3GPP TSG CN Meeting #27 **NP-050130**
9th - 11th March 2005. Tokyo, Japan.

Source: **TSG CN WG1**

Title: **CR to Rel-6 WI "SEC1-SC" for TS 24.109**

Agenda item: **9.3**

Document for: **APPROVAL**

The CR is revised due to changes to cover page (linked CR to SA3 approved change).Original N1 document was N1-050358 (NP-05081).

This document contains **CR on Rel-6 Work Item "SEC1-SC"**, that has been agreed by TSG CN WG1 CN#37 meeting and forwarded to TSG CN Plenary meeting #27 for approval.

CR-Form-v7.1

CHANGE REQUEST

⌘ **24.109** **CR** **12** ⌘ **rev** **2** ⌘ Current version: **6.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☒ Radio Access Network ☐ Core Network ☒

Title:	⌘	PSK TLS updates
Source:	⌘	Nokia
Work item code:	⌘	SEC1-SC
Date:	⌘	16/02/2005
Category:	⌘	**F**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
Ph2 *(GSM Phase 2)*
R96 *(Release 1996)*
R97 *(Release 1997)*
R98 *(Release 1998)*
R99 *(Release 1999)*
Rel-4 *(Release 4)*
Rel-5 *(Release 5)*
Rel-6 *(Release 6)*
Rel-7 *(Release 7)*

Reason for change:	⌘	PSK TLS is kept in Release-6 and minor updates are needed to reflect the changes in the Internet Draft of PSK TLS.
Summary of change:	⌘	- editor's notes are removed. - ServerKeyExchange was changed to optional in PSK TLS ID; this is reflected in section 5.3.3.1
Consequences if not approved:	⌘	Specification is not aligned with changes in PSK TLS internet draft and (conditional, see other comments) with the decision of SA3 to keep PSK TLS in Rel-6.

Clauses affected:	⌘	2, 5.3.3, Annex F

		Y	**N**		
Other specs affected:	⌘	**X**		Other core specifications	⌘ TS 33.222 (CR15)
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘	Conditional approval by CN1: the decision whether PSK TLS is kept in Rel-6 is made in SA3 meeting (next week).			

===== **BEGIN CHANGE** =====

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document.*

[1] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic bootstrapping architecture".

[2] 3GPP TR 33.919: "Generic Authentication Architecture (GAA); System description".

[3] 3GPP TS 29.109: "Generic Authentication Architecture (GAA); Zh and Zn Interfaces based on the Diameter protocol; Protocol details".

[4] 3GPP TS 33.221: "Generic Authentication Architecture (GAA); Support for Subscriber Certificates".

[5] 3GPP TS 33.222: "Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)".

[6] IETF RFC 3310: "Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)".

[7] 3GPP TS 23.003: "Numbering, addressing and identification".

[8] IETF RFC 3023: "XML Media Types".

[9] IETF RFC 2617: "HTTP Authentication: Basic and Digest Access Authentication".

[10] IETF RFC 3548: "The Base16, Base32, and Base64 Data Encodings".

[11] IETF RFC 2246: "The TLS Protocol Version 1.0".

[12] IETF RFC 2818: "HTTP over TLS".

[13] 3GPP TS 24.228: "Signalling flows for the IP multimedia call control based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3".

[14] IETF RFC 2616: "Hypertext Transfer Protocol -- HTTP/1.1".

[15] IETF draft-ietf-tls-psk-0~~1~~5: "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)".

[16] PKCS#10 v1.7: "Certification Request Syntax Standard".

NOTE: ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-10/pkcs-10v1_7.pdf

[17] WAP Forum: "WPKI: Wireless Application Protocol; Public Key Infrastructure Definition"

NOTE: http://www1.wapforum.org/tech/documents/WAP-217-WPKI-20010424-a.pdf.

[18] IETF RFC 3546: "Transport Layer Security (TLS) Extensions".

[19] Open Mobile Alliance: "ECMAScript Crypto Object"

NOTE: http://www.openmobilealliance.org.

[20] Open Mobile Alliance: "WPKI"

NOTE: http://member.openmobilealliance.org/ftp/public_documents/SEC/Permanent_documents/.

[21] 3GPP TS 33.203: "3G security; Access security for IP-based services".

[22] IETF RFC 2234: "Augmented BNF for Syntax Specifications: ABNF".

===== **BEGIN NEXT CHANGE** =====

5.3.3 Shared key-based mutual authentication between UE and NAF

~~Editor's note: If the "Pre-Shared Key Ciphersuites for TLS" Internet Draft [15] does not reach the RFC status by the time when Release 6 is frozen, this clause must be removed and the support for the Pre-Shared Key TLS is postponed to Release 7.~~

5.3.3.1 Authentication procedure

The authentication mechanism described in this clause is optional to implement in the UE and in the NAF.

The Pre-Shared Key Ciphersuites for TLS (PSK TLS) (draft-ietf-tls-psk-01 [15]) can be used with bootstrapped security association as the authentication, confidentiality, and integrity protection method.

The PSK TLS (draft-ietf-tls-psk-01 [15]) handshake shall be used with bootstrapped security association as follows:

- the ClientHello message shall contain one or more PSK-based ciphersuites;
- the ClientHello message shall contain the server_name TLS extension as specified in RFC 3546 [18] and it shall contain the hostname of the NAF;
- the ServerHello message shall contain a PSK-based ciphersuite selected by the NAF;
- the ServerKeyExchange shall be sent by the server and it shall contain the psk_identity_hint field and it shall contain a static string "3GPP-bootstrapping". The psk_identity_hint field may contain a list of psk_identity_hints (see NOTE 1);

NOTE 1: Other psk identity name spaces than "3GPP-bootstrapping" can be supported, however, they are out of the scope of this specification.;

- the ClientKeyExchange shall contain the psk_identity field and it shall contain a prefix "3GPP-bootstrapping"indicating selected psk identity name space, a separator character ";" and the B-TID;
- the UE shall derive the TLS premaster secret from the NAF specific key material (Ks_NAF) in the case of GBA_ME, and the NAF specific external key material (Ks_ext_NAF) in the case GBA_U as specified in draft-ietf-tls-psk-01 [15];

NOTE 2: The NAF specific internal key material (Ks_int_NAF) in the case of GBA_U shall not be used with PSK TLS.

An example flow of the PSK TLS procedure can be found in clause F.3.

5.3.3.2 Authentication failures

Authentication failures are handled as they are described in RFC 2246 [11] and in draft-ietf-tls-psk-01 [15].

5.3.3.3 Bootstrapping required indication

During TLS handshake, the NAF shall indicate to the UE that bootstrapped security association is required by sending a ServerHello message containing a PSK-based ciphersuite, and a ServerKeyExchange message containing the psk_identity_hint field which contains a static string "3GPP-bootstrapping". This shall trigger the UE to run the bootstrapping procedure over Ub interface.

NOTE: The NAF shall select a PSK-based ciphersuite only if the UE has offered one or more PSK-based ciphersuites in the corresponding ClientHello message.

5.3.3.4 Bootstrapping renegotiation indication

During usage of TLS session, the NAF shall indicate to the UE that bootstrapped security association has expired by sending close_notify alert message to the UE. The UE may attempt resume the old TLS session by sending a ClientHello message containing the old session ID. The NAF shall refuse to use the old session ID by sending a ServerHello message with a new session ID. This will indicate to the UE that the bootstrapped security association it used has expired.

During TLS handshake, the NAF shall indicate to the UE that the bootstrapped security association has expired by sending handshake_failure message as a response to the Finished message sent by the UE. This will indicate to the UE that the bootstrapped security association it used has expired.

5.3.4 Certificate based mutual authentication between UE and application server

The authentication mechanism described in this clause is optional to implement in the UE and in the application server.

The certificate based mutual authentication between an UE and an application server shall be based on TLS as specified in IETF RFC 2246 [6] and IETF RFC 3546 [8].

Annex B in TS 33.222 [5] provides guidance on certificate mutual authentication between UE and application server.

5.3.5 Integrity protection

Integrity protection is provided by using authenticated TLS tunnel as described in RFC 2818 [12].

===== **BEGIN NEXT CHANGE** =====

Annex F (informative): Signalling flows for PSK TLS with bootstrapped security association

~~Editor's note: If the "Pre-Shared Key Ciphersuites for TLS" Internet Draft [15] does not reach the RFC status by the time when Release 6 is frozen, this annex shall be removed and the support for the Pre-Shared Key TLS is postponed to Release 7.~~

F.1 Scope of signalling flows

This annex gives examples of signalling flows for using PSK TLS with bootstrapped security association.

F.2 Introduction

F.2.1 General

A bootstrapping session established using a bootstrapping procedure (cf., clause 4 and annex A) is used between a UE and a NAF. The BSF provides to the NAF a NAF specific key material (Ks_NAF or Ks_ext_NAF) which is derived from the key material (Ks or Ks_ext). The NAF uses this key to authenticate and optionally secure (i.e. integrity protect and encrypt) the communications between it and the UE. The BSF will also provide the NAF the expiration time of the bootstrapping session. When the bootstrapping session becomes invalid the NAF will stop using the session, and indicate to the UE that bootstrapping session has expired and that new session needs to be established.

An example of the signalling flows of the authentication procedure using PSK TLS [15] is given in clause F.3.

F.2.2 Key required to interpret signalling flows

The following key (rules) have been applied to TLS handshake signalling flows to improve readability, reduce errors and increase maintainability:

a) The description of TLS messages and their fields are identified by three fields: "TLS.MESSAGE.FIELD":

- "TLS" identifies that the message is a TLS message;
- "MESSAGE" identifies the name of the TLS message (e.g. ClientHello);
- "FIELD" identifies the name of the TLS message field (e.g. client_version).

An example being "TLS.ClientHello.client_version", which identifies TLS message "ClientHello" and its data field "client_version". The possible TLS message and TLS message field names as well as their encoding to the TLS protocol are specified in IETF TLS related specifications such as IETF RFC 2246 [11] and IETF RFC 3546 [18].

b) If multiple TLS messages are sent in sequence from one entity to another this is described as one step.

- the figures describe the sending of multiple TLS messages in one step by listing the TLS message names in separate lines;
- the description of the step contains the explanation of the messages and their parameters as described in bullet a).

c) In order to differentiate between TLS messages and other protocol messages, the TLS messages are marked with simple arrow line, and all *non-TLS* messages with block arrows.

d) The flows show the signalling exchanges between the following functional entities:

- User Equipment (UE);
- Bootstrapping Server Function (BSF);
- Network Application Function (NAF).

F.3 Signalling flow demonstrating a successful PSK TLS authentication procedure

The signalling flow in figure F.3-1 describes the generic message exchange between UE and NAF using PSK TLS.

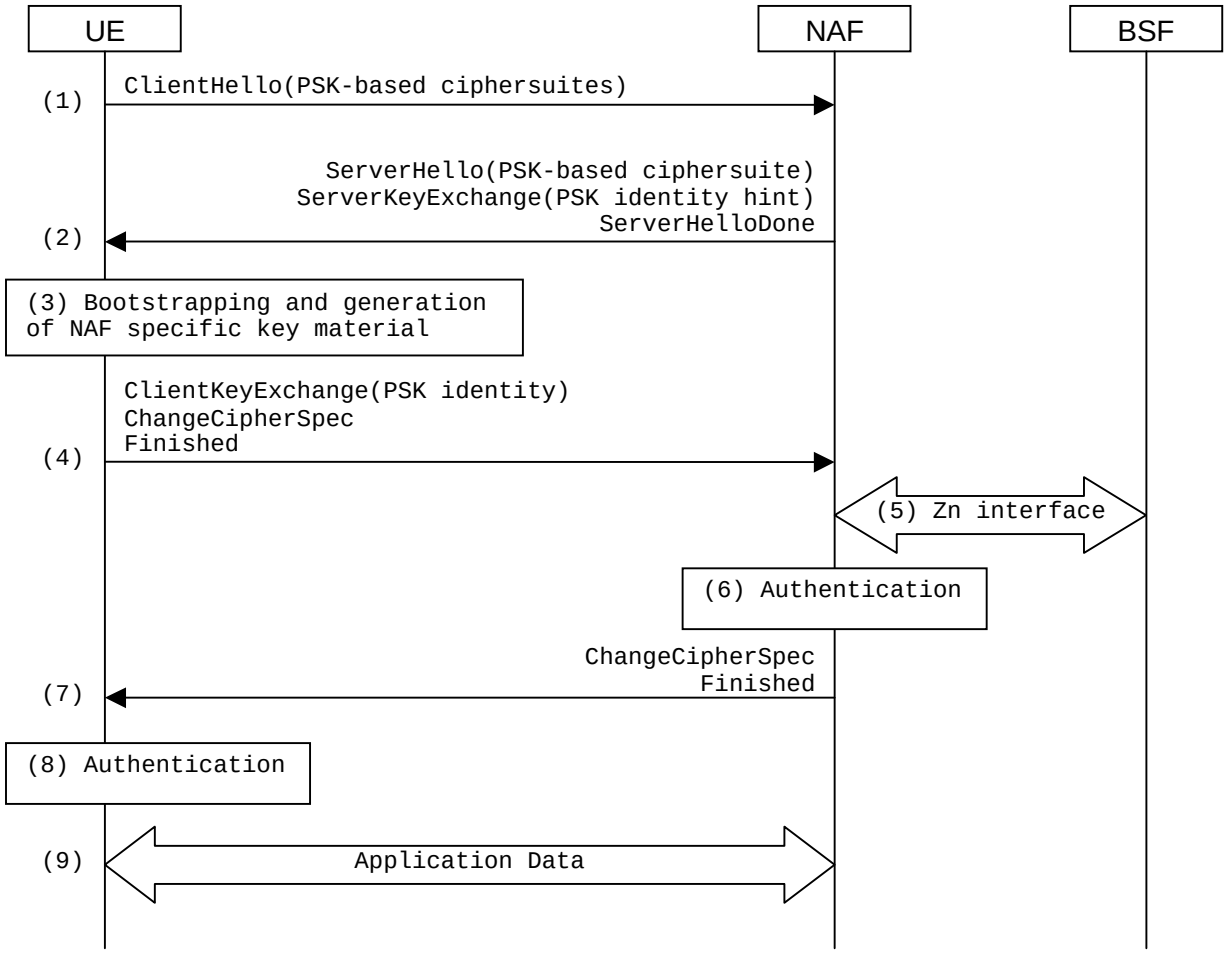

Figure F.3-1: PSK TLS handshake with bootstrapped security association.

1. **TLS handshake message: ClientHello (UE to NAF)**

   The UE sends ClientHello message to the NAF. In order to indicate that the UE is capable of PSK-based authentication it includes the PSK-based ciphersuites to the list of acceptable ciphersuites list. The UE also includes to the ClientHello message the server_name TLS extension containing the hostname of the NAF.

   TLS.ClientHello.client_version: the version of the TLS protocol in the UE is 3.1.

   TLS.ClientHello.random: a UE generated random structure.

   TLS.ClientHello.session_id: the ID of the TLS session is empty, i.e. no previous TLS session is used.

   TLS.ClientHello.cipher_suites: the list of ciphersuites includes one or more PSK-based ciphersuites.

   TLS.ClientHello.compression_methods: a list of the compression methods is null.

   TLS.ClientHello.client_hello_extension_list: list of extensions includes server_name extension that contains the hostname of the NAF.

2. **TLS handshake messages: ServerHello, ServerKeyExchange, ServerHelloDone (NAF to UE)**

   If the NAF wants to use PSK-based authentication, it selects one of the acceptable PSK-based ciphersuites, places the selected ciphersuite in the ServerHello message, and includes an appropriate ServerKeyExchange message. The NAF can help the UE in selecting the correct PSK identity by providing a list of hints in ServerKeyExchange message. That list includes a static string "3GPP-bootstrapping ".

   TLS.ServerHello.server_version: the version of the TLS protocol in the NAF is 3.1.

   TLS.ServerHello.random: a NAF generated random (must be different from ClientHello.random).

TLS.ServerHello.session_id: the identity of the TLS session generated by the NAF.

TLS.ServerHello.cipher_suite: the ciphersuite selected by the NAF is one of the PSK-based ciphersuites listed in ClientHello.cipher_suites.

TLS.ServerHello.compression_method: the compression method selected by the NAF is null.

TLS.ServerHello.server_hello_extension_list: list of extensions is empty.

TLS.ServerKeyExchange.psk_identity_hint: the PSK identity hint contains the constant string "3GPP-bootstrapping".

TLS.ServerHelloDone: this message does not have data fields.

3. **Bootstrapping and generation of NAF specific key material at UE**

   The UE performs the bootstrapping procedure to produce B-TID and Ks_NAF as described in clause A.3. If bootstrapping procedure has been done recently, the UE can use the B-TID and Ks_NAF produced from that procedure.

4. **TLS handshake messages: ClientKeyExchange, ChangeCipherSpec, Finished (UE to NAF)**

   The UE sets concatenated "3GPP-bootstrapping" string, separator character ";" and the B-TID as the PSK identity, and Ks_NAF as the pre-shared key. The UE then sends ClientKeyExchange containing the B-TID, ChangeCipherSpec, and Finished messages to the NAF. The TLS premaster secret is derived from Ks_NAF as specified in draft-ietf-tls-psk-01 [15].

   TLS.ClientKeyExchange.psk_identity: the PSK identity contains concatenated "3GPP-bootstrapping" string, separator character ";" and the B-TID.

   TLS.ChangeCipherSpec.type: contains value 1 (change_cipher_spec).

   TLS.Finished.verify_data: the verify data contains the hash of the handshake messages. For details, see RFC 2246 [11].

5. **Zn: NAF specific key procedure**

   The NAF extracts the B-TID from the ClientKeyExchange message and requests the NAF specific key (Ks_NAF) from BSF. The BSF returns Ks_NAF that corresponds to the B-TID.

   For detailed signalling flows see 3GPP TS 29.109 [3].

Table F.3-1: Bootstrapping authentication information procedure (NAF to BSF)

Message source and destination	Zn Information element name	Information Source in TLS	Description
NAF to BSF	B-TID	ClientKeyExchange.psk_identity	The bootstrapping transaction identifier is encoded in the ClientKeyExchange.psk_identity field according to PSK TLS.

6. **Authentication at NAF**

   The NAF validates the Finished message sent by the UE.

7. **TLS handshake messages: ChangeCipherSpec, Finished (NAF to UE)**

   The NAF sends ChangeCipherSpec, and Finished messages to the UE.

   TLS.ChangeCipherSpec.type: contains value 1 (change_cipher_spec).

   TLS.Finished.verify_data: the verify data contains the hash of the handshake messages. For details, see RFC 2246 [11].

8. **Authentication at UE**

The UE validates the Finished message sent by the NAF.

9. **Application data transfer**

The UE and the NAF initiate application data transfer in the TLS session.

===== END CHANGE =====