Providing for Multiple-Proxy Authentication of a SIP Request

<draft-sparks-sip-multiproxy-auth-00>

Robert Sparks - MCI WorldCom

Motivation

- Service providers or other administrative authorities will want to protect SIP resources using an authenticating proxy
- Behavior under the current SIP specification is not clear when a request traverses more than one such proxy

Request Fails

```
UAC         Proxy1      Proxy2
 |  request() |           |
 |----------->|           |
 |  407 Proxy-Authenticate (challenge1)
 |<-----------|           |
 |  request(challenge1,credentials1)
 |----------->|           |
 |            | request() |  ( Proxy1 strips the
 |            |---------->|    Proxy-Authorization
 |            |           |    header )
 |           407 Proxy-Authenticate (challenge2)
 |            |<----------|
 |  407 Proxy-Authenticate (challenge2)
 |<-----------|           |
 |  request(challenge2,credentials2)
 |----------->|           |
 |  407 Proxy-Authenticate (challenge3)
 |<-----------|           |
 |            |           |
 |            |           |
```

Proposal

- UACs remember and respond to all proxy challenges received in a given call leg
- Challenging proxies search through all responses for any meant for them
- Proxies forward all response material not meant for them

UACs remember challenges

- For the duration of a call-leg (To:,From:,Call-ID:), a UAC will retain any proxy challenge material received and include a response to each challenge in a separate Proxy-Authorization header in each subsequent request in that call-leg. While retaining challenge material, a UAC must be sensitive to the realm of the request, so that stale challenges are replaced with their updates.

Proxies search for responses

- Any proxy requiring authentication that receives a request with multiple Proxy-Authorization headers will search for headers with   challenge parameters matching those it requested. If no such header is found, the proxy will reply with a challenge. If exactly one such header is found, the proxy will verify the credentials and forward the message or issue a challenge/failure. If more than one such header is found, the proxy will reply with a 403 Forbidden (to discourage hunting for valid credentials).

Proxies forward other responses

- A proxy not requiring authentication or a proxy whose challenge has been satisfied will forward all other Proxy-Authorization headers downstream unaltered. A proxy MAY remove the Proxy-Authorization header that was meant for it.

Request Succeeds

```
          UAC         Proxy1      Proxy2          UAS
          |  request() |           |             |
          |----------->|           |             |
          |  407 Proxy-Authenticate (challenge1) |
          |<-----------|           |             |
          |  request(challenge1,credentials1)    |
          |----------->|           |             |
          |            | request() |             |
          |            |---------->|             |
          |            |  407 Proxy-Authenticate (challenge2)
          |            |<----------|             |
          |  407 Proxy-Authenticate (challenge2) |
          |<-----------|           |             |
          |
request(challenge1,credentials1,challenge2,credentials2)
          |----------->|           |             |
          |            |  request(challenge2,credentials2)
          |            |---------->|             |
          |            |           |   request() |
          |            |           |------------>|
```

Notes

- A UAC should be prepared to terminate the deadlock situation caused by a proxy in the chain that expires a challenge after its first successful response.
- Proxies implementing this proposal must accept a valid response to a challenge more than once within the context of a given call-leg.
- Proxies accepting different credential sets must take care to issue unique realm strings.

Discussion Points / Continuing Work

- Parsers must handle multiple field values in one header

  Proxy-Authenticate: DIGEST realm=“a”,nonce=“a”, DIGEST realm=“b”,nonce=“b”

  vs

  Proxy-Authenticate: DIGEST realm=“a”,nonce=“a”

  Proxy-Authenticate: DIGEST realm=“b”,nonce=“b”

- Proxies can challenge with a multiple-choice scheme set.