IRTF SMug, 4-May-2000 Columbia, MD

Agenda
~~~~~~
4-May-2000
1. Intro
2. Presentations
-C.Meadows
-H.Harney
- Break -
3. Problem-Area I (Ran Canetti)
- Lunch -
4. Problem-Area I (continued)
- Break -
5. Problem-Area II (Thomas Hardjono)

5-May-2000
1. Problem-Area II (continued)
2. Problem-Area III (Hugh Harney)

I. Introductions were made and agenda discussed
II. Presentations
1. What Analysts Need From a Protocol Specification
Cathy Meadows, NRL
In her presentation Cathy briefly motivated the benefits of formal analysis of security protocols and reviewed application of techniques such as theorem proving, model checking and high level belief logic. Formal analaysis has been done by NRL and others on IKE and SET. Analysis is needed for group protocols such as the security protocols that SMuG is exploring.

Cathy said that (1) who sends a message, (2) message definition, and (3) message processing must be specified in formal analysis. Assumptions about the message-passing environment and intruder capabilities must be documented to evaluate how well a given protocol meets its goals to thwart attacks by preserving key secrecy, authentication properties, or other documented goals.

Cathy discussed the experiences and shortcomings of some communications and eCommerce security protocols and stressed the importance of clarity in identification of principals. Also, when a security protocol uses services from another security protocol, the analysis must consider how intruders' capabilities are limited in both protocols.

Protocol developers can aid formal analysis when they carefully specify the following information.
- o What are the principals' identities?
- o How are keys bound? Are they bound to principals who are executing programs, to the key management system acting on their behalf, or to some other entity?
- o What do principals know?
- o What is kept secret?
- o What must be "fresh?"
~~~~~~

- What is authenticated?
- What events are guaranteed to happen previous to a given protocol procedure?
- What definitions of the underlying security properties are needed? For example, define "PFS" rather than refer to it.
- DoS/Replay requirements or assumptions should be stated up front.

Authentication protocols are well understood, particularly in the point-to-point case. Less is known about ecommerce and very little is known about group protocols. Protocol designers of group protocols should take care to specify and document their protocols with an eye toward the analysis to be subsequently performed on them.

Cathy's talk was followed by discussion on the applicability of natural language specifications for protocol developers and implementers. Analysts convert these specs to formal models, which require at least the information mentioned above. There was some trailing discussion on the problematic use of timers and synchrony in security protocols.

2. Multicast IPSec Policy - Hugh Harney, SPARTA

Each Security Association used for group key management has policy definitions for the crypto suite used, authentication of principals' identities, and credentials needed by principals. Hugh and Andrea Colegrove mapped the GSAKMP Policy Token to the fields used in the IPSec SPD and SAD. This was done for the GKMBB Category 3(data protocol SA) and Category 1(key pull SA). The Category 2 (key push SA) would not use IPSec or any other underlying security protocol under GSAKMP or GKMBB. Hugh gave us a handout that included Category 3 and Category 2 definitions. He presented the Category 3 policy definitions, and a draft will be forthcoming.

Mapping of Category 3 information assumes that GSAKMP is used to initialize an IPSec SA from group sender(s) to receivers to run ESP, for example. The SA is defined by a 6-tuple.

- Source Address, which may be * for the Group Authentication option
- Source Port, which may be "*" for the Group Authentication option
- Destination Address
- Destination Port
- Transport Protocol
- Name, which is a 4-byte random value to accommodate the fact that an IP Multicast Destination Address/Port may be reused

There was discussion concerning the selectors such as the applicability of a fully-specified Source Address/Source Port, which is needed for the Source Authentication option and will be used independently of the SrcAuth when Single Source Multicast is used. The fully-specified Source Address/Source

Port does not scale well for M:N multicast as M, the number of senders grows. We see no way to avoid supporting both options, however, since many applications will be single source or will want source authentication (presumably for a small number of sources).

The Principals' identities are conveyed in a GID tuple in the key management protocol.

- o Member identity can be IP address (deprecated by IAB), host or user domain name, public key, shared key, and other means.
- o Group identity, the <Destination Address, Destination Port, Name> tuple contained in the selector

There is IPSec-specific information for Situation, Authentication/Integrity, and Encryption. Also specified in an IPSec SPD is the security protocol, which should be ESP and which comes with data-origin authentication, integrity, replay protection, and data confidentiality.

Hugh's handout maps the fields found in the IPSec SPD to the GSAKMP Policy Token.

The IPSec SAD is an instantiation of an SPD entry which includes Replay counter, SPI, and keying material. Replay counter works when source is identified in the selector. It's not possible to have Replay protection for each source of an M:N session.

The SPI, augmented with SRC/Dest selectors identifies packets belonging to the SA at the member-receiver and will change when the SA changes. The MTU is meaningless in multi-sender groups.

The Category 1 SA is the policy for the "secure channel" that we discussed extensively during Thomas' report.

More discussion followed Hugh's report. Joe talked about single source multicast and the fact that multiple secure groups will send to the same multicast address in a single source environment. There was more discussion on the replay issue when there are multiple sources and some speculation on how this can be made to scale.

III. Problem Area I: Multicast Data Security Transforms, Ran Canetti

Ran presented the transforms draft that he wrote with Pankoj Rohatgi and Pau-Chen Cheng from Watson. 3 functional building blocks are defined

- o GA: group authentication
- o GS: group secrecy guarantees that only group memebers have access to communicated date
- o SrA: guarantees data originates with claimed sender and was not modified en-route

These functional building blocks are realized by two protocol

building blocks, MESP and AMESP. Thus the protocol building block transforms satisfy the requirements of multiple functional building blocks. An important functional requirement is support of RM protocols.

- SrA is easier if RM is provided
  FEC-based RM needs SrA on individual packets/frames (i.e. below RM xform) because FEC-based RM is vulnerable to DoS
- Retransmission-based RM needs encryption 'above' the RM xform (this is because routers and non-group members may function as 'repair nodes' in some some schemes.

The functional requirements thus span internetwork, transport and application layers so layer placement of the xforms is an issue. SrA is computationally intensive (but not always). Ofter there are inter-packet dependencies making message-level (transport or app layer) SrA processing preferable. New SrA algorithms continue to be developed so future placement at internetwork layer can't be ruled out though placement at the transport or application layer seems best at the present time.

GA and GS are efficient for both packets and messages and can live at any of the 3 layers. The order in which GA, GS, and SrA are performed maps to the layers, however, although any order is cryptographically sound (except that sign preceeds encrypt if non-repudiation is desired). Here are some recommended orderings.

- GA is easy to do, weak guarantee; makes sense to do first (i.e. to do at the "outer layer" in the notation below) and to do this at internetwork layer when possible to eliminate DoS from non-group members
- SrA before encryption to allow GS in a lower layer for applications where this is desired.

Ran used "outer" and "inner" notation where outer transforms are performed before the inner transforms. For example, GA(GS(SrA(data))) allows GA at internetwork layer, GS at internetwork or transport/appilcation layers, and SrA at transport or application layers. If non-repudiation and GA DoS protection is unneeded, SrA(GS(data)) can be done at transport/ application layers though it does not give non-repudiation. 2 identical transforms, one in IP layer and one in transport/ application layer can provide full security functionality: SrA, GA, GS though group policy specifies where is each functional BB is to be carried out. MESP realizes the functional BB at the internetwork layer and the AMESP protocol BB realizes the functional BBs at the transport or application layer.

The IP-layer transform, MESP, is an extension of ESP where the processing includes not only GA and GS, with no (or minimal?) changes to ESP, but SrA can be added by MESP developers if this is desired in the future. When there is no SrA in MESP, we get IPSec ESP.

Application-layer MESP, called "AMESP," is identical to MESP except that it occurs after the IP and Transport header and makes no use of the next-protocol field. AMESP does everything in

the application layer.

Here are some AMESP transforms.
- o AMESP with GA[GS[SrA[data]]]
- o AMESP with SrA[GS[data]]
- o null MESP

The Group Authentication service may be done entirely in the transport/applicatoin layer. So too for the Group Secrecy service. As described earlier, SrA is probably better done in the transport/application layer.

There was discussion on the various alternatives of header and transforms placement. The proposal was very well received by meeting participants.

4. Problem Area II: Thomas Hardjono

Thomas presented key management message exchanges for the key management building block draft he is writing with Hugh Harney and Mark Baugher. The first draft, draft-irtf-smug-gkmbb-gsadef-00.txt, reviewed objectives, requirements, and properties of the Group Security Association in the context of the SMuG Reference Framework. The GSA is composed of a Category-1 SA (Pull SA), a Category-2 SA (Push SA) and a Category-3 SA (Data SA). Category-1 is used to initialize or re-initialize a member with keying material. The Category-2 SA implements an efficient multicast key management algorithm such as LKH+ to add and remove secure multicast group members. The Category-3 SA would be used with MESP, AMESP or some other packet or message security protocol.

draft-irtf-smug-gkmbb-gsadef-00.txt was presented at the February SMuG meeting at ACIRI. This work was well received and the consensus of the ACIRI meeting was that the authors should proceed to specify flows for the GSA. Thomas presented a foil showing flows for the pull SA and push SA. The flows were high-level and abstract; they are layer-independent and can make use of various group determination algorithms such as LKH, OFT, or LKH+. New payload definitions were introduced and described by Thomas such as the GID, which identifies the secure group, the PT, which is the Policy Token being defined by Hugh, Patrick and others, and the key array, which is defined by the specific group determination algorithm.

Paul pointed out that the exchanges do not include enough authenticating information for AKE. Thomas explained that this is the next layer of detail that will be included in the draft. There was considerable discussion on the role of the "secure channel" and the credentials that are used and exchanged in the secure channel that commences the pull exchange. Ran and others questioned the need for AKE following secure channel establishment. A lengthy discussion ensued regarding whether the credentials used for the secure channel are sufficient for admitting the member into the group. An alternative view is that the credentials used to establish the secure channel enable the member to securely communicate with the GC+KS and that a different set of credentials are needed to admit the member into the secure multicast group. Thus

there are two possibilities. First, there is a single-phase approach where the secure channel is established with the credentials that enable the member to join the particular group. This is much simpler than the scheme Thomas presented. The second scheme, the one that was presented, is a two-phase establishment where the download of policy information is done under the protection of the secure channel. The two-phase approach allows the member to determine whether it has the credentials needed to join the group. The two-phase approach also allows the secure channel with the GC+KS to be reused by the member, who may be likely to join multiple groups.

The secure channel/phase discussion led into two other topics, the trust relationships and the processes that preceed key determination. The group enumerated three distinct processes.

- group announcement: various means may be employed to make potential members aware of the secure multicast group and how/where potential members may get credentials for the group.
- group registration where the potential member gets the needed credential to join a particular group.
- group key determination where the member gets initialized and reinitialized with keying material and includes message exchanges to maintain and delete the group.

The key management building block is concerned with the third step, key determination. How a potential member learns of the group owner and group policy relates to the three trust relationships of the model.

- Group owner establishes policy and signs the policy token that specifies the cryptographic and authentication requirements for the group along with all aspects of group key management such as the algorithms used for adding and removing group members. The group owner delegates the management of the group to the GC+KS, which it trusts for admitting, keying, and removing members from the secure group.
- GC+KS distributes keying material to group members according to the group policy. The GC+KS is the agent of the Group Owner and will remove members at the command of the Group Owner. The GC+KS requires potential members to present credentials as specified by the group owner; these credentials will be used by members who need to be re-initialized with group keying material during the lifetime of the group, which it destroys according to policy or by command by the Group Owner. The GC+KS may delegate key distribution to a key server, according to policy. Either the GC+KS or a designated key server will send authenticated keying material to Group Members, which they trust to use in accordance with Group Policy.
- Group Members present the group credential to the GC+KS to be initialized or reinitialized with keying material. The Group Member checks that the source of the keying material is from a the GC+KS designated by the Group Owner or by some authenticated key server that has been delegated by the GC+KS. These trust relationships are embodied in the Policy Token.

Group Policy is communicated in the Policy Token. The SMuG participants discussed various alternatives for how the potential member learns of the group, gets the group credential, gets the

group policy, and gets keying material.

The discussion went to the end of the first day and through most of the next morning. Thomas said that the team working on the group key management building block will clarify the issues and make proposals in the forthcoming draft document. Two related issues need to be addressed.

- The nature of the credentials used in secure channel establishment
- Whether AKE is needed within the secure channel

5. Conclusion

Hugh and Patrick deferred the Policy Token presentation to the next SMuG meeting, which will be held at the next IETF meeting.