Security for Mobile Peer-to-Peer Systems  (1/2)

**Research Problem**

Part I: Security Bootstrapping

Generation, authentication and distribution of authentic key material is crucial for secure interaction:
- No pre-defined relations: unknown devices, no secure links, no trust relations
- No central service: no administrator, no trusted third party, ...

Part II: Weak Devices

Some devices have limited resources (battery, computing power, communication interface, etc.) such that they cannot co-operate with standard protocols.

**Security Bootstrapping**

We use pairing to build security and trust relations between devices owned by the same user and a security policy to build security relations to devices owned by other users and to assign rights to relations.

Ownership Model
- Transient association to user

Security Policy Definition Language
- Local decisions
- Introduce other devices
- Abstraction to security protocols

Our work also includes support for exception handling (lost devices) and delegation of ownership.
Support for Weak Devices

Some devices have limited resources (battery, computing power, communication interface, etc.) such that they cannot co-operate with standard protocols.

We use the concept of a security proxy to allow capable devices to transparently represent the weak device, and using an adequate protocol to interact with the weak device.

Virtual Services

The security proxy also provides virtual services that are based on the capabilities of the limited device but could not be provided by the limited device itself (e.g., history), or the combination of services provided by different devices.

Example: The thermometer provides current temperature measurements. The security proxy can offer a service that provides a history of measurement values over the last few hours.

Authentication: A Mathematical Analysis

The critical security operation to interact with a weak device is authentication. We investigated the computational complexity of several authentication protocols, and implemented one of them on a mobile phone.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Messages</th>
<th>Cycles A</th>
<th>Cycles B</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSH (Diffie-Hellmann + pwd)</td>
<td>7</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>Fiat-Shamir (1 round)</td>
<td>3</td>
<td>39</td>
<td>25</td>
</tr>
<tr>
<td>Kerberos</td>
<td>6</td>
<td>36</td>
<td>45</td>
</tr>
<tr>
<td>Modified Needham-Schroeder</td>
<td>3</td>
<td>21</td>
<td>21</td>
</tr>
</tbody>
</table>

Implementation of the Fiat-Shamir protocol on a Nokia 6310 in J2ME:

\[
\begin{align*}
A &= s^2 \mod n, n \\
B &= v, n \\
r &= e \in_R [1, n] \\
x &= r^2 \mod n \\
y &= rs^e \mod n \\
\end{align*}
\]

Conclusion: Startup and communication latency dominate the execution time. The calculations only take a few milliseconds.