Minutes of 11 Feb 2000 SMuG Meeting minutes, ICSI ACIRI, Berkeley, CA
~~~~~~

Scribe: Mark Baugher

Agenda
~~~~~~

1. 9-10 presentations
   A.Perrig, multicast source authentication
2. 10-12:20 GKM BB
3. 12:30-1:30 Lunch
4. 1:30-2:30 policy BB
5. 2:30-3 Break
6. 3-5:30 data handling BB

List of Attendees

1. A.Perrig, Efficient Authentication and Signature of Multicast
   Streams over Lossy Channels is a paper prepared by Perrig,
   Canetti, Tygar, and Song; available after 10 March at
   http://paris.cs.berkeley.edu/~perrig/projects.html.
   Approach separates authentication from non-repudiation
   and extends "Guy Fawkes" protocol. 2 sets of protocols
   presented, 5 variations on the authentication protocol and
   one non-repudiation. Authentication schemes feature
   low overhead on the order of one MAC and 2 prf's per packet
   for both sender and receiver; there's much less state and
   application dependence than previous tree-based models,
   which do not separate authentication from signatures.
   The packet overhead is as low as 25 bytes for authentication
   and 50 bytes per packet for non-repudiation. Basic idea is
   that the sender creates and uses a secret to compute a MAC,
   which the receiver caches until the secret is revealed in
   a later packet. Security of the system is based on very
   loose time synchronization between senders and receivers
   to preserve the security condition: Data packet Pi arrived
   safely if receiver can unambiguously decide based on its
   synchronized time a bounded clock drift that the sender
   could not yet send out the corresponding key disclosure
   packets Pj>i. The five multicast source authentication
   schemes deal with multicast distribution, packet loss,
   arbitrary packet rates, variable packet rates, and a
   heterogeneous receivers with a wide range of clock drift.
   Much of the resilience and efficiencies of the schemes result
   from the use of a key chain of the non-invertible PRFs that
   minimizes state and computation as well as permit space-time
   tradeoffs to be made by individual receivers. Note that
   the five schemes have different capabiities and restrictions.
   Whereas scheme I is not tolerant to loss, scheme 2 is but
   is sensitive to delay, and scheme 3 removes the sensitivity
   to delay for fixed-rate streams, etc.

One scheme was presented for non-repudiation that used a similar approach by chaining hash values computed across a set of packets without the need to delay individual packets. Though the receiver may need to wait to verify individual packets. An optimistic authentication approach is discussed under the stream authentication section of the paper where the application is informed of earlier packets that were delivered but not authenticated.

There was some controversy regarding the synchronization requirements of some schemes and the effects of delay on the protocols. Performance results from the paper were also discussed. There were a number of participants expressing the view that this was a very promising approach.

2.R.Canetti, Reliable multicasting of rekey information

Ran led a discussion on the need to complement scalable key distribution with reliable multicast. There is an issue regarding whether an RM protocol should be incorporated into the multicast key distribution protocol or whether an RM protocol should be a layer independent of the key determination. It is possible to piggyback on data flow. This flow may be lossy, in the case where it is not an RM flow, and such an approach breaks the data/control separation.

Ran described a "multi-tiered" solution where each re-key message contains information on several re-key events using FEC. When the group member cannot reconstruct the key messages, it must re-initialize through the Group Controller/Key Server. This may lead to scalability issues. The "multi-tiered" solution has "outposts" of key servers to hierarchically support both push and pull of keying material from sub-populations of group members.

There was a brief, lively discussion regarding layering of RM services versus incorporating RM in key management. Some discussion regarding the type of RM appropriate to key management and whether RM needs to be considered for group key management at all. It was decided to resume the discussion later in the day prior to the data handling building block presentation.

3. T.Hardjono,Group Security Association (GSA)Building Block Concepts

Thomas presented work done on the group key management building block to identify constructs needed to support the distribution of multicast keying material. The presentation was based on the draft document

http://search.ietf.org/internet-drafts/draft-irtf-smug-gkmbb-gsadef-00.txt

The GSA answers the questions of state needed to support three procedures

o The "pull" of keying material by a member who joins a secure

multicast group; in the SMuG Reference Framework, the member is either a sender, a receiver, or both sender and receiver to the secure group. The pull request is made by the member to the Group Controller/Key Server (GC+KS). Following mutual authentication and member authorization, the GC+KS distributes the keys to the requesting member, which repeats the pull request when it needs to be re-initialized with keys.

- The protection of data sent to the secure multicast group using the keying material that is pulled or pushed from the GC+KS.
- The "push" of keying material from the GC+KS to achieve scalable key distribution to members using multicast distribution techniques and an efficient algorithm such as LKH or OFT.

The distribution of keys and protection of data require the establishment of security state. In the "pull" case, security state must be maintained between the GC+KS and each member though the connection used to pull the keying material may be brought down once the member receives the keys. Conceptually, a security association (SA) must be established to accomplish the pull and which initializes the member with keying material needed for a second type of SA that exists between group senders and group receivers. Call this a "type" of SA since there could be multiple SA instances, such as one for each sender to the secure multicast group or a "bundle" of logically-related SAs for a multimedia session. The type of SA between senders and receiver members is called a "Type 3 SA" and the type of SA between GC+KS and members is called a "Type 1 SA." A "Type-2 SA" is also between the GC+KS and all members where keys are distributed using multicast techniques and algorithms such as LKH and OFT. Type-2 is a "push" SA and a Type-1 SA is the "pull" SA. Only Type-3 and Type-1 SAs are strictly required for a secure multicast group as many group key management schemes do not use multicast techniques for distributing keys; scalable key distribution is not needed for some solutions (GKMP, MARKS) and some applications, such as "pay-per-view" applications.

Thomas drew the SMuG Reference Framework using different colored arcs for each SA type. The three types of SAs are aggregated into a GSA, which is established, maintained and destroyed for a secure multicast group by a group key management protocol. Each SA closely resembles an SA from the Internet Security Architecture as the SA has transport address/protocol selectors, keys, policy information, group identifiers and other attributes. Type-2 and Type-3 differ from an IPSec SA in that it's SPI is selected by the source (or the GC+KS on behalf of the source), not the receiver. There are issues with SPI uniqueness in this approach that have been discussed in an earlier draft by Hardjono et. al. along with possible solutions. These issues will be considered in the next stage of the group key management building block work, which will develop protocol exchanges/flows for GSAs. There will be a draft describing the concepts following by another draft describing the exchanges/flows.

In the discussion, it was pointed out that the reliability issue

is concerned with the Type-2 GSA. Some participants pointed out that use of Type names may be confused with U.S. Government Type I and Type II applications. There was some controversy over the need for multiple instances of SAs in the GSA concept. Overall, three seemed to be consensus that the GSA concept is important to group key management and central to work on the building block. Ran asked the for a show of hands of persons interested in working on the group key management building block and there were at least a half dozen people indicating interest. Thomas said he would followup with a next-steps proposal for this group within SMuG.

4. H.Harney, Group policy
   Hugh defined group policy as the cryptographic and group-trust parameters that inform each member the criteria for belonging to a group, how the group will be maintained and by which entities. Group policy must be unambiguous and verifiable whereby all members enforce and adhere to a published policy concerning access to group keying material and authorization to perform security actions such as admitting members, re-keying members, and evicting members. The acceptable infrastructures for group policy include the GSA and PKI infrastructures.

   There are local and group-specific policy sets. Latter includes policy identification, access, authorization, mechanisms/SAs, policy verification (e.g., Antigone) and includ access policy such as an ACL but alternative approaches include rule-based (or role-based) access control. Former (local policy) includes local configuration deata describing requirements of received policies - infrastructure policy of local credentials and trusted parties.

   Hugh presented a tree structure with a root having five descendent objects defining a policy structure: Group Identification, Access, Authorization, Mechanisms and Verification. Each object has various attributes and the tree of objects and attributes defines a group and local policy structure. This work will be written up in a forthcoming draft for the group policy building block.

   During discussion, some participants (Da Mingh, Gene) thought that a smaller set of policy objects and attributes, which was simpler, would be preferable. Specifically, less in authorization policy would be better according to some participants. There were also questions about what policy does the member need to know versus what the GC+KS needs to know. Hugh answered that the member would need the information contained in the tree structure. The sender needs to know the criteria for a member to gain access to group keying material (Access), which entities may admit members and rekey the group (Authorization), how group would be rekeyed and the crypto policy (Mechanisms), the identity of the group (Identity), and the Verification of the source of the policy.

   Hugh was encouraged to write up the presentation in the form of

a draft prior to the Spring SMuG meeting.

5. ReDiscussion on rekey
   Mark Baugher proposed relegating reliability for Type 2 to a reliability layer and not incorporate these mechanisms into the key determination flows for Type 2. Debate over whether there is a 'chicken and egg' problem: Ran suggested that there is an apparent (as opposed to real) chicken and egg problem if it were perceived that GKM relied on the output of RMT. Discussion ensued regarding the idea of tunneling keying material to the source for distribution. Hugh said the issue is an application problem that does not belong in group key management. Mark Handley and others suggested that SMuG building blocks should not recreate RM mechanisms. Further discussion wandered into the need for heartbeat mechanisms and scalability issues of lost re-key messages triggering requests to the GC+KS by potentially a large number of members. In summary, three options were outlined: do nothing (Hugh's proposal), do a minimal RM protocol, much like the reliability mechanisms in IKE, use an available RM protocol. MarkB agreed to post a summary to the SMuG list for further discussion.

6. R.Canetti, P.Rohatgi, Multicast data security transforms (Area 1)
   Ran presented the data handling (Area 1) building blocks for group authentication, data encryption, and source authentication. The latter is the hardest problem in Area 1, possibly the hardest problem for secure multicast.

   Source authentication can be computationally intensive, often require significant state kept across frames (except Rohatgi's), and there are many different algorithms, new ones continually proposed. Current methods seem more suitable for the application layer, but it would be preferable to have this function in the internetwork and/or transport layers to relieve application programs of the need to incorporate such complexity and to better protect against denial of service attacks.

   Whereas most source authentication algorithms are better suited for the application layer, Ran presented "deployment considerations" that assigned group authentication (packet authentication using a group key) to the IP layer. Data encryption, another data-handling service, can also be placed at the IP layer.

   Group authentication and data encryption at the IP layer may make use of ESP (SSL?). The order of application of the various services was also considered under "deployment considerations." Ran reported that any order is cryptographically sound for group auth but source auth is preferable before encryption to allow encryption to be done at a lower layer. Ran presented a table describing suites and ordering.

       I   SA in app, enc+GA in ip layer (esp)
       II SA in app, end+GA in app (new, esp-like transform)
       III SA in IP layer, en_GA in IP layer (ESP) {FEC needs it??}

IV SA+ENC+GA in transport layer (one monolithic transform?)

The Baltimore SMuG meeting proposes doing III and II. Three transforms were also proposed.

1 SourceAuthT: Appl-layer source auth xforms; several instantiations already exist; either with or w/o RM; independent of whether enc in IP or in application
2 ESP: For IP layer group auth, encryption
3 AESP: application-layer ESP, with similar layout (some changes to header fields accommodates new layer placement), The original idea was to always do group authentication with encryption, but a new idat is to do group auth at IP layer with encryption at another layer.

Ran promised that an upcoming draft will review prominent suites for the data-handling building block and present a more detailed specification.

There was discussion concerning the appropriateness of hybrid stream/block ciphers, which would require change to the current ESP specification. There was also discussion on the idea of expediting the work that Adrian Perrig presented on source authentication, possibly as a IETF working group separate from data handling, key management and group policy. Thomas proposed that the Perrig et. al. paper be submitted as an IETF draft as a first step.

7. Next meeting

Thomas will post a message to the list to determine interest in meeting in Australia. If not, we will find another meeting time in April-May timeframe.

Name	Affiliation	Email
Mark Baugher	PassEdge	mbaugher@passedge.com
Ran Canetti	IBM Watson	canetti@watson.ibm.com
Olivier Cevassut	LBL	ochevassut@lbl.gov
Dah Ming Chiu	SUN Microsystems	chiu@east.sun.com
Mark Handley	ACIRI	mjh@aciri.org
Eric Harder	NSA	ejh@tycho.ncsc.mil
Thomas Hardjono	Nortel	hardjono@nortelnetworks.com
Hugh Harney	Sparta	hh@sparta.com
Mike Luby	Digital Fountain	luby@dfountain.com
David McGrew	CISCO	mcgrew@cisco.com
Roger Kermode	Motorola	Roger.Kermode@motorola.com
Amit Kleinmann	NDS	akleinmann@ndsisrael.com
Catherine Meadows	NRL	meadows@itd.nrl.navy.mil
Adrian Perrig	CMU/UCB	adrian@cs.berkeley.edu
Bob Quinn	stardust.com	rcq@stardust.com
Lars Rasmussen	Digital Fountain	lars@dfountain.com
Ignacio Solis	ISI/USC	isolis@isi.edu
Dawn Song	CMU/UCB	dawnsong@cs.berkeley.edu
Paul Syverson	NRL	syverson@itd.nrl.navy.mil

Gene Tsudik	UCI	gts@isi.edu
Brian Weiss	CISCO	bew@cisco.com