3GPP TSG SA WG3 Security#31 **S3-030744**
18-21 November, 2003
Munich, Germany

Agenda Item: **6.9 (GAA) and 6.18 (Presence)**

Source: **Siemens**

Title: **Role of Authentication Proxy (AP-NAF) – Discussion and Pseudo-CRs to TSs on GAA/HTTPS and Presence Security**

Document for: **Discussion and decision**

Abstract

The report on SA3#30 states in the context of the discussion on S3-030551: "An agreed definition of the term "reverse http authentication proxy" is needed." This contribution gives definitions of "forward" and "reverse" proxies and applies their properties to the Authentication Proxy (AP-NAF) environment. Pseudo-CRs to TS ab. cde (GAA/HTTPS) and TS 33.141 (presence security) are proposed.

1. Forward and Reverse Proxy

1.1 Definition of Forward and Reverse Proxy

The following definition is taken from Apache Server Documentation (Apache module mod_proxy):

"A *forward proxy* is an intermediate system that enables a browser to connect to a remote network to which it normally does not have access. A forward proxy can also be used to cache data, reducing load on the networks between the forward proxy and the remote webserver.

A *reverse proxy* is a webserver system that is capable of serving webpages sourced from other webservers - in addition to webpages on disk or generated dynamically by CGI - making these pages look like they originated at the reverse proxy. When configured with the mod_cache module the reverse proxy can act as a cache for slower backend webservers. The reverse proxy can also enable advanced URL strategies and management techniques, allowing webpages served using different webserver systems or architectures to coexist inside the same URL space. Reverse proxy systems are also ideal for implementing centralised logging websites with many or diverse website backends. Complex multi-tier webserver systems can be constructed using an mod_proxy frontend and any number of backend webservers."

1.2 Relevant Properties

The following gives relevant properties of forward and reverse proxies concerning necessary configuration of other network elements and with respect to usage with HTTPS:

Forward proxy:

- *Configuration*: A forward proxy is configured hop-by-hop, i.e. it has to be addressed explicitly by the next element further down in the chain towards the client (other forward proxy or client itself). This is done by local configuration of the proxy one further down (use specific proxy for domain name in request URL).
  In case the DNS name of the forward proxy changes, the configuration entry of the next element down the chain has to be adjusted (or the whole chain, if there is also a change in the whole access path through Internet/Intranet).
  In case only the IP address changes the DNS entry of the forward proxy has to be changed.
- *HTTPS*. A forward proxy is always transparent for SSL/TLS. It cannot terminate a SSL/TLS connection. Termination of SSL/TLS connection is somewhere further up in the chain.

As a result a forward proxy used with TLS cannot issue 401 Unauthorized or 407 Proxy Authentication Required and cannot read or insert WWW-Authenticate, Proxy-Authenticate, Proxy-Authorization, or Authorization header fields.
(Note: A forward proxy may react in an “unusual” way to a CONNECT request, which asks for a transparent tunnel for HTTPS to the next hop. It may answer the CONNECT like an ordinary forward proxy (according to the definition above), but then, instead of establishing the tunnel and switching to transparent pass-through, it answers itself to the following TLS handshake with the element sending the CONNECT request. In this manner it can terminate the SSL/TLS connection and get access to the authorization header fields. The client is not aware of the unusual behaviour of the forward proxy, hence has to be configured as for an ordinary forward proxy and has to use the CONNECT method.)

Reverse proxy:

- *Configuration*: A reverse proxy is never explicitly seen by a forward proxy further down the chain or by the client itself. It always resembles the server itself, given in the request URL. The DNS entry contains the IP address of the reverse proxy. There is no proxy specific configuration in elements further down the chain. (In case of changes regarding the application server the standard procedures apply: on name change → new URL in client, on IP address change → change DNS entry).
- *HTTPS*: A reverse proxy always terminates a SSL/TLS connection, i.e. a TLS connection cannot transparently extend across a reverse proxy.
  As a result it has full control over the connection including authentication header fields.

1.3 Evaluation

From the above properties one can conclude in a straightforward manner that an authentication proxy in the sense of S3-030371 by Ericsson and the new TS GAA/HTTPS (S3-030665) should be a reverse proxy for the following reasons:

- As forward proxy the NAF would not have access to authentication headers in HTTPS connections, and therefore cannot act as authentication proxy.
- The “unusual” forward proxy mentioned above would give the advantage that it can terminate TLS. But this unusual behaviour of a forward proxy does not seem to be compatible with current proxy implementations, and the problem stated in the next bullet still remains.
- For both HTTP and HTTPS connections, (re-)configuration of additional, intermediate forward proxies, e.g. forward proxies typically used in a corporate network, down the chain poses a hard problem, as these proxies may not be known to either NAF operator or the user.
- The main argument against the use of forward proxies (ordinary or unusual), however, is the need to manage proxy configurations in the UE. A solution which minimizes the need to UE configuration is to be preferred. Otherwise, it needs to be specified that UEs need to support easy management of proxy configurations.

2. Proposal (Pseudo-CRs)

It is proposed to include the following requirement in section 5.1 (Use of authentication proxy / requirements and principles) of TS ab.cde v011 (GAA/HTTPS, S3-030665) and in section 5.4.1 of TS 33.141 v020 (Presence Security):

“ The use of an authentication proxy should be such that there is no need to manage the authentication proxy configuration in the UE.

Note*: This requirement implies that the authentication proxy should be a reverse proxy in the following sense: A reverse proxy is a web server system that is capable of serving web pages sourced from other web servers - in addition to web pages on disk or generated dynamically by CGI - making these pages look like they originated at the reverse proxy.”*

It is proposed to include this requirement in both draft TSs, as it is currently unclear whether both TSs will be completed within the Release 6 timeframe. If both TSs are completed in time, SA3 may decide later that all material on authentication proxies is to be contained in TS ab.cde (GAA/HTTPS), and that TS 33.141 (Presence Security) only is to make reference to TS ab.cde (GAA/HTTPS).