Supporting IPsec with Legacy Credentials

Steven M. Bellovin

`smb@research.att.com`

973-360-8656

AT&T Labs Research

Florham Park, NJ 07932

Rationale

“to define a standard mechanism to accomplish human user authentication to an IPSec device running IKE, using legacy authentication mechanisms.

. . .

“The WG strongly prefers mechanisms that require no changes to AH, ESP or IKE protocols.”

–The IPSRA Charter

My Goals and Non-Goals

- To demonstrate that there are many ways to accomplish our objectives without touching IKE.
- To build on existing tools and protocols.
- To avoid producing a standards-track RFC. (If this RFC is ever advanced, I've failed.)

Approach

- Use SSL/TLS.
- Use existing HTTP and HTML syntax.
- *Perhaps* permit use of Web browsers, with added manual steps or automated plug-ins.

Four Suggestions

- Client-side certificate generation.
- Server-side key pair generation.
- Server-side key storage.
- Server-generated shared secrets.

Client-side Certificate Generation

- Server sends (Netscape-standard) `<KEYGEN>` tag.
- Client generates RSA key pair; uploads public key via SSL/TLS.
- Standard HTTP-style authentication is used.
- Server signs and returns certificate.
- Application (or user) conveys certificate and private key to IKE module.
- Server side does nothing — certificates are self-identifying.

Server-side Key Pair Generation

- Server generates high-quality key pairs in its spare time.
- Client uses HTTP authentication and SSL/TLS to request a certificate.
- No risk here to server retaining private key — the server controls all access no matter what, and this certificate is used for nothing else.

Server-side Key Storage

- The user’s long-term certificate and encrypted private key are stored on the server.
- After HTTP-style authentication, both are returned under protection of TLS.
- Client decrypts and uses private key.
- Can be used with global PKI or locally-generated certificates.

Server-generated Shared Secrets

- SSL/TLS required for earlier schemes is expensive; the result is then discarded, to be followed by an equally-expensive IKE exchange.
- Instead, use the authenticated SSL/TLS session to pass back a transient shared secret.
- Authentication server then passes the secret to the IKE server.
- Permits use of cheaper IKE variants.

Issues

- Designed to permit back-end RADIUS servers, including token card authentication.
- Standard Web browsers are a poor match for such cards — but this isn’t standard HTTP, since the user doesn’t return there.
- Must resolve issue of certificate expiration versus SA expiration, and balance against desire for reuse of legacy authentication technique.
- Clients *MUST* verify server-side TLS certificate.

Conclusions

- There are many ways to solve this problem.
- Existing building blocks are quite sufficient.
- A hybrid of the second and third schemes is a big step towards use of a PKI with client-side certificates.
- We don’t need to touch IKE.