages to

original data. If proper integrity is not implemented a form of attack called *man-in-the-middle* can be implemented.

Secure ICCP implements data integrity indirectly by providing a cryptographic checksum. The checksum can logically determine if any part of the payload has been modified or tampered. The data integrity of ICCP data is dependent on the proper implementation of the encryption process.

Data is *confidential* if only authorized parties can read it. Most implementations of confidentiality rely on some form of encryption to prevent the disclosure of the information while the data is "in flight" to its destination.

Secure ICCP provides data confidentiality by encrypting ICCP data exchanges. However, Secure ICCP encryption occurs at the application layer, so it does not provide confidentiality for lower layer protocol information such as port assignments or addressing of ICCP data. For example, an adversary capturing network packets (snooping) relies on a tool called a network analyzer, which uses the standard protocol fields available in the different layers of the communication protocol stack to help in arranging, decoding, and cataloging the information in the captured packets. This protocol information can provide information about end node participation in ICCP sessions, and Secure ICCP doesn't protect it. See section 3.6, *Strategy for the transition from ICCP to Secure ICCP*.

A communication system is *reliable* if it provides intended service a large percentage of the time. The reliability of a network depends on the interconnected network components of the system and the protocols used to provide end-host-to-end-host communication. Communication protocols can improve the reliability of the data communications process. For example, a somewhat noisy network link creating bit errors within a packet will not by itself prevent communication between two end nodes if the communications protocol is able to detect the errors and retransmit the affected packets. The packet communication process can thus remain reliable in spite of bit errors injected by the network link.

Although the ICCP protocol does not itself provide reliable transport of data, it is supported within the implementation of RFC 1006 [25], ISO Transport Services on top of Transport Control Protocol (TCP). TCP provides a data transport reliability service for its encapsulated ICCP payload.

Data is *authentic* if its apparent source and its actual source are the same. Maintaining the relationship of a datum and its associated source in modern network communications is done with the use of public key encryption and a digital signature. A digital signature is a hash[4] created from the datum. For a digital signature, the process used to create the hash is one way and cryptographically strong. A hash created this way is thought of as a signature because it's unique[5] to the original contents of the message. These bits are encrypted with the private key

produce the same hash, but the hash function is chosen so that messages other than the original that generate the same hash will differ from the original to such an extent they will be readily detectable as non-messages.

[4] A *hash code* (or, colloquially, just *hash*) is a bit string, customarily much shorter than the datum itself, generated by applying a mathematical formula to a datum. A hash is also referred to as a *message digest*.

[5] Technically, two different data can generate the same hash, but with extremely small probability.

of the author and sent along with the original message. The recipient is then able to verify the message content by decrypting the message digest with the author's public key and comparing this output with the output of the received message's hash.

The secure version of ICCP has the ability to provide authenticity of data.

3.3 ICCP Use of Public Key Cryptography

One of the tenets of security provided by Secure ICCP is end node authentication. ICCP interactions can be compromised by participants who have not been authenticated. ICCP end node authentication relies on public key cryptography and its underlying Public Key Infrastructure (PKI). See sections 1, *Public Key Cryptography*, and 2, *Public Key Infrastructure*, in *Appendix B: Security Technology* for a description of the mechanisms associated with a public key infrastructure.

3.4 ICCP Use of Public Key Infrastructure Certificates

Public key certificates are used heavily in the Secure Sockets Layer (SSL) protocol and will be an important part of Secure ICCP integration. See sections 3, *Certificates in the Secure Sockets Layer (SSL)*, and 5, *Certificate Management*, in Appendix B: Security Technology for a discussion of the certificate management infrastructure.

3.4.1 PKI Certificate Hierarchy Recommendations for ICCP Networks

Each certification hierarchy has its advantages and disadvantages, and each network is different. See section 1,

Certification Hierarchy Schemes, in Appendix B: Security Technology, for a discussion of the pros and cons. Sometimes, a network is small or self-contained, and the flat hierarchy makes the most sense. Likewise, some networks are so broad and diverse that multiple layers of certification are necessary to amortize the cost of providing PKI services.

The data networks of control systems sharing ICCP data are somewhat isolated and generally small (hundred of nodes, instead of thousands). As such, they lend themselves to a relatively flat Certificate Authority (CA). The advantage of using a hierarchical CA approach is that only one CA hierarchy needs to be established for everyone on the network, reducing the complexity of the configuration. In a tiered approach, each company would maintain its own CA, a proposition that is likely cost-prohibitive and more managerially complex. Since the network is small, one CA could service the entire network while still remaining adequately quick about revocation. It is important to note that the recommendations of a flat hierarchy are with respect to identities associated within a single operational domain or network. This concept will be reanalyzed when there is a need or requirement to communicate between operations domains. See section 3.5.5.1, *Creating ICCP CA Boundaries*, for a discussion of external domain node authentication.

3.5 Secure ICCP Certificate Management Issues

Section 5, *Certificate Management*, in Appendix B: Security Technology, describes certificate management in a typical system using SSL to secure communications. However, some plans for using SSL to Secure ICCP deviate from the typical scenario and introduce significant certificate management problems. Secure ICCP applications use PKI to establish and protect communication channels. Specifically, they use PKI in the SSL tunnels that protect the ICCP traffic and the MMS layer that authenticates ICCP nodes. In this section we examine some of the non-typical uses of certificates and SSL and provide appropriate analysis and recommendations.

3.5.1 Number of Certificates per ICCP Node

One issue that needs to be addressed in a Secure ICCP infrastructure is how many public key pairs, and consequently how many certificates, each node should have. The most basic approach is to give each node a single certificate that is used for multiple purposes. For instance, a single certificate can be used by SSL to secure the network connections and by MMS to secure the ICCP transactions.

However, it is considered best practice to use a certificate for only a single purpose. The classic example is to have one certificate for public key encryption and a different certificate for public key signatures. Having multiple certificates provides more robust security at minimal cost (The cost of managing several certificates at a node is only marginally greater than the cost of managing one). Based on this commonly accepted practice, we recommend using distinct certificates for different purposes, such as one for SSL, one for MMS, etc.

3.5.2 ICCP Security Policy and Certificates

3.5.2.1 ICCP Certificate Update Notifications

Certificates in SSL are exchanged during the SSL handshake at the beginning of every session and are used only temporarily before being discarded, so there is no need to keep track of anyone's certificates. For normal SSL function, nodes do not need to be alerted or updated when another node receives a new certificate. Conversely, ICCP templates are normally configured manually and mapped to specific end nodes participating in an ICCP connection. Other participating nodes need to be alerted when a participating node receives a new certificate.

Having a node alert other nodes when it gets a new certificate is not standard practice in PKI systems because it is costly and usually unnecessary. However, it has been suggested that there may be some application-level policy decisions that require nodes to keep track of other nodes' certificates. For example, it appears that certificates are being mapped to security policy configurations (ICCP system stack File). If that is the case, decisions about access control and policy are decided based on a client's certificate, not the client's identity. This approach is fundamentally flawed and should be avoided.

As stated earlier, policy decisions should not be mapped to certificates, but instead should be mapped to the identity attested to by the certificate. Each entity that has a certificate should have a globally unique distinguished name, and privileges should be mapped to that Distinquished Name (DN) (see section 3 of Appendix B: Security Technology for a discussion of distinguished names and SSL). For example, access privileges should be defined for the DN "`C=US, O=Sandia, OU=Security, CN=John Doe`", not for that node's digital certificate. Certificates are short-term, ephemeral objects that bind public keys to long-term, static identities. As such, it makes much more sense to assign security configurations to the long-term, static identities as signified by DNs. When security controls are mapped to identities, any valid certificate that identifies the user as the DN "`C=US, O=Sandia, OU=Security, CN=John Doe`" will work, and there will never be any need to update the security configuration database.

3.5.2.2 ICCP Initial Configuration with Certificates

In some current ICCP applications, configuration of the security policy database requires that the node have beforehand the certificates of all the other nodes with which it will communicate. The certificates are used by the ICCP application to map access control permissions to each end node. (Note that the policies and permissions should not be mapped to a node's certificate, but to the unique identity—the distinguished name—of the end node. Mapping policies to certificates creates the problems described above.)

Requiring a node to have copies of certificates *a priori* is contrary to the purpose and general use of public key certificates. *Certificates should be sent or acquired on demand, not pre-distributed*. Further, configuring the ICCP security policy should never require certificates; policy decisions should be tied to the identity, as specified by a DN, of the certificate owner. To create a security policy, the node only needs to know the DNs of the nodes with which it will communicate, but it should not need their certificates.

3.5.3 Permanent ICCP SSL Sessions

Normally, SSL sessions are relatively short-lived. SSL sessions are usually created on-demand when data needs to be sent and are closed after the requisite data has been transmitted.

In its use for securing ICCP, SSL sessions are quite different. An SSL session is established between two nodes and is kept alive indefinitely regardless of how much traffic passes between the nodes. The plan for long-lasting sessions introduces several security and certificate management issues that must be considered.

3.5.3.1 ICCP Certificate Revocation or Expiration Vulnerability

The SSL sessions are expected to last long periods of time, perhaps months, and will likely span certificate expiration periods. An obvious question is what happens when the node with which you are communicating has its certificate revoked. Under normal circumstances where sessions are constantly being created then destroyed, the revoked certificate would be used during a session handshake, it could be identified as revoked, and the SSL connection would not be created, thus severing communication with the revoked node.

However, if SSL sessions last indefinitely, there is never any handshake in which the revoked certificate can be identified. A longstanding SSL session has no knowledge of the certificates used long ago when it was first created. Therefore, a permanent SSL session will remain open even after the certificate of one of the session's end nodes is revoked. Obviously, it is a security problem if a session is not ended when one of its nodes has its certificate revoked. Unfortunately, terminating long-lasting sessions is not a typical requirement of SSL, so SSL has no built-in mechanism for identifying or destroying such a session.

Potential Solutions

One way of correcting this issue is allowing the ICCP implementation to maintain a copy of the remote certificate used during the creation of an SSL session. If the remote certificate is revoked or expires, the ICCP implementation may terminate the corresponding SSL session (the precise functionality has not been finalized). Thus, the Secure ICCP implementation is able to terminate longstanding SSL sessions if either of its end node certificates becomes invalid.

Unfortunately, in order for the Secure ICCP implementation to solve this one problem, several other problems are introduced. The biggest problem is that nodes must now store copies of each others' certificates locally. Caching other nodes' certificates creates some certificate management problems when certificates expire or are revoked. If a cached certificate expires, the Secure ICCP implementation can notice the expiration and be able to terminate the SSL session. To prevent terminating sessions that should remain open, the expiring cached certificate needs to be replaced with a renewed certificate. While it may not sound difficult, in practice this certificate replacement means that each node must inform every other node each time its certificate is renewed. As discussed above, it is not standard practice in a PKI to send notifications when new certificates are received, so that *functionality must be added.* Furthermore, this notification process must be performed well in advance of the certificate expiration to ensure that copies of the renewed certificate are

distributed before the old certificate expires. While these problems are not insurmountable, they are clumsy and require additional infrastructure to be remedied.

Instead of caching certificates locally, perhaps a better method of terminating longstanding SSL sessions is to re-perform periodically the SSL handshake. Redoing the SSL handshake is a standard process commonly referred to as renegotiation. To make sure that the two nodes in a given SSL session still have valid certificates, they can perform a renegotiation. The renegotiation process can be configured in such a way that each node is required to supply the other with a valid certificate. If either node no longer has a valid certificate, the session can be terminated. This renegotiation process can be performed as often as is necessary to ensure timely detection of expired or revoked end nodes. Possibly, the SSL sessions can be configured to perform a renegotiation each time a new CRL is received to minimize the window of vulnerability. This solution is desirable because it does not require any sophisticated supporting infrastructure and does not require nodes to alert each other when they receive a new certificate.

3.5.3.2 CBC Rollover vulnerability

There is one additional security caveat related to longstanding SSL connections that should be mentioned. Depending on the cryptographic algorithms used, the CBC counter used to encrypt messages may roll over in old SSL connections that have transferred a lot of traffic. CBC rollover is a significant security issue that can allow an attacker to gather information about the encrypted data. To prevent CBC rollover in longstanding SSL sessions, the end nodes should periodically perform an SSL renegotiation and create new session keys.

3.5.4 ICCP Internet Certificate Authorities

It has been suggested that some ICCP systems will not stand up their own on-network CA, but will use common internet-based CAs, such as VeriSign. The other option, which appears more reasonable, is to have a local CA that sits on the isolated control network and services all the control nodes on that network. The advantage with using internet CAs is that the certificate issuance costs are minimal. While the low cost is attractive of internet CAs, we recommend against using them for a variety of reasons.

Most control networks are not connected to the internet. As such, control center nodes certified by some internet CA will not be have any means of communicating with their CA to receive common CA services, such as certificate renewal. The lack of a communication channel between control center nodes and their CA means that normal operations like certificate renewal, update, or revocation checking become quite challenging. To receive those necessary services, some gateway (manual or automated) must connect to both the internet and the isolated control network and act as an intermediary for the nodes. Creating a gateway that can provide this intercessory service in an efficient and secure manner is challenging and not straightforward.

There are also problems with certificate revocation. If CRLs are used, the CRLs will be maintained and provided by the internet CA. Some intermediary for the control network will need to download periodically the CRLs from the internet and transfer them to the isolated network. Since the internet CA services nodes besides those on the control network, the CRLs will be quite large and comprised almost solely of revoked internet certificates.

Control network nodes do not care about revoked internet nodes (with which they have no contact), so downloading and applying the large CRLs is an especially inefficient process. One solution that has been proposed requires the intermediate gateway node to filter the CRL down to only the certificates that matter. This solution is also challenging since the gateway must therefore know exactly which certificates its subsidiary nodes are interested in.

Another issue with internet CAs is revocation. If a node on the network is subverted and its certificate must be revoked, it is much easier to put that certificate on a CRL maintained by a local CA than a CRL maintained by some distant internet CA. Control system companies will have much less influence on the CRLs of an internet CA than they would have on a local CA that serves only the control network. Due to the large scale at which they operate, internet CAs are necessarily slower and less responsive to CRL changes than a local control network CA could be. As such, internet CAs would impose delays (in addition to the aforementioned delay imposed by the gateway) in certificate revocation.

Finally, there is an issue of trust and reliability associated with using internet CAs to certify control networks. These control networks are high consequence systems and by nature have higher security demands than typical internet nodes. If nodes use different internet CAs for certification services, then the other nodes must also trust those internet CAs. The nodes must trust that the internet CA authenticates and verifies the identity of all its clients in a manner that is commensurate with the control network's high security standards. (For example, if the CA did not adequately verify the identity of its clients, an adversary could assume the identity of a control network node by tricking the CA into signing a bogus certificate.) The security procedures of internet CAs are often unknown and outside of the authority of the control network administrators. Using internet CAs would therefore require entrusting the bulk of the security of the control network to the (possibly unknown) security practices of third-party CAs. *For high consequence systems, it seems more prudent to invest in establishing a user domain CA that serves its local control network, has a strictly defined and managed registration authority process, is verified to be secure, and is related more directly to the control network companies*.

3.5.5 CA and CRL Domains

Section 3.5.4, *ICCP Internet Certificate Authorities*, discussed the structure of CAs and the distribution and management of CRLs within a typical PKI Internet CA service. It was suggested that high consequence networks (such as utility control networks) not participate in publicly available CA services. A more prudent and secure approach to certificate management is to align the CAs more closely with participating utility nodes. This implies that the role of the Central Authority should reside within the business structure of interest and the actual size of the CA domain must be aligned with the required interactions of participating utilities.

Within the United States, the electrical power grid is comprised of three primary networks. Within these networks smaller networks or associations are also created. The primary networks provide connections between multiple utilities to allow the transfer of electricity from different parts of the network. The three primary networks in the United States are the Western Interconnect, the Eastern Interconnect, and the Texas Interconnect. The Eastern Interconnect provides power to most of the eastern part of the Untied States. The Western

Interconnect provides power primarily to the Rocky Mountain region, including the Pacific west and the Southwest. And the Texas Interconnect provides power primarily with the state of Texas.

These regions have limited connections with each other, although the Western Interconnect and the Texas Interconnect are both linked with Mexico. And the Western and Eastern Interconnects are interconnected with Canada. All contiguous United States utilities are connected to one of these primary electrical power networks

The North American Reliability Corporation (NERC) is the responsible body for reliability planning and coordination of the interconnected electric power grid. The NERC is comprised of ten regional councils that are responsible for the reliability and security of the contiguous United States and parts of Canada and Mexico. The NERC boundaries are created by the service areas of the electric utility regions. **Figure 11** displays this structure.

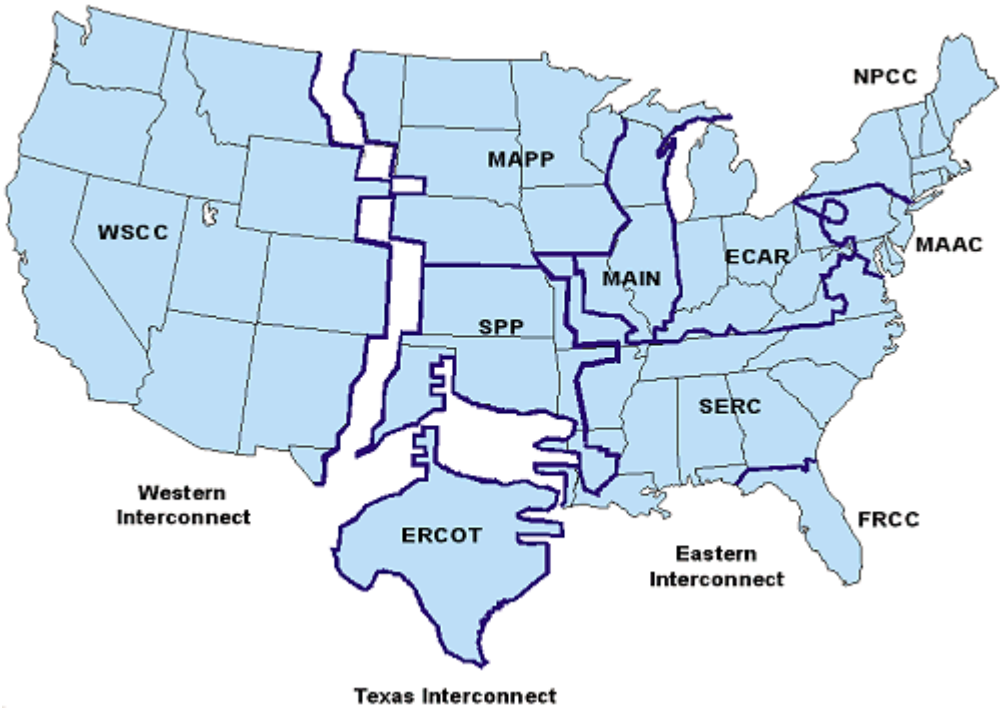

Figure 11. Main Interconnections of the US Electric Grid and Ten NERC Regions[6]

ECAR – East Central Area Reliability Coordination Agreement
ERCOT – Electric Reliability Council of Texas
FRCC – Florida Reliability Coordinating Council
MAAC – Mid-Atlantic Area Council
MAIN – Mid-America Interconnected Network
MAPP – Mid-Continent Area Power Pool
NPCC – Northeast Power Coordinating Council
SERC – Southeastern Electric Reliability Council
SPP – Southwest Power Pool
WSCC – Western Systems Coordinating Council

3.5.5.1 Creating ICCP CA Boundaries

The first challenge when implementing a certificate base authentication scheme is to identify all the necessary end nodes that are required to participate. From this information a boundary can be formed. As seen in **Figure 11**, boundaries may be associated with NERC regions, but this is not a requirement. The identified boundary should include the support of the day-to-day operations associated with the exchange of ICCP related data. Once all participants have been identified then the role of the central authority must be decided. The CA will be responsible for validating participants and issuing and managing the authentication certificates. This includes the creation and access of a CRL database that can be pushed out

[6] The Changing Structure of the Electric Power Industry 2000: An Update, Energy Information Administration, October 2000

to users upon request. In most cases, a single centralized CA architecture works most efficiently, off-loading the majority of authentication management to a single identity.

3.5.5.2 CA Cross-Certification Chain of Trust

In some cases, when information exchange is required between different boundaries for proper electric power coordination, cross-boundary communication is required. These cross-boundary communication requirements may include the major network interconnects that consist of extra-high-voltage connections between individual utilities designed to permit the transfer of electrical energy from one part of the network to another. When these communication channels have been identified, then a CA chain of trust may be needed to authenticate nodes outside of local operation domain.

To allow authenticated and secure communication between nodes that use different CAs, trust must be established between the CAs. Secure ICCP implementations should be able to accommodate a PKI that provides a means of trusting certificate authorities and their associated public keys. These CAs become a chain of trust, and certificates that have been issued and digitally signed by a CA on this chain can be trusted. Section 1,

Certification Hierarchy Schemes, in Appendix B: Security Technology describes a tiered CA implementation. This description is based on a single-domain implementation in which an individual CA is assigned to each utility in a tier architecture that includes a common "root" CA. This approach is not recommended because it requires each utility company to stand up and manage its own separate CA, which prevents trustworthy cross-domain communication; trust between domains cannot be established because each domain has its own CA and there is no root CA.

There are two primary means to extend trust between CAs. The first is *peer-to-peer* cross-certification and the second is *hierarchical* or *tiered* cross-certification.

Peer-to-Peer Cross-Certification

In the peer-to-peer approach, each independent domain provides a CA that has self-signed its own certificate and is considered the *trust anchor*—the authoritative CA against which all certificates are validated—for the domain. Each node within a domain practicing peer-to-peer cross-certification will have the certificate and public key of the trust anchor CA for the domain. The local domain CA establishes trust with an external domain CA by signing the certificate of the external domain CA. When a local node establishes communication with a node from the external domain, its certificate can be validated because the signature of its (external) domain CA has been signed by the local domain CA. Figure 12 depicts this structure.

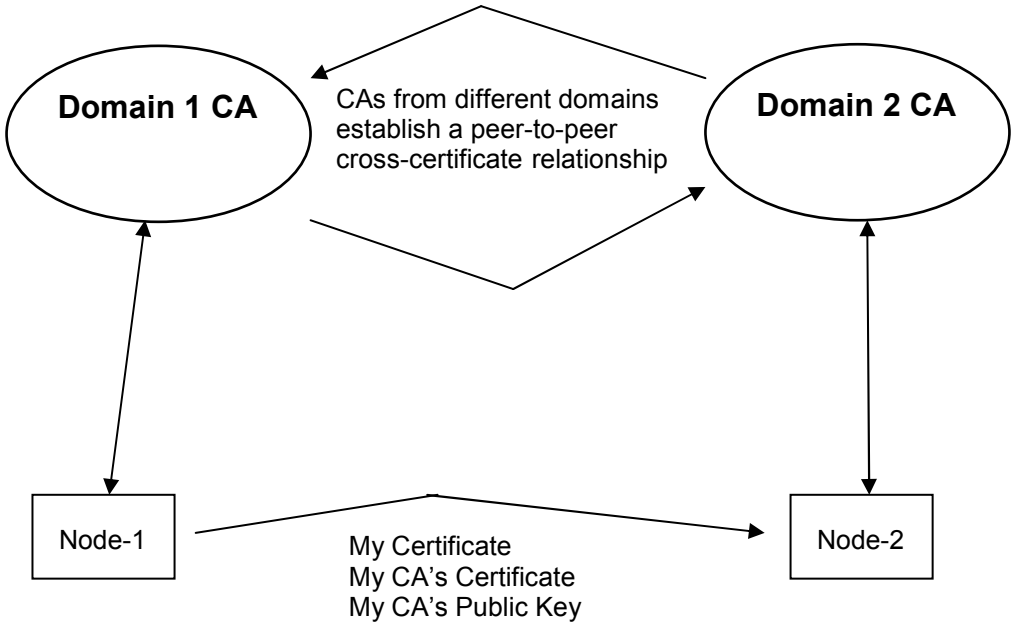

Figure 12. Peer-to-Peer Cross-Certification

For example, consider peer-to-peer cross-certification established between CA1 and CA2, as in Figure 12. If Node-1 (in domain 1) wants to communicate with Node-2 (in domain 2), it sends a signed message to Node-2. The Node-1 user sends the Node-2 user a copy of his certificate signed by his Domain 1 CA, a copy of his Domain 1 CA certificate signed by Domain 2 CA, and a copy of Domain 1 CA public key. This signed message will be

successfully validated by Node-2 because the Node-2 CA has cross-certified with the Node-1 CA by signing Node-1 CA's public key. Likewise, the Node-1 CA has cross-certified with the Node-2 CA by signing the Node-2 CA's public key.

Advantages of using Peer-to-Peer Cross-Certification

The advantage of peer-to-peer cross-certification is that each individual domain CA is autonomous in creating and revoking cross-certification relationships. Such an autonomous CA does not need another CA as a trust anchor, and site security policies can be based on business needs, rather than having to depend on an external root CA (and any necessary subordinate CAs) for administering certificates. This is more flexible than a hierarchal structure and more appropriate for business relationships that are dynamic in nature.

Hierarchical or Tiered Cross-Certification

Another approach to cross-certification is the hierarchical or tiered approach. As the name implies, a hierarchical structure consists of a single trust anchor within a series of CAs. The trust anchor is the root CA from which all subordinate CAs branch out. This structure can be as deep as needed, with additional root CAs being designated below the primary root. A root CA is responsible for signing the CA certificates of all CAs below the root. The primary difference with this approach vs. the peer-to-peer approach is the location of the trust anchor CA. The root CA is the trust anchor for all subordinate CAs which must use the root CA public key for certificate validation. Figure 13 depicts this structure.

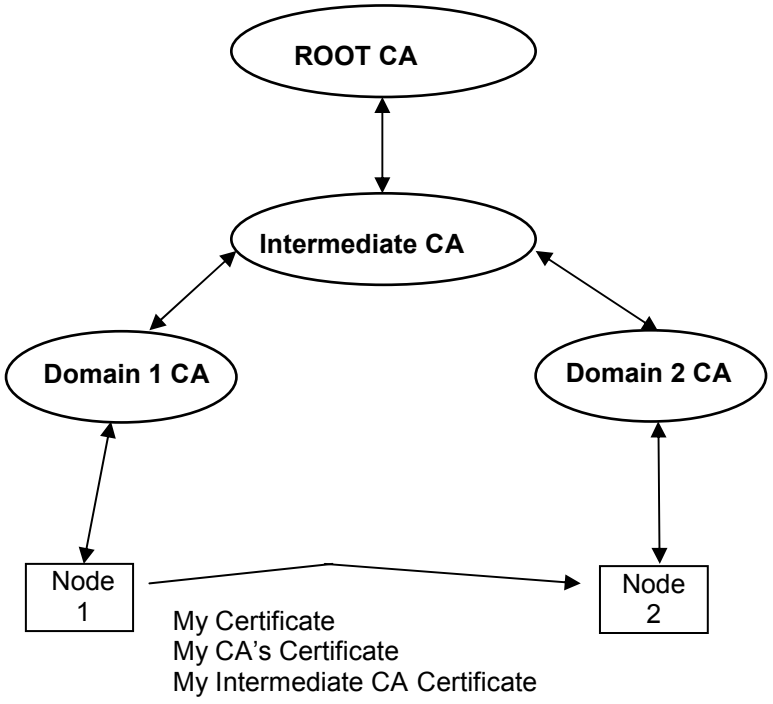

Figure 13. Hierarchical Cross-Certification

For example, in Figure 13 when Node-1 registers with his domain 1 CA, the PKI process will download the root CA public key and domain 1 CA certificate signed by the root CA. The

same is true for Node-2 in domain 2. When registering with the PKI, it will download the root CA public key and domain 2 CA certificate signed by the root CA. When a node associated with a subordinate CA registers with the PKI, it receives a copy of the root CA public key and will use this as its CA trust anchor.

When a participating node (such as Node-2) receives a request to establish communications with another node outside of his local domain, such as Node-1, then he must validate the external node's certificate with a higher order CA. The external party node (Node-1) will have its certificate signed by its direct CA (Domain 1 CA), which is validated by Node-2 using the public key of Domain 1 CA, which is sent with the certificate from Node-1. The Domain 1 CA will have its certificate signed by its direct CA the "Intermediate CA" which has its certificate signed and validated by the highest order CA (the root CA) the Intermediate CAs certificate is also sent from originating Node-1. Upon receiving Node-1's certificate, Node-2 validates it with Node-1's Domain CA public key. Then Node-1 must validate Domain CA 1's signature with the public key of its direct higher order CA, the Intermediate CA. Using the public key of the Intermediate node sent with its certificate from Node-1, Node-2 validates the signature. And finally, the signature of the Intermediate CA is validated by the public key of the Root CA resident on Node-2. The chain of trust has now been validated and Node-2 can trust Node-1. This seems like a long and arduous process, but, with a properly designed architecture, it can be efficient.

Advantages of using a Hierarchical Cross-Certification Process

Hierarchical cross-certification is appropriate where multiple CAs need to be created and an organization requires complete control over all CAs in the hierarchy. The Root CA can control the policy of all subordinate CAs including revoking CA's that do not comply with published policy. The hierarchical structure lends itself to a business model that is mostly static where the organizations participating in cross domain communications are known and fixed.

The previous example described inter-domain communications between two nodes that required the sending of multiple certificates for validating a chain-of-trust. In lieu of sending multiple certificates from an originating node, another approach that can be implemented for cross-certification of a higher-order CA is by the use of an extension field within the sending node's certificate.

This extension field called the Authority Information Access (AIA) field contains the specifics of requiring a certificate of a higher order CA. This is normally a URL that provides the link to a certificate repository. In the previous example shown in Figure 13, the receiving Node-2 would follow the link in the extension field of Node-1's certificate and automatically retrieve a copy of the sending nodes CA certificate. Since this is a lower-order CA and not a root CA, the participating node must look at the AIA extension of the lower-order Domain 1 CA which points to the higher order Intermediate CA certificate. This has within its extension field the URL or directory of the highest order certificate which is the same root certificate or trust anchor that is resident on the participating node and thus can be trusted for certifying the sending nodes domain CA's signature. The receiving node can then use the sending node's domain CA certificate to validate the sending node's certificate. This

technique is normally referred to as path validation, because the validating end node follows a path to validate the originating node's CA's certificate.

Regardless of which approach is used to provide cross validation for cross domain certificates, the question arises: What designated party should be responsible for providing a higher-order or "root" CA? Referring to Figure 11. *Main Interconnections of the US Electric Grid and Ten NERC Regions*, it appears that NERC would be a good candidate for providing the proper PKI root-level CA service. This allows any subset of participating groups within a local region to select the NERC regional authority as its immediate higher-order CA. An addition of one layer to the hierarchy can be created by the NERC to allow cross regional communications to be validated by a multi-regional or national level root CA. Figure 14 depicts this structure.

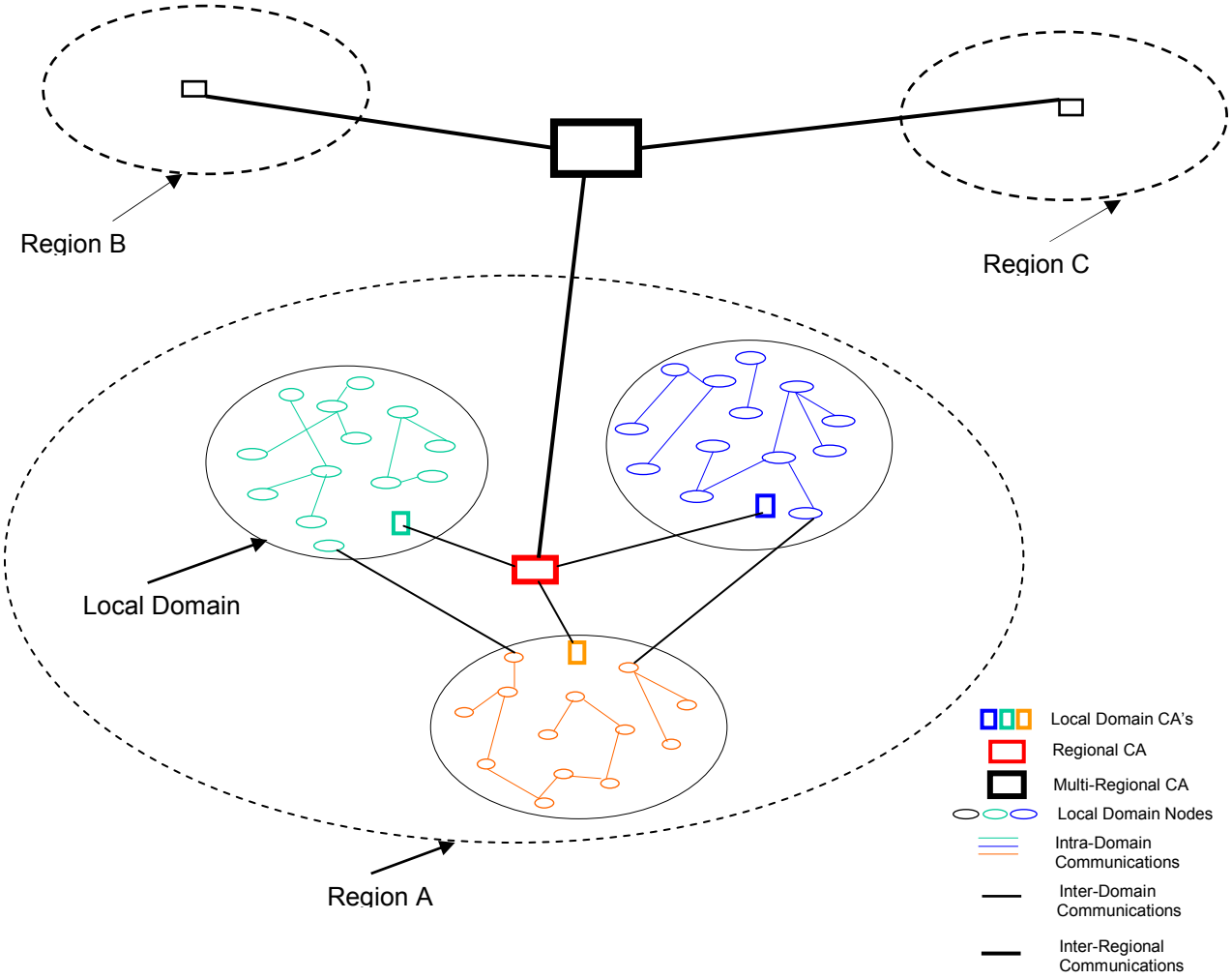

Figure 14. Central Authority Chain-of-Trust Structure

3.5.5.3 CA Domain and Cross-Certification Summary

The previous sections discuss the concept of CA domain design and implementation. Two primary architectures are discussed, the peer-to-peer and the hierarchical, typical interactions are provided and operational examples given. It is difficult to provide a blanket statement

about which implementation is best for the entire utilities industry however, based on the need for the most secure approach in implementation, the hierarchal structure is most appropriate for the following reasons:

Using a Root CA to certify and issue policy to subordinate CAs can enhance the security of the root CA. By not using the root CA to support end users, the root CA will be less exposed to end users and can be physically controlled more restrictively.

Policy consistency is enforceable with the hierarchical model but not with the peer-to-peer model. When nodes in different communication domains administered by a peer-to-peer autonomous CA model need to communicate, there is no assurance that the external participating node has developed and implemented a secure approach to communications. Each time a local node needs to communicate with a different domain, it risks being compromised due to improper configuration or management of an external domain CA. Because of this, autonomous peer-to-peer models provide more avenues for compromise

Since the root CA is the "trust anchor" for all users and CAs within the hierarchy, the maximum physical security policies and practices are required only for the root CA and not for all the subordinate CA's. This reduces the overall risk for proper management security for those domains that need a CA service.

3.5.6 Secure ICCP Stale Certificate Detection

It is apparent that some current ICCP practice is incompatible with the intent of providing a dynamic approach to end node authentication as envisioned by the Secure ICCP protocol. Practical changes in current ICCP implementation need to take place. Two potential solutions have emerged.

The first, which is not recommended, requires participating ICCP nodes to cache certificates locally. Caching the certificates of other nodes creates certificate management problems when certificates expire or are revoked. If a cached certificate expires, the Secure ICCP implementation can notice the expiration and is able to terminate the SSL session. To prevent terminating sessions that should remain open, the expiring cached certificate needs to be replaced with a renewed certificate. While it may not sound difficult, in practice this certificate replacement means that each node must inform every other node each time its certificate is renewed. *This approach is not recommended*. It is not standard PKI practice to send notifications when new certificates are received, so that functionality would need to be added to the PKI infrastructure if this approach were used. Furthermore, this notification process must be performed well in advance of the certificate expiration to ensure that copies of the renewed certificate are distributed before the old certificate expires.

The second solution, which is recommended, is to periodically re-perform the SSL handshake, which is a standard process commonly referred to as *renegotiation*. Two nodes in an SSL session can perform renegotiation to make sure that their certificates are still valid. The renegotiation process can be configured so that each node is required to supply the other with a valid certificate. If either node no longer has a valid certificate, the session can be terminated. This renegotiation process can be performed as often as necessary to ensure timely detection of expired or revoked end nodes. Possibly, the SSL sessions can be

configured to perform renegotiation each time a new CRL is received to minimize the window of vulnerability. *This approach is recommended* because it does not require alteration of the existing PKI infrastructure and does not require nodes to alert each other when they receive new certificates.

3.6 Strategy for the transition from ICCP to Secure ICCP

For some utility sites the conversion from standard ICCP to Secure ICCP will occur over time. This section describes some alternatives to simulaneously converting all nodes to Secure ICCP and is applicable when communication needs to be protected within a group of nodes where (1) all are using standard (non-secure) ICCP or (2) some are using standard ICCP and some are using Secure ICCP.

Virtual Private Tunnels (VPN) can occur at multiple levels within the communication stack. Essentially, the secure form of ICCP provides a transport layer VPN to protect its data payload. But there are other VPN technologies that can be used to create secure virtual private tunnels between Utilities without the implementation of Secure ICCP. Additional communication layers within the OSI communication stack can be utilized to afford protection for non-Secure ICCP nodes sharing information from local area networks across wide area networks.

3.6.1 Layer 3 Link Protection

Internet Protocol Security (IPSec) is a standard developed by the Internet Engineering Task Force (IETF) for providing secure communications over public Internet Protocol (IP) networks, i.e., the Internet. At the network level, IPSec supports peer authentication, data origin authentication, data confidentiality, data integrity, and replay protection. IPSec is normally used with IKE (Internet Key Exchange) for key management. IPSec supports most modern encryption algorithms such as advanced encryption standard (AES), data encryption standard (DES), its more secure 3DES version, and Rivest cipher (RC4). It also provides integrity support using popular integrity HASH algorithms such as message digest (MD5) and secure hash algorithm (SHA-1), and authentication using X.509 certificates. IPSec can be implemented either Host-to-Host or gateway-to-gateway. For a more detailed description of IPSec and IKE refer to IETF RFCs 4301 [26], 4303 [27], 4835 [28], and 4306 [29]

3.6.1.1 Phases of IKE

IKE negotiations have two phases:

Phase one

The two gateways negotiate and set up a two-way Internet security and key management protocol (ISAKMP) security association (SA) which they can then use to handle phase two negotiations. One such SA between a pair of gateways can handle negotiations for multiple tunnels.

Phase two

Using the ISAKMP SA, the gateways negotiate IPSec (ESP and/or AH) SAs as required. IPSec SAs are unidirectional (a different key is used in each direction) and are always negotiated in pairs to handle two-way traffic. There may be more than one pair defined between two gateways.

Both phases use the UDP protocol and port 500 for their negotiations. After both IKE phases are complete, you have IPSec SAs to carry your encrypted data. These use the ESP or AH protocols.

IPSec can be used in one of two different modes: encapsulated security payload (ESP) or authentication header (AH), described and discussed in [27] and [28]. These modes are called, respectively, *transport mode* and *tunnel mode*. In tunnel mode, the IP datagram is fully encapsulated by a new IP datagram using the IPSec protocol. Tunnel mode provides authentication, data stream integrity, and confidentiality. In transport mode, only the payload of the IP datagram is handled by the IPSec protocol, which inserts the IPSec header between the IP header and the upper-layer protocol header. Transport mode provides only data stream integrity and authentication, not confidentiality. Figure 15 shows these two modes.

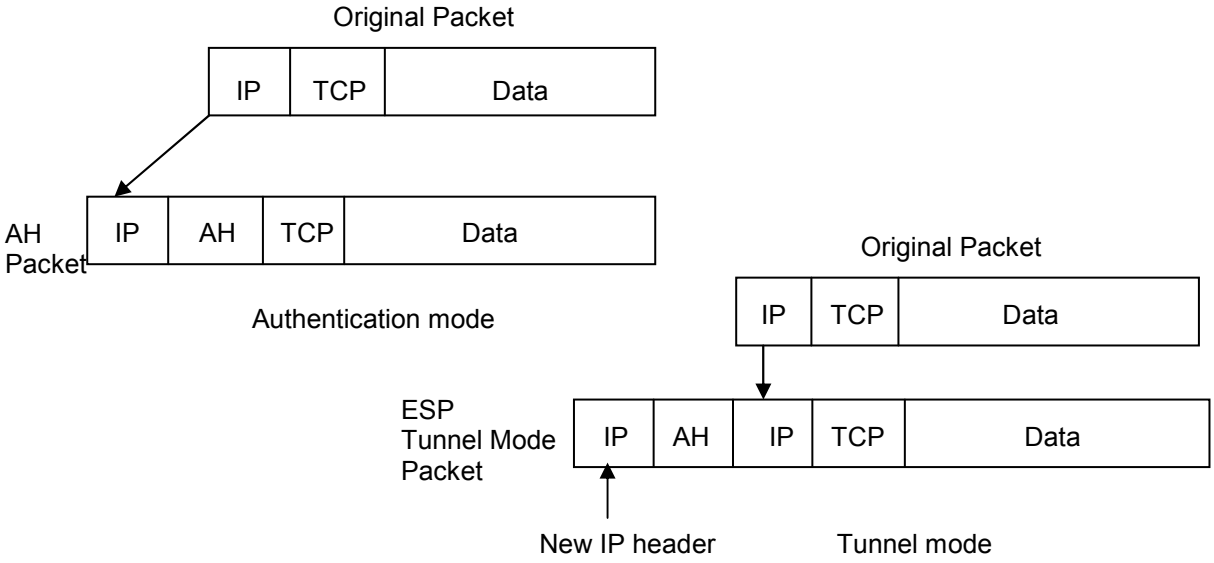

Figure 15. IPSec modes of operation

Security gateways are required to support tunnel mode connections. In this mode the gateways provide tunnels for use by client machines behind the gateways. The client machines need not do any IPSec processing; all they have to do is route data to gateways.

IPSec transport mode can also be implemented between two chosen hosts (ICCP client & server). Each end host must support IPSec security and be able to negotiate an authenticated link between host machines (as opposed to security gateways).

IPSec is implemented at Layer 3 of the OSI network stack to encapsulate IP packets. After a VPN tunnel has been established per tunnel mode, application data such as ICCP can be encapsulated and sent through the tunnel. IPSec is popular in site-to-site VPN implementations because it can be realized in network devices, such as a gateway router, without modifying any client or server applications. Figure 16 shows an ICCP application flow through an IPSec VPN tunnel.

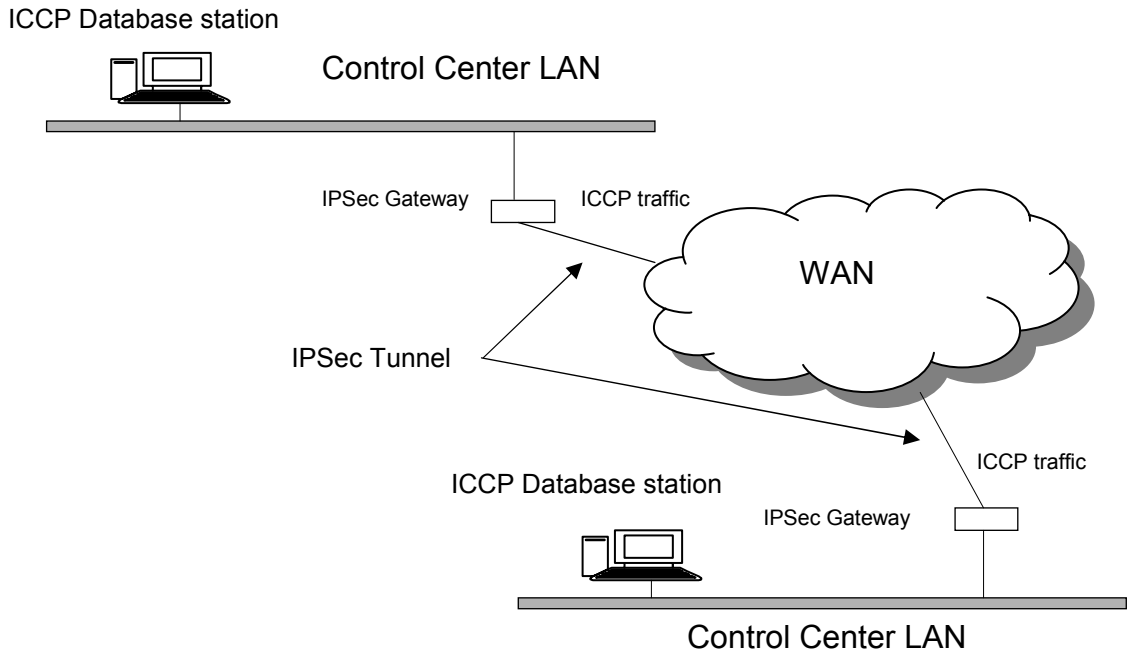

Figure 16. IPSec Gateway Center to Center Communications

3.6.1.2 IPSec encryption and protection

As previously mentioned, IPSec supports multiple types of encryption algorithms, these algorithms can be divided into two categories: stream encryption algorithms, such as RC4; and block encryption algorithms, such as Triple DES (3DES) and Cipher Block Chaining (CBC), both of which are commonly used in VPNs.

A stream cipher is a symmetric encryption algorithm that is designed for efficient processing, making it faster than standard block ciphers. A stream cipher combines a generated key stream with the clear text data to create a cipher data stream. The popular RC4 stream cipher from the company RSA has been used in the Wired Equivalent Privacy (WEP) protocol for 802.11 wireless systems. Because the RC4 stream cipher creates a key that combines private bit selection with a clear text initialization vector it is susceptible to adversarial compromise

Block encryption is less vulnerable to traffic analysis than stream encryption. A block cipher transforms a fixed-length block, called the block size, of clear text data into a block of cipher text data. The conversion is based on a user selected secret key. To decrypt the blocks of data, the reverse conversion on the cipher text block using the same secret key is performed. Since different clear text blocks are mapped to different cipher text blocks, a permutation of all possible block messages provides much more security when encrypting the clear text data.

The following is a list of some of the protections IPSec provides against adversary activities:

- IPSec prevents *man-in-the-middle* attacks by protecting the integrity checksum calculated and inserted into the encrypted portion of the payload.
- IPSec prevents *message replay* attacks by inserting sequence numbers within each packet. This allows stale correspondence to be discarded.

- IPSec is much more immune to *denial of service* attacks than any TCP-implemented security service. This is because IPSec uses connectionless services such as IP and UDP (IKE) which are easier to ignore than TCP SYN floods, which create and fill up session tables and can exhaust the allowed number of simultaneous sessions.
- IPSec can protect end points from *address spoofing* because packet end points are authenticated prior to the flow of data.

3.6.1.3 Configuration Guidance and Protection

During the IPSec configuration, cryptographic access lists are created to provide a form of access control. This allows the end user control over which remote end host can participate in an IPSec session. Access can be limited to a single server (as in the case of an ICCP server) or to an entire private subnet. Packet filters can be constructed that only allow a specific data stream to be inserted or received for an individual session

To implement authentication, IPSec employs Internet Key Exchange (IKE), using digital certificates or pre-shared secrets for two-way authentication. Potential operational problems can occur if network address translation (NAT) is implemented within the gateway router. NAT is used to translate a non-routable IP address frequently used within a private LAN to a routable IP address for public network (Internet) transport. Since the IPSec header provides an integrity checksum, the NAT process, which swaps private addresses for public "routable" addresses, changes the result of the integrity check sum which causes the IPSec process to discard the packet.

It's possible to overcome this problem by creating a static NAT that precedes the IPSec process as long as the same private address is associated with the protected end node. Also, some IPSec products can implement a NAT traversal extension to overcome this limitation, but, to prevent possible interoperability problems, both end nodes participating in IPSec should have the same product implementation.

The following steps provide some configuration guidelines when building an IPSec VPN.

1) Determine network design details to include the encryption policy, identified host and networks that will be protected, and the IPSec features that will be used. Allow any preconfigured firewalls to pass IPSec negotiation ports UDP port 50 & 51.
2) Configure the mode for creating security associations, static or dynamic. The process of securing data between multiple users using IPSec starts with the defining of a security association (SA). An SA, uniquely identified by a multiple-bit number called a Security Parameter Index (SPI), is constructed by identifying the following parameters in a transform set.
    - Source and destination IP address of the peers participating in the creating and termination of the IPSec tunnel.
    - The encryption algorithm and secret key used by the IPSec protocol.
    - The authentication algorithm used to authenticate IPSec packets
    - IPSec mode (transport or transport)
    - Lifetime of the security association.

Static configurations of SAs are prone to error. It is suggested that the dynamic form of SA establishment be used. This is done by selecting the internet security association key management protocol (ISAKMP).

3) Configure ISAKMP for IPSec and select key distribution method, a peer-to-peer method or a certificate authority.
4) Define the transform set parameters that will be used to negotiate a security association with a peer node (see step 2 list).
5) Create a crypto map. A crypto map is a file that associates all the parameters of the VPN. One of the important features of the crypto map is associating a pre-defined data filter that will identify and filter specific data flows into and out of the VPN tunnel
6) Apply the crypto map to the selected interface that will represent the ingress and egress point of the VPN.
7) Test and verify the VPN.

3.6.1.4 Non-Secure ICCP Fallback Configuration

One of the transitional problems associated with the integration of Secure ICCP is the matter of secure data flow negotiation. The ICCP protocol has the ability to "fall back" into a non-secure form of transport if both end nodes do not support a secure profile. This means that ICCP can be configured to allow a fallback to transmit ICCP data in the clear. The following provides a recommended approach to configuring a network connection to provide a mixed-mode operational scenario when both secure and non-secure forms of ICCP co-exist on a network.

The primary purpose of an IPSec gateway is to decide which flows are to be protected between two distant end points. Profiles are created to provide the ability to isolate communication between hosts, such as trusted servers, and any pre-determined end devices. Thus, regardless of the means of communicating, private WAN or public Internet, the remote egress gateway must use IPSec to negotiate trust and to secure IP traffic end-to-end with the destination computer located behind the corresponding ingress gateway.

With respect to ICCP, there are two ways to approach this profile configuration. The first is to use a less granular configuration that provides IPSec encryption *for all* communications between identified end hosts. The second is to use fine granularity in the form of a port filter, which can identify non-secure forms of ICCP and provide IPSec encryption *for only* those forms.

Single Host Isolation, no port filtering

As previously described in section 3.6.1.3, *Configuration Guidance and Protection*, a filter is constructed within each gateway to identify communicating endpoints that are allowed in and out of the encrypted tunnel. In the case of an ICCP server and a distant Host or Hosts, each connection will be identified and authenticated by its IP address. This provides a bulk approach to data confidentiality by encrypting all communications between end points regardless of whether a higher layer of encryption is being applied, as is the case of Secure ICCP implementations. This double encryption can provide an additional layer of protection by obfuscating the original IP addresses of the end host participating in the communications

but may cause additional processing burdens and delays associated with the transmission of data. See Appendix B: Security Technology for additional discussion on this topic.

Single Host Isolation with port filtering

Another approach that can be pursued to isolate data flows in the scenario of secure and non-Secure ICCP data streams originating and terminating at the same server is using port filtering for flow identification. As part of the IPSec configuration profile an access control list is created that identifies each host allowed into a protected domain to communicate with a particular host. In the case of ICCP this could be the ICCP server. To identify the type of communications taking place between the two endpoints, an additional filter can be enabled that allows the gateway to peer into the transport layer header and identify the port being addressed by the client/server session. When it is seen that the communication is using the secure form of ICCP, the stream is not forwarded through the IPSec tunnel (no double encryption). If the port address is determined to be a non-secure form of ICCP it is then pushed through the tunnel for data encryption. Figure 17 shows this inspection.

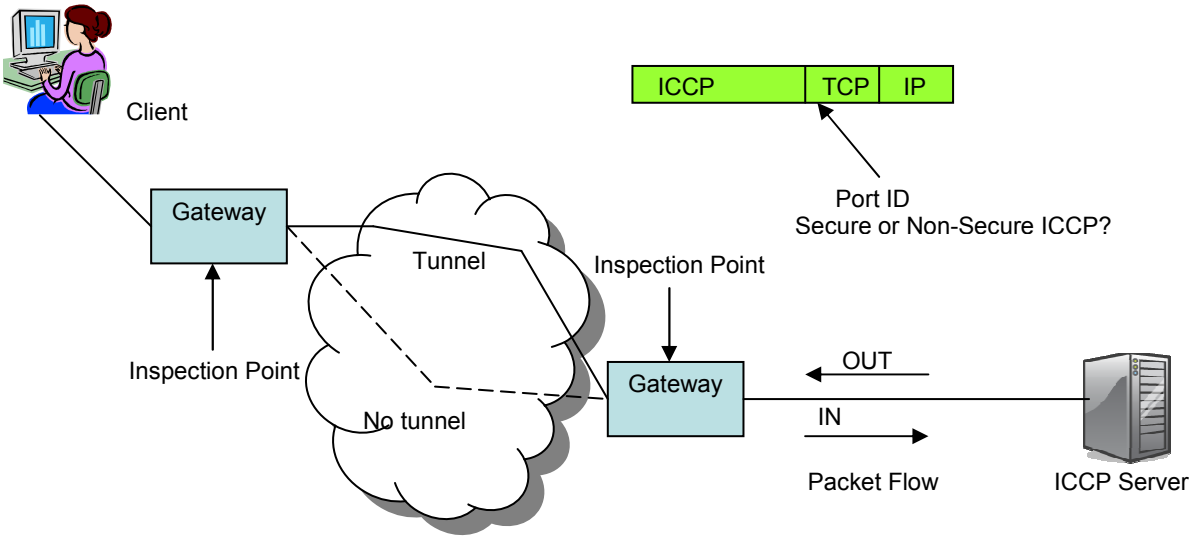

Figure 17. IPSec Port Filter Implementation

The purpose of IPSec domain isolation is to mitigate the risk posed to trusted resources. Implementation technologies such as gateways for filtering and authentication can help protect a utility's data assets. The gateway solution allows only those end nodes that can meet some specific security profile to interact with trusted resources. End nodes that are untrusted are denied access. By creating this trusted environment and restricting the permitted communications inside and outside of this environment, the utility company can reduce the overall risk to its data assets.

One additional security asset that needs to be mentioned as part of the inspection architecture shown in Figure 17 is the addition of an intrusion detection system (IDS). Because the encrypted IPSec data stream is not encrypted until it reaches the demarcation point, represented by the gateway, an IDS has complete knowledge of all activity on its protected domain. This is the advantage of using IPSec in tunnel mode as opposed to transport mode.

3.6.1.5 IPSec Administration Issues

The Internet Key Exchange (IKE) protocol which is used to configure automatically and setup IPSec protected data tunnels uses UDP port 500. The associated IPSec Encapsulated Security Payload (ESP) and Authentication Header (AH) protocols use UDP ports 50 and 51. As part of the authorized port access list, it is important to verify these port numbers are allowed to pass through any of the restricted interfaces configured to prevent unauthorized access.

IPSec administrators must create security policies for each authorized network connection. This information becomes part of the "transform set" that is used to negotiate secure connections between two end nodes. Information needed to create communications policies includes the IKE authentication method, Diffie-Hellman Group, data encryption algorithms, hash authentication type, and security association lifetimes. Many IPSec network product vendors have created user-friendly proprietary management systems that help the user automate policy distribution. These systems can be helpful as long as the IPSec networks are kept to as low as possible.

A gateway which represents the ingress/egress point of a protected domain normally has its IPSec processes within the gateway's route function and any management configuration will be provided by the gateway's administrative port. The port normally takes the form of a console that can be accessed by multiple protocols, i.e., Telnet, HTTP, SSH, SNMP, etc. To provide proper protection of the administration port, any remote management of the gateway device should be allowed only from a management station connected to a trusted network. It is also recommended that different levels of access be defined to ensure only authorized administrative personnel are assigned configuration tasks that provide the security profiles for remote access and data throughput.

The administrator responsible for the configuration and management of the IPSec device, if it is not directly connected at the console port of the gateway device, must ensure the following:

- Ensure a connection from a management station to the gateway is connected to a trusted network
- Provide multiple user and access level control configurations on the gateway device
- Provide a policy that describes the means to authenticate and confiscate packet flows transmitted over untrusted networks.
- Provide the means to protect cryptographic keys

When developing an IPSec policy include the following in your review:

1. Determine the state of your network infrastructure: Review current architecture and determine its applicability to an IPSec implementation by identifying the trusted domain to be protected, any firewalls that may hamper IPSec negotiation, and the identification and location of the IPSec hosting gateway.
2. Design and test the IPSec policy prior to deployment: Test IKE configurations for proper connectivity and negation prior to installing configuration on active network.

3. Identify supporting security elements such as intrusion detection systems to ensure LAN traffic is visible on the LAN prior to gateway injection. This allows all IPSec streams to be monitored for improper activity.

Additional administrative observations about IPSec

IPSec cannot ensure the security of a system if the system is not secured. End host that have been subverted, undermine the protection mechanisms administered of IPSec configured on gateway hosts.

IPSec can provide a good security service when encrypting data between gateways that transmit data over an untrusted WAN, but it provides only gateway-to-gateway authentication. An additional authentication process must be included in the security configuration tasks associated with protecting the communicating end-points. For example, to control which users access an ICCP database server, you need to implement some independent user authentication mechanism along with some sort of data access level, e.g., bi-lateral table configuration.

As a reminder, the IPSec gateway configuration does not protect the contents of the packets from being viewed by observers on the protected network. However, if the protected network is instrumented with an IDS, this situation is quite acceptable.

Although IPSec is used to provide confidentiality (encryption) to its payload it does not prevent traffic analysis. Traffic analysis is a technique that is used while "sniffing" data flows. It attempts to identify characteristics of the data flow based on visible header information, packet size, packet frequency, and event and time correlation.

3.6.1.6 Layer 3 Link Protection Summary

IPSec can provide a means of protecting ICCP data exchanges prior to a full deployment of Secure ICCP. It provides many of the secure attributes needed to protect data traffic exchange including mutual authentication. The implementation does not require any changes to client or server applications and provides protection from some forms of transport layer attacks. It prevents unexpected hosts from initiating communications with hosted servers.

IPSec systems can be designed with no burden on the end user. No additional username and passwords are needed for client and servers to connect through an established tunnel. Host names or IP addresses are used to filter appropriate traffic into a tunnel. Tunnel negotiated end-points use authentication protocols to verify each other and can complement other security mechanisms implemented to protect any undefined commuter or device.

IPSec can protect protocols above the IP layer such as UDP or TCP and any combination of applications. It can complement other security mechanisms used to protect application data, although additional complexity and processing delays need to be analyzed prior to their implementation. See section 3.7, *Security Configurations and Performance*, for more information about use and integration.

3.6.2 Layer 2 Link Protection

When Secure ICCP is not available or will not be implemented, an alternative approach to securing ICCP data exchange across the WAN is providing an encryption technique that is

implemented at the data link layer of the communication path. This approach can provide data security for the Frame Relay packets that are transmitted across a Frame Relay wide area network.

3.6.2.1 Securing Frame Relay Communications

Encryption devices that support Frame Relay layer protection can provide data integrity (data tampering detection), data confidentiality, and protection from replay attacks for all non-Secure ICCP data flows. Most data link encryption devices support modern forms of encryption algorithm implementations, to include Data Encryption Standard (DES), triple DES, and Advanced Encryption Standard (AES). Additionally, identifying and filtering the source and destination addresses at the ingress/egress of the network allows for tighter control of the WAN connection. Providing intrusion detection monitoring and managing the cryptographic keys are important parts of securing communications.

Because a Frame Relay connection through the Frame Relay network is associated with the setting up and partitioning of a Permanent Virtual Circuit (PVC), it is recommended that each PVC have its own encryption key and this key should be changed on a regular basis. Many of the Frame Relay encryption devices allow for the system to change automatically the data encryption key without user intervention thus reducing the dependency of a user remembering and initiating regular changes to the encryption key.

Data link encryption can take two different forms, “bulk” encryption or “data field only” encryption. In bulk encryption the entire frame is encrypted, including the header where addressing information is contained. With a bulk encryption approach, only point-to-point implementations are feasible because of the obfuscation of the header. This is normally implemented when a lease line is used to connect two distinct endpoints and does not require address fields to be assessable. Figure 18 shows this implementation.

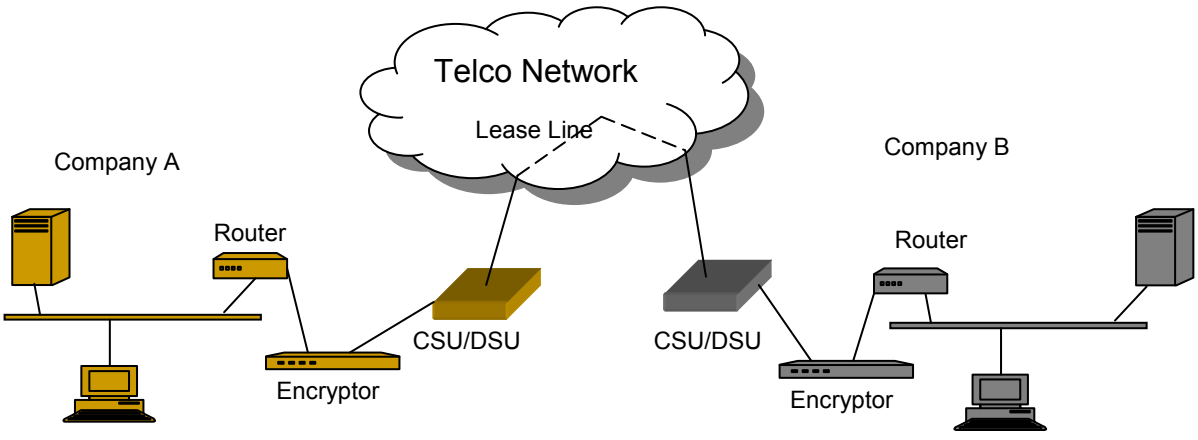

Figure 18. Point-to-Point WAN Connection

A data link encryptor that provides data field encryption encrypts only the data field in the Frame Relay packet, leaving the header information, which contains the address, in the clear. This allows the Frame Relay packet to be processed and switched through the Frame Relay switch nodes during transmission. This in turn allows the ICCP data to be encrypted once at the ingress of the network and then decrypted at the destination. Figure 19 shows this setup.

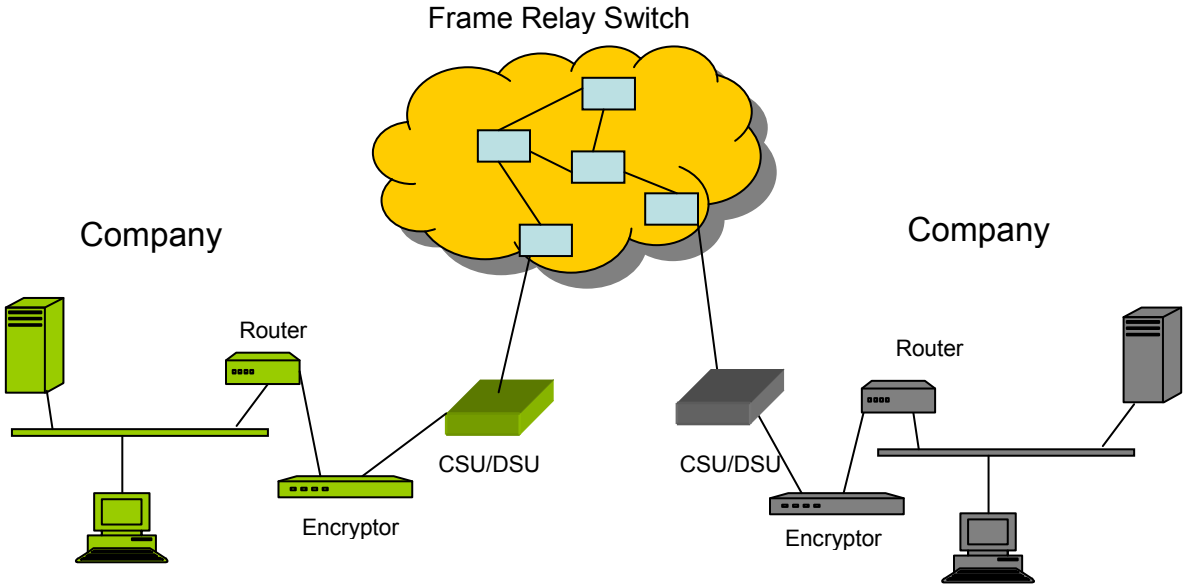

Figure 19. Frame Relay WAN Connection

As previously mentioned, data field encryption encrypts only the data in each frame, not the header which contains address information, allowing the routing of the frame through the network. Since the encryption occurs at Layer 2 of the OSI network stack, the encryption technique can also protect upper layer communication protocols. Standalone encryption devices, as depicted in Figure 19, can generally operate with any type of router and network topology. When selecting an encryption scheme, be aware that encryption integrated into a router may often require the same router at every communicating point in the network. This prevents a more open approach to system construction because of the proprietary nature of the technology, requiring the same units to be used at all connecting end points.

A hardware encryptor-based solution, as depicted in Figure 19, can be more secure than software-only solutions that rely on the operating system in which they are deployed. Hardware implementations, being designed on a standalone platform, are not susceptible to vulnerabilities associated with the underlying operating system. Hardware solutions can also offer tamper-proof protection along with both physical key and password access control.

One of the advantages of using a Frame Relay network is the circuits are pre-established by the use of Data Link Connection Identifiers (DLCI). This allows for the monitoring and filtering of the Frame Relay traffic at the encryption device. Encryption devices can be assigned access control criteria for each DLCI, essentially adding an additional layer of security. This can provide more granularity to each DLCI circuit. For example, DLCI circuit #1 originating from one particular utility company could always be encrypted using a specific encryption algorithm, while DLCI circuit #2, originating from another utility company, may transmit its data in the clear.

3.6.2.2 Key Management

The primary purpose of key management is to generate, distribute, and protect the cryptographic keys required to encrypt data. The security requirements are based on the level of protection that is needed for the data being protected. The implementation or techniques chosen must be flexible enough not to hinder operations but sustain the security of the system. The approach that key management describes should enable the secure processing of keys in both a manual and automated fashion. The approach should include the management of the key life cycle of generation, distribution, updating, and deleting.

One reason a utility node may not be able to participate in any secure form of ICCP is because of the need to upgrade its software suite to a form of ICCP that can provide the ability to participate in a public key management scheme that includes dynamic key validation and distribution services. Because these features are not available on current releases of ICCP, an out-of-band key distribution process must be identified to provide a data encryption key for participating end nodes. This process is needed to establish a secure session between two encryption units. Once the encryptor end devices are initiated with a common "master key," session keys can be derived in an dynamic fashion between participating end nodes. Following are some key management observations that can be used as a design guide when selecting a Layer 2 encryption scheme.

- Identify an out-of-band solution for initial master key distribution. Once a session is established, communicating endpoints should support the ability to change data session keys automatically.
- Separate session keys should be used for each unique connection (DLCI identifier) between utility end points. Any chosen authentication scheme should use a key that is different from the data session key.
- A session key between any set of utilities should remain secure or isolated from any other communicating end nodes.
- A compromised session key between a pair of participating end nodes should not result in all other sessions being compromised
- Session initiation should be able to be established from any participating end node
- Any data link encryptor solution should include the ability to change session keys in an automatic and deterministic way.
- A process needs to be established that provides recovery from key loss or key disclosure events.

3.6.2.3 Layer 2 Link Protection Summary

As part of an ICCP-to-secure-ICCP transition strategy, incorporating a secure data link approach to security can provide ICCP end nodes with a secure form of data transport. This works well when the numbers of end nodes remains small however, because the initial key distribution is out-of-band and requires additional external coordination, the approach does not scale. Frame Relay encryption can be a cost effective way of providing a network transport mechanism between participating end nodes. It is important to note that any Frame Relay architecture needs to be associated with a service level agreement (SLA) with the service provider, an SLA that takes into account all necessary quality of service (QoS)

elements of transport reliability (see section 3.1.6.3, *Frame Relay Congestion and QoS Management*). It is also important to identify key management aspects of a solution to determine if it can be applied across all participating end nodes.

It is important to note that when not using a secure form of ICCP a Layer 2 WAN protection mechanism as described in this section provides protection only from the ingress to the egress of the WAN. ICCP Packets within the originating LAN will be transmitted in the clear prior to injection onto the WAN. It is important to institute additional layers of protection to provide situational awareness of all LAN traffic. See section 3.1.3, *Infrastructure Design and Protection*, for a description of additional security practices.

The previous discussion did not include any form of centralized key management and distribution architecture. The rationale for excluding this information is that building and maintaining a centralized form of key management within the data link protection approach can entail as much analysis and integration effort as implementing Secure ICCP. See section 2, *Public Key Infrastructure,* in *Appendix B: Security Technology* for a description of key management and distribution. It is recommended that, when it becomes apparent that a centralized approach to authentication and key management is needed, the transitional data link encryption approach be relinquished for a Secure ICCP standard approach that can support a PKI infrastructure.

3.7 Security Configurations and Performance

3.7.1 Introduction

As discussed in section 3.6, *Strategy for the transition from ICCP to Secure ICCP*, there are other security technologies that can be applied to protect ICCP transactions prior to the ubiquitous implementation of Secure ICCP. What this section provides is a summary of the previously described technologies and their protection mechanisms to provide the reader a sense of how much security is enough and what is the performance hindrance created by the introduction of security layers to protect ICCP data transactions.

It is not the intent of this report to provide a blanket statement about how many layers of security are sufficient in the protection of ICCP transactions, but to provide information on security technologies that can be leveraged within the communication process and the protections that each layer of security provides. Once an understanding is established of the protections that are afforded by implementing a security technology, then policies can be created to govern the choice and implementation of the technology. Also, in some instances the operational and performance impacts that can be encountered when introducing security techniques to the operational environment are also provided. This information will allow each asset owner to make informed decisions concerning the development of security based policies that will govern the data exchange interaction between participating end nodes

3.7.2 Security Layer overview

The purpose of this section is to provide the reader an understanding of what the security technology described in this report provides for the protection of data communications.

3.7.2.1 SSL/TLS Public Key Certificates

Prior to transmitting ICCP data from one node to another, the trustworthyness of both participating end nodes must be verified. In ICCP, each node has a predefined mechanism that can be validated to prove it is the node that it claims to be. As part of Secure ICCP, it is hoped that the dynamic mechanism to accomplish the authenticity of communicating end nodes is through the use of digitally signed certificates. This process relies on the installation of a public key infrastructure (PKI), described in section 2, *Public Key Infrastructure*, of Appendix B: Security Technology. This technique of proving each other's identity will take place prior to any data exchange between nodes. When an ICCP application calls the Secure ICCP layer to protect its data exchange, the Secure ICCP layer will initiate the request for a certificate exchange. How long it takes prior to the resolution of trust will depend on the following constraints the speeds of the processors on each node performing the certificate exchange, the transmission delay caused by all the intermediate communication infrastructure nodes that are responsible for relaying data associated with the information exchange, and any additional layers of security that must be engaged to process the transmitted data.

A properly configured certificate exchange provides the ICCP transaction user with only end-node authentication and negotiated data confidentiality. It doesn't provide the following:

- Application software validation (Software version security)
- Application Identification (Does not hide port numbers)
- User Identification (An actual person)
- Network tunnel protection (Entire Layer 4 and above protection)
- Data link tunnel protection (Entire Layer 3 and above protection)

3.7.2.2 PKI Architecture Design Protection

Along with providing a certificate exchange process for end-node authentication, is the need to build a secure supporting architecture to provide the means of distributing secure authentication certificates. As described in the PKI section, there are two primary ways to build a PKI infrastructure: using a peer-to-peer structure that expands trust autonomously and, alternatively, the use of a hierarchical structure that expands trust by subordination.

The advantages and disadvantages of these architectures are described in section 2, *Public Key Infrastructure*, of Appendix B: Security Technology, and are not restated here. Strictly speaking, with respect to security, the hierarchical or "tiered" structure is more secure because of the management policy restrictions it implies. The peer-to-peer mode has more entities involved in its management, so the likelihood of disparate security policy administration is increased. This provides a rich avenue for unauthorized entry into the structure. Conversely, security policy is strictly controlled and administered in a hierarchical structure. This protected structure enables the distribution of authentication certificates, as described earlier in section 4.2, *Tiered Hierarchy*, which provides proper authentication and data confidentiality.

3.7.2.3 User Access Control

The next logical step in the protection of information is to verify who—in other words, which people—have a Need To Know (NTK) the information that is to be shared. To unauthorized assess to the ICCP service, there must be a process in place to identify each user who logs in. This process, as specified by policy, should require user authentication as a condition for access to the workstation and/or to the server where the ICCP application resides. This can provide the proper restrictions to application access on a per-user basis. User authentication can be implemented locally for each machine or more globally by the use of user role-based authentication services, which provides a role-based access control (RBAC) which essentially means translating a "user's role" to application permission.

3.7.2.4 Application Authentication

After an end node has been authenticated and any user role validated, the next layer of protection comes in the form of software validation. The Association Control Service Element (ACSE) which is layered with the Manufacturing Message Specification (MMS) layer, is responsible for establishing an application association between two application programs. An application-association is a cooperative relationship between two application entity interfaces. It provides the necessary frame-of-reference in terms of the application service services. This relationship is formed by the communication of application protocol control information between application entities through their use of the presentation. This service provides the identification of the peer application entity and protection from replay of previous connection information. It is invoked at the establishment of an application association and uses a message authentication code to validate information generated form the source node and verified at the receiving node to detect any modification of application information during the lifetime of the application association. This service can provide Software object authentication and data integrity.

3.7.2.5 Network Communication Protection

Another means of protecting ICCP communications in a network environment is the insertion of data encryption services at the data link (Layer 2) and at the network link (Layer 3) of the OSI communication stack. As described in section 3.6, *Strategy for the transition from ICCP to Secure ICCP*, inserting these encryption services allows for the protection of data streams from a demarcation point between the LAN and the WAN. Although the Layer 3 protection mechanism could be deployed at the workstation or server, it was recommended to insert this protection at the entry point of the WAN. This allows for other security monitoring technologies, such as intrusion detection systems and intrusion protection systems, to continually monitor the data transmission and reception streams for abnormal content or behavior. Layer 2 and Layer 3 encryption services provide data integrity, data confidentiality, and application port confidentially. In the case of Layer 3, these services can also provide end node authentication.

3.7.2.6 Network Performance Protection

Another important aspect of network communications is performance metrics. Data transmission performance is dependent on the network and the end-to-end delay, data throughput, inter-packet delay, and data loss encountered. One means of providing assurance in the end-to-end communications process is providing a quality of service (QoS) guarantee

for near real-time data flows, such as ICCP. Regardless of how well the data stream is protected from unauthorized access or disclosure, if it cannot be reliably delivered then in essence it results in a denial of service situation. *A properly QoS-partitioned network will protect data flows in congested denial of service situations*.

3.7.3 ICCP Network-Based Performance Testing

The Inter-control Center Communications Protocol (ICCP) supports modes of secure and unsecured communication for inter-utility transactions. Two distinct ICCP configurations, a Regional Transmission Operator (RTO) using an Energy Management System (EMS) that implements ICCP and a utility using an EMS that implements ICCP will be utilized to conduct an examination of the end-to-end communication that is most typical with ICCP transactions. The objective, using both the secure and non-secure forms of ICCP, is to identify the operational and performance impact of using the secure version of ICCP. Several operational configurations have been identified with the intent of measuring the computational loads (workloads) for each configuration. This will provide the operators deploying Secure ICCP a base-line of measured performance.

This testing directly supports tasks associated with the scope of work to be performed in the *Secure ICCP FY06 Work Package* (SNL).

3.7.3.1 Test Configuration

In order to answer some of the questions concerning the introduction and configuration of a secure version of ICCP, a test template has been constructed. The test template identifies candidate ICCP configurations. Each of these configurations will be tested to determine its performance impact. The purpose of these tests is to attempt to get time-rate-performance ratings from the applications that are doing the authentication and encryption on the ICCP. The factors and values for this testing are the performance issues related to the execution of SSL encryption and IPSec tunneling. The performance information can then be used to help guide the selection and integration of the best candidate ICCP configurations.

The following ICCP configurations were tested for performance on a Local Area Network (LAN) using the National SCADA Test Bed (NSTB) network at Sandia National Laboratories. All tests were performed with an SNL client as one end node and and SNL server as the other.

1. No security (baseline)
2. Client unsecured; Server secured with SSL/TLS
3. Client and Server both secured with SSL/TLS
4. Client unsecured; Server secured with MACE
5. Client and Server both secured with MACE and SSL/TLS

The workload parameters are determined by the systems environment, which is Windows Server 2003 for both the client and the server in this configuration. These systems are configured with a common server-client architecture that uses ICCP to transport data back and forth. The operating system is installed on a 3GHz processor with 1 GB of memory. For IPSec tunneling there are two Cisco 3600 series routers without specialized encryption hardware.

3.7.3.2 Network Configuration

The network used for testing was configured as shown in Figure 20. It consists of an ICCP client and an ICCP server, each running Windows Server 2003, connected through two Cisco 3600 series routers and conjoined by a hub, which is used for timing purposes. This configuration allows testing to be performed between the client and server while timing analysis may be done between the client and server, between the client and hub, and between the server and hub. An additional machine, not shown in this figure, contains two Ethernet interfaces, so that it may be connected to two points at once. This allows for a single clock to be used for timing and avoids synchronization issues inherent in using two independent clocks. The connection of the Ethernet interfaces depends on the timing analysis being performed and is shown in the appropriate figures.

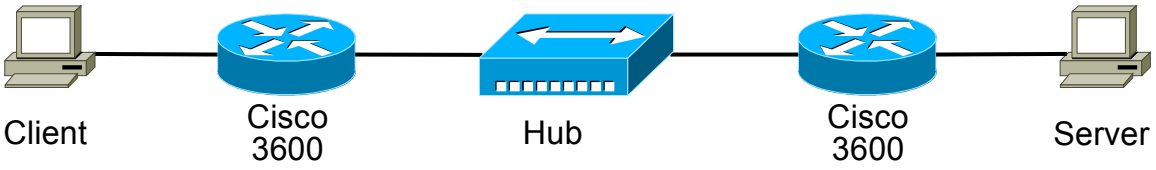

Figure 20. ICCP Network Layout

3.7.3.3 IPSec Network-Based Testing

The goal of network-based testing is to investigate the performance issues related to tunneling ICCP through an IPSec tunnel. System boundaries are defined as the network communication links and the processing performance related to the encryption used in IPSec by the Cisco 3600 series routers. An additional machine, shown as “Timer” in Figure 21 and Figure 22, was added to the network configuration for timing.

Figure 21 shows the configuration for measuring the end-to-end transmission time of a packet across the test network. This configuration connects the Ethernet interfaces of the timing machine to the client/server side of the Cisco routers. This allows the timing machine to see a packet as soon as it leaves the source machine and to determine exactly when it arrives at the destination machine. If an IPSec tunnel is not being used, the end-to-end transmission time is the standard latency for the packet to be processed by the outgoing router, repeated by the hub, and processed by the incoming router. If an IPSec tunnel *is* being used, the end-to-end transmission path contains everything in the non-tunneled case plus IPSec encryption on the outgoing router and IPSec decryption on the incoming router. This difference between the tunneled and the non-tunneled case is considered to be the added latency of the IPSec tunnel.

Figure 22 illustrates the configuration for measuring the end-to-midpoint transmission time of a packet across the test network. This configuration connects the Ethernet interfaces of the timing machine to the client side of the Cisco routers and to the hub. This allows the timing machine to see a packet as soon as it leaves the source machine and, at the same time, determine exactly when it departs the Cisco router towards the destination machine. If an IPSec tunnel is not being used, the end-to-midpoint transmission time is the standard latency for the packet to be processed by the outgoing router. If an IPSec tunnel is being used, the end-to-midpoint transmission time contains the element of the non-tunneled measurement with the addition of the IPSec encryption or decryption depending on the direction the packet is traveling. If the packet is traveling from client to server, then the measured latency will

include the encryption time. If the packet is traveling from server to client, then the measured latency will include the decryption time.

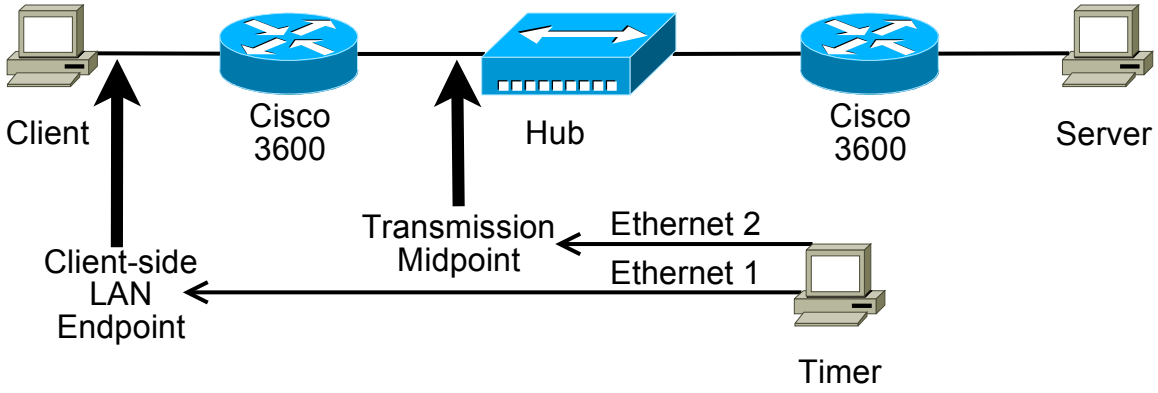

Figure 21. End-to-End Measurement Configuration

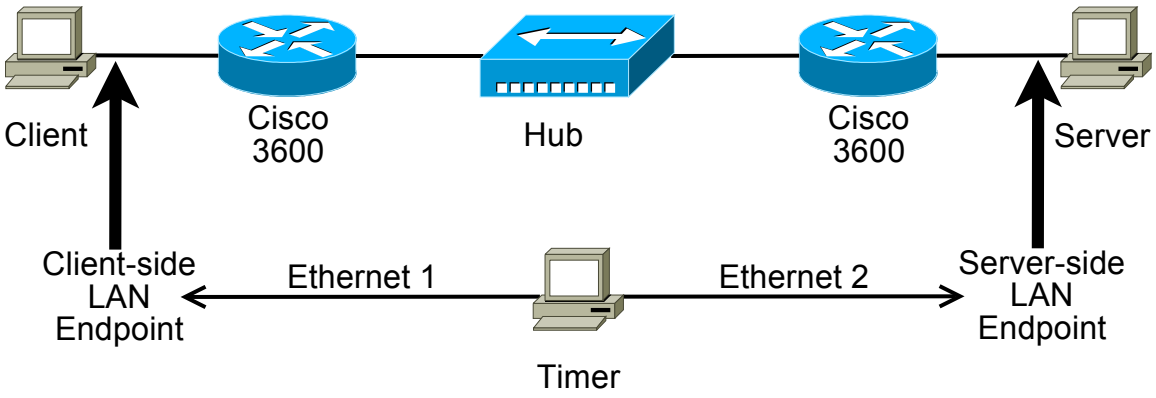

Figure 22. End-to-Midpoint Measurement Configuration

3.7.3.4 Network-Based Results

The average time to travel across the non-secure network (i.e., without the IPSec tunnel) was 0.59 milliseconds. Due to the small observed values and network traffic, actual minimum and maximum times may vary from those observed if the end points are more distant than those of the test environment. Distant values are almost certain to vary because of differences in the number of router and switch hops between endpoints. The intent of measurements presented in this section is to provide the reader with representative processing delays on some common network devices without bandwidth saturation. Impacts of bandwidth saturation associated with ICCP application flows for IP routed networks and Frame Relay switched networks are described in section 3.1, *Impact*. When measured to the midpoint the latency in sending a packet out of the router (i.e., no routing necessary) was 0.14 ms. The latency for receiving a packet, which includes appropriate routing, was 0.36 ms. As an additional check these values may be added and compared to the average full trip time. This comparison yields a difference of 0.09 milliseconds. This value is reasonable due to the repeat time of the hub and/or nominal variation due to small time scale. Figure 23 and Figure 24 show the actual measured average for each communication path.

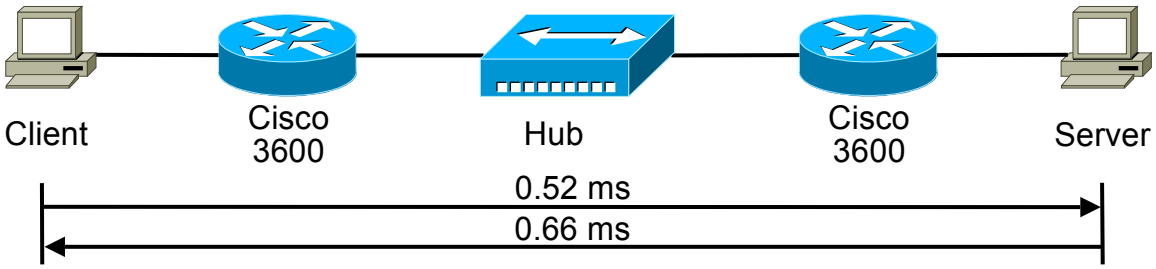

Figure 23. End-to-End Non-Secure Measurements

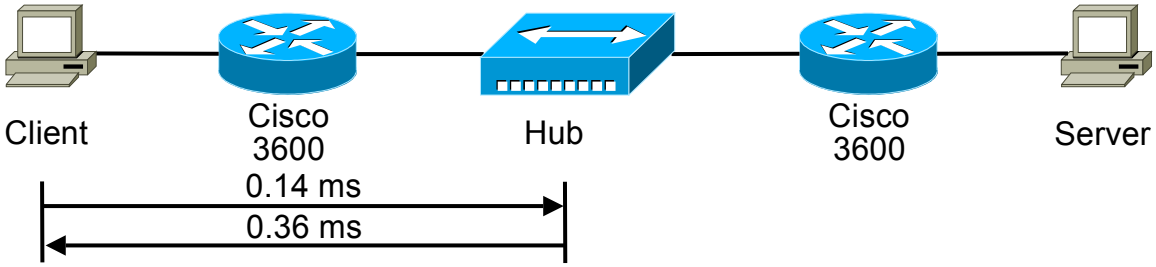

Figure 24. End-to-Midpoint Non-Secure Measurements

The average time to travel across the network *with* the IPSec tunnel was 6.10 ms. The 5.51 ms difference is a direct reflection of the encryption and decryption delays. When measured to the midpoint, the latency in sending a packet out of the router (i.e., no routing necessary) was 3.55 ms, including encryption. The latency for receiving a packet, which includes appropriate routing, was 2.52 ms, including decryption (with a known key value, decryption is typically faster than encryption). As a reasonability check, these values may be added to get 6.07 ms and compared to the average full trip time, yielding a difference of 0.56 ms. This value is reasonable due to the hub repeat time and/or variation due to small time scale. Figure 25 and Figure 26 show the actual measured average for each communication path measured.

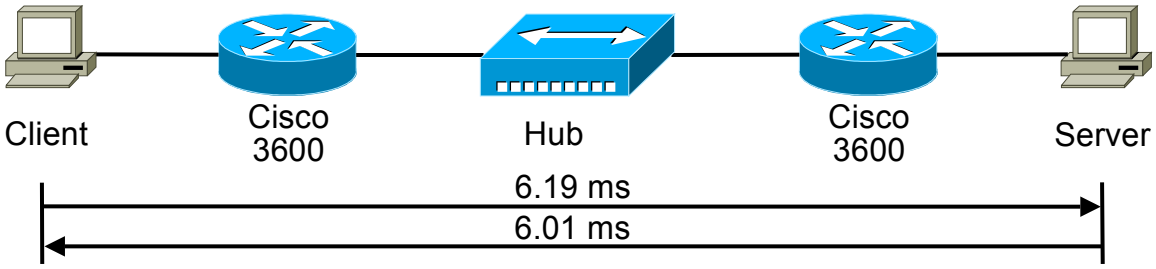

Figure 25. End-to-End Secure Measurements

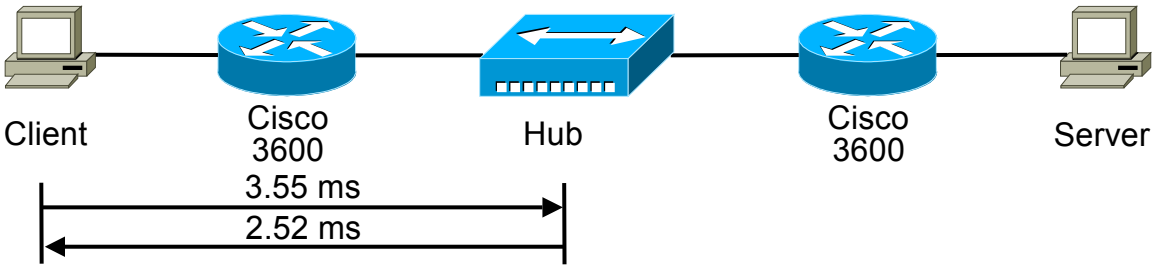

Figure 26. End-to-Midpoint Secure Measurements

3.7.4 ICCP Software-Based Performance Testing

Software-based testing is used to provide performance timing in milliseconds for the encryption/decryption routines used in securing ICCP. Another value of this type of testing is the design of special tools used to test timing of the central processor unit (CPU) and the actual loading of our own function into a running process, then retrieving the timing of a SSL encryption within an executable program.

ICCP utilizes SSL/TLS as one of its security methods. Timing of the application cannot be performed due to on specific implementations of SSL/TLS associated with vendor applications concerning NDA licensing restrictions. Timing analysis was performed using OpenVPN. OpenVPN is an open source tunneling technology built on OpenSSL. OpenSSL is the implementation of SSL utilized within the version of ICCP that is being tested. Thus, by timing OpenVPN and OpenSSL, we can determine an estimate of the delays incurred by ICCP.

Unfortunately, MACE encryption cannot be tested due to licensing restrictions and a suitable replacement technology could not be found to provide reliable estimates of the timing of this functionality.

Software-based timing is performed by inserting into a running application a small amount of code that has been specifically designed to calculate the runtime of the original code. Custom tools were written to insert this specialized code into the OpenVPN process, thus allowing the encryption and decryption routines to be timed with high accuracy. The code that is inserted into the application replaces a function call, reads the current timestamp from the processor (in clock ticks), calls the original function, re-reads the new timestamp and reports the difference in timestamp. The difference in clock ticks is the number of clock ticks spent inside the original function. This value along with the processor speed allows the total millisecond value to be calculated. The reading of the timestamp and the associated storage adds a negligible amount to the overall timing. This methodology provides very accurate timing of the running function.

Figure 27 shows an example of the custom timing analysis tool. In this example, the process identification number, or PID, of the OpenVPN process is `2788`, the location in virtual memory of the function to be timed is `0x00409c51` and the code is to be injected at virtual memory location `0x0044fde4`. How these numbers are calculated is beyond the scope of this paper. The output of the tools is in comma-separated value (CSV) format. The first three values, shown in hexadecimal format, are the timestamp when the function was called, the timestamp when the function completed, and the difference in the timestamps. The final column, shown in decimal milliseconds format, is the time difference calculated using the timestamp difference and the processor speed.

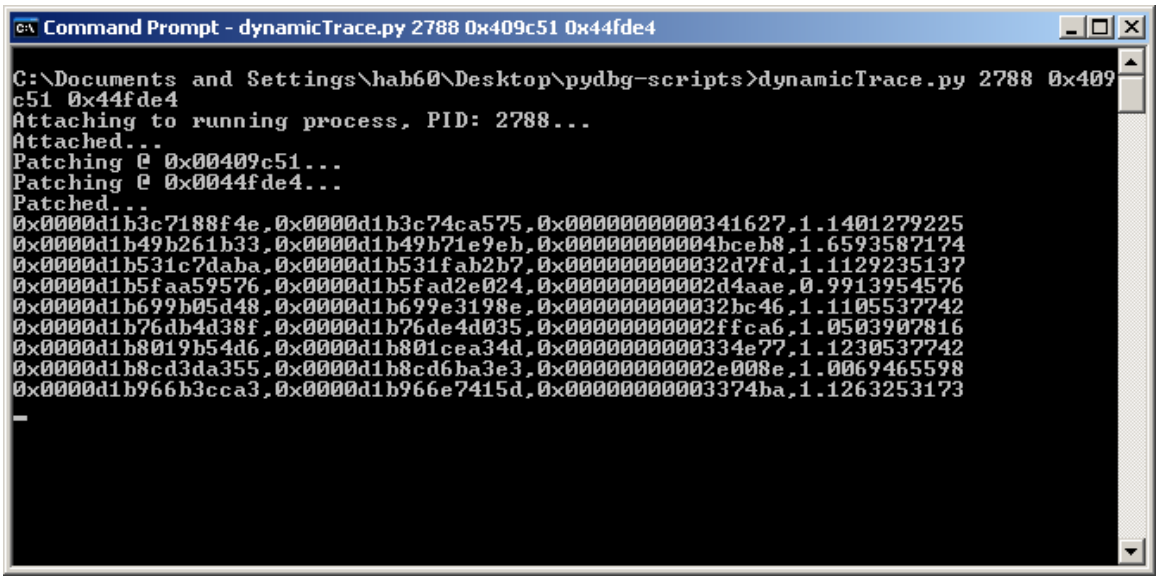

Figure 27. Example Use of Timing Analysis Tool

3.7.4.1 Software-Based Testing Configuration

Software-based testing was performed for three variations on the configuration as shown in Table 2. The configurations tested here are considerably different from the network-based testing because all timing is done inside one CPU, and no outbound or inbound network resources are affected by OpenVPN tunneling. Configuration number one, "Unencrypted Tunnel," measures a baseline of absolutely no protection. Configuration number three, "Public-key Encryption," measures the most secure method tested. This method corresponds most closely to that used by Secure ICCP.

In addition to utilizing different configurations, a custom tool was written to simulate ICCP traffic. Each configuration adds an incremental amount of data to the overall packet size. Because ICCP has already included this data in its packets, a custom application was written to take into account this additional data while maintaining the size of the packets in order to simulate the average size observed from ICCP. This average value will change relative to the amount of information being sent between ICCP client and server. We will use our configuration for a baseline of a typical ICCP system. Over a run of approximately 1300 packets, the average size observed was 165 bytes. Accordingly, our tool will maintain a final size of 165 bytes, including the data added in each configuration.

Table 1. Software-Based Testing Configurations

Configuration Name	OpenVPN Tunnel	Encryption	Authentication
Tunneled	Yes	None	None
Default Encryption	Yes	Shared-key	Yes
Public-key Encryption	Yes	Public-key	Yes

3.7.4.2 Software-Based Testing Results

Each test consisted of sending 300 packets through the network stack. For each packet, the timing was calculated for the time taken to prepare the packet. This includes adding tunnel

information, signing the packet for authentication and encrypting the packet. The worst-case time for the unsecured tunnel was 1.39 ms, the best-case time was 0.80 ms, and the average-case was 0.96 ms. A full graph of the observed packet preparation times is shown in Figure 28. When OpenVPN was configured to sign and encrypt the packets with the default method (Blowfish in cipher-block-chaining mode), the average time grew slightly. The worst-case then became 1.17 ms, the best-case 0.89 ms, and the average-case 1.01 ms. For the default, shared-key encryption, OpenVPN and OpenSSL displayed a difference of 0.05 ms on average. When pushing ICCP through this same tunnel, no ill effects were observed, indicating that for a typical system 0.05 ms is an allowable delay. A graph of the observed preparation times for shared-key encryption and signing is given as shown in Figure 29.

Next, similar analysis was performed for OpenVPN in Public-Key Infrastructure (PKI) mode. This is very similar to that of ICCP, where authentication is based on certificates and encryption keys are negotiated at connection time. The per-packet preparation time for this test is shown in Figure 30. In this case delays again grew slightly. The worst-case timing was 1.13 ms, the best-case was 0.87 ms, and the average timing was 0.99 ms. Again, these values were within limits necessary for our ICCP configuration to work properly. The results from all three tests are summarized in Table 3.

Table 2. Software-Based Testing Results for Send Preparation

Configuration Name	Worst-case Time	Best-case Time	Average Time
Unencrypted Tunnel	1.39 ms	0.80 ms	0.96 ms
Default Encryption	1.17 ms	0.89 ms	1.01 ms
Public-key Encryption	1.13 ms	0.87 ms	0.99 ms

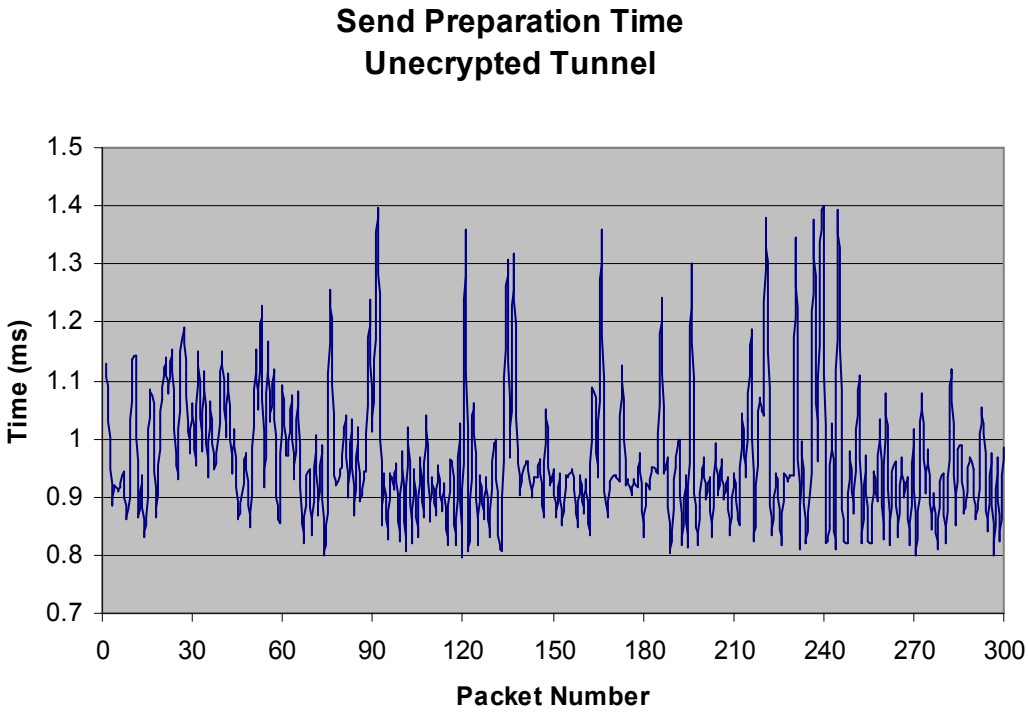

Figure 28. Observed Send Preparation Times for Unencrypted Tunnel

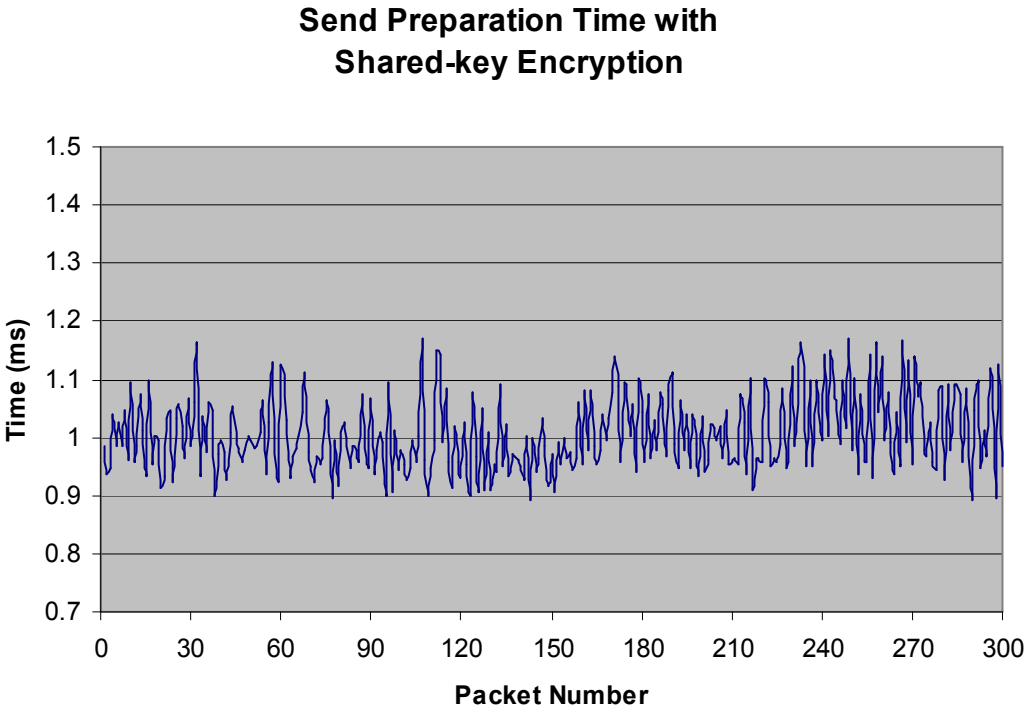

Figure 29. Observed Send Preparation Times for Shared-key Encrypted Tunnel

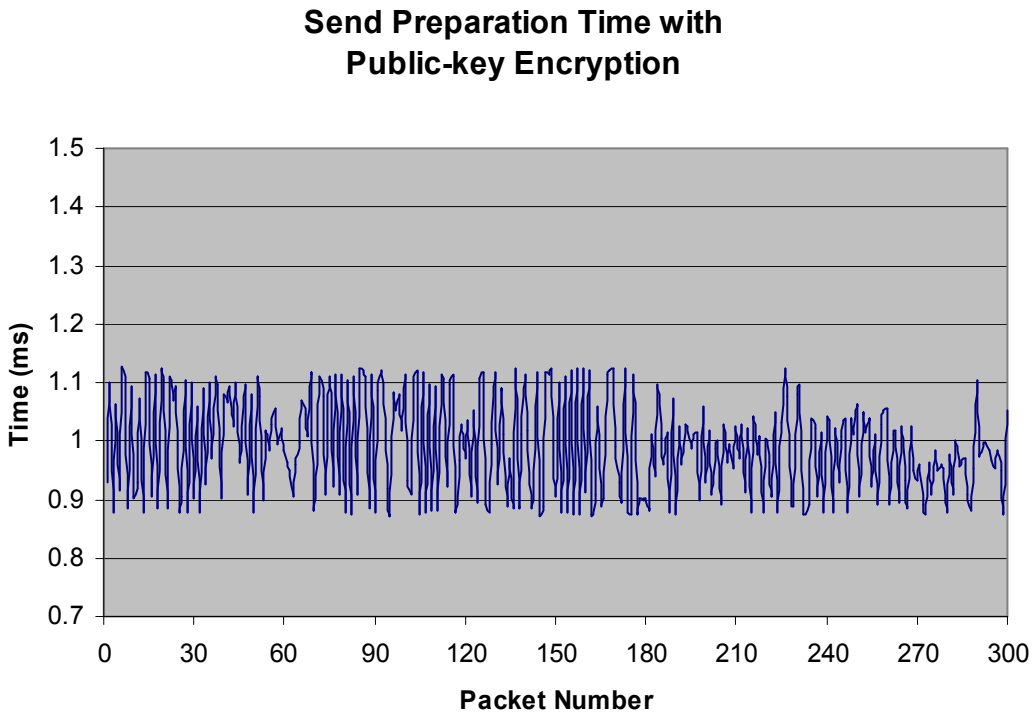

Figure 30. Observed Send Preparation Times for Public-key Encrypted Tunnel

Testing of the latency of send preparation for the packets determined that values were within limits necessary for normal operations. However, this does not provide adequate results if the receiving process timing (including possible decryption) was beyond limits. As mentioned earlier, most decryption algorithms perform as fast or faster than the corresponding encryption algorithm if the proper key is known. This leads to the expectation that the receiving process latency should be as small as or smaller than the send preparation time. Individual results for these tests are shown in Figure 31, Figure 32, and Figure 33. The most important details revealed during testing are that the worst-case timing is appropriate for proper functionality during either encryption type and, more importantly, the average timing is nearly identical in all three cases. The observed values for all three tests are summarized in Table 4.

Table 3. Software-Based Testing Results for Receive Processing

Configuration Name	Worst-case Time	Best-case Time	Average Time
Unencrypted Tunnel	1.05 ms	0.81 ms	0.84 ms
Default Encryption	0.85 ms	0.82 ms	0.84 ms
Public-key Encryption	0.85 ms	0.83 ms	0.84 ms

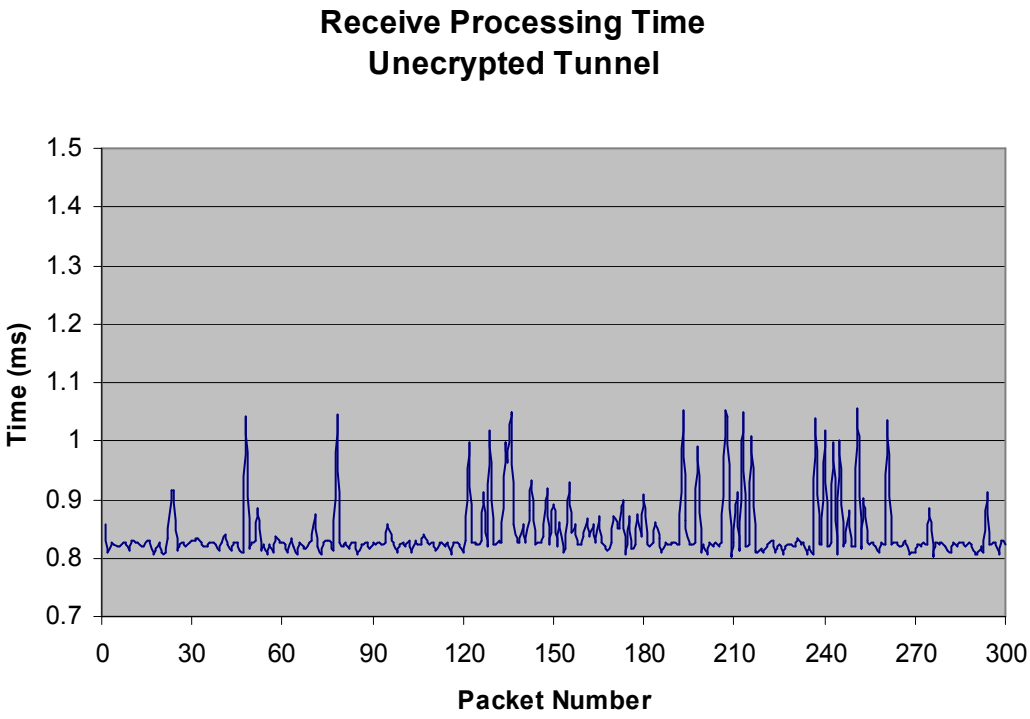

Figure 31. Observed Receive Processing Times for Unencrypted Tunnel

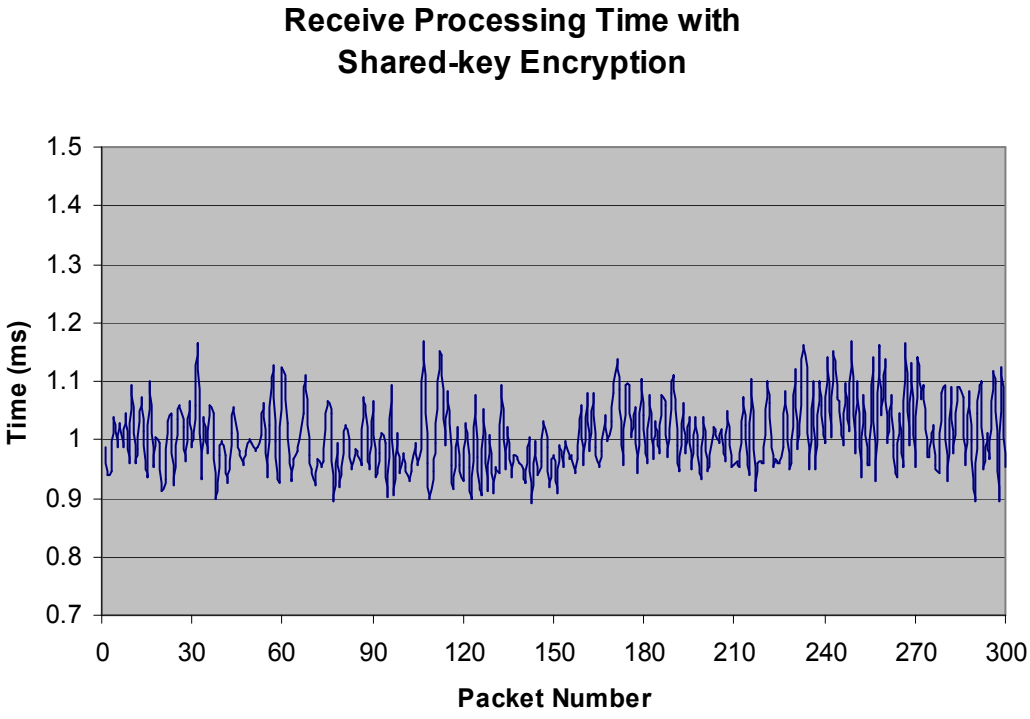

Figure 32. Observed Receive Processing Times for Shared-key Encrypted Tunnel

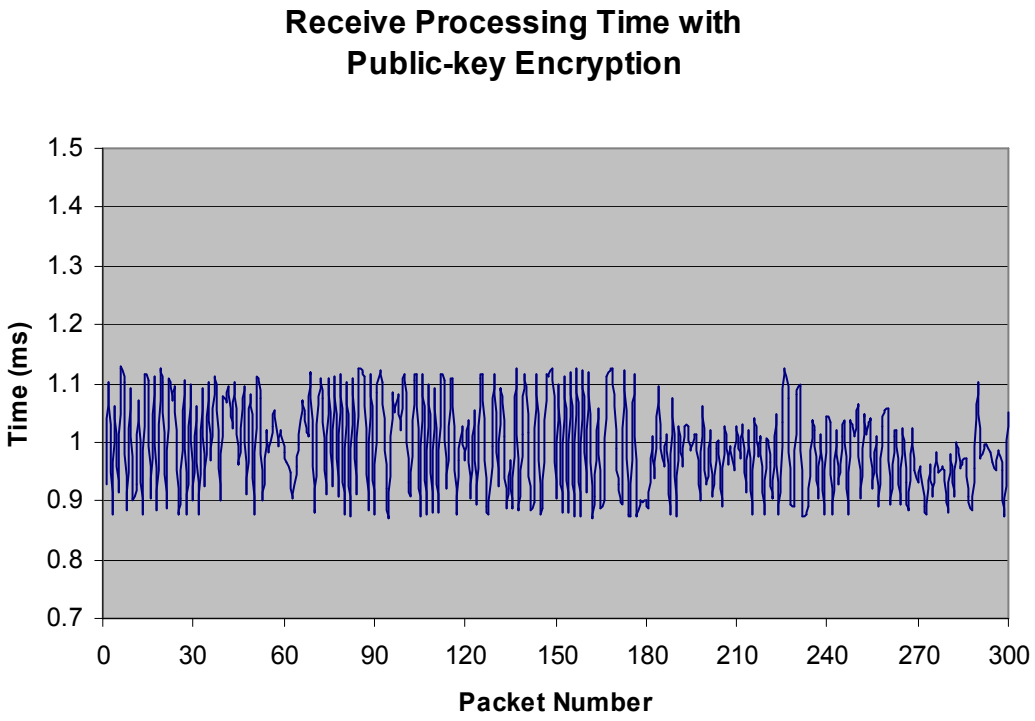

Figure 33. Observed Receive Processing Times for Public-key Encrypted Tunnel

In terms of additional system resources, the needs of OpenVPN were fairly minimal. The memory usage of OpenVPN never exceeded 4 megabytes and the processor usage averaged 2 percent. However, when using ICCP, most of this overhead is already included in the software itself. Switching from non-secure to Secure ICCP should have a smaller impact on these system resources than using OpenVPN with non-Secure ICCP.

3.7.5 Overall ICCP Performance Testing Summary

Testing of Secure ICCP features yielded several interesting results. Table 4 summarizes the key findings of the network-based testing results. The values shown here measure the latency of using an IPSec tunnel external to the ICCP endpoints. These values are not significantly affected by the security settings of ICCP and may be implemented in addition to or instead of built-in security features of Secure ICCP. The overall observed average difference in latency of the Cisco 3600-based IPSec tunnel was 5.51 milliseconds. Comparisons of the end-to-end and end-to-midpoint timings, for both non-secure and secure modes, are shown in Figure 34 and Figure 35.

Table 4. Network-Based Testing Results

Configuration	ICCP Client to ICCP Server	ICCP Server to ICCP Client	Average
End-to-End Non-Secure	**0.52 ms**	0.66 ms	0.59 ms
End-to-End Secure	6.19 ms	**6.01 ms**	6.10 ms
End-to-End Difference	5.67 ms	5.35 ms	5.51 ms

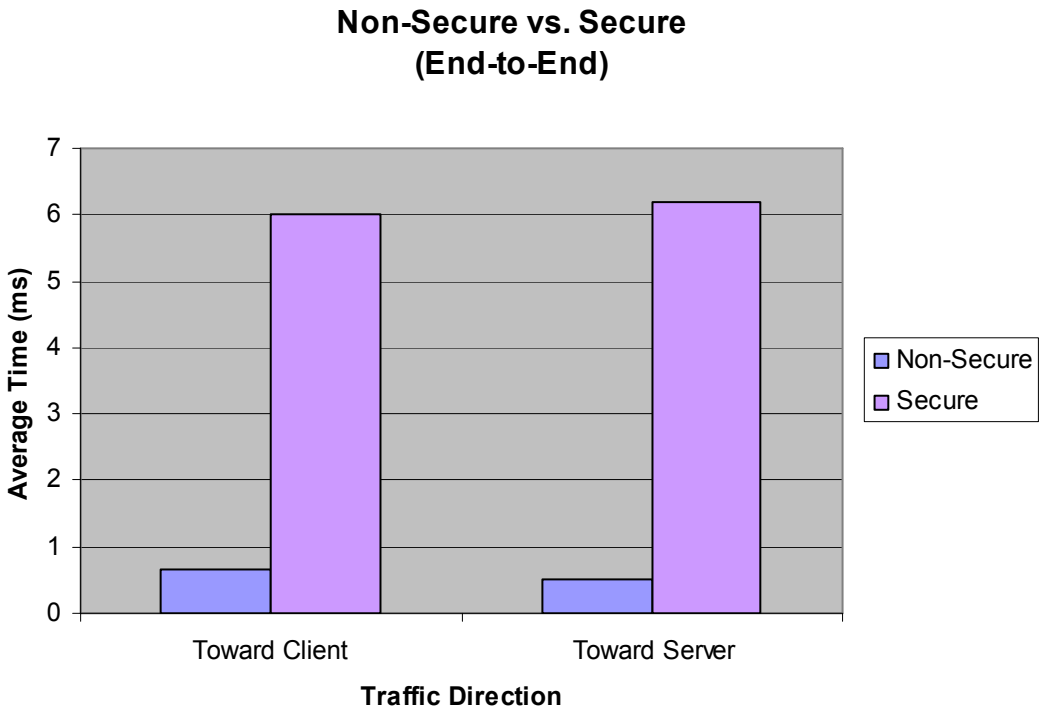

Figure 34. Non-Secure versus Secure End-to-End Latency with Cisco 3600 VPN

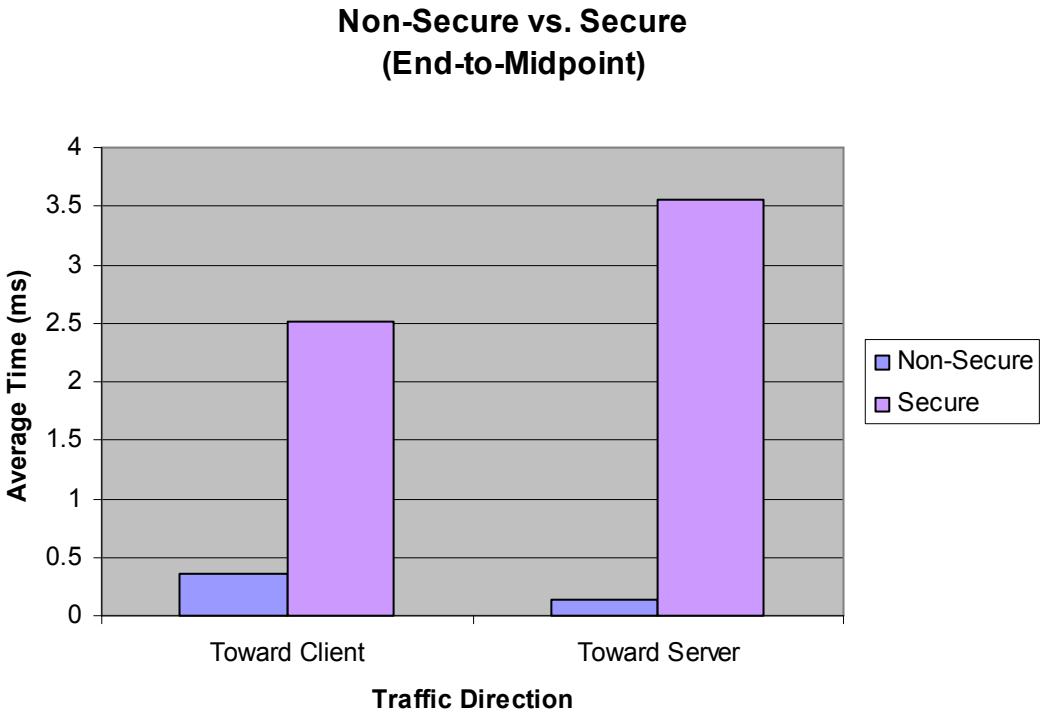

Figure 35. Non-Secure versus Secure End-to-Midpoint Latency with Cisco 3600 VPN

Table 5 shows the key findings of the software-based testing results. The values shown in bold are the minimum values for each category in the table. It can be seen that the additional latency experienced due to the addition of packet tunneling, signing, and encrypting was 0.05 milliseconds at worst from an unsecured tunnel and 1.01 milliseconds from traditional plaintext transmission. Each value is affected by the processor speed of the machine performing the OpenVPN processes.

The values shown here are for a modern machine running Windows Server 2003. Secure ICCP timing will not be identical to the values shown, but it is based on the same software (OpenSSL) as OpenVPN. This provides a strong estimate of the actual latency experience with Secure ICCP. Additional graph-based comparisons of the difference in average timing are shown in Figure 36 and Figure 37. The difference in latency in all cases is on the scale of microseconds (µs). When summed together, the send preparation and receive processing latencies provide the overall latency of a particular configuration. For the public-key configuration, which is most similar to Secure ICCP, this gives an average latency of 1.83 ms.

Table 5. Software-Based Testing Results

Configuration Name	Average Send Preparation Time	Average Receive Processing Time
Unencrypted Tunnel	**0.96 ms**	**0.84 ms**
Default Encryption	1.01 ms	**0.84 ms**
Public-key Encryption	0.99 ms	**0.84 ms**

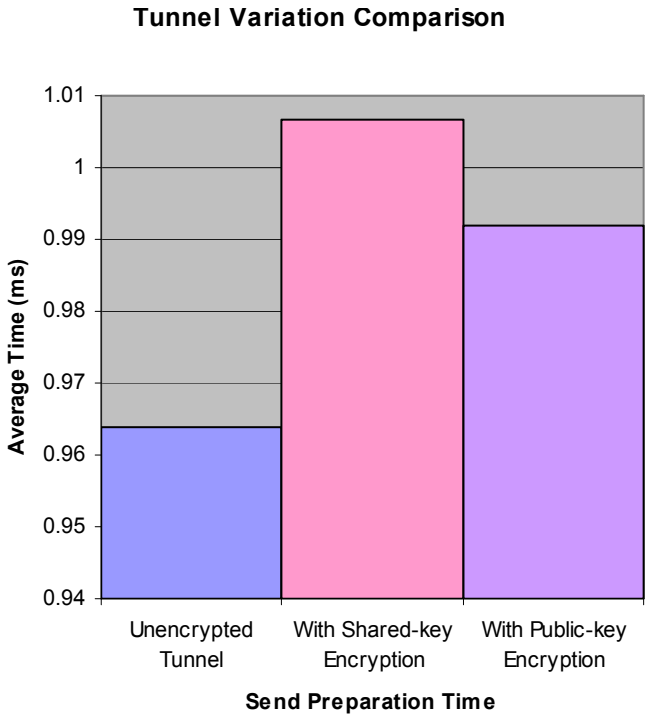

Figure 36. Average Observed Send Preparation Time by Configuration

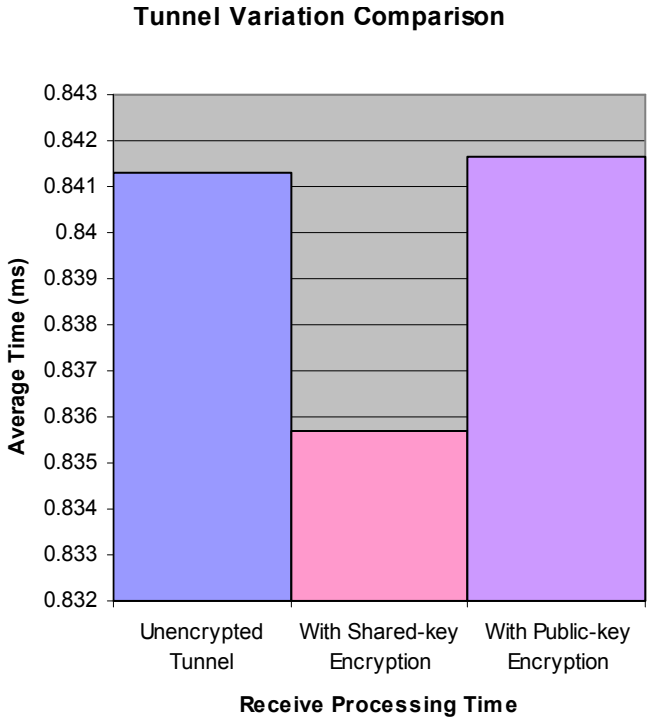

Figure 37. Average Observed Receive Processing Time by Configuration

A key observation to take away is that OpenSSL-based OpenVPN on modern computing resources performs better than Cisco 3600 routers without the appropriate hardware modules. The difference of 5.51 ms with the Cisco versus 1.83 ms with the OpenVPN in public-key mode is evidence of this fact. Accordingly, utilizing the security features of Secure ICCP, which in our testing environment is also OpenSSL-based, will likely yield similar results. However, it should be noted that using the appropriate hardware encryption module with the Cisco 3600 would possibly level these values, if not allow it to outperform OpenSSL. Also note that the hardware encryption modules may be more cost-prohibitive than Secure ICCP in many cases.

In addition, experimentation has shown that neither of these solutions have noticeable negative impacts on the operation or performance of the ICCP client or server. If absolute best performance is a primary concern and costs prohibit hardware-enabled IPSec tunneling, Secure ICCP is the clear choice.

3.7.6 Frame Relay Performance Discussion

The following provides a discussion of some of the important aspects of identifying a Frame Relay encryption product that can be used for layer 2 communication protection. Specific performance data for a Frame Relay encryption schemes were not performed in the NSTB laboratory.

When implementing a Frame Relay encryption process it is important to make sure the data throughput is not significantly reduced while being injected into the network. A properly selected system should add minimal overhead to each frame while performing the encryption and decryption activities. For most applications, standalone Frame Relay encryptors provide

higher performance values than those integrated into Frame Relay access devices (FRADs) or than software encryption schemes. When surveying Frame Relay encryption products, take note on how the data processing is implemented. Devices with multiple processors that can off-load tasking or provide parallel tasking have better response figures than single processor techniques. Also, designs that incorporate Very Large Scale Integration techniques for providing signal processing in firmware can provide tremendous efficiency to encryption algorithm processing. The processing speed and the instructions per seconds, normally measured in millions (MIPS) can also provide a means of comparing and contrasting different product designs.

One of the attributes of a high-performance Frame Relay encryptor is the ability to process the frames at full line rate. What this implies is originating Frame Relay data on the local area network is encrypted and pushed out on the wide arena network at the transmission rate available at the WAN interface. But when comparing Frame Relay products associated with line rate throughput, it is important to understand the aspect of this statement. The important aspect is the size of the frames that were used to measure this performance - not all frames are created equal. For example two products may claim that they can process Frame Relay data at a full line rate of, for example, 2.048 Mbps full duplex (E1 rate). That statement alone is not sufficient to determine if both products are equal in performance. It is important to note the size of each frame used during the test or calculation. Some of these figures are based on minimum Frame Relay byte size of 64 bytes. With larger frame sizes the “line rate throughput” will not hold up, and buffering will be needed.

Another aspect of Frame Relay performance is associated with a service provider’s service level agreement (SLA). Within the contents of an SLA are references to an attribute called bursting. A bursting allocation provides the user with the ability to take advantage of the Frame Relay WAN during light loading periods for no additional cost. For example, an SLA can be constructed such that a service provider could offer a service rate of 512 kilobits/second with a burst rate up to a T1 rate of 1.544 megabit/second available on an intermittent time frame when network bandwidth is available. Note that the burst rate would not be part of the Committed Information Rate (CIR), and there will always be a probability that frames may get discarded during times when the burst rate is being utilized.

Depending on the encryption process being deployed, the potential for dropping frames in an encrypted Frame Relay environment can cause synchronization and recovery degradation for some Frame Relay encryption schemes. Encryption techniques that are reliant on fixed order and small inter-frame delays require renegotiation of crypto handshakes to recover from lost frames. These implementations do not include crypto header information to be inserted for each frame, relying mostly on frame arrival timing. Crypto header information allows for more resiliencies during times of dropped or re-ordered frames that occur during transmission. Encryption without encryption headers should not use any “bursting” features that may be offered by carrier providers.

4 Conclusions

4.1 Conclusions relating to Overall ICCP Network System Design

4.1.1 Design Components

This report discusses design considerations and components. Its primary contribution is to identify and discuss security technologies that can enhance a network administrator's ability to protect the network, particularly in the context of ICCP.

4.1.2 Conclusions Relating to Quality of Service and Service Level Agreements

The primary issue here is the requirement that ICCP data traffic have a certain end-to-end quality of service. Achieving this requirement means accommodating the Wide Area Network (WAN) that will provide the node interconnections. One of the primary weaknesses of a routing approach is that the most efficient and highly available routes will, over time, become congested, as discussed in section 3.1.4, *SCADA Wide Area Networks*. Without some means of dealing with this congestion, communication between participating end nodes, e.g. SCADA control centers, can be severely delayed or/and lost. This would constitute denial of service (DoS) even if no active denial were occurring. Section 3.1.5, *IP Congestion and QoS management*, points out the importance of service level agreements (SLAs) with WAN providers to protect in-transit ICCP traffic and identifies important SLA attributes associated with providing a level-of-service guarantee for ICCP data streams.

4.2 Conclusions Relating to Secure ICCP Certificate Management

Section 3.4, *ICCP Use of Public Key Infrastructure Certificates*, and Section 3.5, *Secure ICCP Certificate Management Issues*, discuss the ramifications of various approaches to PKI certificate management in the context of ICCP.

4.2.1 PKI domain design Conclusions

Before the integration of a PKI solution for the distribution of Secure ICCP certificates, an architecture must be identified. Based on best practice implementations, two primary PKI domain designs, a flat hierarchy and a tiered hierarchy, are identified and analyzed. For control systems within an established domain, a flat hierarchy is preferred for the distribution of authentication certificates. This preference is based on the number of endpoints sharing ICCP data. For the most part, the control system networks are more isolated and generally small (at most a few hundred nodes) and as such, lend themselves better to flat hierarchies. The advantage is that only one CA needs to be established for everyone on the internal domain network, reducing the complexity of the configuration. In a tiered approach, each company would maintain its own CA, a proposition that is likely cost-prohibitive and more complex managerially.

4.2.2 Inter-Domain Communication Conclusions

Another important issue associated with the introduction of certificates for the authentication of Secure ICCP end nodes is the requirement for authenticating end-nodes between different communication domains. This also engenders the need to identify an architecture for its construction and a management approach for its execution. The preferred architecture for inter-domain communication is a tiered hierarchy. This is based on a desire to provide the most secure implementation. Creating a single "root" CA allows more restrictive security policies to be enforced at the root while alleviating some of stringent security requirements on subordinate CAs.

4.2.3 Secure ICCP application issues

Current implementations of certificate-based schemes within ICCP applications are primarily static. This implies that any certificate update or renewal process requires action by an operator. This mechanism does not fit modern techniques of end node authentication. Web-based forms of certificate authentication do not require that computers involved in the process be informed of the certificate update because the new certificate is sent at the beginning of each SSL handshake. Because an initiating node sends its certificate at the beginning of each session, no node should need to store local copies of anyone else's certificate. Therefore, when a node is issued a new certificate for any reason (expiration, key update, etc.), the operation is transparent to other nodes in the network and they do not need to be notified. These techniques should be designed into all applications intended to support Secure ICCP.

4.3 Conclusions Relating to Transition from ICCP to Secure ICCP

For some utility sites conversion from standard ICCP to Secure ICCP will occur over time, which implies that ICCP and Secure ICCP will coexist in some networks. Section 3.6, *Strategy for the transition from ICCP to Secure ICCP* describes configuring network connections to provide mixed-mode operation when both secure and non-secure forms of ICCP co-exist on a network.

For sites that must provide security without using Secure ICCP, either because they are in transition or because they do not plan to upgrade to the secure form of ICCP, This section also discusses potential alternatives to assure ICCP data protection. Both IPSec (see section 3.6.1, *Layer 3 Link Protection*) and data link encryption (see section 3.6.2, *Layer 2 Link Protection*) can provide protection for in-flight ICCP data.

4.4 Performance of Networks Incorporating Secure ICCP

Measurements were taken to characterize the impact of using different security layers associated with securing the ICCP data. The processing and transport delays were characterized to provide the user with a sense of the operational impact when adding a technology to the protection of ICCP. Associated implementations such as OpenSSL, for characterizing Secure ICCP, and IPSec, for characterizing a Layer-3 encryption, are documented. The overall results show that the integration of secure protocols had minimum effect on the end-to-end performance of an application, but the overall management complexity increased with each added layer of protection.

5 Recommendations

Secure ICCP Certificate Management

PKI domain design

Before the integration of a PKI solution for the distribution of Secure ICCP certificates, an architecture must be identified. Based on best-practice implementations, two primary PKI domain designs were identified, a flat hierarchy, and a tiered hierarchy. For control systems within an established domain, a flat hierarchy was recommended for the distribution of authentication certificates. This recommendation was based on the number of endpoints sharing ICCP data. For the most part, the network is more isolated and generally small (at most a few hundred nodes) and as such, it lends itself better to a flat hierarchy. The advantage is that only one CA needs to be established for everyone on the internal domain network reducing the complexity of the configuration. In a tiered approach, each company would maintain its own CA, a proposition that is likely cost-prohibitive and more managerially complex.

Inter-Domain Communication

Another important issue associated with the introduction of certificates for the authentication of Secure ICCP end nodes is the requirement for authenticating end-nodes between different communication domains. This issue also revolves around the need to identify an architecture for its construction and a management approach for its execution. The architecture recommended for inter-domain communication was a tiered hierarchy. This recommendation was based on the need to provide the most secure implementation. Creating a single "root" Certificate Authority (CA) allows more restrictive security policies to be enforced at the root while alleviating some of stringent security requirements on subordinate CA's.

Secure ICCP application issues

Current implementations of certificate-based schemes within ICCP applications are primarily static in nature. This implies that any certificate update or renewal process requires actions by an operator. This mechanism does not fit modern techniques of end node authentication. Web based forms of certificate authentication do not require machines (computers) to be informed of the certificate update because the new certificate will be sent at the beginning of each SSL handshake. Because a node's certificate is sent at the beginning of each session, nodes should not need to store local copies of anyone else's certificate. Therefore, when a node is issued a new certificate for any reason (expiration, key update, etc.), the operation is transparent to other nodes in the network and they do not need to be notified. It is recommended that these techniques be designed into all supporting Secure ICCP applications.

Network System Design

Design Components

Within the report, design considerations and components of the design were discussed. The primary observations were associated with the identification and integration of security

technologies that can enhance the ability of a network administrator in the protection of the network.

Quality of Service and Service Level Agreements

The primary issue associated within this topic is the requirement for providing ICCP data traffic an end-to-end quality of service. This must account for the Wide Area Network (WAN) that will provide the node interconnections. One of the primary weaknesses of a routing approach is that the most efficient and highly available routes will, over time, become congested. Without a means of accommodating for this congestion, communications between participating end nodes, i.e. SCADA control centers, can be severely delayed or/and lost creating a denial-of-service (DoS) situation. The recommendation provided within this section was the identification of important attributes for the creation of service level agreements (SLA's) with WAN providers to protect in-transit ICCP traffic. These attributes were associated with providing a level of service guarantee for ICCP data streams.

Transition Strategy

Layer 2 & Layer 3 protection mechanisms

For some utility sites the conversion from the standard ICCP to the secure version will not be rapidly achieved. For those sites that do not plan to upgrade to the secure form of ICCP, a section of this report discussed some potential alternatives to provide the security needed to assure ICCP data protection. Both IPSec and data link encryption were suggested as means to provide the necessary data surety for the protection of in-flight ICCP data.
A technique was also described to configure a network connection to provide a mixed-mode operational scenario when both secure and non-secure forms of ICCP co-exist on a network.

Performance

Measurements were taken to characterize the impact of using different security layers associated with securing the ICCP data. The processing and transport delays were characterized to provide the user with a sense of the operational impact when adding a technology to the protection of ICCP. Associated implementations such as OpenSSL for characterizing Secure ICCP, and IPSec for characterizing a Layer-3 encryption were documented. The overall results showed that the integration of secure protocols had a minimum effect on the end-to-end performance of an application, but the overall management complexity increased with each added layer of protection.

Appendix A: References

[1] T. Saxton, D. Ambrose, and F. Kendall, *ICCP User Guide*, prepared for the Electric Power Research Institute by KEMA-ECC, October 1996. http://www.sisconet.com/downloads/usrguid5.doc

[2] E. Rescorla, *SSL and TLS: Designing and Building Secure Systems*, Addison-Wesley, 2001.

[3] T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.1*; IETF RFC 4346, April 2006.

[4] B. Stojkovic and M. Vukasovic; "A new SCADA system design in the power system of Montenegro – ICCP/TASE.2 and Web-based real-time electricity demand metering extensions", *Proc. IEEE/PES Power Systems Conference and Exposition*, October 2006.

[5] R. Lee, H. Cal, A. Beard, and J. Scott, *ICCP Implementation for Distributed Control Systems*, Public Service Company of New Mexico, January 2005. http://www.livedata.com/images/stories/pnm_iccp.pdf

[6] C.-C. Lin and S.-J. Huang; "Enhancement of power system communications via real time transport protocol", *Proc. IEEE Power Engineering Society Winter Meeting*; January 2002.

[7] D. Popovic; "Open electric power control system communication"; *Elektroprivreda*; vol.51, no.2, p.27-33; April-June 1998.

[8] G. Maestri and P. Scalera; "Network reliability and continuity of supply"; *Elettrificazione*; vol.56, no.1-2, p.70-4; January-February 2006.

[9] J. Weiss; "Information security needs for T & D equipment"; *Proc. IEEE Power Engineering Society Transmission and Distribution Conf.*; January 2002.

[10] J. Hughes (proj. mgr.), *The Integrated Energy and Communication System Architecture*, Electric Power Research Institute Inc., 2004.

[11] J. Hamilton; *The Use of Authentication across Borders in OECD Countries*; Industry Canada; OECD Document JT00194846; November 2005.

[12] E. Byres, D. Hoffman and N. Kube; "On Shaky Ground – A Study of Security Vulnerabilities in Control Protocols", *Proc. 5th American Nuclear Society Int. Mtg. on Nuclear Plant Instrumentation, Controls, and HMI Technology*; November 2006.

[13] S. Bradner, *The Internet Standards Process, Rev. 3*, IETF RFC 2026, October 1996.

[14] *Infrastructure Protection Challenges and Efforts to Secure Control System*, Report to Congressional Requesters, GAO-04-354, March 2004.

[15] *Roadmap to Secure Control Systems in the Energy Sector*, U.S. DOE and U.S. DHS, prepared by Energetics Incorporated, January, 2006. http://www.controlsystemsroadmap.net/

[16] *The Changing Structure of the Electric Power Industry 2000: An Update*, Energy Information Administration, October 2000.

[17] J. Michalski, C. Price, E. Stanton, E. Lee, K.S. Chua, Y.-H. Wong, and C.-P. Tan; *Network Security Mechanism Utilizing Network Address Translation LDRD project*, SAND Report, SAND2002-3613, November 2002.

[18] Single-User Network Access Security TACACS+, Cisco Systems Inc. Copyright 1996. http://www.cisco.com/warp/public/614/7.html

[19] C. Rigney, S. Willens, et al. *Remote Authentication Dial In User Service (RADIUS)*, IETF RFC 2865, June 2000.

[20] D. McDysan, *QoS & Traffic Management in IP and ATM Networks*, McGraw-Hill, 2000.

[21] K. Nichols, S. Blake, F. Baker, and D. Black; *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, IETF RFC 2474, December 1998.

[22] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus; *Requirements for Traffic Engineering Over MPLS*, IETF RFC 2702, September 1999.

[23] *Configuring and Troubleshooting Frame Relay*, Document ID 16563, Cisco Inc., updated November 2005. http://www.cisco.com/warp/public/125/12.html

[24] C. Brown, A. Malis, *Multiprotocol Interconnect over Frame Relay*, IETF RFC 2427, September 1998.

[25] M. Rose and D. Cass, *ISO Transport Service on top of the TCP Version 3*, IETF RFC 1006, May 1987.

[26] S. Kent and K. Seo, *Security Architecture for the Internet Protocol*, IETF RFC 4301, December 2005.

[27] S. Kent, *IP Encapsulating Security Payload (ESP)*, IETF RFC 4303, S. Kent, December 2005.

[28] V. Manral, *Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)*, IETF RFC 4835, April 2007.

[29] C. Kaufman (ed), *Internet Key Exchange (IKEv2) Protocol*, IETF RFC 4306, December 2005.

[30] R. Housley, W. Polk, W. Ford, and D. Solo, Internet *X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, IETF RFC 3280, April 2002.

[31] W. E. Anderson, J. Michalski, and B. Van Leeuwen, *Enhancements for Distributed Certificate Authority Approaches for Mobile Wireless Ad Hoc Networks*, SAND Report, SAND2003-4395, December 2003.

[32] M. Meyers, R. Ankey, A. Malpani, S. Galperin, and C. Adams, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol (OCSP)*, IETF RFC 2560, June 1999.

Appendix B: Security Technology

1 Public Key Cryptography

In modern information systems, regardless of specific security or cryptographic needs, the security supplied by cryptography is provided by a *secret key*. In the traditional model, when two parties Alice and Bob want to communicate, they share a common secret key. Using that shared secret key and a corresponding cryptographic algorithm (e.g. an encryption algorithm), Alice and Bob can secure their communications by processing (e.g. encrypting) their messages with that key. Since both Alice and Bob use the same secret key to secure their communications, this approach is called *symmetric key* cryptography.

The obvious question is: How do Alice and Bob establish or agree upon that shared secret key? To ensure the security of the key, it must be distributed over a secure, out-of-band channel. Unfortunately, these requisite secure channels are difficult to establish and sustain. For instance, it could be required that all communicating parties meet face to face in order to establish a shared secret key, but this requirement is often onerous and impractical. Consequently, the symmetric key model is simple and works fine for small situations; however, it has serious scaling issues. If Alice wants to communicate with other people besides Bob, she needs to establish a new shared key with every person with whom she wants to communicate. Bob must do the same. As networks become large, it soon becomes excessively difficult to negotiate a key with everyone in a secure out-of-band fashion.

Public key cryptography was designed in part to solve this key management problem. The Internet's Public Key Infrastructure (PKI), the Internet Key Exchange (IKE), is described in [29]. Public key cryptographic algorithms are distinguished by the fact that the algorithms use two different keys, one of which is kept secret, while the other is made public. The keys are mathematically related to each other, and there is a unique private key associated with each public key. For encryption schemes, the encryption key is made public while the decryption key is kept secret and known only to its owner. If Alice wants to send an encrypted message to Bob, she can use Bob's public key to encrypt the message. Bob (and only Bob) knows the corresponding private key, so only he can decrypt the message from Alice. Since Bob's public key is publicly available, there is no need for Alice to exchange keys with Bob. By allowing Bob's encryption key to be publicly known to anyone (even an attacker), the key management problem of symmetric keys is avoided.

2 Public Key Infrastructure

2.1 Registration Authority

In a public key system, each entity is bound to its own public key/private key pair (or pairs). This system is commonly known as a Public Key Infrastructure (PKI). A fundamental requirement of any PKI is a mechanism by which public keys can be distributed and bound to their owner. This binding is done via digital certificates. While digital certificates come in different formats, they often contain the following fields: the identity of the owner of the

public key; the public key itself; the intended use of the key; the validity period of the certificate; and the identity of the certificate issuer. There have been some attempts at standardization of certificates. The International Telecommunication Union (ITU) issued X.509, a standard for PKIs which includes a specified digital certificate format [30]. A digital certificate is valid only if it is signed by a trusted Certification Authority. The public key infrastructure governs both the use and management of these digital certificates.

The following are core components of a PKI.

2.2 Registration Authority (RA)

The Registration Authority (RA) registers public key owners by creating their digital certificates, which are then signed by the Certification Authority. The RA confirms the identity of the public key owner and may even generate the keying material on behalf of the owner. The requirements for verifying owner identity vary among different Registration Authorities. Thus, the trust in the digital certificate is dependent on the trust in the RA's identification verification procedures. Furthermore, if the RA generates the keying material on behalf of the public key owner, this may undermine the non-repudiation aspect of public key cryptography since both the registration authority and the public key owner will have access to the secret keying material.

2.3 Certification Authority (CA)

The Certification Authority (CA) certifies the identity of a public key's owner by signing the digital certificate generated by the RA. The validity of the CA's signature is verified using the CA's public key, raising the obvious question: "Who certifies the Certification Authority?" As will be described below, there may be a hierarchy in Certification Authorities, and the public key of a given CA is verified by another CA.

2.4 Certificate Repository

For public key cryptography and key exchange to work, the public keys and digital certificates must be available to users. One option is to store these keys and certificates in a Certificate Repository. Examples of Certificate Repositories include the X.509 server, the LDAP server, and corporate databases. SSL takes an alternative approach. In SSL, certificates are exchanged by the clients at the beginning of each session, so no Certificate Repository is needed.

2.5 Certificate Revocation Mechanism

When a public key pair is compromised, or when there is a change in any digital certificate field, the certificate needs to be revoked and placed on a Certificate Revocation List (CRL). Certificate validity should be checked against these lists whenever a certificate is used, but in practice this is rarely done. Issues associated with using a CRL are discussed in section 5.3.1, *Certificate Revocation Lists*, of this appendix, *Appendix B:* Security Technology.

When two nodes need to communicate in a secure manner over SSL, each must first authenticate the other to ensure the communication is not with an impostor. Digital certificates are the cornerstone of this authentication.

3 Certificates in the Secure Sockets Layer (SSL)

In SSL, the entity who initiates a network connection with a remote machine is called the *client*, and the remote machine that accepts the remote connection is referred to as the *server*. To be consistent with the SSL documentation, we will adopt the same terminology. When a client initiates an SSL session with a server, the pair exchanges a series of handshake messages that establish (1) their identities, (2) the preferred cryptographic algorithms, and (3) short-term symmetric keys that will be used to protect the newly-formed SSL session. The short term session key is sent by the client and is encrypted with the public key of the server. Public key cryptography is used primarily during this handshake phase to establish the temporary symmetric session keys, and all later communications are protected using those session keys with much faster symmetric key algorithms.

It should be noted that in the basic SSL handshake, the authentication is only one-way; the client authenticates (i.e., is assured of the identity of) the server, but the server does not authenticate the client. Fortunately, it is possible for the server to request client authentication as well. For the networks relevant to reader, this mutual authentication is likely to be required and should be used. In this discussion it is assumed that both client and server authenticate each other.

Regardless of who is authenticating whom, the procedure is fundamentally the same. Figure 38 depicts the process. In order for the server (S) to authenticate itself to the client (C), the server sends its certificate to the client. The client has a cached copy of the CA's trusted certificate, which it implicitly trusts. Since the client trusts the CA, the client will also trust certificates signed by the CA. Therefore, if the server possesses a valid certificate signed by the CA, the client can trust the certificate and likewise the server.

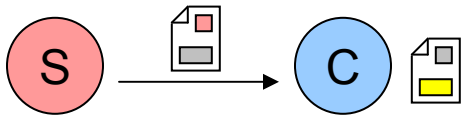

Figure 38. The server (S) sends its certificate to the client (C)

When the client receives the server's certificate, it checks whether the server's certificate was indeed signed by the CA using its copy of the CA's public key. If the certificate is invalid (e.g. the signature is incorrect, the name on the certificate does not match, the certificate has expired, etc.), the client will not trust the server and can end the SSL session. On the other hand, if the certificate was correctly signed by the trusted CA, the client can be assured that the server is the entity named in the certificate. In this case, the server is successfully authenticated to the client. The client can then discard the certificate and finish the SSL session handshake. Note that there is no need for the client to cache or store copies of the server's certificate.

This discussion of node authentication and the use of certificates in SSL is a simplification, but it highlights the fundamental issues. It should be noted that the client can identify the server only by the entity named on the certificate. It is therefore imperative that the names on the certificates be unique for every network entity. There are standards that define the specific naming conventions for these *distinguished names* (DNs). For example, a possible DN could be composed of the entity's country, organization, organizational unit, and common name, such as:

```
C=US, O=Sandia, OU=Security, CN=John Doe
```

These distinguished names are intended to prevent any naming collisions and to allow network entities to be uniquely identified. Any policy or access control decisions made at higher layers in the stack (e.g., the application layer) should usually base their policy on these distinguished names, *not* on the actual certificates. Certificates are short-term tokens that merely bind unique identities (as denoted by DNs) to public keys. Assigning privileges or access controls to certificates is something for which certificates were not intended. Instead, the policy should be mapped to the *identity* (e.g. to the DN) to which the certificate is bound.

4 Certification Hierarchy Schemes

Each node in the network needs to have access to a certification authority to receive the fundamental PKI services (e.g. signing, caching, and revoking certificates). There are novel ways to manage certificates in challenging environments [31], but the two basic ways in which the PKI certification hierarchy can be structured remain the *flat* hierarchy and the *tiered* hierarchy. In the simplest scenario, the entire network is serviced by a single root CA; each node communicates directly with the root CA to receive the required services. This is the the *flat* hierarchy. Alternatively, it may be desirable for each company on the network to manage its own nodes independently. The companies' individual PKIs are in turn certified by a single root CA. This is the *tiered* hierarchy. In the following subsections, each of these schemes is examined in detail along with its associated strengths and weaknesses.

4.1 Flat Hierarchy

In the most basic case, there is a single certificate authority that provides PKI services to the entire network. Each node, regardless to which company it belongs, connects directly to the lone CA to receive certificates. In this scenario, all nodes are assumed to have a direct connection to the CA. This flat PKI structure is depicted below in Figure 39.

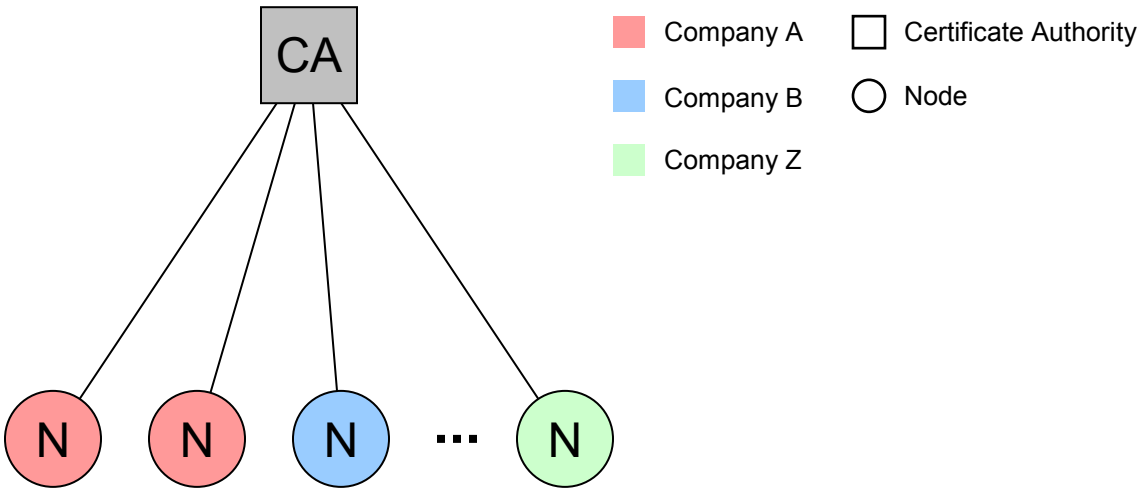

Figure 39. Flat certification authority hierarchy

In this scenario, validating a certificate is quite straightforward. Since all certificates are signed by the single CA, nodes need to cache only one trusted certificate, that of the CA. With that single trusted certificate, any other node's certificate can be verified, as shown below in Figure 40. Here, when Node-1 needs to authenticate itself to another node, it transmits its certificate. The authenticating node can then use its local copy of the CA's certificate, which contains the CA's public key, to check the signature on Node-1's certificate. If the signature verifies, the certificate is assumed to be valid. The rest of the handshake can then proceed to verify that Node-1 indeed has the key pair attested to by the certificate.

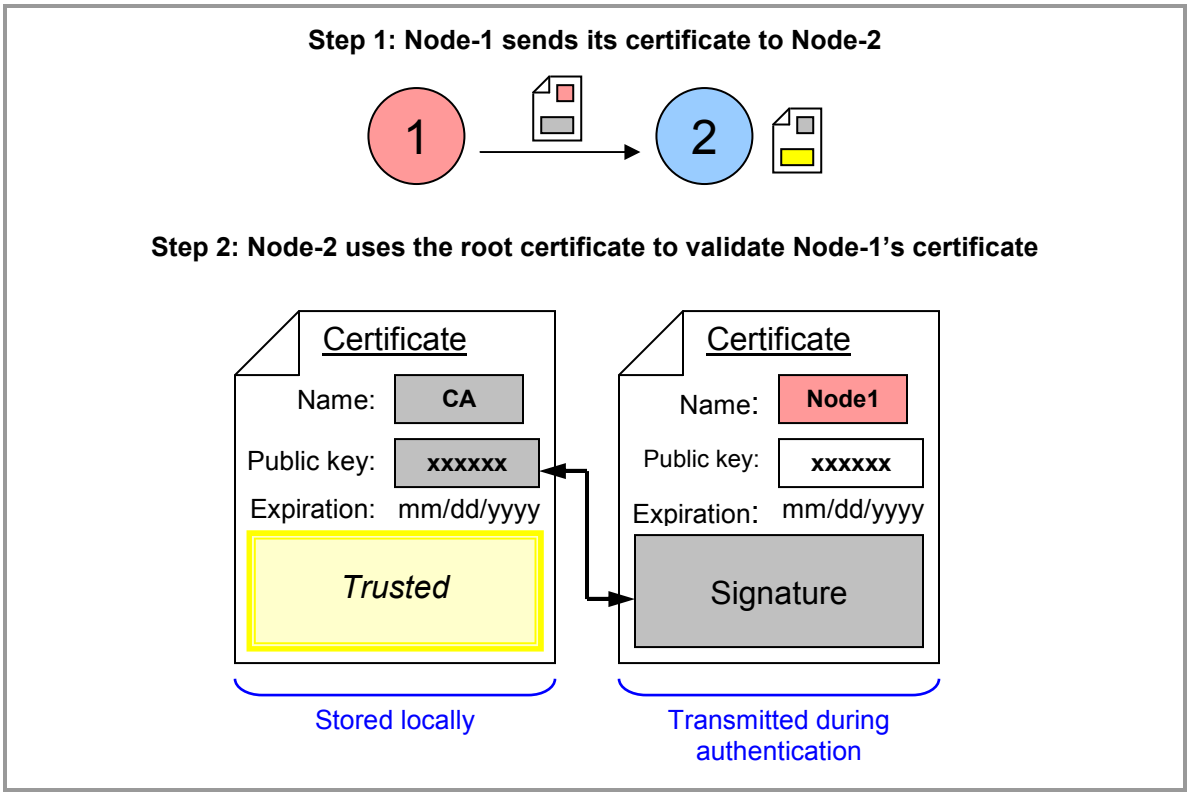

Figure 40. Certificate exchange in a Flat PKI

This centralized approach has several strengths and weaknesses:

Strengths

- Certificates are managed by one central site, relieves burden from individual companies
- A node needs to send only its own certificate in the handshake
- CRLs are simpler and valid across the system
- CRLs are managed at one central site
- Simplicity; getting PKI services is very straightforward

Weaknesses

- Does not scale to large networks (500+ nodes)

- Centralized solution provides a single point of failure
- "One size fits all" model of security for all nodes across different companies. Changes to the security policy must be more formal and restrictive since they affect all nodes.
- Companies must trust the single CA to manage everyone fairly
- Single node responsible for CRLs can experience heavy load
- The process of adding a node to the CRL can be complicated

As networks grow larger, the flat PKI structure becomes difficult for a single entity to manage and service. Furthermore, either out of convenience or distrust, organizations may prefer to manage their own PKI nodes themselves. To satisfy these issues, a tiered PKI hierarchy can be implemented.

4.2 Tiered Hierarchy

An alternative approach is to create a layered, or tiered, hierarchy of certificate authorities. In this model, each company runs its own CA that is responsible for providing PKI services for its own nodes only. For example, the PKI network would contains as many CAs as there were participating utility companies. Certificates held by each node are signed by the node's local (i.e company-specific) CA. Figure 41 depicts the tiered PKI structure with two levels of CAs. The top level is the root CA, and below each company (A-Z) has its own local CA that issues certificates for its own nodes.

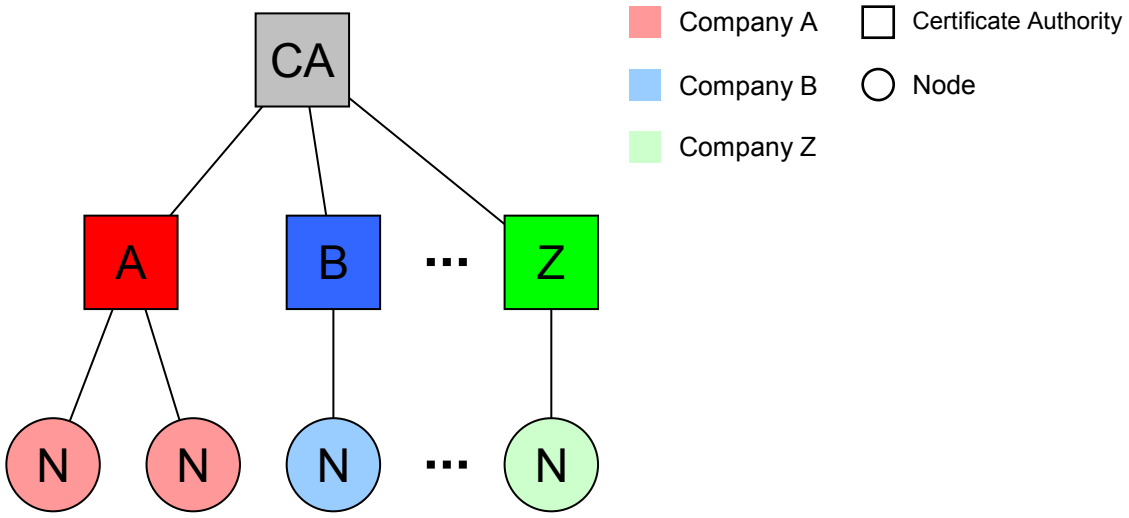

Figure 41. Tiered certification authority hierarchy

Since each company signs its own certificates, verification of cross-company (also called cross-domain) certificates is slightly different. If we use the same certificate exchange as before, when a node from company B receives a certificate signed by company A, it has no way of determining whether the signature is trustworthy. Company B only trusts itself and the root CA. As shown below in Figure 42, a node from company B (Node-2) has no way to verify a certificate from company A (Node-1). In the figure, Node-2 cannot verify the signature on Node-1's certificate because he does not trust company A's CA.

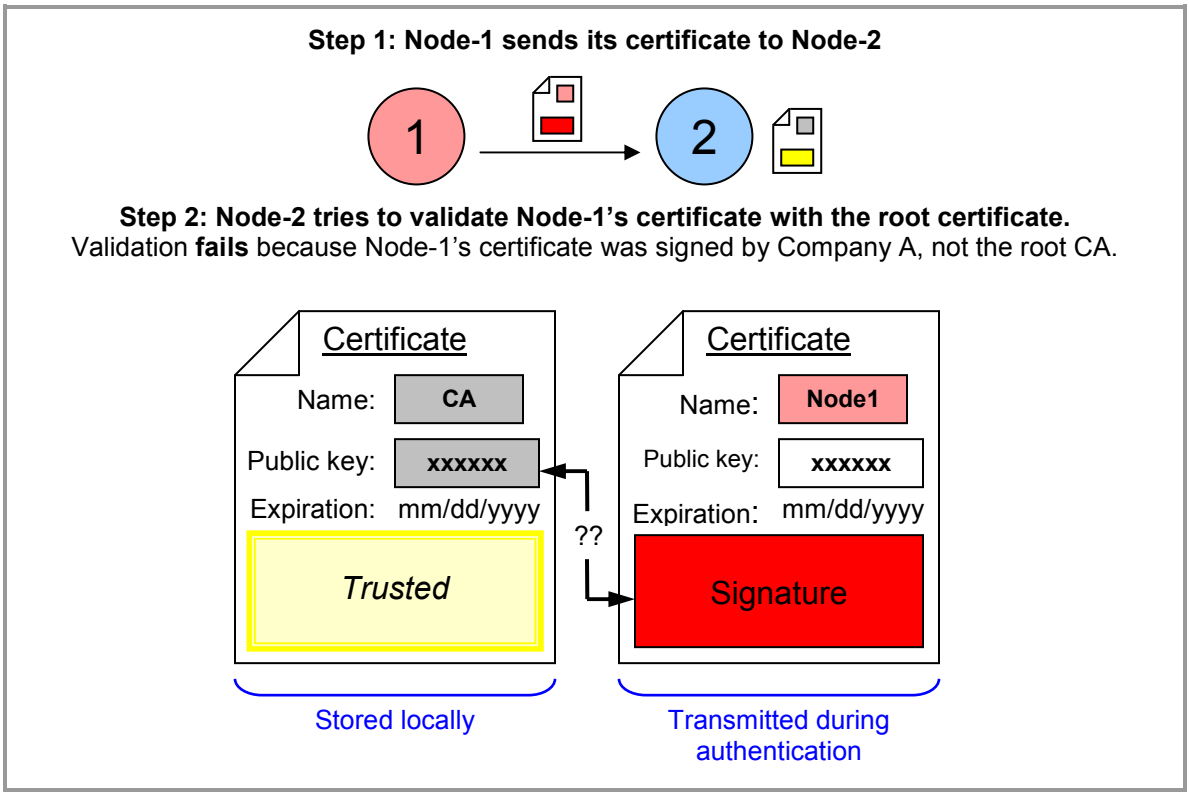

Figure 42. Incorrect certificate exchange in a tiered PKI

Instead, a *chain* of certificates must be transmitted to the verifying node. The chain of trust includes the node's certificate, signed by company A and its company A's certificate, which is signed by the root CA. Since the root CA is trusted, the verifying node at company B can verify that company A's certificate was signed by the trusted CA, and that Node-1's certificate was signed by the now trusted company A. The chain of certificate verification is illustrated below in Figure 43. While the chain of trust depicted below is only two certificates long, in practice the chain can be of any length.

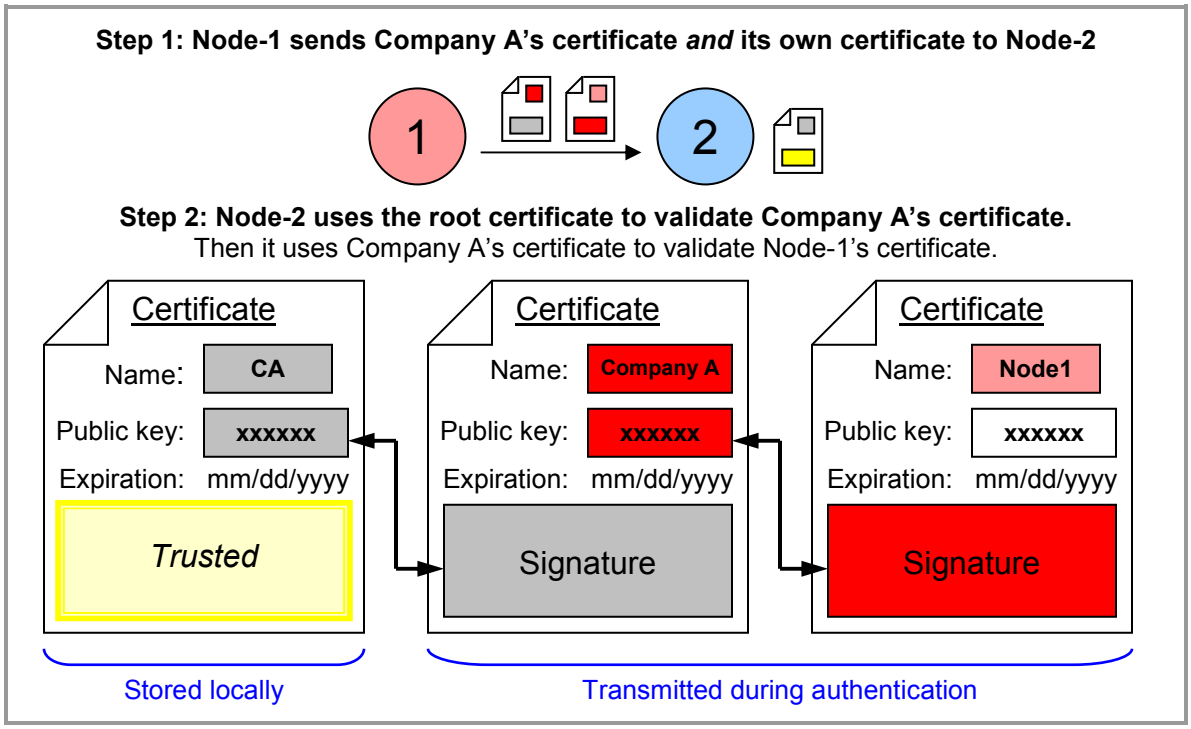

Figure 43. Correct certificate exchange in a tiered PKI

In Figure 43, Node-1 from company A sends two certificates to Node-2 at company B: (1) company A's CA certificate signed by the root CA, and (2) Node-1's personal certificate, signed by company A's CA. Node-2 can then use its local copy of the root CA's certificate to verify the identity of company A's CA. If verification is successful, Node-2 can now trust the certificate of company A's CA. Next, Node-2 uses the newly trusted copy of company A's CA certificate to verify the certificate of Node-1. If that second verification is successful, Node-2 can trust the certificate of Node-1.

This de-centralized approach has several strengths and weaknesses.

Strengths

- Scales very well for larger networks. There can be multiple tiers of CAs between the root CA and the end node.
- Each company can independently manage its own nodes
- There is much less stress on the single root CA
- The burden of circulating the CRL is no longer on a single node

- A failure or compromise at a company's CA will only affect that company's nodes

Weaknesses

- Complexity. This scenario is slightly more complex. The company CAs must occasionally have their certificates updated, which in turn must be distributed to all the nodes.
- There are multiple CAs that must be secured from failure and attack
- Each company must maintain its own CRL and distribute certificates.
- Multiple certificates must be passed to provide proper authentication
- Lower-tiered CAs must have reachback to the primary "root" CA.

5 Certificate Management

In the normal operation of a public key, certificates are constantly used and exchanged between nodes. The certificates are temporary objects that attest to the identity of a node. Over time, certificates will expire or be revoked, and a process must be in place to manage the system's certificates. This section describes the management of certificates in a typical PKI.

5.1 Certificate Issuance

The challenge in issuing certificates is having the CA verify that an entity is in fact who it claims to be. As such, to authenticate a physical entity, such as a control node computer, certificates cannot be issued online but rather require some level of physical interaction. For instance, if a node claiming to be "control Node-1" at Company B asks for a certificate online, the CA has no way of determining the real node from an impostor; the CA needs some type of physical interaction with "control Node-1" at Company B, such as hand-delivering the certificate, to be sure that the certificate was issued to the correct entity. *For the Utilities the Internet model of asking for and receiving an initial node certificate should be much more stringent than initiating a web browser application.*

5.2 Certificate Expiration and Renewal

Every certificate has a specific period of time for which it is valid. After that period of time elapses, the certificate expires and should no longer be accepted. The validity period of certificates is defined in the security policy of its surrounding PKI and is ultimately a policy decision made by the system administrators.

Regardless of how long the validity period is defined, certificates will eventually expire. When a node's certificate is due to expire, it is necessary for that node to request and be granted a new certificate. The node communicates with its issuing CA and asks for a renewed certificate with a new expiration date. Assuming the node is still a valid node, the CA will issue the node an updated certificate.

With the new certificate in hand, the node can start using it during the SSL handshake to authenticate itself to other machines. It is important to note that the other machines do not

need to be informed of the certificate update because the new certificate will be sent at the beginning of each SSL handshake. *Because a node's certificate is sent at the beginning of each session, nodes should not need to store local copies of anyone else's certificate*. Therefore, when a node is issued a new certificate for any reason (expiration, key update, etc.), the operation is transparent to other nodes in the network, and they do not need to be notified.

5.3 Certificate Revocation

As described above, certificates are naturally invalidated when their validity period expires; however, it may be necessary to invalidate a certificate before its validity period expires. For example, if a node leaves the system, the node (and its certificate) should no longer be considered valid. Similarly, if a node is compromised and its private key is stolen, the corresponding certificate should no longer be accepted as valid since the key is potentially exposed to an attacker. Unfortunately, in either case, other nodes have no way of determining that the certificate should be considered invalid; the certificate has not expired and was valid when it was first issued, so other nodes will trust it. To combat this problem, the PKI structure needs an additional mechanism in place for invalidating, or *revoking*, otherwise valid certificates.

5.3.1 Certificate Revocation Lists

The classic mechanism for marking certificates as revoked is the *Certificate Revocation List* (CRL). The CRL is a list published by the CA of certificates that should no longer be considered valid. Certificates listed on the CRL have been deemed to be untrustworthy and should be treated as invalid certificates. When a node receives someone else's certificate, the node should ensure that the certificate is not on the CRL. If the certificate is listed on the CRL, it can no longer be trusted and should be considered invalid.

Certificate revocation lists are desirable primarily for their simplicity. The CRL is signed and published periodically (e.g., weekly) by the CA. Nodes, in turn, periodically download the latest copy of the CRL and store it locally. The process of publishing and downloading CRLs is very straightforward, is simple to implement, and does not require any additional infrastructure. The window of vulnerability of CRLs is the time between CRL publications. The window of vulnerability can be shrunk by publishing CRLs more often, but the rapid updates impose a significant bandwidth and computational cost on the CA.

The primary drawback of CRLs is their scalability issues. In larger networks, it becomes increasingly difficult for the CA to provide timely CRL services for its myriad nodes. Furthermore, the larger the network is, the larger the CRL becomes. For networks with a huge number of nodes, the CRL can grow to be megabytes in size, imposing a significant bandwidth cost on the CA. The bandwidth issue is compounded by "CRL request implosion." That is, the nodes on the network may become synchronized around the CRL publication time and will request the new CRL near the moment of publication in order to minimize the window of vulnerability. This synchronization will inundate the CA with numerous simultaneous CRL requests and may cause network congestion and additional latency.

5.3.2 Online Certificate Status Protocol

An alternative to CRLs is the Online Certification Status Protocol (OCSP) [32]. OCSP is designed to provide timely certificate status checking. OCSP introduces trusted third parties called OCSP Responders. OCSP Responders answer certificate status queries on behalf of the CA. When a node is presented with a certificate, a request is sent to an OCSP Responder to determine the validity of the certificate. The OCSP Responder sends back a signed status response of either "Good," "Revoked," or "Unknown." All certificate validity checking is performed real-time, so there is no need for nodes to preemptively download or cache anything.

OCSP is an IETF standard and has several commercial implementations because it provides several desirable properties that are not found in a traditional CRL. The main advantage of OCSP is the small window of vulnerability. Since certificate revocation status is checked real-time, the window of vulnerability can essentially be made zero. The other primary advantage is the scalability. The certificate revocation information is distributed to a sub-network of trusted OCSP Responders who answer on behalf of the CA. As such, the certificate status requests by the nodes are distributed across the OCSP Responders, allowing the system to scale adequately (although not perfectly) in large networks.

The drawbacks of OCSP are the need for the trusted third parties and the necessity for them to remain online. The OCSP Responders must all be trusted, secure entities similar to the CA. Furthermore, the OCSP Responders must remain online at all times in order for the scheme to work properly. If the OCSP Responders go down or are severed from the network, nodes will not be able to identify revoked certificates.

5.3.3 Impetus for Certificate Revocation

Sometimes it is not clear why a certificate revocation mechanism is necessary. The following scenarios are intended to illustrate the practical impact certificate revocation mechanisms can have on real life systems.

Scenario 1: *Secure node communications without an implemented CRL*

Multiple nodes within three Utility companies Alpha, Bravo, and Charlie communicate daily and exchange information needed to conduct business. They rely on a PKI-enabled communication system that employs user certificates but does *not* implement an accompanying CRL.

A valid node within company Alpha was stolen by an adversary during a weekend evening. The theft was discovered by security personnel a few hours later and was reported to the local authorities. During the subsequent week, the adversary set up the communication node in a distant location using an IP spoofing scheme in order to trick nodes in company Bravo and Charlie into believing that it was still communicating from its primary location. Each time a communication request took place between company Alpha's stolen node and the other participating companies, a PKI certificate exchange took place. Each of these exchanges was accepted because the certificate presented by the stolen node was valid and the adversary successfully completed the required challenges. The private key was located on the stolen node, so the adversary was able to confirm its

identity by encrypting proposed challenges presented to him by the other company nodes. These challenges were verified and confirmed by the public key of the stolen node. Even though the theft of the stolen node was reported, the Alpha node appeared to be secure and legitimate from Bravo's and Charlie's perspectives, and so communication occurred unabated. Unfortunately, by communicating unknowingly with the adversary, sensitive information was exposed resulting in severe data compromise.

Scenario 2: *Secure node communications with an implemented CRL*

Multiple nodes within three Utility companies Alpha, Bravo, and Charlie communicate daily and exchange information needed to conduct business. They rely on a PKI communication mechanism that provides user certificates and an accompanying CRL.

As before, a valid node within company Alpha was stolen by an adversary during a weekend evening. The theft was discovered by security personnel a few hours later and was reported to the local authorities *and to the network security officer* who immediately informed the CRL administrator to *add the nodes certificate to the CRL database***.** During the subsequent week, the adversary set up the communication node in a distant location using the aforementioned IP spoofing technique to mask his location. Each time a communication request took place between company's Alpha stolen node and the other participating companies, a PKI certificate exchange took place. Like before, each of these exchanges was initially accepted because the certificate presented by the stolen node was valid and the adversary successfully completed the required challenges. However, the certificate of the stolen node was then compared to the system's CRL. Since the network communications officer in company Alpha added the stolen node's certificate to the CRL database, the certificate was identified as revoked. The stolen node's certificate was thereby deemed invalid, communication with the stolen node was immediately terminated, and a security alert was logged.

In Scenario 1, without some mechanism for certificate revocation, there was no way for company Bravo or Charlie to recognize the stolen node as being untrustworthy. Thus, it was only natural for them to engage in normal communications with the stolen node. Without a certificate revocation mechanism, the compromised certificate could not possibly have been recognized to prevent unauthorized release of sensitive information. On the other hand, Scenario 2 employed a CRL for certificate revocation. Consequently, the compromised certificate was immediately identified, and all communication with the stolen node was stopped before it ever began. While this scenario is intentionally simplified, it provides a fundamental depiction of the risks associated with deploying a PKI in high consequence systems without a method for certificate revocation.

Appendix C: Acronyms

ACSE	Association Control Service Element
BECN	Backward Explicit Congestion Notification
BPS	Bits Per Second
CIR	Commited Information Rate
CSU	Customer Service Unit
DE	Discard Eligible – data frames above the Commited Information Rate (CIR)
DLCI	Data Link Connection Identifier
DSU	Data Service Unit
EASE	Embedded Application Service Element. Older term used for MMS
FECN	Forward Explicit Congestion Notification
FRAD	Frame Relay Access Device
ICCP	Inter-control Center Communications Protocol
LATA	Local Access Transport Area
LEC	Local Exchange Carrier
MFLOPS	Millions of FLoating-point Operations Per Second
MIPS	Millions of Instructions per Second
MMS	Manufacturing Message Specification
NAS	Network Access Server
NTK	Need To Know
PPS	Packets Per Second
RADIUS	Remote Authentication Dial-in User Service
RBAC	Role-Based Access Control
RTU	Remote Telemetry Unit
SLA	Service Level Agreement
TACACS	Terminal Access Controller Access Control System
TASE	Telecontrol Application Service Element
TPS	Transaction Per Second
VLAN	Virtual Local Area Network
VTP	VLAN Trucking Protocol

Appendix D: Glossary

Cost/ performance ratio	A metric for comparing two or more systems. For ICCP testing we might consider the estimated time for configuring the different implementations of Secure ICCP and TLS.
Efficiency	The ratio of usable capacity to theoretical capacity. Also, the ratio of the performance of an n-processor system to that of a single-processor system.
IPsec	IP Security. IPsec is used widely to implement Virtual Private Networks.
Knee Capacity	Throughput at the "knee" (point of maximum curvature) of the response time curve. Considered the optimal operating point.
MACE	MMS Application Certificate Exchange. MACE provides application authentication as well as anti-replay for non SSL/TLS connections
Network Throughput	Data transfer rate through a component, connection, or system. Usually given in units (bits, bytes, or packets) per second.
Nominal Capacity	The expected maximum throughput of a link under standard load.
OSI	Open Systems Interconnection. A standard for how messages should be transmitted between two nodes in a telecommunication network.
Reaction time	Time between submission of a request by the client and the beginning of its execution by the server.
Response time	Time between server beginning execution of a request and reception of the response by the client. (For a batch stream, responsiveness is measured as the Turnaround time, which see)
Round-trip time	Time between submission of a request by the client and reception of the response from the server. Round trip time = Reaction time + Response time.
SSL	Secure Sockets Layer. SSL provides client-server authentication and data encryption. Note that SSL version 3.0 has officially changed names to TLS.
Stretch Factor	The ratio of response time at a particular load to response time at the minimum load. Response time generally increases as the load on the system increases.
Throughput	The rate at which requests can be serviced by the system.
TLS	Transport Layer Security (SSL version 3.0)
Turnaround time	Time between the submission of a request by a client and completion of the output by the server.
Usable Capacity	The maximum throughput achievable without exceeding a specified response time limit.
Utilization	The fraction of time a resource is busy servicing requests.

Appendix E: For More Information

Author	John Michalski (jtmicha@sandia.gov) Critical Infrastructure Systems Department Sandia National Laboratories P.O. Box 5800 Albuquerque, New Mexico 87185
National SCADA Testbed (NSTB) Project	Jennifer DePoy, Manager (jdepoy@sandia.gov) Critical Infrastructure Systems Department Sandia National Laboratories P.O. Box 5800 Albuquerque, New Mexico 87185