3GPP TSG SA WG3#30 **S3-030576**
6-10 Okt 2003
Povoa de Varzim, Porugal

Agenda Item: **GBA**

Source: **Alcatel**

Title: **Comparison of different solutions for GBA and AP based AS: standard TLS versus shared secret based TLS**

Document for: **Discussion**

1. Introduction

At the SA3 ad hoc meeting in september in Antwerp progress was made related to the specification of a generic bootstrapping architecture(GBA) to be used by applications to acquire a shared secret key between an application server and a UE and perform authentication. The shared secret key is bootstrapped from the security association shared between USIM and HSS, as part of a 3G subscription. Applications that are likely to use this GBA include distribution of subscriber certificates, Presence, MBMS and perhaps WLAN/3G interworking. It may also be useful to apply the GBA to features defined outside 3GPP (e.g. by OMA).

It was also agreed that there will likely be situations were, if available, the use of public keys and digital certificates is more appropriate for authentication between UE and application server. However it was agreed that if applicable or desirable, the use of public keys and digital certificates should be described in the appropriate application TS and will not be part of the GBA.

During the ad hoc meeting Siemens proposed in Tdoc S3z030011 an architecture that is a compromise between solutions that were in previous meetings proposed by Nokia and Ericsson (e.g. S3-030371, S3-030359) to secure the Ut interface under the Presence WI and under the work item Support for Subscriber Certifictaes (SSC) (S3-030317, S3-030397). It was agreed to continue with the architecture presented in S3z030011 for GBA. This architecture had still some open issues among others related to optimizing the flows in case an Application Proxy (AP) is used between UE and Application Server (AS). This contribution compares some of the alternative approaches for ASs that use and AP.

2. Archtiecture overview related to support for use of a reverse http proxy

It was agreed that the use of a reverse http proxy has imprtant advantages and should be supported by the GBA. However there were different options of how to incorporate the Application Proxy (AP) in the GBA architecture as discussed in this section..

2.1 Separate BSF and AP and standard TLS

A first option is shown in Figure 1 (taken from [2]) where BSF and AP are a priori separate network elements. When the UE wants to access one of the application servers, which are attached to the AP, on the right hand side of the figure, then the sequence of events is as follows (overview):

1) the UE starts http digest aka (rfc3310, protocol A) with the BSF. The BSF may contact the HSS to fetch authentication vectors (protocol C). After step 1), the UE and the BSF share a secret key, cf. TS SSC, section 4.3.1.

2) The UE sends an http request towards an application server. The http request is intercepted by the http authentication proxy (AP). The UE establishes a TLS tunnel with the AP. The AP authenticates to the UE by means of a server certificate.

3) The UE authenticates to the AP by means of HTTP Digest and a shared secret that results from protocol A (step 1)). In this process, the AP fetches the agreed key from the BSF (protocol D), as described in TS SSC, section 4.3.2.

4) The UE runs the application protocol with the application server through the AP.

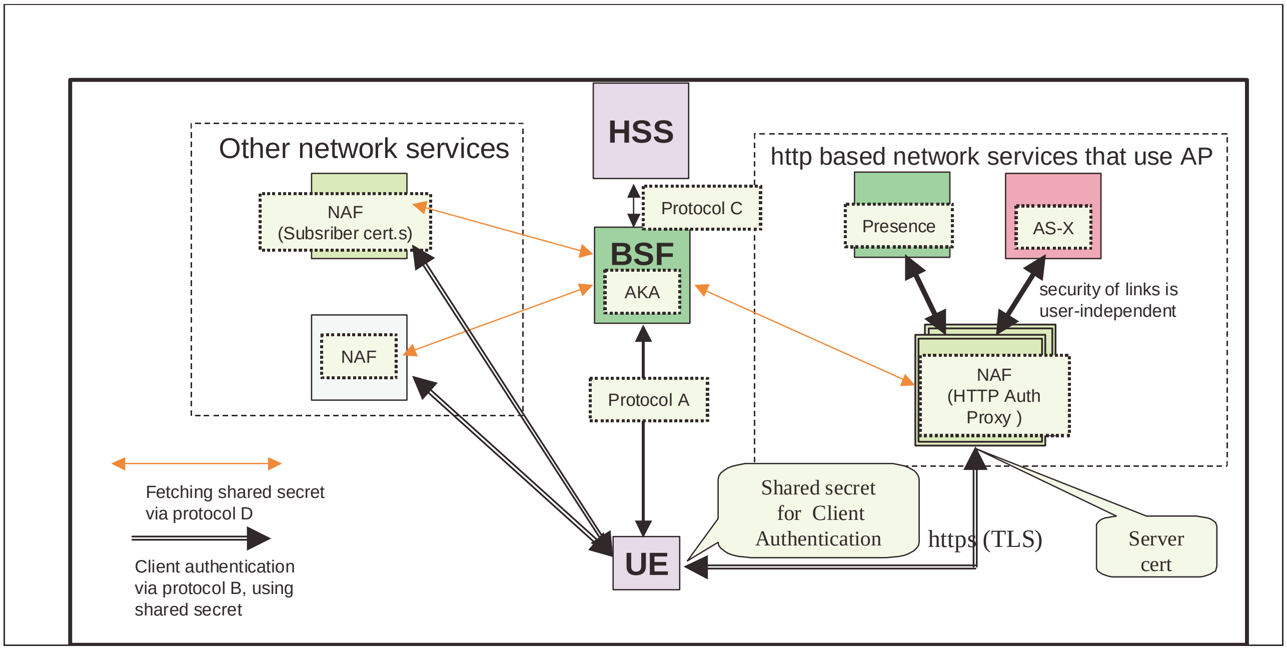

Figure 1 GBA overview with separate BSF and AP

This sequence of messages is rather heavy since http digest is run twice (once with BSF and once with AP inside TLS tunnel).

2.2 AP and BSF functionality merged and standard TLS

An alternative sequence of messages was proposed in [2] to optimise the authentication phase in the case where the BSF and the AP are merged in one network element. Merging BSF and AP has the additional advantage that the interface between them over which protocol D runs to fetch the shared secret must not be specified for the AP case. The architecture with merged BSF and AP is shown in Figure 2.

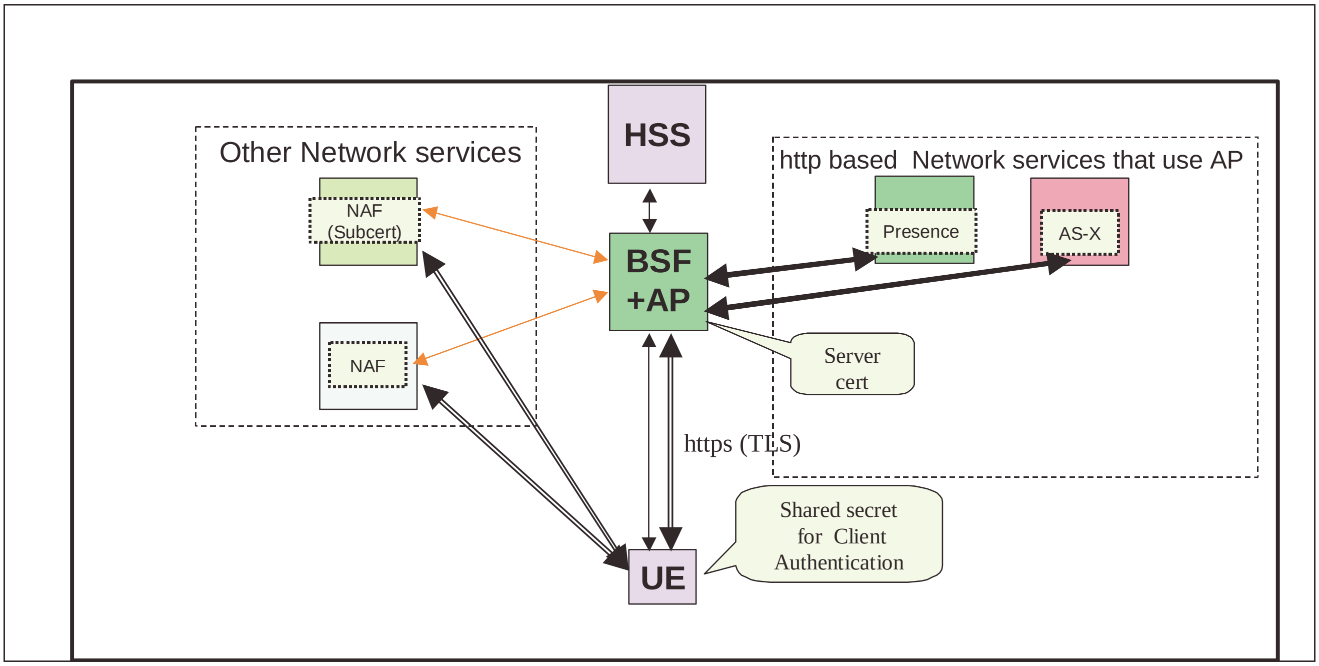

Figure 2 GBA overview with merged BSF and AP for protocol optimisation

When the UE wants to access one of the http-based application servers connected to the AP the sequence of events can be as follows (using standard TLS):

1) The UE sends an http request towards an application server. The http request is intercepted by the http authentication proxy (AP). The UE establishes a TLS tunnel with the AP. The AP is authenticated to the UE by means of a server certificate.

2) the UE starts http digest akav2 with the AP. Some binding is required to prevent MitM attacks. The BSF may contact the HSS to fetch authentication vectors (Cx-like protocol). After step 2), the UE and BSF/AP are mutually authenticated, share a secret key (from AKA) and have a TLS tunnel installed.

3) The UE runs the application protocol with application server through the AP.

2.3 Separate BSF and AP and shared secret based TLS

There has been discussion over the mailing list that pointed SA3's attention to an IETF draft (see [1]) that proposes to allow peer authentication based on shared secrets in TLS rather than only supporting authentication based on public keys. This would allow for an efficient and yet uniform mechanism for both general NAFs and NAFs based on HTTP and using APs (left and right sight in the figures).

When using TLS with shared secret authentication as descibed in [1] is used, the sequence of events can be as follows when the UE wants to access one of the http-based application servers connected to the AP:

1) the UE starts http digest aka (rfc3310, protocol A) with the BSF. The BSF may contact the HSS to fetch authentication vectors (protocol C). After step 1), the UE and the BSF share a secret key, cf. TS SSC, section 4.3.1.

2) The UE sends an http request towards an application server. The http request is intercepted by the http authentication proxy (AP). The UE establishes a TLS tunnel with the AP. UE and AP mutually authenticate in TLS by means of the shared secret that results from protocol A (step 1)). In this

process, the AP fetches the agreed key from the BSF (protocol D), as described in TS SSC, section 4.3.2.

3) The UE runs the application protocol with the application server through the AP.

3. Comparison

In this section we compare the third solution that uses TLS with shared secrets with the two previous approaches where standard TLS is used.

Advantages of TLS with shared secret

- Efficiency. The extra run of http digest inside the TLS tunnel of section 2.1 is avoided.
- Uniform structure. Run of protocol A between UE and BSF is independent of the type of NAF and whether or not the NAF uses an AP.
- Related to the previous point, the UE does not need a mechanism to decide whether it needs to start with the set up of a TLS tunnel or with HTTP digest AKA towards the BSF (or BSF/AP)
- MitM attack problem is solved in an elegant way by integrating the peer authentication in the TLS set-up procedure. Neither modification to http digest akav2 is required nor is there a need for implementation of both akav1 and akav2.
- No data traffic needs to pass through BSF.

Disadvantages of TLS with shared secret

- Status of [1]. The document that proposes to support shared secret based authentication in TLS is an IETF draft that is still in an early stage. It needs to be investigated in what time frame this could proceed to RFC if it is to be adopted in GBA
- Adaptation of TLS implementation required. According to the authors of [1] support for shared secret based authentication would imply very few modifications, but some adaptations will anyhow be needed.

Remark: the authentication between UE and NAF/AP is in this case not base on http digest which could be argued to decrease uniformity. However during th Antwerp ad hoc meeting it was argued that SA3 does not want to impose a requriement for http support on the NAF. This was one of the main arguments against a proposal of Ericsson to use an AAA server as BSF and drop the direct interface between UE and BSF (protocol A) (Tdoc S3z030024). Hence, if http support is not mandated for the NAF then http digest cannot be the only authentication option for protocol B either

4. Conclusions and open issues

Alcatel suggests that for the moment BSF and AP are a priori considered to be different network elements. The status of the IETF draft [1] that specifies TLS with peer authentication based on shared secret should be investigated in more detail and it needs to be checked whether this IETF draft can proceed to standard RFC within the release 6 timeframe. If that is the case then this should be the preferred solution for UE authentication towards the AP as it allows for the most uniform appraoch with nevertheless an optimized message flow.

5. References

[1] “Use of Shared Keys in the TLS Protocol”, P. Gutmann, http://www.ietf.org/internet-drafts/draft-ietf-tls-sharedkeys-01.txt

[2] S3z030011 “Generic Authentication Architecture evaluation”, document presented by Siemens during the september ad hoc meeting in Antwerp