PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 9/32 | A2

(11) International Publication Number: **WO 99/25093**

(43) International Publication Date: 20 May 1999 (20.05.99)

(21) International Application Number: PCT/FI98/00869

(22) International Filing Date: 10 November 1998 (10.11.98)

(30) Priority Data:
974186 10 November 1997 (10.11.97) FI

(71) Applicant *(for all designated States except US)*: NOKIA TELECOMMUNICATIONS OY [FI/FI]; Keilalahdentie 4, FIN–02150 Espoo (FI).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: IMMONEN, Olli [FI/FI]; Tuohuskuja 16 A 5, FIN–00670 Helsinki (FI).

(74) Agent: KOLSTER OY AB; Iso Roobertinkatu 23, P.O. Box 148, FIN–00121 Helsinki (FI).

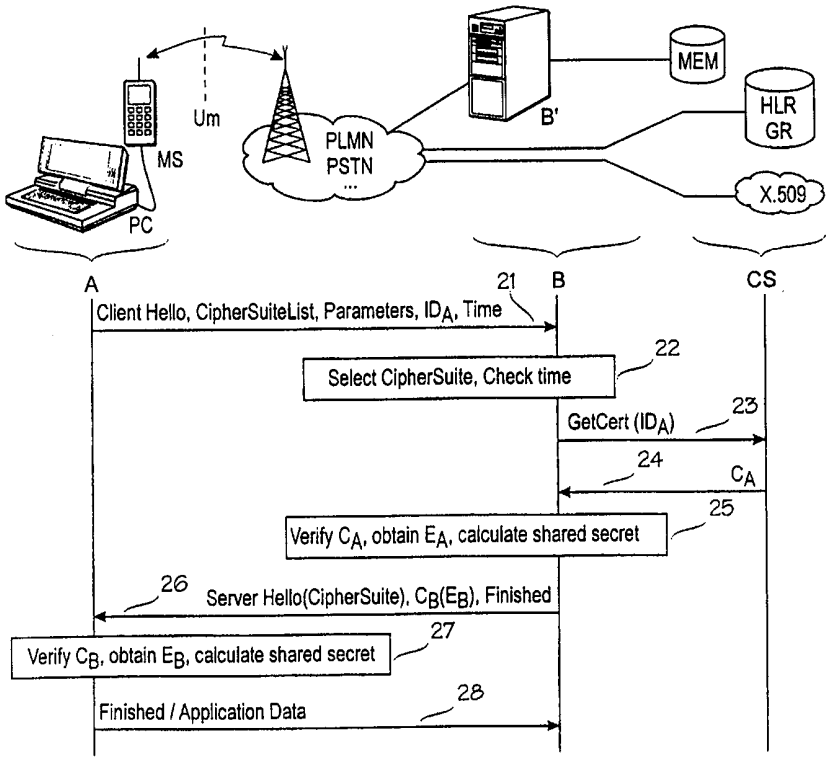
(81) Designated States: AL, AM, AT, AT (Utility model), AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), EE, EE (Utility model), ES, FI, FI (Utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: SECURE HANDSHAKE PROTOCOL

(57) Abstract

Method for a secure handshake protocol between A and B, connected by a slow channel (Um). A sends a first message (21) indicating a set of cipher suites with parameters, and its identifier (ID_A). B selects a cipher suite, obtains A's certificate (C_A) over a fast connection, verifies A's certificate (C_A) and obtains A's public key (E_A). Next B sends a second message (26) comprising B's certificate (C_B), and indication that B has verified A's certificate (C_A), and an indication about the selected cipher suite. A begins to use the selected cipher suite, verifies B's certificate (C_B) and obtains B's public key (E_B). Next A sends a third message (28) indicating that A has verified B's certificate (C_B). Application data can be sent from A to B in the third message (28), whereby a two–way key–exchange and mutual verification is achieved with an effective overhead of two messages (21, 26) between A and B.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

Secure handshake protocol

Background of the Invention

The present invention relates in general to a secure handshake protocol for telecommunications networks. More particularly, the invention relates to a method and an apparatus for providing secure handshake between call parties with minimal overhead before actual data transmission.

Within this application, "TLS" refers to Transport Layer Security. One such protocol is described in *"The TLS Protocol"*, May 21, 1997, by Tim Dierks and Christopher Allen, Consensus Development. This document has been published as "draft-ietf-tls-protocol-03.txt", incorporated herein by reference. More particularly, the present invention proposes an improved handshake protocol which is applicable i.a. in protocols like TLS.

A TLS-type protocol comprises several layers, such as:

Upper layer protocols

Handshake protocol/Alert protocol/Application protocol

Record protocol

Transport protocol

Lower level protocols

Figure 1 is based on section 7.3 of said TLS draft protocol, and it illustrates a prior art handshake method. In order to keep the specification consistent with said draft, parties A and B are also referred to as "client" and "server", respectively. (Terms like "hello" and "finished" are also used consistently with said TLS draft.) In step 11, the client A sends a client hello message. This client hello message comprises a list of cipher suites and compression methods supported by the client. Additionally, the message may also comprise a time stamp. In step 12, the server B selects a cipher suite and a compression method. (Optionally, B may also check the timestamp to make sure that the message is not an old message being retransmitted.)

In step 13 the server B responds with a server hello message. The client hello and server hello messages 11 and 13 establish security between the parties, typically by establishing the following attributes: protocol version, session ID, cipher suite and compression method. In connection with the server hello message, the server B sends its own certificate C_B to the client A and it requests the client A to send its client certificate C_A to the server B. In response to this, in step 14 the client A verifies B's certificate and obtains B's

public key E_B. In step 15 the client A sends B a finished message, indicating that A has been able to verify B's identity. Additionally, A sends its own certificate C_A to B. In step 16, B uses C_A to obtain A's public key E_A. In step 17, B sends its own finished message to the client A. In connection with verifying its peer's identity, each party independently calculates a shared secret key for this session. Now both parties have exchanged keys, agreed on a cipher suite/compression method and verified the identity of the other party. In step 18, the client A can start transmitting application data.

An essential component in the above protocol are the certificates C_A and C_B. By means of certificates signed by a mutually trusted authority, each party can verify its peer's identity. A certificate comprises at least its owner's identity (A/B) and public key(s) (E_A/E_B), period of validity, the issuer of the certificate and the issuer's digital signature. It may also comprise the rights granted to its owner. A suitable mechanism for digital signatures is a reversal of public-key encryption: the issuer signs the certificate with its private key and whoever wants to verify the certificate, does so by using the issuer's public key. A suitable structure for a certificate is specified in ISO standard X.509.

A problem with this prior art handshake protocol is the high overhead required. As seen in Fig. 1, the actual data transmission does not begin until step 15, or after four messages have been transmitted between the parties. In a wireless multiple access system, where the parties A and B are separated by an air interface Um and a public land based mobile network PLMN, the actual messaging is much more complicated than the one shown in Fig. 1. This is because Fig. 1 only shows the actual messages and omits (for clarity) the resource reservation and release steps which are routine for a person skilled in the art, but which are nevertheless indispensable.

Disclosure of the Invention

Based on the foregoing description, it is an object of the present invention to create a method and suitable network elements (nodes and terminals) for providing a secure handshake protocol with a low overhead, i.e. a small number of messages over the air interface. This object will be achieved with a method and network elements which are characterized by what is disclosed in the appended independent claims. Advantageous embodiments of the present invention will be presented in the dependent claims.

The invention is based on a novel distribution of operations between the parties A and B. In addition, some messages over the air interface

can be eliminated by using a land-based certificate store or service, and performing an inquiry to this store. Further, the invention is based on the vision that the last message of the handshake proper should be sent from A to B, whereby actual data transmission can be concatenated with the last handshake message, whereby the net overhead is minimised.

The invention is applicable to telecommunication systems with a slow and/or unreliable transmission channel acting as a bottleneck between the parties.

Brief Description of the drawings

In the following, the invention will be described by means of preferred embodiments with reference to the accompanying drawing, in which:

Figure 1 shows a signalling diagram illustrating a prior art handshake protocol; and

Figure 2 is a combination wherein the bottom portion is an interleaved signalling diagram/flowchart illustrating an embodiment of the invention and the top portion is a block diagram showing how the inventive functionality can be mapped to various network elements.

Detailed description of the invention

Referring now to Fig. 2, an embodiment of the invention will be described. The lower portion of Fig. 2 is an interleaved signalling diagram/flowchart illustrating an embodiment of the invention. The upper portion of Fig. 2 is an associated block diagram, illustrating a possible mapping between call parties and physical network elements.

In step 21 the client A sends a first inter-party message comprising all the elements of the message of step 11. (An inter-party message is a message from A to B or vice versa.) Additionally, the message of step 21 comprises an identifier ID_A of the client A, and encryption parameters (such as random numbers and/or initialisation vectors) if required by any of the indicated cipher suites. The identifier ID_A will be studied later in more detail. In response to the client hello message, in step 22 the server B selects a cipher suite. Preferably, it also checks the timestamp of the message sent by A. In step 23, instead of requesting A's certificate C_A from A itself, the server B uses the ID_A sent by A to retrieve A's certificate C_A from a certificate store CS. The connection between B and CS should be significantly faster than the air interface Um. In step 24, the trustee CS returns A's certificate C_A. Alternatively or

additionally, B can also maintain a local memory MEM of certificates and omit the inquiry to CS if A's certificate is found in the local memory. In step 25, B verifies C_A, obtains A's public key E_A and calculates the shared secret key. In step 26, B sends a second inter-party message to A. The second inter-party message comprises B's certificate C_B. It also indicates that B has been able to verify A's certificate. (However, this indication can be an implicit one, meaning that B only sends its certificate if it has verified A's certificate.) In step 27, A verifies B's certificate C_B, obtains B's public key E_B and calculates the shared secret key. In step 28, A sends B a third inter-party message comprising a finished message which indicates that it has been able to verify B's certificate.

For clarity, Fig. 2 only shows what happens when the handshake is successful, i.e. both parties act according to the protocol. If a departure from the protocol is detected, this is usually a fatal error and the handshake terminates.

It should be noted that the last inter-party message (comprising the finished message in step 28) points from A to B. This is in marked contrast to the prior art handshake shown in Fig. 1. An advantage of this property of the invention is that application data can be concatenated with the third inter-party message in step 28. Thus the effective overhead of the handshake protocol according to the invention is only two inter-party messages, compared to an overhead of four messages in the prior art handshake. In order to achieve this, an appropriate key exchange mechanism must be used. Suitable key exchange algorithms include Diffie-Hellman (DH) with fixed parameters certified with Digital Signature Algorithm (DSA). The DH algorithm can be found in most textbooks on cryptography. Additionally, the original Diffie-Hellman algorithm (DH) is described in US Patent 4 200 770 and the Digital Signature Algorithm (DSA) is a U.S. standard and a de facto international standard. Another good combination is Elliptic Curve Diffie-Hellman (ECDH) with fixed parameters certified with Elliptic Curve Digital Signature Algorithm (ECDSA). The difference between standard DH and ECDH is only different mathematics in obtaining and using encryption and decryption keys. Such differences are not essential to the invention.

Additionally, RSA (Rivest-Shamir-Adlemann) and ECES (Elliptic Curve Encryption Scheme) algorithms can be used with appropriate modifications. With these algorithms, a server key exchange takes place as follows. B generates a random number, which is a pre-master secret, encrypts it with A's

public key, and sends the result to A. Thus the message in step 26 would comprise ServerHello, C_B, ServerKeyExchange, Finished. Now A decrypts this pre-master secret. This server key exchange procedure resembles a mirror image of the one used in TLS, whereby the handshake can still be accomplished with two messages over the air interface.

The handshake method described above uses public keys. As is well known, public-key cryptography is much slower than symmetric cryptography. Therefore, it is preferable to use the public-key handshake only for exchanging parameters which are used for computing a shared key for symmetric cryptography, such as DES. The parameters (random numbers) sent in message 21 can be used for this purpose.

Although the inventive handshake somewhat limits the available key-exchange mechanisms during the handshake phase, the invention does not limit the available mechanisms used for the actual data transmission. In other words, the invention does not limit the choices available for symmetric cryptography, although it requires that the parameters for the symmetric cryptography first used are exchanged by using a key-exchange mechanism with fixed parameters. The encryption parameters sent in message 21 (and 26) can be combined with private keys to create pre-master secrets which in turn are used to create master secrets, etc. Thus, to each application data message following message 28, a separate message can be concatenated. This separate message can be used for changing the selected cryptography mechanism.

The identifier ID_A of client A should be unique to each A. Suitable identifiers are e.g. a network number, such as MSISDN or an X.509 number. The ID_A is not protected by the handshake protocol proper, although it may be protected by a lower level protocol. Therefore, it is preferable to create the ID_A using a one-way function, such as a hash function. One-way functions are functions that are much easier (at least by several orders of magnitude) to perform in one direction than in the reverse direction. Examples of one-way functions are multiplying large prime numbers, discrete exponentiation, elliptical functions and hash functions. The advantage of one-way functions is that they hide the identity of A from possible eavesdroppers. As is well known, hash functions reduce information. Hashed numbers are thus not necessarily unique. However, a good combination is achieved by using a hash of the cli-

ent's public key E_A and assigning public keys such that they do not produce identical hash values.

The upper portion of Fig. 2 shows how the functionality of the invention can be mapped to various network elements. The invention can be used in a wireless communication system, such as a mobile communications system. The client A can be a mobile station MS, possibly having a portable computer PC connected or integrated thereto. The server B can be a computer B' providing financial services, or granting access to confidential information, etc. A and B can communicate over an air interface Um and via a public land based mobile network PLMN, possibly also via a public switched mobile network PSTN.

The trustee CS could be implemented in one of the registers of the PLMN, such as a home location register (HLR), or a GPRS register GR. Alternatively, the trustee services can be implemented as disclosed in said ISO standard X.509.

Instead of retrieving A's certificate from CS, or in addition to it, B can maintain a local memory MEM of certificates and omit the inquiry to CS if A's certificate is found in the local memory. B can e.g. be connected to a local area network and the certificates of all the clients A are maintained over the local area network. A local memory MEM can also be used as a cache memory for storing recently used certificates. In real-time applications, if a certificate is revoked, the computer B' must be informed and it must also delete the revoked certificate from its cache.

An important advantage of the invention is that the overhead over the slow communications channel, such as the air interface, can be halved compared to prior art protocols. Another advantage is that the client's certificate C_A does not have to stored in the client itself. Since the client A is typically a mobile station, its memory capacity is limited. This also reduces the information gained by dishonest third parties in case the client hardware gets lost or stolen, or is used by unauthorised persons. Also, because the client's certificate C_A is not transmitted over the air interface, less information is leaked to possible eavesdroppers.

The invention has been described in its preferred embodiments. However, the specifications for telecommunications technology are developing rapidly. Such developments may require additional modifications to the invention. Therefore, all words and expressions should be interpreted broadly, and

they are intended for illustrating rather than limiting the invention as specified in the appended claims.

Claims:

1. A method for a secure handshake protocol between a first party (A) and a second party (B), connected via a communications channel (Um, PLMN) wherein each party supports a respective set of cipher suites and for each party, a respective certificate (C_A, C_B) is defined, each of the certificates (C_A, C_B) comprising a public key (E_A, E_B) of its respective owner (A, B); the method being characterized in that

the first party (A) sends a first inter-party message (21) indicating the set of cipher suites supported by it, parameters required by the cipher suites, and an identifier (ID_A) of the first party (A);

in response to the first inter-party message (21), the second party (B):

- selects one of said indicated cipher suites which is also supported by the second party (B);
- uses said identifier (ID_A) to obtain the certificate (C_A) of the first party (A) over a connection which is significantly faster than the communications channel (Um, PLMN) connecting said parties;
- verifies said obtained certificate (C_A) the of the first party (A) and obtains the public key (E_A) of the first party (A);
- sends a second inter-party message (26) comprising the certificate (C_B) of the second party (B), an indication that the second party (B) has verified the certificate (C_A) of the first party (A), and an indication about said selected cipher suite;

in response to the second inter-party message (26), the first party (A):

- begins to use the selected cipher suite;
- verifies the certificate (C_B) of the second party (B) and obtains the public key (E_B) of the second party (B);
- sends a third inter-party message (28) indicating that the first party (A) has verified the certificate (C_B) of the second party (B);

whereby information not needed for the above steps can be sent from the first party (A) to the second party (B) in the third inter-party message (28), thus providing a two-way key-exchange and mutual verification with an effective overhead of two inter-party messages (21, 26).

2. A method according to claim 1, characterized in that said step of obtaining the certificate (C_A) of the first party (A) comprises retrieving it from a source (CS) external to the second party.

3. A method according to claim 2, characterized in that said external source (CS) is a register (HLR, GR) of a telecommunications network or a directory service substantially conforming to ISO standard X.509.

4. A method according to claim 1, characterized in that said step of obtaining the certificate (C_A) of the first party (A) comprises retrieving it from a local memory (MEM).

5. A method according to any one of the preceding claims, characterized in that said identifier (ID_A) of the first party (A) is formed by means of a one-way function, preferably a hash function.

6. A method according to any one of the preceding claims, characterized in that said second inter-party message (26) comprises a pre-master secret which the second party (B) obtains by generating a random number and encrypting it with the public key (E_A) of the first party (A).

7. A telecommunications apparatus (A) being adapted to act as a first party in a secure handshake protocol between said apparatus (A) and a second party (B), said apparatus being characterized in that it is adapted to:

- send a first message (21) to the second party (B), said first message indicating a set of cipher suites, parameters required by said cipher suites, and an identifier (ID_A) of the apparatus (A);
- receive a second message (26) from the second party (B), said second message comprising an indication about a cipher suite selected by said second party, a certificate (C_B) of the second party, an indication that the second party (B) has used said identifier (ID_A) of the apparatus to obtain and verify a certificate (C_A) of the apparatus (A), and;
- use the cipher suite indicated by said second message (26);
- verify the certificate (C_B) of the second party (B) and obtain a public key (E_B) of the second party (B);

- send a third message (28) to the second party (B), said third message indicating that the apparatus (A) has verified the certificate (C_B) of the second party.

8. A telecommunications apparatus (A) according to claim 7, characterized by being adapted to insert information not needed for the above operations in said third message (28).

9. A telecommunications apparatus (B) being adapted to respond to a secure handshake protocol initiated by a first party (A), said apparatus (B) being connectable to said first party (A) by a communications channel (Um, PLMN), said apparatus (B) being characterized in that it is adapted to:

- receive a first message (21) from the first party (A), said first message indicating a set of cipher suites, parameters required by the cipher suites, and an identifier (ID_A) of the first party (A);
- select one of said indicated cipher suites;
- use the identifier (ID_A) to obtain a certificate (C_A) of the first party (A) over a connection which is significantly faster than said communications channel (Um, PLMN);
- verify said obtained certificate (C_A) of the first party (A) and obtain a public key (E_A) of the first party (A);
- send a second message (26) to the first party (A), said second message comprising a certificate (C_B) of the apparatus (B), indicating that the apparatus (B) has verified the certificate (C_A) of the first party (A), and indicating said selected cipher suite;
- receive a third message (28) from the first party (A), said third message indicating that the first party (A) has verified the certificate (C_B) of the apparatus (B).

10. A telecommunications apparatus (B) according to claim 9, characterized by being adapted to extract information not needed for the above operations from said third message (28).

Fig. 1

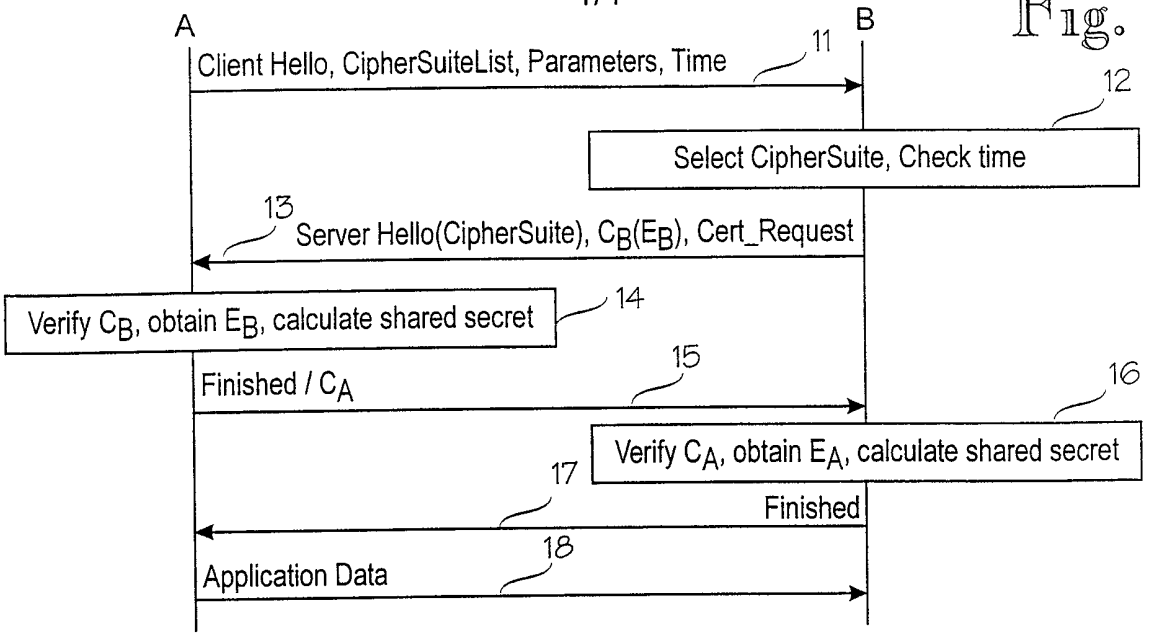
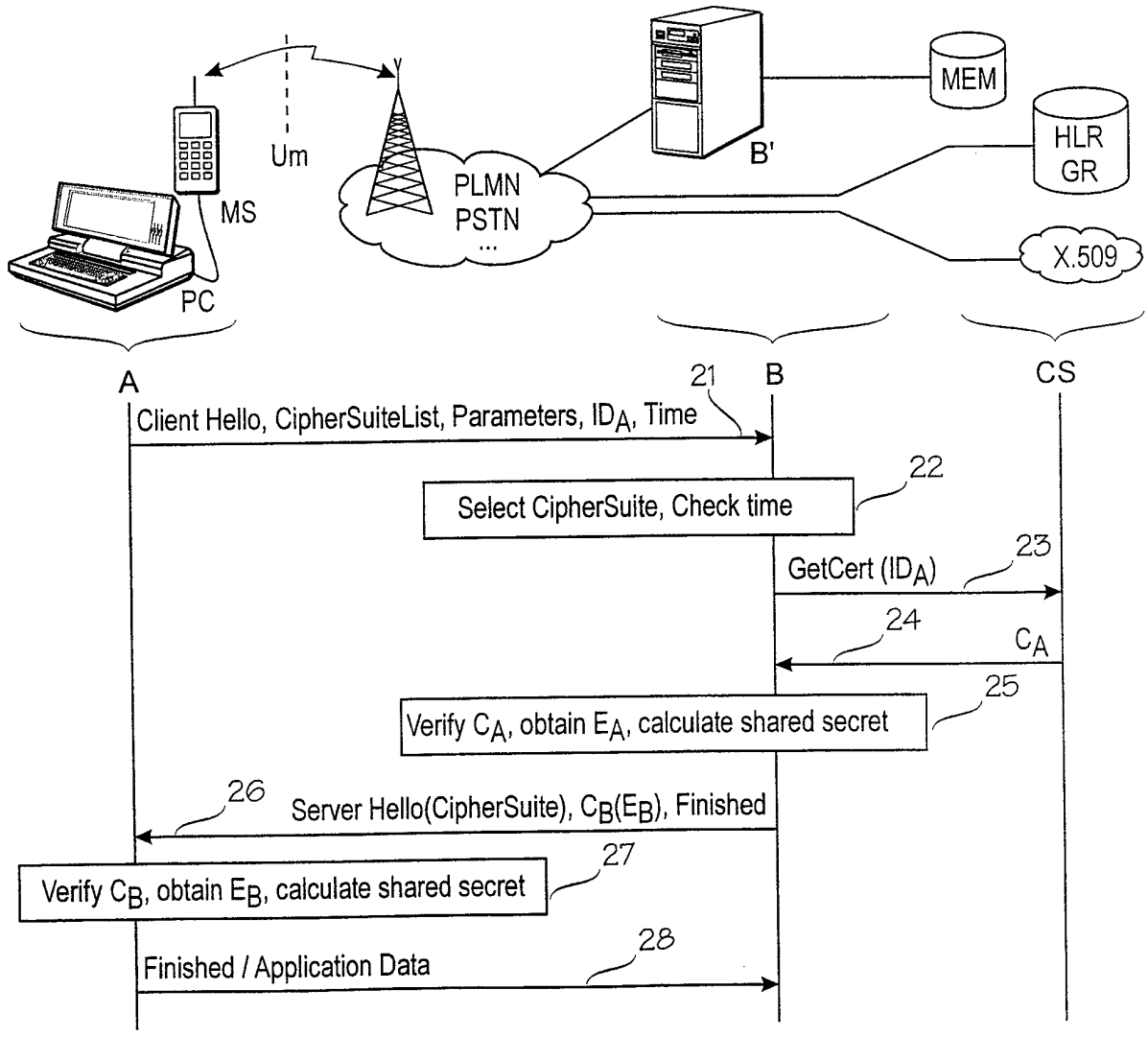

A
B
Client Hello, CipherSuiteList, Parameters, Time
11
12
Select CipherSuite, Check time
13
Server Hello(CipherSuite), $C_B(E_B)$, Cert_Request
Verify C_B, obtain E_B, calculate shared secret
14
Finished / C_A
15
16
Verify C_A, obtain E_A, calculate shared secret
17
Finished
18
Application Data

Fig. 2


Um
MS
PC
PLMN
PSTN
...
B'
MEM
HLR
GR
X.509
A
B
CS
21
Client Hello, CipherSuiteList, Parameters, ID_A, Time
22
Select CipherSuite, Check time
23
GetCert (ID_A)
24
C_A
25
Verify C_A, obtain E_A, calculate shared secret
26
Server Hello(CipherSuite), $C_B(E_B)$, Finished
27
Verify C_B, obtain E_B, calculate shared secret
28
Finished / Application Data