Motivation

- Users continue to use the familiar password login method
- Password is no longer sent in the clear
- Advanced approaches – the server doesn’t store the password or password equivalent data
- Eavesdroppers and impersonators obtain no information for an online attack
Lamport’s Hash
Lamport’s Hash

Simple beginning: one time password scheme using iterated hashes

see http://lodestone.org/people/hoss/ops/node5.html

Alice

Bob

BOB knows \( \langle n, \text{hash}^n(\text{password}) \rangle \)

Compares \( \text{hash}(x) \) to \( \text{hash}^n(\text{password}) \);

If equal, replaces \( \langle \text{hash}^n(\text{password}) \rangle \) with

\( \langle n-1, x \rangle \)

Alice, pwd

Alice

n

x = hash\(^{n-1}\)(pwd)

Alice’s Workstation

Trusted
Attack on Lamport’s Hash

- **Small n attack**
  - Active attacker intercepts servers reply message with n and changes it to a smaller value
  - Attacker can easily manipulate the response (repeatedly) to impersonate Alice

- Eavesdropper captures Alice’s hashed reply and conducts off-line attack

- Replay Alice’s response to other servers where Alice may use the same password
  - Thwart using salt at the server – server hashes pw || salt and sends n and the salt to Alice during login
  - Salt also permits automatic password refresh when n reaches 1
Encrypted Key Exchange (EKE)
Encrypted Key Exchange (EKE)

- Both parties know $W$
- Can an off-line attack be done?

"Alice", $W\{g^a \mod p\}$

$W\{g^b \mod p, C_A\}$

can compute $K_{AB} = g^{ab} \mod p$

$K_{AB}\{C_A, C_B\}$

$K_{AB}\{C_A\}$

Alice

Bob
Augmented EKE

- EKE vulnerable to database disclosure since Bob stores $W$ in the clear.
- Defense: Augmented EKE – Alice knows the password, Bob knows a one-way hash of it.
  - Bob stores: $g^W \mod p$

Diagram:

- Alice sends: $g^a \mod p$.
- Bob responds with: $g^b \mod p$, $H(g^{ab} \mod p, g^{bW} \mod p)$.
- Alice confirms with: $H'(g^{ab} \mod p, g^{bW} \mod p)$. 

“Alice”, $g^a \mod p$
Strong Remote Password (SRP)
n A large prime number. All computations are performed modulo n.
g A primitive root modulo n (often called a generator)
s A random string used as the user's salt
P The user's password
x A private key derived from the password and salt
\nu The host's password verifier
\nu Random scrambling parameter, publicly revealed
a, b Ephemeral private keys, generated randomly and not publicly revealed
A, B Corresponding public keys
H() One-way hash function
m, n The two quantities (strings) m and n concatenated
K Session key

Table 3: Mathematical Notation for SRP
SRP Setup

- Carol picks password P and salt s
- Carol computes the following:
  \[
  x = H(s, P) \\
  v = g^x
  \]
- Steve stores v and s as Carol’s password verifier and salt
- See authentication steps on next slide
<table>
<thead>
<tr>
<th>Carol</th>
<th>Steve</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>C --&gt;</td>
</tr>
<tr>
<td>2.</td>
<td>x = H(s, P)</td>
</tr>
<tr>
<td>3.</td>
<td>A = g^a</td>
</tr>
<tr>
<td>4.</td>
<td>B, u</td>
</tr>
<tr>
<td>5.</td>
<td>S = (B - g^x)^{(a + ux)}</td>
</tr>
<tr>
<td>6.</td>
<td>K = H(S)</td>
</tr>
</tbody>
</table>

Table 4: The Secure Remote Password Protocol
SRP Benefits

- An attacker with neither the user's password nor the host's password file cannot mount a dictionary attack on the password. Mutual authentication is achieved in this scenario.
- An attacker who captures the host's password file cannot directly compromise user-to-host authentication and gain access to the host without an expensive dictionary search.
- An attacker who compromises the host does not obtain the password from a legitimate authentication attempt.
- An attacker who captures the session key cannot use it to mount a dictionary attack on the password.
- An attacker who captures the user's password cannot use it to compromise the session keys of past sessions.
Summary

- Observe how DH idea has been extended to strong password protocols.
- Patents have slowed the adoption of these ideas, but they are starting to expire. We may see a resurgence soon.
- SRP can run without an SSL connection.
- There is a proposed TLS-SRP extension to do an SRP mutual authentication handshake. Potential to deter phishing attacks.