SAND REPORT

SAND2001-3252
Unlimited Release
Printed March 2002

Key Management for SCADA

C. L. Beaver, D.R. Gallup, W. D. NeuMann, and M.D. Torgerson

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 under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Sandia National Laboratories

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

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

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.doe.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/ordering.htm

SAND2001-3252
Unlimited Release
Printed March 2002

Key Management for SCADA

Cheryl Beaver, Donald Gallup, William Neumann and Mark Torgerson
Cryptography and Information Systems Surety Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0785

Abstract

In this paper we discuss various security aspects and requirements of the Supervisory Control and Data Acquisition (SCADA) system for the electric power grid. In particular we discuss a method of managing cryptographic keys and give sample cryptographic algorithms that are appropriate for the SCADA system. We also describe a simulated SCADA network that we have implemented and discuss the items concerning its efficiency and compatibility with the requirements of the SCADA network.

This page intentionally left blank.

Contents

1. Introduction 7

2. Model of the SCADA Network 8
   - 2.1 Cryptographic Authority 8
   - 2.2 Master Station 9
   - 2.3 Substation 9
   - 2.4 Intelligent Electronic Device 9
   - 2.5 Security Aspects 10
   - 2.6 Functional Requirements 10
   - 2.7 Communication Restriction 11

3. Communications 14
   - 3.1 Packet Formats 15
   - 3.2 Controller-to-Subordinate Communications 16
   - 3.3 Peer-to-Peer Communications 18

4. Key Management 22
   - 4.1 Manual Key Management Duties 22
   - 4.2 Automated Key Management Duties 23

5. Simulation Details 26
   - 5.1 Hash Functions 26
   - 5.2 MAC 27
   - 5.3 Encryption 27
   - 5.4 Elliptic Curves 27
   - 5.5 Public Key Signature 27
   - 5.6 Key Exchange 28
   - 5.7 The Simulation 28

6. Practical Considerations 28

7. Conclusion 29

8. References 29

Figures

Figure 1 A Model SCADA Network……………………………………………………………15

Tables

Table 2-1 Benchmarks for Various Authentication Methods 11
Table 3-1 Key Storage Requirements for Controller to Subordinate Communications 18
Table 3-2 Key Storage Requirements for Elliptic Curve Implementation of Peer-to-Peer Communications 21

Introduction

Traditionally, the security features of the Supervisory Control and Data Acquisition (SCADA) system of an electric power utility consist of physical security, system complexity, dedicated communication channels, proprietary communication protocols, and minimal communication between components. An adversary that broke into a building in a remote location had the option of flipping a few switches, turning a few knobs, etc. The impact of such an act was generally limited to a reasonably small geographic location. The need to physically penetrate the system combined with little potential payoff has been a sufficient deterrent for any sort of advanced malicious adversary. Security breaches have generally come from unsophisticated adversaries, such as mischievous youngsters with too much time on their hands, or from those who wish to reduce their billing by tampering with their meters.

The SCADA network has been evolving and will continue to do so. The advent of high power, low cost computing has opened the door for utility companies to replace legacy systems with smart devices. The devices are powerful enough to take on duties that legacy components could never assume. There is a sufficient economic pressure to have various system components communicate with each other and make SCADA network decisions without ever communicating with a higher network authority such as the master station.

As we look to the future, it is likely that the physical security and system complexity of the major sites will remain at relatively the same level. However, what is changing is that the SCADA network is turning to standards based communications devices that use widely available standards based protocols and public communication channels. Further, the devices are distributed throughout the physical network, but electronically connected to each other. This means that an adversary will more easily have the ability to disrupt a vastly increased area by manipulation of data streams emanating from a component of the system and do so while never leaving the comfort and privacy of home.

The ease of intrusion and the magnitude of the possible disturbance make the SCADA network an inviting target. Undoubtedly, the SCADA network will be the target of increasingly sophisticated adversaries. Data security features must be added to SCADA in order to mitigate some of the vulnerabilities that the enhanced technology has brought.

To date, most of the efforts directed to SCADA security have not discussed key management. The main focus of this paper is an investigation of key management issues in the SCADA network. To properly discuss key management we develop a simple model of the SCADA network. Based on the model we describe, we discuss communication requirements and restrictions that have a bearing on the types of cryptographic algorithms that can be used in the SCADA network. We do not provide new cryptographic procedures. Rather, we focus on well-known methods and show that these algorithms are sufficient to meet the specific needs of a SCADA network. The key management procedures we provide are minimal in nature and yet meet the needs of the model. The model is an approximation to the actual SCADA network and the methods will have to be modified to fit any particular utility's needs. **This work shows that a large, complex (and hence expensive to implement and maintain) cryptographic infrastructure is not necessary to provide data security for any particular utility.**

We discuss various directions of possible evolution of the network and raise cautions against certain possibilities. Finally, we discuss a specific implementation of our proposed methods and provide performance figures for the implementation.

2. Model of the SCADA Network

The actual SCADA network is a highly complex entity. We have made no effort to cover every one of the many scenarios to be found within the SCADA network. However, we have gathered the essentials of the network to begin a cryptographic testbed.

A particular utility must assume the responsibility of securing its assets, and those assets may include elements that are not directly related to the proper generation and distribution of power. From a security standpoint those elements that do not directly contribute to proper function of the power system must be physically excluded from the SCADA network. For instance, it would be sheer folly to allow a computer within the utility's advertising department to exhibit superior control over the power management or power generation systems. As such, we will assume that a utility has two separate networks: a SCADA network and a non-SCADA network. The protections we describe are to protect the SCADA network from the non-SCADA network as well as from any other network.

The security requirements needed for a utility's non-SCADA network are varied and are not in the scope of this report, except that we make specific mention that access to the non-SCADA network must not provide any advantage for would-be attackers of the SCADA network. When we say network we mean the SCADA network.

We assume that there are four different types of entities in the SCADA network. Each serves a different function and has unique security needs. These entities are the following:

- The Cryptographic Authority (CA).
- Master station (MS).
- Substation (SS).
- Intelligent Electronic Device (IED).

For ease of communication, we say that a node is any one of the four entities mentioned above.

2.1 Cryptographic Authority

The CA is responsible for providing cryptographic support for the SCADA network. The CA generates and distributes a majority of the keying material used by the nodes within the network. It oversees the security of the network's security policy. We assume that the CA has sufficient computational resources, is physically secure, and has a high quality random number generator. We assume that the CA has no duties within the SCADA network other than providing cryptographic support to the network. Due to the need for access control it is essential that the lines of communication between the CA and other network entities be limited. We assume that the CA communicates only with the master station. In reality, the CA will most likely sit within a secured room within the master station. A detailed description of the key management protocols used by the CA is given in Section 4.

2.2 Master Station

The master station is the brain of the entire SCADA network. Its duties are varied. The master station communicates with the CA for cryptographic purposes. It controls and oversees the various substations. It is assumed that a master station has access to the outside network, and that communications with the other nodes in the SCADA network range over various communication media such as a LAN, dedicated channels or even a public network. The MS must have reasonable computational resources.

It may be that a particular utility maintains a backup master station that can assume all or part of the responsibilities of the master. From a cryptographic standpoint transition from master to backup control will be seamless provided that there is a backup CA as well. When network functions are switched to the backup, any necessary cryptographic controls may also be switched.

From a security standpoint, the backup CA is a weakness. Its very existence provides an avenue of attack that must be jealously guarded against. The CA will store the keys for the entire system and so must the backup CA (if it exists). The CA and the backup CA must have the same level of protections.

2.3 Substation

Each substation communicates with and controls a certain number of IED's. Besides IED's, a substation communicates with the master station, and possibly other substations. A substation may include a power generation facility, a major distribution point etc. Basically it is any point that has a high degree of complexity and function.

2.4 Intelligent Electronic Device

The network contains many control elements. These elements may be sensors, relays, and remote terminal units to name a few. In legacy systems these elements tended to be analog and manually controlled. These elements are being replaced with digital devices that have reasonable computational and decision-making power. We lump these digital elements into the category of Intelligent Electronic Device (IED). We do not assume that all IED's need cryptographic support. Indeed, some may not be important enough to invest the resources to secure. Further, not all IED's are capable of supporting cryptographic functions. However, there is a very strong trend to make IED's smarter and more capable. With added functionality comes added responsibility. As technology evolves, protecting critical IED's will become a greater issue. For the purposes of this report, we consider only the subset of IED's that are able to support the cryptographic operations required.

Those IED's that are unable to support security features must have an extremely limited importance in the SCADA network. Here importance is defined by the ability to communicate with and influence other nodes in the network.

There are two types of IED's of concern: those that reside within either an MS or an SS, and those external to either of the stations. The elements that reside within a station may form their own subnetwork. However, when viewed externally (or functionally) the subnetwork will be viewed as the substation. All communications from internal IED's to a point external to the station will be aggregated with other station element communications and routed as necessary. We focus our efforts on securing IED’s that are external to any station.

2.5 Security Aspects

We assume that the SCADA network exhibits reasonable physical security. The level of security associated with each node is commensurate with the importance of the node within the network. The master station and substations are assumed to have adequate physical, operational and network security. Communication channels also must be limited to those required to make the SCADA network function properly.

An IED may sit on the top of a pole in some remote location. Unless there are very strong measures to protect the data within the IED from probing, one has to assume that any cryptographic secret that the box contains has been or could easily be compromised. As mentioned above, some IED's may not have security features. The amount of potential SCADA network control exhibited by those IED's must be strictly limited.

We concern ourselves with securing the links between physically secure sites. However, the techniques that we propose can be used to help secure the communications of station subnetworks.

From a security standpoint, it is important that the damage that can be caused due to the compromise of a collection of nodes be limited to the portion of the network directly under the control of the compromised nodes. For instance, if a substation is compromised, this compromise should not give any significant advantage for the compromise of another substation. However, the IED's under the control of the compromised substation are effectively compromised.

2.6 Functional Requirements

In this section we indicate a number of physical communication requirements that must be met by the SCADA network. These requirements impose certain restrictions on the type of cryptographic protocols that can be used to secure the SCADA communications.

According to [IEC01] and [IEEE00] there are many critical SCADA communications that have maximum delay times on the order of 2-4 milliseconds.

A well-accepted set of speed benchmarks for cryptographic protocols provided by Wai Dai can be found at [WD]. In regards to the implementation platform Wai Dai says, “All were coded in

C++ or ported to C++ from C implementations, compiled with Microsoft Visual C++ 6.0 SP4 (optimize for speed, blend code generation), and ran on a Celeron 850MHz processor under Windows 2000 SP 1. Two assembly routines were used for multiple-precision addition and subtraction."

The following is a sample of relevant benchmarks.

Table 2-1 Benchmarks for Various Authentication Methods

MD5	100 megabits/sec (5.12microseconds/512bit block)
SHA-1	48 megabits/sec (10.7 microseconds/512bit block)
RSA-512 Signature	1.92 milliseconds
RSA-512 Verification	.13 milliseconds
RSA-1024 Signature	10.29 milliseconds
RSA-1024 Verification	.30 milliseconds
DSA-512 Signature	1.77 milliseconds
DSA-1024 Signature	5.5 milliseconds

Each and every sent SCADA message must be authenticated through the application of some sort of signature. Each sent message may also need to be encrypted. The signature of every received message must be verified and decrypted if needed. The time to sign and then verify using the public protocols such as DSA, RSA run on the order of 2-11ms for moderate security. These times are with a full function processor. The signing and verification times may be many times lower than these values if a less powerful processor is used. At worst, the times may be on the order of 1000 times slower if an 8-bit processor is used. Further, these times do not account for any sort of message queuing. If a substation has a thousand IED's underneath of it, then even a 2ms processing time would overwhelm a full function processor. This communication requirement precludes the possibility of using the given public key protocols for performing authentication.

2.7 Communication Restriction

As mentioned above, decision making power that is cheaply available allows for external IED's to communicate directly with each other. To date, this does not seem to be a widely accepted practice. However, it is a reasonable SCADA network evolutionary direction that must not happen. In this section we describe several issues that indicate why there must be certain communication restrictions placed on the SCADA network.

To be precise we insist that an external IED communicate only with a single (its) substation. If for a particular functional reason an external IED must communicate with an external IED, the communication must be forwarded through the sender's substation, and then through the receiver's substation.

When a substation receives a communication from a node (IED or otherwise) intended for one of the IED's under its control the substation must choose whether or not to forward the message.

There are four main reasons that we impose this restriction on external IED communications:

- Simplicity
- Damage control
- Communication constraints
- Cost

The more complex a system is, the harder it is to analyze and determine where the vulnerabilities lie. A large all-to-all network is highly complex and requires a set of complex mechanisms to secure. To assess the vulnerabilities of this type of network one would have to determine the vulnerabilities of each security component. Then one would have to determine if new vulnerabilities arise when the components are joined. Finally, each and every connection between nodes must be assessed. The more complex the network the less comfortable one can be that all the vulnerabilities have been found and accounted for.

Good security practices include a minimization of damage that a compromised node can cause to a network. If a single IED has the ability to directly affect the operation all the IED's under the control of a substation, then an adversary need subvert just one IED rather than the substation. It is almost certain that the security measures at the IED are less than at a substation.

The type of cryptography best suited to support a large all-to-all network is different than the support required for the simple network that we have described. In theory, public key protocols are an excellent choice for securing large networks. Unfortunately, they place exceptional drain on computational and bandwidth resources as well as require significant managerial resources. As explained above the communication throughput requirements are too high to support public key protocols. On the other hand, supporting and managing a large all-to-all network with symmetric keys is an almost impossible task. So, the physical communication constraints either force the network to be simple, or else force the external IED's to have significant specialized resources.

The issue of cost of securing the network will, in practice, be a driving force behind the network topological decisions. The cost to secure a network is really a function of the type and amount of security that one places on the system. As mentioned above, if an IED has the potential to affect a large number of nodes, then it is a particularly inviting target. To help mitigate the risk, the IED must have extremely high levels of physical security. If the security of the entire network relies on the physical security of a single point then that point must be highly secure. That intense physical security comes at a price and can easily increase the price of the IED many fold.

With enough hardware enhancements the IED computational throughput requirements could be met. If public key cryptography were used, each IED would either have to have a full function, high-speed processor or else a specially designed cryptographic accelerator. In either case, the cost to support the public key infrastructure would be prohibitive.

The cost of managing the security of the network is another issue. The more complex the network is, the more costly it is to manage. Complex networks need a complex set of rules for governance and must have someone skilled enough to oversee the security operations. Public key protocols not only require intricate key generation, they also require that the system parameters be carefully initiated and incorporated into the network.

A network composed of many small items will have component failure. Thus, configuration of the lower portion of the network may change periodically for functional reasons. This means that each node must have a method of accurately determining current network membership in order to facilitate proper authentication of messages. Network membership accounting is a service that must be readily provided to the network. In a public key infrastructure (PKI) this is generally accomplished through the use of certificate revocation list (CRL). Certificates and proper management of the CRL are costly in terms of bandwidth and communications with the CA. This is another feature that costs the utility when the system is designed and operated.

Commercial vendors are a possible source for providing security services. The billing services come in two general types; pay per seat or else a blanket contract. The pay per seat in a PKI usually includes generation and distribution of keys and certificates and vendor maintenance of the system. A blanket contract usually assumes that the vendor provides the initial goods and services, but the business provides the maintenance and most if not all of the support services. Blanket contracts tend to be more cost effective when the number of users is very large, the business absorbs much of the cost by providing the training and support for its own maintenance personnel. In either case vendor provided services for a fully connected external IED network would be expensive. We will see that the needs of the SCADA network do not warrant such a cost intensive approach.

One option to mitigate the need for highly specialized cryptographic accelerators is to allow the IED's to pass from the public key domain to the symmetric key domain via a key exchange algorithm. Since there may be many thousands of IED's each IED must either have the resources to securely store considerable information about each other node or each pair of nodes must initiate a key exchange before communication can begin. Because of the multiple communication requirements that key exchange algorithms require, the time that it takes to complete a key exchange can easily be many fold the time that it takes to verify a single signature. Thus, a key exchange cannot be a replacement for public key signatures on a message by message basis. This means that each external IED would have to initiate a key exchange and store the key for multiple use. This means that each node must have sufficient secure memory to store the exchanged keys of each of the nodes that it communicates with. This extra secure memory comes at a cost, but it may not be quite the cost of the specialized cryptographic accelerator. Nor does this solution reduce the cost to manage the network, as the nodes still must participate in a public key infrastructure.

The security issues associated with allowing inter (external) IED communications make one pause to consider the worth verses the security risks. However, it is easily seen that the cost to secure the all-to-all network would override any functional or financial benefit that such a network could provide.

In rare instances it may be highly beneficial from a corporate point of view to allow certain external IED's the ability to communicate directly. Allowing this to take place on an extremely limited basis may be allowable, however each and every instance must be individually considered. Special protections must be in place to safeguard these special IED's. Depending on the level of connectivity the type and level of protections required might be as great as those needed to support a substation.

3. Communications

The communication restriction that we impose allows for a simple network with reasonable and efficient security solutions. The security protocols we discuss are to provide security for the following four paths of communication.

- Cryptographic Authority to Master Station (CA-MS).
- Master Station to Substation (MS-SS).
- Substation to Intelligent Electronic Device (SS-IED).
- Substation to Substation (SS-SS).

A functional diagram of the communication paths for a SCADA network is given in the following figure.

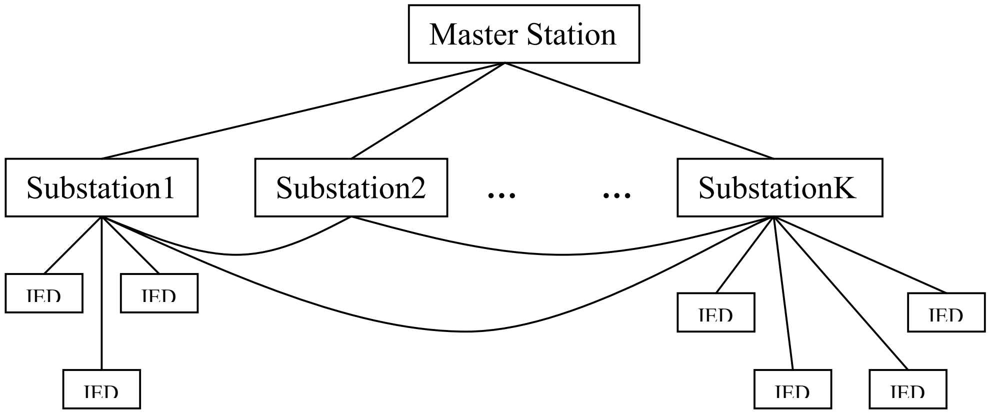

Figure 1: A Model SCADA Network

The three types of communications CA-MS, MS-SS and SS-IED are fundamentally different from SS-SS communications. In the SCADA network the MS-SS and SS-IED relationships are that of a controller to subordinate (C-S). Instructions from the higher are to be followed by the lower. Autonomy of the lower is limited. On the other hand, SS-SS communications are peer-to-peer (P-P) communications.

There is little need for the cryptographic model to deviate significantly from the functional model. Lower entities are controlled in a cryptographic sense by the higher entities. Only SS-SS communications use public key protocols, which are suited to peer-to-peer communications. And this is only for key exchange, which occurs on a periodic basis.

3.1 Packet Formats

Communications within the SCADA network have essentially one packet type, namely (FLAG,ID,TVP,LEN,message).

The specifics of the packet fields are defined below.

- **FLAG** is a few-bit field indicating certain special aspects of the packet. The FLAG field indicates to the recipient of the packet what type of packet has been received and thus allows proper processing of the packet. For example, one value in the FLAG field may indicate that the packet is encrypted, another may indicate that the packet is unencrypted.

- **ID** is unique sequence identifying the sender of the packet.

- **TVP** is a field of time varying parameters, which provide message uniqueness. The field consists of a **Timestamp** and a **Sequence Number**.

- **LEN** is a field that indicates the number of bytes in DATA. We arbitrarily set this to be a two byte field. In practice, this field may be unnecessary if the SCADA network uses only a few set length data options. In this case, the flag field may indicate the length of the data.

- **message** is a field that has different formats depending on the flag field. There are three different possibilities:

message=(DATA,MAC), message=(ENCR) or else message=(Key Exchange Information).

- **DATA** is the data field of the packet. Under normal operations this field carries the communications for the SCADA network in the format required by the network. Otherwise, it carries security upkeep information.

- **MAC** is a keyed message authentication code covering the non-MAC portion of the packet (FLAG,ID,TVP,LEN,DATA).

- **ENC** is the encrypted fields (DATA,MAC,PAD).

- **PAD** is a field of semi-random bytes; enough to provide an integral number of blocks to be processed by the encryptor.

- **Key Exchange Information** The specification of the key exchange information varies depending on the progress of the key exchange protocol. The details are given in section 5.6.

When the receiver receives a packet, it must determine that the encrypted portion of the packet is the smallest number of blocks that is larger than the number of bytes defined by LEN plus the number of bytes in the MAC. If not, the packet is rejected.

3.2 Controller-to-Subordinate Communications

Controller-to-subordinate communications is ideal for symmetric key techniques. The symmetric techniques provide efficient authentication and encryption. The symmetric techniques also provide efficient methods of producing, changing and auditing keying material.

We allow three different types of C-S communication. In normal operations mode the (DATA, MAC,PAD) portion of the packets are encrypted. In abnormal operation mode we assume that (DATA,MAC) are unencrypted. The third type of communication is a key update mode. During the key update mode the data must be encrypted. The packet format is similar to the format for the normal operation mode. However, the MAC and encryption keys are derived in a different fashion.

3.2.1 Keys for C-S Communications

The controller shares a key with each of its subordinates. This key removes the need for a costly key exchange. The keys that are used for C-S communication are of three types.

- Each subordinate is given a **Long Term Key** (LTK) that is shared with its controller. The shared LTK must be given to both parties in a secure, non-cryptographic, manner. This key is used to update General Keys.

- A controller requires a **General Seed Key** (GSK) that is also used to compute the general key.

- The controller assigns each subordinate a **General Key** (GK). The GK's are encrypted before transmission using the subordinate's LTK. The GK's are known to both controller and subordinate and are used to derive session keys. The length of time that a GK is used is determined by the security policy of the SCADA network. In any event general keys are to be updated periodically by the controller.

- Before a C-S communication occurs, the sender derives a **Session Key** (SK) from the GK. The SK is used for the encryption/decryption of the (DATA,MAC,PAD) portion of the packet. The SK's are a function of the general keys, the sender's ID and the time variant parameters of the message being sent.

- The MAC is keyed with the SK.

All C-S keying material is 128 bits in length.

3.2.2 General Key Generation

Throughout, let H be a cryptographic hashing function with a 128-bit output. The general key is computed by the controller with the help of a 128 bit random seed generated by the CA and assigned to the controller. We call the 128-bit number the General Seed Key (GSK). When ID needs a new GK the controller randomly chooses a 128 bit number Random, then computes GK=H(GSK,Random).

The frequency that the general keys must be updated is a function of the amount of data encrypted with the general key. Once a year should be sufficient. However, we suggest a key update whenever normal operations are resumed after the unencrypted mode is entered. A backup of keying material is needed. The CA must hold a copy of all long-term keys. It need not retain a copy of the general key. A new general key can be created from the long-term key. The role of the CA in the C-S key generation is to provide a source of high quality random numbers and to provide a backup of the LTK's.

Every communication between the CA and the master station has to do with the security of the network. CA-MS communications must not have an unencrypted mode.

3.2.3 Session Key Generation

Each session requires the computation of the session key. The SK is used for encryption/decryption of (DATA,MAC,PAD) and is computed as follows:

SK=H(GK, FLAG, ID, TVP, LEN).

Session keys are always generated as described above except during a general key update. In that case, the controller generates a session key from the LTK to encrypt the message and MAC containing the GK. The key update session key (KUSK) is computed as follows:

KUSK=H(LTK, FLAG, ID, TVP, LEN).

The value of the FLAG field informs the receiver as to the method of session key generation.

3.2.4 Key Storage

For C-S communications a controller must store a single general seed key. The controller must also store a general key and a long-term key for each subordinate under its control. So, if a controller controls N subordinates it must securely store 2N+1 keys. This does not include any keys required for other types of communication. For instance, a substation is both a controller and a subordinate depending on the communication. The keys for each role are generated independently.

The following table summarizes the storage requirements needed for C-S communication.

Table 3-1 Key Storage Requirements for Controller to Subordinate Communications

Key	Subordinate	Controller
Long Term Key	1	1 per Subordinate
General Key	1	1 per Subordinate
General Seed Key	None	1

Both substation and IED must securely store their keys. However, there are a couple of different levels of security needed for a substation to store its keys. The general keys are to be used by the substation on a frequent basis and must be readily accessible for use. This implies that the keys must be readily accessible on the machine on which they reside. This means that the primary method to secure the general keys resides on the ability to secure the machine on which they reside. This comes through operational security and network security.

The use of the long-term keys is different than that of the general keys. The long-term keys are only needed on a periodic basis, when the cryptoperiod has expired and it is time for a key replacement. They are also used when some sort of compromise in the substation has occurred and an IED's need an emergency key replacement. The fact that the long-term keys are to be used after a substation compromise has occurred means that they CANNOT be secured in the same way that the general keys are secured. If the long term keys are secured in the same way that the general keys are secured and the adversary were able to circumvent the security of the substation and recover the general keys, then one has to assume that the long term keys are compromised as well.

If all the secrets of a symmetric key cryptosystem are compromised, then there is no way to securely reestablish the system without manual intervention. This means that if the long term and general keys were similarly protected and if a substation compromise were to occur each and every IED would have to be manually rekeyed. The point of the long-term keys was to avoid the need for a manual rekey. It must be that the long-term keys be stored so that they are not accessible to an adversary even if they have full run of the substation. How the storage of long term keys is actually implemented is beyond the scope of this paper. We make mention of it because it is vitally important for proper function of the security features.

3.3 Peer-to-Peer Communications

In keeping with the restriction to limit external IED communication, it would simplify the communication architecture to also restrict inter-substation communications. However, this restriction may be unreasonable from a functional point of view. We assume that inter-substation communications must take place.

The characteristics of the substations are vastly different than an IED. These differences make securing SS-SS communications feasible. In fact the special attributes of the substation network pave the way for certain efficiencies that otherwise would not exist.

- The importance of a substation in the network is significant.

- The monetary value of a substation is significant.

- Each utility operates a comparatively small number of substations.

- The existence and functionality of the substations in the network is extremely stable.

A breach in the security of a substation could have catastrophic consequences for the utility and the public community at large. A utility will have to expend sufficient resources to secure those facilities. However, because of the size and value of the stations the security features needed are a small percentage of the cost of the station. This will be true no matter whether the network uses a public key infrastructure for inter-substation communications or not.

Further, the restricted size and stability of the substation-to-substation topology allows for certain efficiencies that the network would not otherwise have. A full-fledged public key infrastructure is not needed to allow for inter-substation communication. For instance, a utility need not maintain a typical certificate revocation list (CRL) for their systems. CRL's are costly to manage and maintain and also tap communication resources to distribute to the nodes on a periodic basis. The nature of the SCADA network allows for other options that are vastly more efficient and less expensive to implement and manage.

3.3.1 Keys for P-P Communications

As mentioned above, the stability of the SCADA network as well as its inherent centralized authority base, provides the possibility of modifying some of the common features found in a more general public key infrastructure. Many of the protocols that we propose in this and the subsequent sections are specifically tailored to the needs and resources of the SCADA network and are not necessarily applicable to all networks.

The keys and other requirements for P-P communications are as follows:

- Each SS is given the **CA's Public Key** as well as all the other infrastructure information. This can be done using out of band techniques or by using the symmetric techniques discussed below.

- Each SS is given the **Public Key Signature Key** (PKSK). The PKSK is a secret system parameter known to all SS's, the MS, and the CA.

- The CA assigns each SS a **Public** and **Private** key pair (PU,PR) and issues a certificate of the public key. The CA records the public and private values.

- Before two substations can communicate in a secure manner they must obtain **a Common Key** (CK). The common key is obtained through a key exchange algorithm, which uses the keys in the previous three bullets. The CK plays the same role in the P-P communications as the general key plays in C-S communication.

- The derivation and use of a **Session Key** from the CK for P-P communication is identical to the derivation and use of session keys from GK's in general C-S communications.

All of the public parameters and keys for such are passed to the substations with the symmetric C-S techniques mentioned above. These communications carry keying information and must be encrypted. For efficiency, each substation stores a certain number of CK's and their time of expiration. The length of time that a CK is valid is determined by the security policy of the network. For the sake of simplicity, we suggest that there be a small number of well-known times a year for which the keys are valid until. The time of expiration must be held as securely as any of the other keys.

Ideally, each SS shares a CK with each of the other SS's. Due to lack of memory or expiration of valid CK's this may not always be case. If an SS does not have a valid key that is common with another SS, then a new key exchange session must be initiated before data can be transferred. The communication protocol must allow for the case where two SS's have a differing opinion as to the status of their common key. For instance, one SS may have forgotten the CK while the other has not. If the sender does not have the common key, then the sender initiates a key exchange before the data is sent. On the other hand, if the sender has what it thinks is a valid common key, it sends a message using the symmetric key techniques. It must receive an acknowledgment or a request for a key exchange before the communication can be completed.

3.3.2 Certificate Revocation List

In a general public key infrastructure (PKI) the certificate system must be robust enough to allow for many different options. A Certificate Revocation List (CRL) is generally maintained. Periodically the nodes obtain a valid copy of the CRL in order to determine if any certificates have been revoked prematurely. In a large dynamic network this process can be costly in terms of communication and computational overhead. The frequency that a node must communicate search out and validate the CRL is a function of the amount of security needed by the system. If a breach occurs, there is a window of opportunity to be had by an adversary. The window is the time between the breach and the time that a node checks the CRL. For high surety systems the time between CRL verifications must be quite small. Frequent communication with the CRL database imposes severe communication overhead for each of the nodes. Infrequent communication with the CRL opens a window for an adversary who has compromised a node to further corrupt the network. Further, the CRL must be kept free from errors. Thus, installing and securely maintaining a CRL is not a trivial matter.

If a substation in the SCADA network has been compromised, then that information must be immediately and proactively disseminated through the network. The utility cannot wait for a substation to get around to checking the CRL before suspending communications with the compromised substation. Nor does it make sense for each substation to maintain a dedicated link to the CRL. Maintaining a CRL in the SCADA environment is a waste of resources. The substation network is small enough that it is easy to propagate compromise messages as needed. Below we describe how to invalidate compromised certificates.

3.3.3 Certificates

X.509 compliant certificates are a very common standard for certificate format. The X.509 certificates have gone through several revisions. The current version is self-formatting that allows multiple levels of CA control, it allows for many different signature methods, plus it contains hundreds of bits of control information. Implementing X.509 compliant certificates is an arduous task and the final product is unnecessary.

The SCADA network is relatively stable. In fact, a substation is not generally allowed to leave the network for anything short of catastrophic circumstances. If some sort of substation compromise does occur, then the utility must invest a certain (large) amount of recourses to rectify the breach. Rather than load the normal network communications down with complex mechanisms necessary to support general certificates, we provide a more efficient certificate geared toward the SCADA network. The certificate is simply (ID,PU,SIG). Here SIG is the CA's public key digital signature of (ID,PU).

The signature is a function of the signature key. That is, the signature is really a signature of H(PKSK,ID,PU,PKSK). In order for the CA to revoke a certificate the CA simply issues a new PKSK to the substations and reissues a certificate to the good substations. Here the premise is that it is easier to reissue new certificates when necessary than it is to support (in the traditional way) the ability to selectively exclude a member when the need arises. Later we will show the network procedures necessary to revoke a certificate.

3.3.4 Storage

The amount of information each substation must store in order to accomplish peer-to-peer communications is a function of the type of public key protocols used. Various choices may require vastly different amounts of storage. We used elliptic curves for the public key protocols in our simulations. Elliptic protocols are a reasonable choice for needing small amounts of storage and fast computations for a desired level of security. The following table summarizes the storage requirements for peer-to-peer communications for our implementation. We give further cryptographic details below. Note that a node must store a number of Common Keys to speed up the communication process. The number is determined by the implementation.

Table 3-2 Key Storage Requirements for Elliptic Curve Implementation of Peer-to-Peer Communications

Elliptic Curve	283 bits
Base Point	566 bits
Order of the Base Point	283 bits
CA’s Public Key	566 bits
Node’s Private Key	283 bits
Node’s Public Key	566 bits
Public Key Certificate	About 1200 bits
Public Key Signature Key	128 bits
Total Storage	About 3900 bits
Common Keys (# by implementation)	

4. Key Management

Key management encompasses various procedures that span the life cycle of a network in order to facilitate keying relationships between the communicating nodes. These procedures encompass the following:

- Initialization of the system and users.
- Generation, Distribution and Installation of keying material.
- Oversight of operational use of keying material.
- Update, Revocation, and Destruction of keying material.
- Storage and Recovery of keying material.

There are a number of key management procedures that are best accomplished through human intervention, while others are well suited for automatic processing. Below we distinguish some of these duties.

4.1 Manual Key Management Duties

Application of key management techniques comes on two fronts: Procedures that can be automated and those that cannot. Human intervention must be a part of any high surety system. For instance, to have the highest level of confidence in the system keys must be initially installed with out of band methods. This is particularly important if the node being initialized is a substation or master station. The procedure for system initialization proceeds in exactly the same way as for cryptoperiod expiration (described below) with one exception: Before initialization can begin the CA must create and store a LTK for each controller-subordinate pair. The LTKs then need to be manually installed in the appropriate master and substations. Once this is done, the CA can proceed as if the cryptoperiod has expired.

If a station is compromised then the complexity and depth of the compromise must be determined. To date, automatic methods do not exist that accurately and precisely measure these issues. Nor are there sufficient measures to automatically detect when a compromise has occurred. Human intervention is required to monitor the network and to reestablish nodal trust.

The use of Long Term Keys will help with the automatic recovery of compromised nodes provided that the Long Term Keys have not also been compromised. As mentioned above the LTK's need to be stored in a very secure fashion or else their use adds little to the security and ease of rekeying.

Another concern is that a malicious modification to a node's programming has occurred during a compromise. This must be both detectable and repairable or else preventable. Prevention is the most likely solution for an IED. If an IED is designed so that it can not reveal the LTK nor can its functionality be modified beyond certain bounds, then electronic rekey with the LTK is reasonable.

Unless specialized hardware is used at the master station and the substations, the much-varied functionality will prevent the simple reassignment of General Keys that is available to an IED. Human intervention is most likely a necessary part of a station's recovery process.

4.2 Automated Key Management Duties

The CA is designed to handle the key management duties that can be handled in an automatic fashion. Once the SCADA network has been initialized the CA will handle all of the mundane and periodic key management procedures. The CA is not required to participate directly in the keying of all nodes. Rather, the CA is the source of randomness required for valid keying material. This randomness must be generated in a secure manner and must not be based on phenomenon observable by an adversary.

Another duty of the CA is to keep track of the current system time and to determine if the current cryptoperiod has expired. If it has, the CA should perform the protocol of section 4.2.2. Additionally, at specified periods, the CA should ensure that all of the nodes in the network see the same approximate ``network time". This needs to be done, or the time varying parameters used in the generation of the session keys could cause the recipient of a message to treat the message as invalid. The CA will perform clock updates in a very simple manner.

Once the network has been initialized the CA is free to supervise any keying updates, revocation and or recovery of keys as necessary. We now give a list of protocols that explain the CA duties in greater detail. Note that the CA must monitor the processes described. If any error condition occurs (including unreceived completion messages), the CA must respond appropriately.

4.2.1 Clock Management

- When it becomes time to perform a clock update, the CA reads its current system time and sends a message to the master station with a command to update its clock with the time given in the message.

- The master station updates a variable that holds a differential between the time given in the message from the CA and its clock. Every time it needs to access the ``network time" it will read its system clock and add this differential to the result.

- The master station will construct similar messages with the current ``network time" and send them to each substation, which in turn update their time differentials. Once the update is complete, the substations will send a confirmation message to the master station.

- After the master station receives confirmation from all of the substations, it sends a confirmation message to the CA, ending the time update protocol.

4.2.2 Cryptoperiod Expiration

Periodically (say once a year), or after the system enters an abnormal (unencrypted) communications state, the CA executes the protocol below to replace most of the keying material used in the system.

- The CA generates a GK and a new GSK for the master station from its high quality source of random bits. These keys are encrypted using a new KUSK and sent to the MS.

- Upon receipt of the GK and GSK, the master station computes a new GSK for each of its substations using GSK=H(GSK, random) where H is a cryptographic hash function with a 128 bit output, and random is a 128 bit value culled from the master station's pseudo-random number generator (PRNG). Each of these GKs for the substations is encrypted using freshly generated KUSKs and sent across the network to the substations. Once the MS finishes these updates, it informs the CA of the completion.

- Once the substation GK rekeying has been completed, the CA generates a new PKSK and new (PU,PR) keypairs for each substation using its source of random bits. The CA then creates a new certificate for each substation [see section 3.3.3] and sends the PKSK and certificates to the master station. The master station then distributes them to the substations.

- Upon receipt of the new PKSK and asymmetric information, the substations send a confirmation message to the master station that in turn sends a confirmation message to the CA after hearing from all of the substations.

- Upon receipt of the asymmetric information, each substation initiates a key-exchange protocol with all of the substations with a higher ID that itself in order to establish new CKs that correspond to the new asymmetric information, and deleting the old CKs as they are replaced.

- A short period of time (on the order of a few minutes) after the CA receives confirmation that the substations were given the new asymmetric information, the master station generates from its PRNG a GSK for each substation to use to rekey its IEDs. These keys are encrypted using a KUSK and sent to the substations.

- Upon receipt of the new GSKs, the substations generate new GKs for its IEDs and informs the master station when this process is complete. Upon receiving confirmations from all of the substations, the master station sends a message of completion to the CA, completing the system rekey protocol.

4.2.3 Substation Compromise

When a substation, S, has been compromised, the following protocol is run to recover from the intrusion:

- The CA distributes a message to the master station that S has been compromised. The master station distributes this message to all substations. This message signals an immediate suspension of communication between substations and S.

- Upon receipt of the compromise message, each substation deletes all stored information relating to S, including certificates and CKs.

- When S has been restored, the CA informs the master station.

- The master station issues S a new GSK and GK, and informs the CA upon completion.

- Upon receipt of the new GSK, S rekeys all of its IEDs and informs the master station upon completion. The master station then forwards this information on to the CA.

- The CA generates a new PKSK and new asymmetric information for S and sends it to S through the master station.

- Upon confirmation of the receipt of S's new asymmetric information, the CA creates and distributes new certificates (using the new PKSK) to all of the uncompromised substations, along with a message that allows the other substations to communicate again with S.

- In the systematic manner described in section 4.2.2, the substations establish new Common Keys with each other, removing the old Common Keys in the process.

4.2.4 Session Key Generation

The process to generate new Session Keys, Key Update Session Keys, and Peer-to-peer Session Keys are all very similar, being computed as a function of some other key, and information that is related to the message being protected by the particular key. To reiterate the three functions as given above:

SK = H(GK,FLAG,ID,TVP,LEN)

KUSK = H(LTK,FLAG,ID,TVP,LEN)

PSK = H(CK,FLAG,ID,TVP,LEN)

Where in each function, H is a cryptographic hash function with a 128 bit output, and FLAG, ID, TVP, and LEN are all fields from the header of the message being protected as described in section 3.1.

4.2.5 Key Storage

The CA will need to store all of the following information:

- Every LTK used in the system.

- The GK it shares with the master station.
- The PKI information (the elliptic curve, the base point and it's order, etc.)
- The (PU,PR) key pairs for each substation. The current PKSK.

All of the above should be stored in a secure manner.

4.2.6 Certificate Management

Unlike a more general, less structured network environment, the SCADA network is a relatively stable, well-trusted environment. As such, we can do away with most of the functionality of a traditional PKI setup using full-blown X.509 certificates for the public keys in the system. We are instead able to use simple certificates of the form, (ID,PU,SIG), where SIG is the CA's public key signature of H(PKSK,ID,PU,PKSK), which binds the identity of substation ID to the public key listed in the certificate. The presence of PKSK in SIG ensures the freshness of the certificate, invalidating any old certificates that were created with old values of PKSK.

Additionally, since the SCADA network is a comparatively small network, certificates are all revoked and replaced at once. This is done on a periodic basis, or when a substation has been compromised. The process used to revoke the current certificates follows that of steps 6-8 in the protocol laid out in section 4.2.3.

5. Simulation Details

In this section we discuss a simulation of the ideas given above and describe particular cryptographic primitives used in the simulation. The choice of primitives was driven by network functional requirements. The time intervals required for many of the control functions of the SCADA network are down in the few milliseconds range. This all but precludes public key cryptography from being used within the network, except for an infrequent use of key exchange. We have chosen well-known protocols and show that these protocols are sufficient for the needs of the network.

5.1 Hash Functions

Because the timing constraints require that the entire cryptographic processing take significantly less than a millisecond, we have chosen the keyed hash approach for providing verification. This approach takes on the order of a few microseconds to process a 512-bit block of data. Two common candidates for a hash function are MD5 and SHA-1. To simplify coding, our simulations used SHA-256 for both the symmetric and public key operations.

The cryptographic hash function used in the symmetric key applications of our simulation is the high order 128 bits of SHA-256 [FIPS-ZZY]. That is, H(M)=SHA-256(M) / 2^{128}.

With our choice of elliptic curves, the P-P signatures call for a hash function with an output larger than provided by either MD5 or SHA-1. In this case we used the full SHA-256 (with the Signature Key pre and post-pended to the value to be hashed).

5.2 MAC

In the symmetric key communications the MAC is the high order 128 bits of: H(SK, DATA). In the unencrypted mode, the MAC is the 128 high order bits of H(Key, FLAG, ID, TVP, LEN,DATA). The value Key is the GK in the case of SS-IED and MS-SS communications and is the CK in the case of P-P communications. All CA-MS communications are in the encrypted mode. In the key update mode of all C-S communications the MAC is: MAC=H(KUSK, GK).

5.3 Encryption

There are many choices for an encryption algorithm. DES has long been the standard to be included in most applications. However, DES is thought to be insecure for serious commercial applications. The cipher Rijndael has been chosen as the winner of the AES competition. It is generally faster than DES and supports larger key sizes. In our simulations we use the 128 bit AES in CBC mode [NIST-ZZZ]. As mentioned above, if (DATA,MAC) is less than an integral multiple of 128 bit blocks, pad with exactly enough semi-random bytes to fill out a block.

5.4 Elliptic Curves

There are many possible choices for the public key operations. However, we have chosen a 128-bit encryption algorithm. To attain equivalent security with RSA or ordinary discreet log systems one would have to have extremely large parameters. The large parameters translate into network inefficiencies. Since the public key operations are few and far between, the network would be able to support the inefficiencies provided by these methods. A nice alternative to RSA or DSA is the elliptic curve analog of DSA. Currently it is believed that a good curve with modulus on the order of 256 bits will give roughly 128 bits of security. In our simulations we used the NIST curve B283 for all public key transactions.

5.5 Public Key Signature

The CA must sign each certificate that it passes out. Also the information used by the key exchange algorithm must be signed. The signature algorithm that we used in our simulations is nearly the elliptic analog of DSA [IEEE01]. The only difference is that a hash keyed with the Public Key Signature Key is used rather than an ordinary hash of the message. That is, each signature is a signature of the hash of (PKSK, message, PKSK).

5.6 Key Exchange

As with the other algorithms there are many choices for a key exchange. Our only requirement is that the key exchange provides the security needed by the key. The key exchange algorithm used in our simulation is MQV [Law98, IEEE01]. If K denotes the value obtained by the MQV algorithm, then the Common Key is the high order 128 bits of H(K).

5.7 The Simulation

Many of the processes we have explained have been implemented and tested for functionality. Because the C-S communication protocols for SS-IED and MS-SS are the same, we did not include SS-IED communications in our simulation. The MS and the CA were simulated with a 1.3 GHz Pentium IV running Windows 2000. The substations were simulated with four 800 Mhz Pentium III's each of which could assume the role of up to 256 different substations (also under Windows 2000).

The time for a key exchange from substations originating from differing machines took about a second from a manual command to start to the time to finish. Public-key key exchange should occur infrequently, so this is a probably sufficient.

Time critical SCADA messages are very short, being on the order of a few to few dozen bytes in length. Our simulations for MS-SS communications arbitrarily set the data message length to 48 bytes. The master station was able to receive and process messages from 256 different substation at a rate of about 6000 messages per second. Processing includes, reading the header, computing key, decryption, authentication, and outputting the message to the screen. There are a number of ways to increase the number of messages per second that the master station can handle. Optimization of the coding would help. Using MD5 or SHA-1 would also speed the processing up. However, our simulations indicate that reasonable throughput is possible even at an aggregation point.

6. Practical Considerations

We have not attempted to devise a full-fledged security standard applicable to SCADA without modification. Rather, we have examined the network to evaluate the communication needs and propose an efficient method for key management that has been tailored to the network. The algorithms that we have chosen for our simulations are widely known and suitable for our communication model. However, there are other possible choices for the cryptographic protocols. IPSEC is a well-known, well-reviewed set of protocols that has a broad acceptance base [IPSEC]. IPSEC allows for both public key and symmetric key authentication and is robust enough to useful in a very wide set of applications and on a wide set of platforms. At the time of this writing AES is not included in the list of default encryption algorithms. A fully compliant version of IPSEC may require on the order of a megabyte of code. The full functionality of IPSEC and its accompanying many bytes of code is not necessary for an IED to have all needed security features. Certainly, it is not inconceivable that a pared down version of IPSEC be used to handle the specific needs of the SCADA system. However, even with a "right sized" version of IPSEC, key management is still an issue. SCADA must have a manual initialization of critical components. Further, IPSEC does not directly specify methods for compromise recovery and the like. So, IPSEC may be an alternative choice for the basic cryptographic blocks, but our key management protocols would still be applicable.

7. Conclusion

The entire SCADA network is a vast, complex entity that is becoming increasingly vulnerable to adversarial manipulation. Legacy devices are being replaced with smart devices with decision-making power. With increased component capabilities comes increased component responsibility and increased network vulnerabilities. Ultimately each utility in the SCADA network must play a roll in securing its own resources. However, it is unreasonable to assume that each utility will become a security expert able to design, implement and maintain its own security features. We have described the basic security needs of a utility's SCADA network. We have provided a simple and cost effective method of providing security features and key management techniques that are also efficient enough to keep up with the communication needs of the network.

8. References

[FIPS-ZZY] NIST, *Secure Hashing Algorithm,* pre-draft description, 2001.

[FIPS-ZZZ] NIST, *Advanced Encryption Standard*, draft standard, February 2001.

[IEC01] IEC, *Communication Networks and Systems in Substations Part5: Communication Requirements for Functions and Device Models.* DRAFT IEC 61850-5, 2001.

[IEEE00] IEEE, *Draft Standard for Substation Integrated Protection, Control and Data Acquisition Communications,* IEEE P1525D4r3, December 2000.

[IEEE01] IEEE P1363, *Standard specification for Public-Key Cryptography*, 2001.

[Law98] L. Law, A. Menezes, M. Qu,j>solinas, S. Vanstone, *An Efficient Protocol for Authenticated Key Agreement,* Combinatorics & Optimization, University of Waterloo, March 1998.

[Men97] A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography,* CRC Press, 1997.

[WD] Wai Dai *webpage* http://www.eskimo.com/~weidai/

This page intentionally left blank.

DISTRIBUTION:

	MS	
1	0785	C. L. Beaver, 6514
1	0455	R. E. Carlson, 6517
10	0785	D. R. Gallup, 6514
1	0785	T. S. McDonald 6514
1	0785	W. D. Neumann, 6514
1	0455	R. S. Tamashiro, 6517
1	0741	M. L. Tatro, 6200
10	0785	M. D. Torgerson, 6514
1	0455	J. J. Torres, 6517
1	0784	R. E. Trellue, 6501
1	0451	S. G. Varnado, 6500
1	0188	LDRD Program Office, 1030 (Attn: Donna Chavez)
2	0899	Technical Library, 9616
1	0612	Review & Approval Desk for DOE/OSTI, 9612
1	9018	Central Technical Files, 8945-1