3GPP TSG SA WG3 Security — S3#22 **S3-020101**

25 - 28 February 2002

Bristol, UK

Source: **Alcatel**

Title: **Authentication Scheme Negotiation in OSA**

Document for: **Adoption**

Agenda item: **T.b.d.**

1 Introduction

This contribution discusses the mechanism defined in TS 29.198-3 v4.2.0 to negotiate the authentication scheme used between the client application and the framework/services. A new mechanism is proposed in this contribution to really implement negotiation of authentication mechanisms between the client and the framework/service.

2 Current mechanism

As per TS 29.198-3, the negotiation of the authentication mechanism is achieved with the initiateAuthenticate() method, which enables the client to indicate which (single) authentication scheme it is willing to use. Currently, two methods have been defined: P_OSA_AUTHENTICATION indicates the use of CHAP (challenge-based authentication with MD5) and P_AUTHENTICATION indicates use of an underlying mechanism (eg CORBA). Other authentication schemes can be defined by service providers and be identified with prefix "SP_".

New authentication schemes under the SP_ prefix are therefore reserved for service providers and would therefore not appear in the standard. Two different service providers may also well assign their own (different) names to the same authentication scheme. This limits the extensibility of the whole mechanism.

In addition, the current mechanism does not enable negotiation of the authentication scheme, since the client indicates a single chosen scheme as a parameter to the initiateAuthentication() method. This limits the scalability of the whole mechanism.

The current specification does not either enable to negotiate the signing algorithm to be used with the terminateAccess() function. A separate contribution discusses this issue further but proposes no solution. We are here proposing a solution in the context of the initial negotiation mechanism.

3 New negotiation mechanism

Several alternative solutions can be designed to solve the above issues.

1. The P_OSA_AUTHENTICATION method can be extended to apply to any authentication method defined in OSA, not only CHAP_with_MD5. A new method, selectAuthenticationMethod(), is defined that enables to negotiate which mechanism to use (CHAP_with_MD5, CHAP_with_HMAC_SHA1, digital signature schemes, …). This new method is then used after initiateAuthentication(). With this solution, the selectAuthenticationMethod() function can also be used to negotiate, as a second parameter, the signing algorithm for the terminateAccess().

2. The authType parameter of the initiateAuthentication() method is modified to carry a list of proposed authentication schemes. The return result must then also contain the scheme chosen by the framework. New authentication types are then defined in table TpAuthType to cover other authenticaton schemes such as digital signature-based schemes, use of HMAC with MD5 or SHA1 in CHAP, … With this solution, the signing algorithm for the terminateAccess() function cannot be negotiated except if the authentication scheme negotiated is always a digital signature scheme, which would then also apply to the terminateAccess() function. To be able to do so, the authType parameter must be made compound to contain two lists of proposals: one for initial authentication and one for the signing algorithm of the terminateAccess() function.