Cross-realm issues could be a charter item?

Shoichi Sakane
Shouichi.Sakane@jp.yokogawa.com
The 68th IETF meeting
Purpose of this presentation

• Introduction of current activities

• (Introduction of some of current approaches.)

• (Discussion about problems.)

• Approval of adding it into the charter
Current activities

- 4 documents AFAIK
  
  1. draft-sakane-krb-cross-problem-statement-01.txt
  
  2. draft-kamada-krb-client-friendly-cross-01.txt
  
  3. draft-ietf-cat-kerberos-pk-cross-08.txt
  
  4. draft-zrelli-krb-xtgsp01.txt
Problem statement

- draft-sakane-krb-cross-problem-statement-01.txt
- Introduced an actual environment.
- Listed requirements and constraints.
- Specified issues if krb is employed.
Issues that are defined

1. Client's Performance
2. Unreliability of authentication
3. No PFS
4. Scalability of the direct trust model
5. Exposure to DoS attacks
6. Applicability to roaming scenario
Approaches

• Client friendly model
  – Draft-kamada-krb-client-friendly-cross-01.txt
  – Proposed a model with two modes.

• XTGSP
  – Draft-zrelli-krb-xtgsp-01.txt
  – Proposed a solution with new extensions.

• PKCROSS
  – Draft-ietf-cat-kerberos-pk-cross-08.txt (expired)
  – Proposed a protocol to establish inter-realm key.
Steps to go

1. Finding out issues
2. Defining problems
3. Adding it into the charter
4. (Repeating 1 and 2)
5. Evaluating approaches
6. Proposing protocols
Question

• Could this item be a charter item?

• Do we need more discussion before adding it?

• Could the problem statement be a working group document?

• [off topic] Are you interested in the approaches?
End of presentation
Reliability of chain

When an intermediary KDC downs, the authentication will fail.
No PFS in indirect trust model

Intermediary KDCs can learn session keys.

Scalability of direct trust model

- When realms make a direct authentication path, they need to maintain each inter-realm key.
Exposure to DoS attack

Not easy to set up filters to protect KDC.
– KDC handles TGS exchanges with remote clients from different realms.
Client's performance

Client centralized exchanges causes unacceptable delay.

– Client must perform TGS exchange with each KDC of the trust path.

☆ Not scalable if number of realms increases especially for small/embedded devices.
## Processing time of Kerberos on embedded devices

*measured by Yokogawa Electric Corporation 04 through 06*

<table>
<thead>
<tr>
<th>CPU</th>
<th>DS5250 (8051 arch., 8-bit, 22MHz, w/DES H/W)</th>
<th>H8 (16-bit, 20MHz) + Crypt H/W (AES, 3DES, SHA1, MD5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Krb lib</td>
<td>MIT-1.2.4</td>
<td>MIT-1.2.4</td>
</tr>
<tr>
<td>Crypt H/W</td>
<td>Enable</td>
<td>Enable</td>
</tr>
<tr>
<td>TGT</td>
<td>4650ms</td>
<td>74ms</td>
</tr>
<tr>
<td>TGS</td>
<td>4579ms</td>
<td>195ms</td>
</tr>
</tbody>
</table>

In the table above, the times refer to:

- **Including waiting time**
- **Excluding waiting time**
Applicability to roaming scenario

Roaming users can not access to home KDC from the visited realm due to chicken-and-egg problem.

– Maybe due to the policy of the realms.

![Diagram showing the difference between home and visited realms regarding KDC access.](image)