                Report of the IAB Security Architecture Workshop

1. Status of this Memo

   This memo provides information for the Internet community. It does
   not specify an Internet standard of any kind. Distribution of this
   memo is unlimited.

2. Copyright Notice

3. Abstract

   On 3-5 March 1997, the IAB held a security architecture workshop at
   Bell Labs in Murray Hill, NJ. We identified the core security
   components of the architecture, and specified several documents that
   need to be written. Most importantly, we agreed that security was
   not optional, and that it needed to be designed in from the
   beginning.

3.1. Specification Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119.

4. Motivations

   On 3-5 March 1997, the IAB held a security architecture workshop at
   Bell Labs in Murray Hill, NJ. The ultimate goal was to design a
   security architecture for the Internet. More concretely, we wished
   to understand what security tools and protocols exist or are being
   developed, where each is useful, and where we are missing adequate
   security tools. Furthermore, we wanted to provide useful guidance to
   protocol designers. That is, if we wish to eliminate the phrase
   "security issues are not discussed in this memo" from future RFCs, we
   must provide guidance on acceptable analyses.

There were twenty-four attendees (their names are listed in Appendix
A). Perhaps not surprisingly for such a group, the overwhelming
majority used some form of cryptography when connecting back to their
home site from the meeting room. But the situation on the rest of
the Internet is not nearly as good; few people use encryption, even
when they should.

The problem is that the rate of attacks is increasing. Apart from
the usual few elite hackers -- the ones who find the new holes --
there are many canned exploit scripts around. ("Click here to attack
this system.") Furthermore, the attackers have gotten smarter; rather
than going after random university machines, more and more are
targeting the Internet infrastructure, such as routers, high-level
name servers, and the like.

The problem is compounded by organizational laziness. Users and
system administrators want "magic security" -- they want whatever
they do to be secure, regardless of whether or not it is, or even can
be.

5. General Philosophy

We concluded that in general, end-to-end security is better. Thus,
one should use something like PGP or S/MIME for email, rather than
relying on an IPsec layer. In general, relying on the security of
the infrastructure is a bad idea; it, too, is under attack. Even
firewall-protected intranets can be subverted. At best, the
infrastructure should provide availability; it is the responsibility
of individual protocols not to make unreasonable demands on the
infrastructure during an attack.

6. IETF Structure

Our security problem is compounded by the IETF's inherent structure
(or, in some cases, the lack thereof). By intent, we are a volunteer
organization. Who should do the security work? The other protocol
designers? Often, they have neither the time nor the interest nor
the training to do it. Security area members? What if they are not
interested in some subject area, or lack the time themselves? We
cannot order them to serve.

To the extent that the IETF does have management, it is embodied in
the working group charters. These are in essence contracts between
the IESG and a working group, spelling out what is to be done and on
what schedule. Can the IESG unilaterally impose new requirements on
existing working groups? What if security cannot be added on without
substantial changes to the fundamental structure of a protocol that
has been reworked over several years?

Finally, there is a perception problem: that IPsec will somehow
solve the security problem. It won't; indeed, it can't. IPsec
provides excellent protection of packets in transit. But it's hard
to deploy on individual hosts, does not protect objects that may be
retransmitted (i.e., email messages), does not address authorization
issues, cannot block against excess resource consumption, etc.

7. Documents to be Written

   Collectively, we decided on several documents that need to be
   written:

      Taxonomy of Attacks
         In order to defend a protocol against attacks, one must, of
         course, know the kinds of attacks that are possible. While the
         specifics differ from protocol to protocol, a number of general
         categories can be constructed.

      Implementation Hints and Pitfalls
         Even if a protocol is sound, a host running it can be
         vulnerable if the protocol is implemented improperly. A
         variety of common errors can and do subvert the best designs.

      Firewall Issues
         Firewalls are both a common defense and a much-reviled wart on
         the Internet. Regardless, they are unlikely to go away any
         time soon. They have both strengths and weaknesses that must
         be considered when deploying them. Furthermore, some protocols
         have characteristics that are unnecessarily firewall-hostile;
         such practices should be avoided.

      Workshop Report
         This document.

8. Working Group Charters

   The actual text in the working group charter is likely to be
   something fairly simple, like

      Protocols developed by this working group will be analyzed for
      potential sources of security breach. Identified threats will be
      removed from the protocol if possible, and documented and guarded
      against in other cases.

   The actual charter text represents a policy enjoined and enforced by
   the IESG, and may change from time to time and from charter to

charter. However, it essentially references and asks for text in
documents conforming to the following, which may be very appropriate
to include in the RFC.

9. Guidelines on writing Security Considerations in an RFC

   A "threat" is, by definition, a vulnerability available to a
   motivated and capable adversary. CERT advisories are quite
   predictable given a knowledge of the target of the threat; they
   therefore represent an existence proof, but not a threat analysis.
   The point is to determine what attacks are possible ("capabilities"
   of a potential attacker) and formulate a defense against the attacks,
   or convincingly argue that the attack is not realistic in some
   environment and restrict use of the protocol to that environment.

   Recommended guidelines:

      All RFCs - MUST meaningfully address security in the protocol or
      procedure it specifies. It MUST consider that it is giving its
      data to "the enemy" and asking it to be delivered to its friends
      and used in the manner it intended. Consideration MUST be given to
      the ramifications of the inherent danger of the situation.

      - MUST do "due diligence" to list the threats to which the
      protocol is vulnerable. Use of legal term does not imply legal
      liability, but rather the level of responsibility expected to be
      applied to the analysis. This discussion might occur throughout
      the document or in the Security Considerations section; if it
      occurs throughout, it MUST be summarized and referenced in the
      Security Considerations section.

      - MUST discuss which of those threats are
         * Ameliorated by protocol mechanisms (example: SYN attack is
         ameliorated by clever code that drops sessions randomly when
         under SYN attack)

         * Ameliorated by reliance on external mechanisms (example: TCP
         data confidentiality provided by IPSEC ESP)

         * Irrelevant ("In most cases, MIBs are not themselves security
         risks; If SNMP Security is operating as intended, the use of a
         MIB to change the configuration of a system is a tool, not a
         threat. For a threat analysis of SNMP Security, see RFC ZZZZ.")

         * Not addressed by the protocol; results in applicability
         statement. ("This protocol should not be used in an
         environment subject to this attack")

10. Core Security Mechanisms

   A variety of security mechanisms exist today. Not all are well-
   designed; not all are suitable for all purposes. The members of the
   workshop designated a number of protocols as "core". Such protocols
   should be used preferentially, if one of them has properties that
   match the needs of your protocol. The following were designated as
   core:

      IPsec [RFC 1825] is the basic host-to-host security mechanism. It
         is appropriate for use any time address-based protection would
         have been used, including with such programs as rsh and rlogin.
         If and when platforms support user-based keying, this scope may
         be expanded.

         One particular technique used by IPsec, HMAC [RFC 2104], is
         more generally useful. If cryptographic authentication but not
         secrecy is needed, and IPsec is not applicable, HMAC should be
         used.

      ISAKMP/Oakley [ISAKMP drafts] is the basic key negotiation
         protocol for IPsec. As such, it should be deployed when IPsec
         is used. With the appropriate "domain of interpretation"
         document, it should be used to negotiate pairwise keys for
         other protocols.

      DNSsec [RFC 2065] is not only crucial for protecting the DNS --
         cache contamination is the easiest way to launch active attacks
         -- it's also needed in many situations when IPsec is used.

      Security/Multipart [RFC 1847] is the preferred way to add secured
         sections to MIME-encapsulated email.

      Signed keys in the DNS. There is, as noted, widespread agreement
         that DNS records themselves must be protected. There was less
         agreement that the key records should be signed themselves,
         making them in effect certificates. Still, this is one
         promising avenue for Internet certificates.

      X.509v3 is the other obvious choice for a certificate
         infrastructure. Again, though, there was no strong consensus
         on this point.

      TLS [TLS draft] was seen by some as the preferred choice for
         transport-layer security, though there was no consensus on this
         point. TLS is less intrusive to the operating system than
         IPsec; additionally, it is easier to provide fine-grained
         protection this way.

Some protocols were designated as "useful but not core". These were
insufficiently general, too new, or were substantially duplicative of
core protocols. These include AFT/SOCKS, RADIUS, firewalls, GSS-API,
PGP, Kerberos, PGP-MIME, PKIX-3, the various forms of per-hop
authentication (OSPF, RSVP, RIPv2), *POP, OTP, S/MIME, SSH, PFKey,
IPsec API, SASL, and CRAM/CHAP. Obviously, entries on this list may
move in either direction.

A few protocols were considered "not useful". Primarily, these are
ones that have failed to catch on, even though they've been available
for some time. These include PEM [RFC 1421, 1422, 1423, 1424] and
MOSS [RFC 1848]. (The phrase "not useful" does not imply that they
are incorrect, or are lacking in important information. However,
they do not describe protocols that we are currently encouraging the
use of.)

One security mechanism was deemed to be unacceptable: plaintext
passwords. That is, no protocol that relies on passwords sent over
unencrypted channels is acceptable.

11. Missing Pieces

   Participants in the workshop identified three significant missing
   pieces: object security, secure email, and route security.

   Object security refers to protection for individual data objects,
   independent of transport. We have one such already -- Secure DNS --
   but we need a me general scheme. MIME is a candidate object
   framework, in part because it is the core of the two most widely used
   and deployed applications: the web and email. However, securing
   email has been problematic and the web is not that far in front.

   Secure email is a critical need and has been for some time. Two
   attempts to standardize secure email protocols (PEM and MOSS) have
   failed to be accepted by the community, while a third protocol (PGP)
   has become a de facto standard around the world. A fourth protocol,
   an industry standard (S/MIME), has been gaining popularity. Both of
   these latter of entered the Internet standards process.

   Route security has recently become a critical need. The
   sophistication of the attackers is on the rise and the availability
   of attacking toolkits has increased the number of sophisticated
   attacks. This task is especially complex because the requirement for
   maximum performance conflicts with the goal of adding security, which
   usurps resources that would otherwise enhance the performance of the
   router.

12. Security Considerations

   Security is not and cannot be a cookie cutter process. There is no
   magic pixie dust that can be sprinkled over a protocol to make it
   secure. Each protocol must be analyzed individually to determine
   what vulnerabilities exist, what risks they may lead to, what
   palliative measures can be taken, and what the residual risks are.

13. Acknowledgments

   This RFC is largely based on the minutes compiled by Thomas Narten,
   whose work in turn was partly based on notes by Erik Huizer, John
   Richardson, and Bob Blakley.

14. References

      [RFC 1825] Atkinson, R., "Security Architecture for the Internet
         Protocol", RFC 1825, August 1995.

      [RFC 2104] Krawcyzk, H., Bellare, M., and R. Canett, "HMAC:
         Keyed-Hashing for Message Authentication", RFC 2104, February
         1997.

      [ISAKMP drafts] Works in Progress.

      [RFC 2065] Eastlake, D., and C. Kaufman, "Domain Name System
         Security Extensions", RFC 2065, January 1997.

      [RFC 1847] Galvin, J., Murphy, S., Crocker, S., and N. Freed,
         "Security Multiparts for MIME: Multipart/Signed and
         Multipart/Encrypted", RFC 1847, October 1995.

      [TLS draft] Dierks, T., and C. Allen, "The TLS Protocol Version
         1.0", Work in Progress.

      [RFC 1421] Linn, J., "Privacy Enhancement for Internet Electronic
         Mail: Part I: Message Encryption and Authentication
         Procedures", RFC 1421, February 1993.

      [RFC 1422] Kent, S., "Privacy Enhancement for Internet Electronic
         Mail: Part II: Certificate-Based Key Management", RFC 1422,
         February 1993.

      [RFC 1423] Balenson, D., "Privacy Enhancement for Internet
         Electronic Mail: Part III: Algorithms, Modes, and Identifiers",
         RFC 1423, February 1993.

      [RFC 1424] Kaliski, B. "Privacy Enhancement for Internet
          Electronic Mail: Part IV: Key Certification and Related
          Services", RFC 1424, February 1993.

      [RFC 1848] Crocker, S., Freed, N., Galvin, J. and S. Murphy, "MIME
          Object Security Services", RFC 1848, October 1995.

Appendix A. Attendees

      Ran Atkinson   rja@inet.org
      Fred Baker   fred@cisco.com
      Steve Bellovin   bellovin@acm.org
      Bob Blakley   blakley@vnet.ibm.com
      Matt Blaze   mab@research.att.com
      Brian Carpenter   brian@hursley.ibm.com
      Jim Ellis   jte@cert.org
      James Galvin   galvin@commerce.net
      Tim Howes   howes@netscape.com
      Erik Huizer   Erik.Huizer@sec.nl
      Charlie Kaufman   charlie_kaufman@iris.com
      Steve Kent   kent@bbn.com
      Paul Krumviede   paul@mci.net
      Marcus Leech   mleech@nortel.ca
      Perry Metzger   perry@piermont.com
      Keith Moore   moore@cs.utk.edu
      Robert Moskowitz   rgm@htt-consult.com
      John Myers   jgm@CMU.EDU
      Thomas Narten   narten@raleigh.ibm.com
      Radia Perlman   radia.perlman@sun.com
      John Richardson   jwr@ibeam.jf.intel.com
      Allyn Romanow   allyn@mci.net
      Jeff Schiller   jis@mit.edu
      Ted T'So   tytso@mit.edu

Appendix B. Author Information

   Steven M. Bellovin
   AT&T Labs Research
   180 Park Avenue
   Florham Park, NJ 07932
   USA

   Phone: (973) 360-8656
   EMail: bellovin@acm.org

Full Copyright Statement