Salted Challenge Response (SCRAM) HTTP Authentication Mechanism

draft-melnikov-httpbis-scram-auth-00

Alexey Melnikov <alexey.melnikov@isode.com>
• Based on SASL SCRAM (RFC 5802), which is used by XMPP, email protocols (IMAP/SMTP), LDAP, etc.
• Provides functionality comparable to HTTP Digest, but
  – simpler to implement (based on experience of implementing HTTP Digest),
  – uses more modern crypto,
  – addresses some defects in HTTP Digest design (e.g. protects the whole authentication exchange, username change doesn't require secret update)
  – speed of authentication (and thus resistance to offline brute force attacks) can be controlled by changing the per user iteration count.
• (see RFC 6331 for more details)
realm, username, client nonce,
<support for channel bindings>
1
---------------------------------->
realm, session id, user salt,
client nonce + server nonce,
user iteration counter
2
----------------------------------

session id, <channel bindings>,
client nonce + server nonce,
client proof
3
---------------------------------->

session id, server verifier
4
----------------------------------

7-Nov-2012
Protocol Features

- Mutual authentication is supported, but only the client is named (i.e., the server has no name).
- Server can store salted SCRAM hash or cleartext password
- The salting prevents a pre-stored dictionary attack if the database is stolen.
### Open Issue

<table>
<thead>
<tr>
<th>Client</th>
<th>Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>realm, username, &lt;channel bindings&gt;</td>
<td></td>
</tr>
<tr>
<td>client nonce + server nonce,</td>
<td></td>
</tr>
<tr>
<td>client proof</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>session id, server verifier</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Q: Can the number of round trips be reduced for reauthentication (similar to HTTP Digest reauthentication).

A: Yes, but the draft doesn't currently show how.