SECURITY HUB ARCHITECTURE SUPPORT FOR IEC61850 INFORMATION EXCHANGE PROTOCOLS

Suhas Aggarwal
Indian Institute of Technology Guwahati

suhasagg@gmail.com

ABSTRACT
In this paper, we give a brief idea about substation devices, substation network and communication model used in IEC 61850. We propose security hub architecture support for IEC 61850 Information exchange protocols. Transmission of GOOSE and SMV messages in substation network follow publisher/subscriber model. Security hub architecture support for publisher/subscriber model is discussed. Group communication possibilities among SIEDs are explored and use of security hub for secure group communication is proposed. Role of Security Hub as a Key Manger and Distributor is discussed. Possibilities of group dynamics behavior among substation SIEDs are explored. Use of key graphs for secure group communication among substation SIEDs are also discussed. Finally, we show certain flaws present in security hub architecture and propose attacks and solutions to these attacks. A modification to IEC 61850 protocol stack is also suggested.

1.INTRODUCTION AND BACKGROUND

1.1. Intelligent Electronic Device

To help understand the logical concepts of IEC 61850, we need to give explaination of intelligent electronic devices (IED) [2], the necessary hardware hosting all the logical objects. Basically, the term intelligent electronic device refers to microprocessor-based controllers of power system equipment, which is capable to receive or send data/control from or to an external source. An IED is usually equipped with one or more microprocessors, memory, possibly a hard disk and a collection of communication interfaces (e.g. USB ports, serial ports, Ethernet interfaces), which implies that it is similar to a computer as those for everyday use. IEDs can be classified by their functions. Common types of IEDs include relay devices, circuit breaker controllers, recloser controllers, voltage regulators etc.. It should be noted that one IED can perform more than one functions, taking advantage of its general-purpose microprocessors. An IED may have an operating system like Linux running in it which may run programs like IEC 61850 Server program

1.2. Substation Architecture

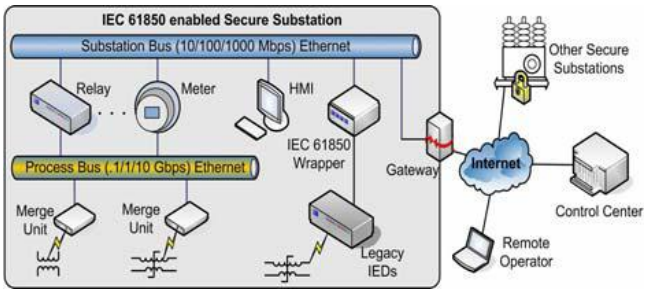

Figure 1

A typical substation architecture is shown in Figure 1. The substation network is connected to the outside wide area network via a secure gateway. Outside remote operators and control centers can use the abstract communication service interface (ACSI) to query and control devices in the substation. There is one or more substation buses connecting all the IEDs inside a substation. A substation bus is realized as a medium bandwidth Ethernet network, which carries all ACSI requests/responses and generic substation events messages (GSE, including GOOSE and GSSE). There is another kind of bus called process bus for communication inside each bay. A process bus connects the IEDs to the traditional dumb devices (merge units, etc.) and is realized as a high bandwidth Ethernet network. A substation usually has only one global substation bus but multiple process buses, one for each bay.

ACSI requests/responses, GSE messges and sampled analog values are the three major kinds of data active in the substation network. Substation architecture Interactions inside a substation automation system mainly fall into three categories:data gathering/setting, data monitoring/reporting and event logging. The former two kinds of interactions are the most important — in the IEC 61850 standard all inquiries and control activities towards physical devices are modeled as getting or setting the values of the corresponding data attributes, while data monitoring/reporting provides an efficient way to track the system status, so that control commands can be issued in a timely manner.

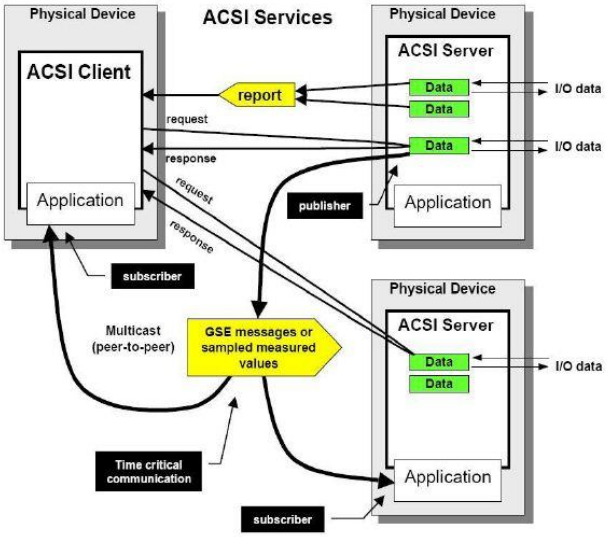

Figure 2 : ACSI Communication model

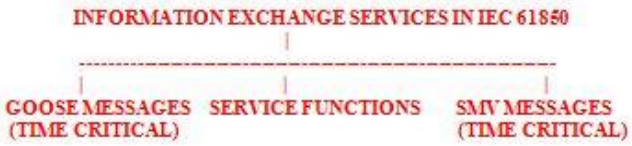

Figure 3

1.3. COMMUNICATION MODELS

GOOSE MESSAGES – follow PUBLISHER SUBSCRIBER MODEL - Transmission using MULTICAST APPLICATION ASSOCIATION

SERVICE FUNCTIONS – follow CLIENT SERVER MODEL - Transmission using TWO PARTY APPLICATION ASSOCIATION

SMV MESSAGES – follow PUBLISHER SUBSCRIBER MODEL - Transmission using MULTICAST APPLICATION ASSOCIATION

GOOSE:(Generic Object Oriented Substation Event)is used for fast transmission of substation events, such as commands, alarms, indications, as messages. A single GOOSE message sent by an IED can be received by several receivers. Examples:–Tripping of switchgear,providing position status of interlocking

SERVICE FUNCTIONS :

Services Operating on Data-

server model-
Getserverdirectory

logical device model-
Get logicaldevice directory

logical node model-
Get logicalnode directory
Get alldata values

data model-
get datavalues
set data values
get data directory
get data definition

data set model-
get data set values
set data set values
create data set
delete data set
get dataset directory

Sampled Measured Values :
A method for transmitting sampled measurements from transducers such as CTs, VTs, and digital I/O.

1.3. SECURITY HUB ARCHITECTURE

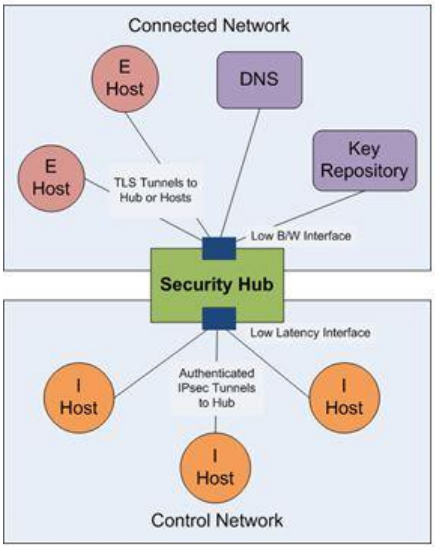

Figure 4 : Security Hub Architecture

A security hub [12] is a network element whose purpose is to provide broad access to a distinguished collection of hosts while assuring latency requirements between these distinguished hosts. The concept is illustrated in the figure.

The security hub links two networks: a connected network such as the Internet or enterprise network and a control network such as the digital communication bus in a power station.Hosts in the connected network are said to be external hosts (ehosts) whereas hosts in the control network are called internal hosts (ihosts). The security hub manages communication with each ihost so that a packet from an ehost to any ihost must pass through the hub and a packet from any ihost to any other ihost must pass through the hub. The security hub provides a "hub-and-spokes" network architecture for the ihosts while also acting as a gateway between the ehosts and the ihosts. It provides a low bandwidth interface to the connected network and a low latency interface to the control network. The hub maintains IPsec-authenticated tunnels between itself and each of the ihosts and takes responsibility for checking the authenticity of the origin of any packet sent between ihosts. It routes packets from the connect network to the ihosts through these tunnels, but leaves their authentication to the ihosts, which use TLS to provide both authentication and encryption for these links. Each ihost is given a DNS name and a certificate with a private key bound to this name. These keys are managed by a key repository which is located in the connected network. The nodes use the Domain Name Server (DNS) to get IP addresses for routing, including routing within the control network. The hub provides multicast addresses and routing for use by the ihosts. An ihost can declare a new multicast address and other ihosts can subscribe to it. Finally, the hub keeps a repository of the names of all of the ihosts attached to it in the control network and is able to provide this list to hosts authorized to receive it. The hub is also able to enforce basic partitioning of the ihosts by being configured so to allowing only some communication between the ihosts, thereby implementing an analog of a VLAN. There is need to provide for secure multicast communications in which messages are authenticated and possibly even encrypted. One could use security technology such as the Internet Security Protocol (IPsec) to address this need, but there are two problems: (1) Due to increasing complexity of substation configurations and the complexity of IPsec configuration there is need to provide automation for security configuration and (2) the latency requirements of substation communications must not be burdened respected by security protocols.It is a discovered fact that a trivial implementation of point-to-point IPsec using a hub-and-spokes model is not efficient enough to maintain substation latencies[14].

2. SECURITY HUB ARCHITECTURE SUPPORT FOR PUBLISHER SUBSCRIBE MODEL

If a SIED wants to be a PUBLISHER, it sends a message to SECURITY HUB. SECURITY HUB broadcasts this message to all SIEDS in the control network. Any SIED wishing to SUBSCRIBE sends a message to security hub. Security hub check its security association policies before setting up a PUBLISHER SUBSCRIBER relationship between them. SECURITY HUB contains a Database which keeps track of publisher subscriber relationships and various publisher subscriber systems active at a given time.

2.1 ROLE of SECURITY HUB as a KEY MANAGER and DISTRIBUTOR

Generates a group key for each publisher subscriber group and transmits it to each group member.
GROUP JOIN OPERATION-If any SIED wants to join an established group it receives the group key of group.
GROUP LEAVE OPERATION-If any SIED wish to leave the group, a new group key is generated and unicasted to each group member .
Use of group keys will save cryptographic checksum calculations as packets meant for the same group members will just have to be duplicated.

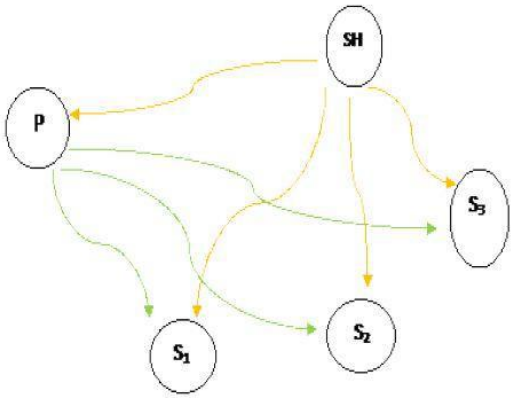

Figure 5: Group Control + Management Via Security Hub
SH:Security Hub P:Publisher Si:Subscriber

Group Control + Key Management Flow
Data Flow

PAIRWISE KEY SETUP -Generate pairwise session keys for members of the group. No. of keys to be generated , ${}^{n}C_{2}$, n is no. of group members. Unicast set of keys to each group member. This way they are aware of other group members as well. Each group member receives 'n' keys in a unicast message (a group key and n-1 session keys corresponding to each group member)In case of a group leave, leaving member sends a message to security hub and security hub multicasts this information to the group, so group members can flush the session key established with this group member from their cache. I think it is a good idea to cache the keys in local cache of publisher-subscriber group members. It is OK to be a bit sloppy here (in terms of security) to save some time. As these SIEDs can be mutually interdependent for the period they are part of a publisher subscriber group, if one SIED gets compromised, others will be affected naturally as they are functionally dependent on each other.

Additional Advantage-
Members which are a part of publisher subscriber system, can also be mutually dependent on each other for that period. So, there is a very high probability that they may issue some service commands to each other as well using client – server model. As member SIEDs can communicate directly via session keys established between them, we save some time again here.
Role of security hub* cuts down to key management to most extent and sort of decentralization is achieved with various publisher subscriber systems operating almost independently. Any communication between SIEDS (belonging to different publisher-subscriber systems or we can say belonging to different domains at that time) is achieved via SECURITY HUB as an intermediate node. This also helps in detecting suspicious behaviour and immediate security actions.
* Security hub may be replicated in the control network to enhance reliability, external connectivity and performance

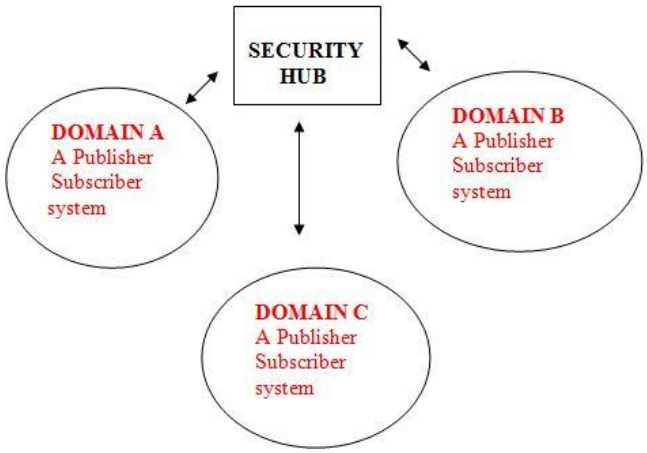

Figure 6

Communication between members of a different domain is achieved via Hub and Spoke principles.

AN ON DEMAND GROUP GENERATION SCHEME-
Implementation of a counter in a security hub which keep tracks of interactivity between SIEDs belonging to different domains. If the activity exceeds a certain set threshold level in some predecided time interval1 a group is established and keys can be distributed accordingly. If the group activity reaches below threshold level in some predecided time interval2 group dissolves.

3.GROUP DYNAMICS
Might be present in Substation SIEDs (This behaviour may be observed among different publisher subscriber groups and might be periodic in nature)

DOMAIN COALESCING-

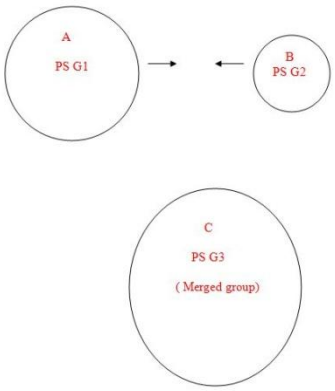

Figure 7

DOMAIN SPLIT-

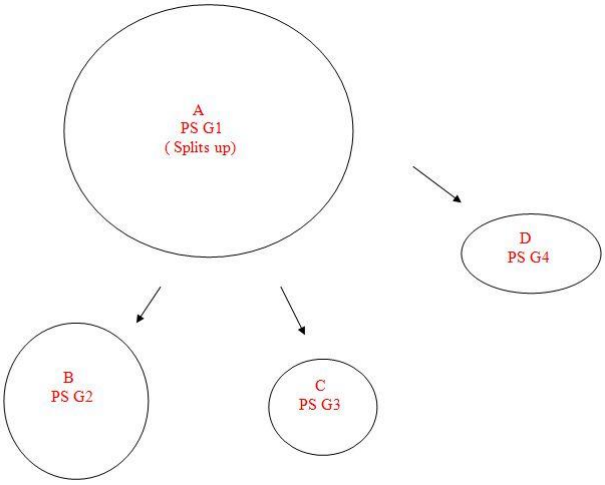

Figure 8

SUB DOMAIN GENERATION WITHIN A LARGE DOMAIN –

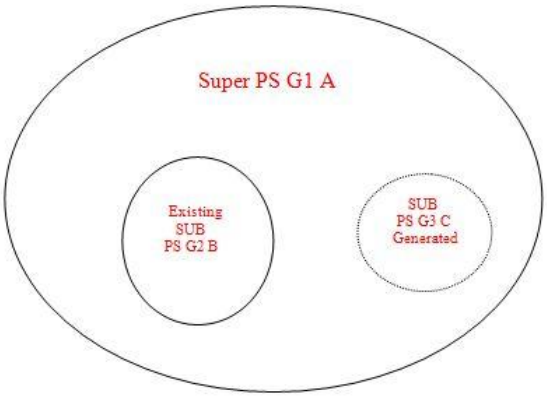

Figure 9

Effect on key distribution-

In case of a domain merge-
A new group key is generated for new group. Also, pairwise session keys are generated between members in domain1 and domain2 respectively. Suppose there are 'a' members in domain1 and 'b' members in domain2. So, total no. of session keys to be generated 'ab'. Each member in domain A receives b+1 keys ('b' pairwise session keys + 1 group key) and unicast message for domain B members contains a+1 keys ('a' pairwise session keys + 1 group key). Total no. of unicasts a+b. In absence of dynamics- No. of keys to be generated ${}^{a+b}C_2$ pairwise session keys + a group key.

In case of a domain split-
It is not required to generate pairwise session keys among group members, as they already exist . Only new individual group keys need to be generated and invalid session keys need to be flushed from local cache of group members.

In case of a sub domain generation-
A new group key should be transmitted to each group member

In absence of dynamics - No. of keys to be generated ${}^{n}C_2$ pairwise session keys, where 'n' is no. of group members in a newly generated sub domain + a group key

Example of rekey message transmission in case of Domain merge –

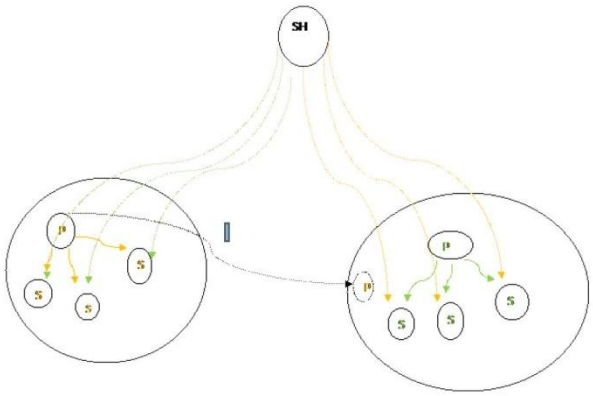

Figure 10

Publisher/Subscriber group in Domain A
Publisher/Subscriber group in Domain B
Rekey Messages sent from Security Hub to Subscribers of domain A (Only group key and pairwise session keys between members of domain A and B needs to be sent)
Rekey Messages sent from Security Hub to Subscribers of domain B (Only group key and pairwise session keys between members of domain A and B needs to be sent)

4. KEY GRAPH FOR GROUP COMMUNICATION AMONG SUBSTATION SIEDS

Some key graphs [7] -

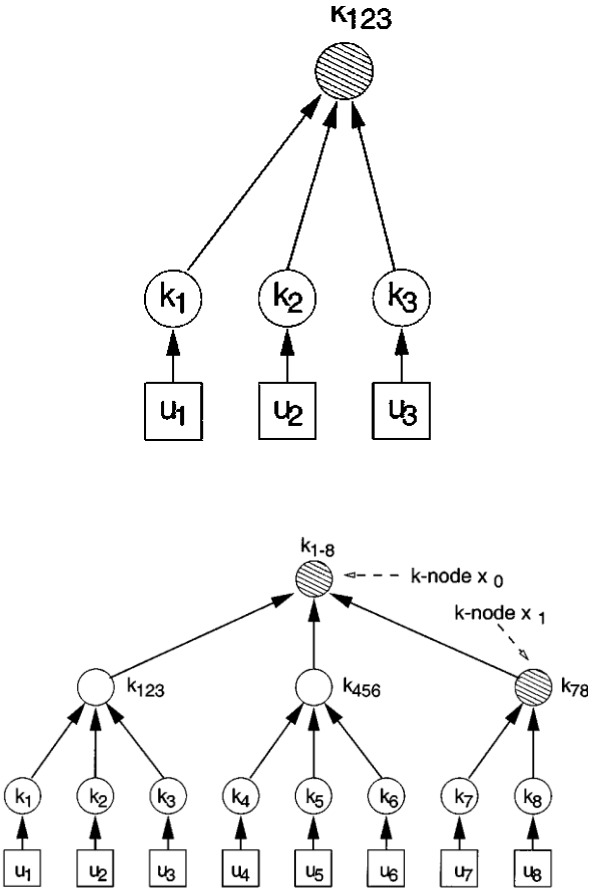

Figure 11

Advantages-
1)Reduction in no. of messages to distribute group key when a group leave operation is performed.
2)Enables sub-group communication via sub group keys.

Overheads-
1)Key graph generation, each time a group is created.
2)Number of keys to be changed when a group leave operation is performed.

Some fundamental questions/challenges which arise are Which key graph to use? A significant parameter on which choice of key graph might depend-Order of group. Is it feasible to make the nature of key graph dynamic?

Some Efficiency Measures to be considered
1)Network traffic (should be in a suitable range)

2)Network latency (should satisfy the requirements, try to achieve as minimal as possible)

3)Key management and distribution overheads (should be manageable)

5. POSSIBLE ATTACKS ON EXISTING SECURITY HUB ARCHITECTURE

As TLS Protocol is used in the connected network, various e-hosts use TLS tunnels to communicate with security hub, both of the attacks described here, exploit TLS protocol to launch a DOS attack.

1) TCP data injection -While TLS itself may be as secure as possible, the use of TCP without additional cross-layer functionality makes it almost impossible to protect TLS against Denial-of-Service (DoS) attacks on the TCP layer. For, TLS the insertion of any data will denote a DoS attack. TCP assembles IP packets into a data stream and hands this stream over to TLS. To suppress duplicate packets it hands over a certain range of bytes only once. If an attacker is able to inject some amount of random bytes into the data stream which TCP hands over to TLS, the data integrity check of TLS will fail, and the data has to be discarded. As TCP's decision is based on the unsecured entries of the TCP header, it will confuse injected and original data as duplicates and can drop original. This will lead to TLS layer needing data that the TCP layer believes to have already given, and therefore detects as duplicates. A TLS implementation will need to restart the entire TLS connection in such a case.

2)Forcing the TLS server to perform large number of illegitimate RSA decryptions-An attacker attempts to incapacitate a TLS server by initiating large no. of TLS handshake requests per second as the number of RSA decryptions the server can perform per second. A high end server can process upto 4000 RSA decryptions per second. If we assume that a partial TLS handshake takes 200 bytes, then 800 KB/s is sufficient to bring down the server.

6. SOLUTIONS

1)Adding authentication at the TCP layer [9]-The prominent problem of TLS is TCP offers a reliable service and delivers data to the upper layer protocols only once because it is built to suppress duplicates. TCP's decision is based on the unsecured entries of the TCP header, otherwise it could figure out that the received packet is a injected one. Even if TLS could figure out that a given data segment was injected, TCP would not pass TLS the original data since it seems to be duplicate. Solution is to add authentication at the TCP layer using TCP MD5 option. Authentication is attained by combining advantages of TLS and TCP MD5 option. TLS allows to setup a secure connection to an unknown peer, (though peers, ehosts and ihosts may not be unknown in this scenario) but misses authentication of data below the TLS layer. The MD5 option adds the missing data authentication but cannot set up connections with unknown peers. Two approaches can be combined, start TLS without protection first and use it to set up the MD5 option for the TLS connection as soon as possible.

Protocol description-

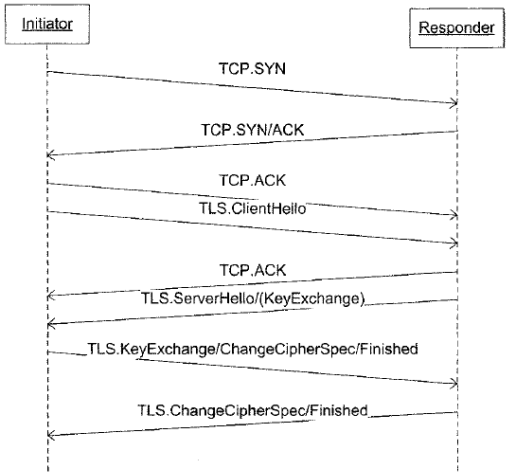

Figure 12

So,the time during which an attack is possible is reduced to the first packets, namely the TCP three way handshake and the first phase of the TLS key exchange. After the TLS handshake phase, the attacker would not be able to successfully inject packets into the TCP. Therefore, the TLS connection is secure for the rest of its lifetime

.

2) Client puzzle solution [8]-The idea of client puzzles is to slow down the attacker sufficiently that a denial of service is no longer possible. In order to prevent the denial-of-service attack against TLS, a new message is added after the Server Hello message and before the Server Done message. This message contains a cryptographic puzzle and is only sent when the server is under load. The server will then wait on a response message before continuing with the handshake protocol. The type of client puzzle used consists of inverting a hash function when given the hash digest and a certain portion of the pre-image. After the client has sent its client hello message, the server chooses a random bit value s and inputs it to a cryptographic hash function. It then includes the hash digest t = hash(s) along with the b first bits of s (where b < a) to the client in the server hello message. Using these b bits, the client solves the client puzzle via brute-force and finds a value s' that hashes to the desired t. With knowledge of the first b pre-image bits, the client only needs to attempt approximately 2 candidate values before finding a valid solution s' that satisfies t = hash(s'). The client then includes s' in its client key exchange message. Only if s' verifies - i.e, it is of correct

length and its hash output is t - will the server proceed with the SSL handshake and decrypt the encrypted session key submitted by the client.The addition of client puzzles to the SSL handshake protocol has the advantage of making DoS attacks more elaborate to carry out. A single client machine will in this case will no longer be able to easily overload an SSL server by sending consecutive SSL initiation requests, as it would need to solve the appropriate client puzzles, which will demand some time, thereby limiting the number of valid requests it could send per second.

3)CLIENT AIDED RSA [10]-The main idea is to shift some computational burden from the ihosts to the ehosts. Specifically, clients should perform the bulk of the work in RSA decryption, thereby allowing the server to accept and process more incoming requests.
Computation transfer- Represent the server's private exponent as d = f1d1 + f2d2 + ... + fkdk (mod φ(n)), where the fi 's and di's are random vector elements of c and |n| bits, respectively. The following processes take place when a server wants to transfer the computation x^d (mod n) to a client:

1. Server sends vector D = (d1 , d2, ..., dk) to client
2. Client computes vector Z = (z1, z2 , ..., zk), where $z_i = x^{di}$ (mod n),and sends it back to server.

3.Finally, server computes

$$\prod_{i=1}^{k} z_i^{fi} = \prod_{i=1}^{k} x^{fidi} = x^d \pmod n$$

Note that, assuming that it is computationally difficult to crack RSA, parameter selection should not introduce any attacks that compromise the security of the above computation by the server, namely x^d (mod n). An attacker can attempt to exhaust all possible vector values fi thereby deriving d. Thus, a minimal requirement for c and k is that a brute force attack (which requires $2^{c \times k}$ steps) should be as difficult as breaking underlying RSA. The client hello and server hello messages remain unchanged, although the server's certificate (which is sent as part of the server hello message) now includes the vector D = (d1, d2, ..., dk). The client chooses a random value x, which is then used to derive the TLS session key, and uses the server's public exponent to encrypt it : $y = x^e$ (mod n). Next, the client uses D to construct a vector Z by computing the individual vector elements $z_i = y^{di}$ (mod n), for $1 \leq i \leq k$. This vector is included in the client key exchange message. The server, upon receiving this message and derives $y^d = (x^e)^d = x$ (mod n).
Note: Chinese remainder theorem can be used to speed up RSA secret key exponentiations.

7. MODIFICATIONS TO EXISTING IEC 61850 PROTOCOL STACK

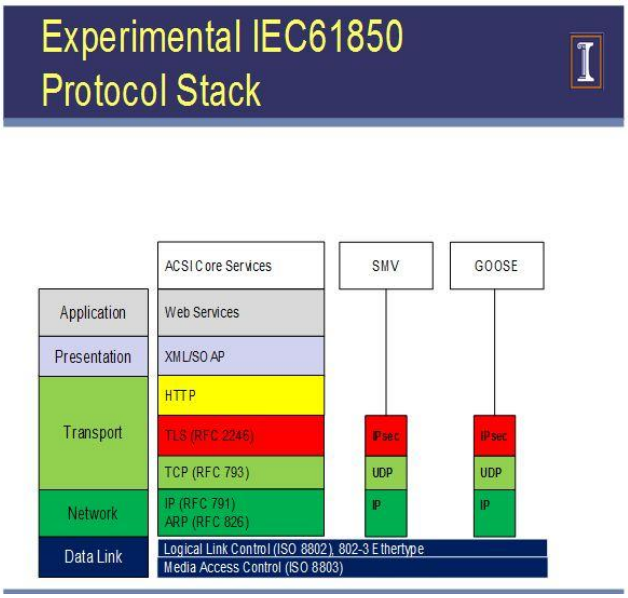

Figure 13

Existing IEC 61850 protocol stack -

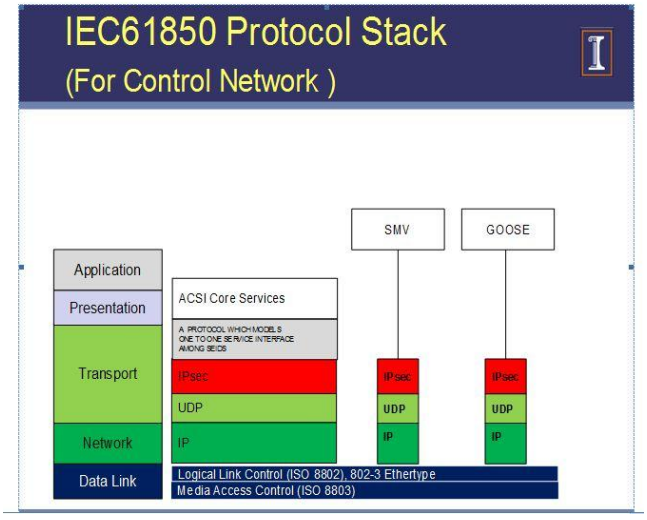

Figure 14

A New Component added to protocol stack, to meet one to one communication need among SIEDS. -

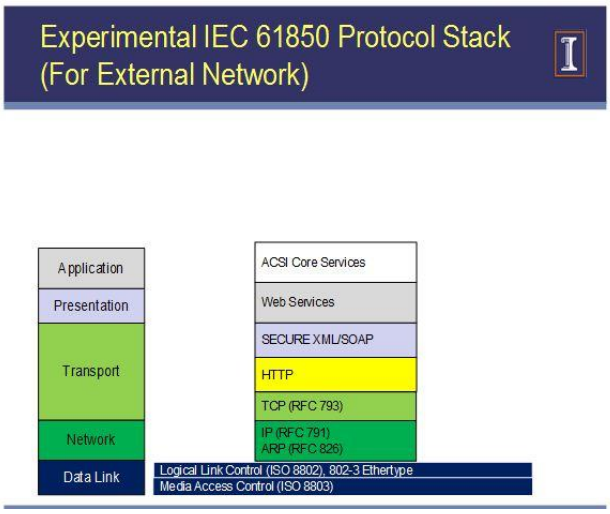

Figure 15

7.1 How XML security will work in this scenario and suits the needs of the architecture?

As information between ihosts and ehosts is exchanged in XML format, security can be embedded in the document itself, eliminating the need of TLS.XML document being sent to the ihost is digitally signed by the respective ehost.ihost on receiving the document verifies the digital signature of ehost. Respective public keys of ehosts can be obtained easily in a secure manner by utilizing DNSSEC architecture .

Some possible benefits as compared to TLS-

Suppose, an XML document say a substation configuration file is to be transferred whose contents are filled by some specific ehosts i.e there are different authentication requirements for different sections of a document. On using TLS, each ehost will establish a separate TLS connection with the ihost in the control network and send the required information. On using XML security options, it can be easily accomplished by using digital signatures. Each ehost will sign its section in the document and this document will be propagated to the control network. On receiving the document, ihost can easily verify, individual signatures. Also, it will be possible to encrypt specific sections of document. SOAP accounts[11] should be included in the soap messages to provide protection against XML rewriting attacks.

8.ACKNOWLEDGEMENTS

I would like to thank Professor Carl Gunter and Jianqing Zhang for their immense guidance and support.

9. CONCLUSIONS

In this paper we explored group communication possibilities for SIEDs. We discussed security hub architecture support for publisher/subscriber model used for transmission of messages in substation network. We studied role of security hub in secure group communication, its role as Key Manger and Distributor. Group Dynamics among SIEDs were observed. Use of key graphs for secure group communication among substation SIEDs was also discussed. Finally, we discussed attacks on existing security hub architecture. Both of the attacks ,exploited TLS protocol to launch a DOS attack. Certain solutions to these attacks were discussed such as adding authentication at the TCP layer, client puzzle solution, client added RSA. We also suggested a modification to IEC 61850 protocol stack. In future we aim to explore behaviour of groups in a substation,design, experiments to observe factors such as group order, group dynamics, study key graph issues for group communication in SIEDs and continue to refine security hub architecture.

10. REFERENCES

[1]IEC 61850 Communication Networks and Systems In Substations, Technical Committee 57, International Electrotechnical Commission

[2] Secure Intelligent Electronic Devices (SIEDs).C. A. Gunter, S. T. King, J. Zhang. *PSERC 2007*

[3]Overview of IEC 61850 and Benefits, R. E. Mackiewicz. PES TD 2005/2006

[4]IEC 61850 Communication Networks and Systems In Substations: An Overview for Users. D. Baigent, M. Adamiak and R. Mackiewicz. *SIPSEP 2004*

[5] A Survey of Multicast Security Issus and Architectures : Kuris

[6]Multicast security issues : Canetti and Pinkas 1998

[7]Secure group communications using key graphs Chung Kei Wong; Gouda, M.; Lam, S.S.Networking, *IEEE/ACM Transactions* Volume 8, Issue 1, Feb 2000 Page(s):16 - 30

[8]Using client puzzles to protect TLS

Source : USENIX Security Symposium archive *Proceedings of the 10th conference on USENIX Security Symposium* - Volume 10 ,Drew Dean Adam Stubblefield

[9]Vlker, L. and Schller, M., Secure TLS: Preventing DoS Attacks with Lower Layer Authentication. *In: Kommunikation in Verteilten Systemen*, Springer Berlin Heidelberg. pp. 237-248.

[10]Improving Secure Server Performance by Rebalancing SSL/TLS Handshakes, Cryptology ePrint Archive 2005/037, IACRby Claude Castelluccia, Einar Mykletun, Gene Tsudik in Proceedings of the 10th *Annual USENIX Security Symposium*

[11]SOAP - based Secure Conversation and CollaborationRahaman, M.A.; Schaad, A.Web Services, 2007. *ICWS 2007. IEEE International Conference* Volume , Issue , 9-13 July 2007 Page(s):471 - 480

[12]Towards secure SOAP message exchange in a SOA by:Mohammad A Rahaman, Andreas Schaad, Maarten Rits(2006), pp. 77-84.

[10]CISCO press - Basic IPsec VPN Topologies and Configurations

[11]CISCO IOS security configuration guide

[12]Security hub architecture Draft July 2007,UIUC
Carl A. Gunter, Sam King, Jianqing Zhang

[13] Understanding and Simulating the IEC 61850 Standard .Yingyi Liang Roy H. Campbell.IDEALS,UIUC Tech Report.

[14] Secure Multicast for Power grid communications Jianqing Zhang. Doctoral Thesis, University of Illinois, September 2010.

[15]Application-Aware Secure Multicast for Power Grid Communications . Jianqing Zhang, Carl A. Gunter. The *1st IEEE International Conference on Smart Grid Communications*, Gaithersburg, MD. October 2010

[16]On the Latency of IPsec Multicast in Power Substation Local Area Networks Jianqing Zhang, Carl A. Gunter. Manuscript, Urbana, IL. June 2009

[17]Evaluating A Secure Protocol Scheme for Control Networks on the DETER Test Bed Jianqing Zhang, Carl A. Gunter. Manuscript, Urbana, IL. June 2008

[18]The Protection of Substation Communication
S Fuloria, R Anderson, K McGrath, K Hansen, F Alvarez. *In proc. of SCADA Security Scientific Symposium*, Jan 2010

[19]Key Management for Substations: Symmetric Keys, Public Keys or No Keys? S Fuloria, R Anderson, Fernando Alvarez, K McGrath. *PSCE 2011: the IEEE Power Systems Conference & Exposition*, March 2011, Phoenix, Arizona, USA