ETSI TR 133 969 V16.0.0 (2020-08)

Digital cellular telecommunications system (Phase 2+) (GSM);
Universal Mobile Telecommunications System (UMTS);
LTE;
Study on security aspects of Public Warning System (PWS)
(3GPP TR 33.969 version 16.0.0 Release 16)

Reference
RTR/TSGS-0333969vg00

Keywords
GSM,LTE,SECURITY,UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Legal Notice

This Technical Report (TR) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document studies security features and mechanisms for protection against false Base Stations broadcasting False Warning Notifications.

The default terminal behaviour is to accept all Warning Notifications even if their authenticity is unknown (i.e. no security protection). The default terminal behaviour is therefore open to the presentation of false Warning Notifications issued by false BSs even in countries without a Public Warning System (PWS) deployed.

Examples of false BS risks include, but are not limited to:

- False Warning Notifications to induce panic;
- Abuse of warning system broadcast channel to send advertising / spam.

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 TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 22.268: "Public Warning System (PWS) requirements".

[3] 3GPP TS 23.041: "Technical realization of Cell Broadcast Service (CBS)".

[4] 3GPP TS 48.049: "Base Station Controller - Cell Broadcast Centre (BSC-CBC) interface specification; Cell Broadcast Service Protocol (CBSP)".

[5] 3GPP TS 25.419: "UTRAN Iu-BC interface: Service Area Broadcast Protocol (SABP)".

[6] 3GPP TS 23.251: "Network sharing; Architecture and functional description".

[7] 3GPP TR 33.859: "Study on the Introduction of Key Hierarchy in Universal Terrestrial Radio Access Network (UTRAN)".

[8] 3GPP TS 33.102 "3G security; Security architecture".

[9] 3GPP TS 35.206 "3G Security; Specification of the MILENAGE Algorithm Set: An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 2: Algorithm specification".

[10] FIPS 186-3: "Digital Signature Standard (DSS)".

[11] SP 800-57 Part 1: "Recommendation for Key Management – Part 1: General (Revision 3)".

[12] SP 800-56A: "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography".

[13] M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols", ACM CCS 1993.

[14] V. Shoup, "Lower Bounds for Discrete Logarithms and Related Problems", EUROCRYPT 1997.

[15] S. Vaudenay, "The Security of DSA and ECDSA", PKC 2003.

[16] P. Paillier and D. Vergnaud, "Discrete-Log-Based Signature May Not Be Equivalent to Discrete Log", Asiacrypt 2005.

[17] D. Brown, "The Exact Security of ECDSA". Technical Report CORR 2000–34, Certicom Research, 2000.

[18] D. Brown, R. Gallant, and S. Vanstone, "Provably secure implicit certificate schemes", Financial Cryptography 2001.

[19] D. Brown and D. Johnson, "Formal Security Proofs for a Signature Scheme with Partial Message Recovery", CT-RSA 2001.

[20] D. Brown, M. Campagna, and S. Vanstone, "Security of ECQV-Certified ECDSA Against Passive Adversaries", IACR eprint Archive, http://eprint.iacr.org/2009/620.

[21] 3GPP TS 31.115: "Secured packet structure for (Universal) Subscriber Identity Module (U)SIM Toolkit applications".

[22] 3GPP TS 31.102: "Characteristics of the Universal Subscriber Identity Module (USIM) application".

[23] 3GPP TS 31.116: "Remote APDU Structure for (U)SIM Toolkit applications".

[24] 3GPP TS 31.111: "Universal Subscriber Identity Module (USIM) Application Toolkit (USAT)".

[25] 3GPP TS 29.168: "Cell Broadcast Centre interfaces with the Evolved Packet Core; Stage 3".

[26] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)".

[27] 3GPP TS 33.223: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA) Push function".

[28] 3GPP TS 33.224: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA) push layer".

[29] 3GPP TS 23.090: "Unstructured Supplementary Service Data (USSD); Stage 2".

[30] 3GPP TS 23.202: "Circuit switched data bearer services".

[31] 3GPP TS 33.203: "3G security; Access security for IP-based services".

[32] 3GPP TS 33.234: "3G security; Wireless Local Area Network (WLAN) interworking security".

[33] 3GPP TS 24.301: "Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3".

[34] Cornell University, "GPS receivers can be 'spoofed,' say researchers ", http://phys.org/news141300510.html.

[35] 3GPP TS 33.402: "3GPP System Architecture Evolution (SAE); Security aspects of non-3GPP accesses".

[36] 3GPP TS 43.318: "Generic Access Network (GAN); Stage 2".

[37] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[38] 3GPP TS 23.234: "3GPP system to Wireless Local Area Network (WLAN) interworking; System description".

[39] 3GPP TS 23.402: "Architecture enhancements for non-3GPP accesses".

[40] 3GPP TS 33.310: "Network Domain Security (NDS); Authentication Framework (AF)".

[41] 3GPP TS 44.060: "General Packet Radio Service (GPRS); Mobile Station (MS) - Base Station System (BSS) interface; Radio Link Control / Medium Access Control (RLC/MAC) protocol".

3 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply.
An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

BS	Base Station
CA	Certificate Authority
CBC	Cell Broadcast Centre

CBE	Cell Broadcast Entity
CMAS	Commercial Mobile Alert System
DSA	Digital Signature Algorithm
eBATS	ECRYPT Benchmarking of Asymmetric Systems
ECDSA	Elliptic Curve DSA
ECQV	Elliptic Curve Qu-Vanstone
EU-Alert	European Emergency Alert System
GWCN	Gateway Core Network
ETWS	Earthquake and Tsunami Warning System
KPAS	Korean Public Alert System
MOCN	Multi-Operator Core Network
NSUC	Network Security Use Counter
NDS	Network Domain Security
PWS	Public Warning System
RISC	Reduced Instruction Set Computing
TLS	Transport Layer Security

4 Recommendations on security requirements of PWS

Requirements for PWS and PWS Security identified by SA1 are specified in 3GPP TS 22.268 [2].
Of special interest for the PWS Security work are:

- TS 22.268 [2], clause 4.6.4 on "Enabling and disabling of Warning Notifications" and
- TS 22.268 [2], clause 4.8 on "Security requirements".

The requirements for PWS Security are optional since there are regions and countries that do not require this functionality. Additional **potential requirements** for PWS Security identified by SA3 are listed below.

NOTE: When the "potential requirements" from the present document might be introduced into normative document (Technical Specifications - TSs), the word **"should"** may need to be changed into **"shall"** (with bold characters).

- For UE that are enabled to receive Warning Notifications from the VPLMN in roaming areas, it **"should"** meet the security requirements listed above.

Editor's Note: The above requirement need further clarification.

- The authentication solution **"should"** be robust against errors in the key distribution and overload so that genuine messages do not get rejected due to some error or overload in the network or in the authentication mechanism itself.
- If the UE has not been configured for PWS message security, PWS warning messages **"should"** always be presented to the PWS application on the PWS-UE.
- The result in evaluating the authenticity/integrity of the Warning Notification **"should"** be provided to the receiving PWS application on a PWS-UE. Depending on the national regulation the user may have access to this information from the PWS application (the mechanism to specify this preference is out of scope of the present document).
- There **"should"** be a mechanism to indicate to the PWS-UE which credential is to be used for the verification of the integrity or the authenticity of the Warning Notification.
- The origin and integrity of the public key to verify signed warning messages **"should"** be ensured when the public key is made available to the UE.
- Solutions mitigating the PWS Security circumvention attack described in the threats clause of the present document **"should"** be provided.

NOTE: SA3 has agreed on the working assumption that SIM subscribers are excluded from PWS Security.

5 System architecture of PWS

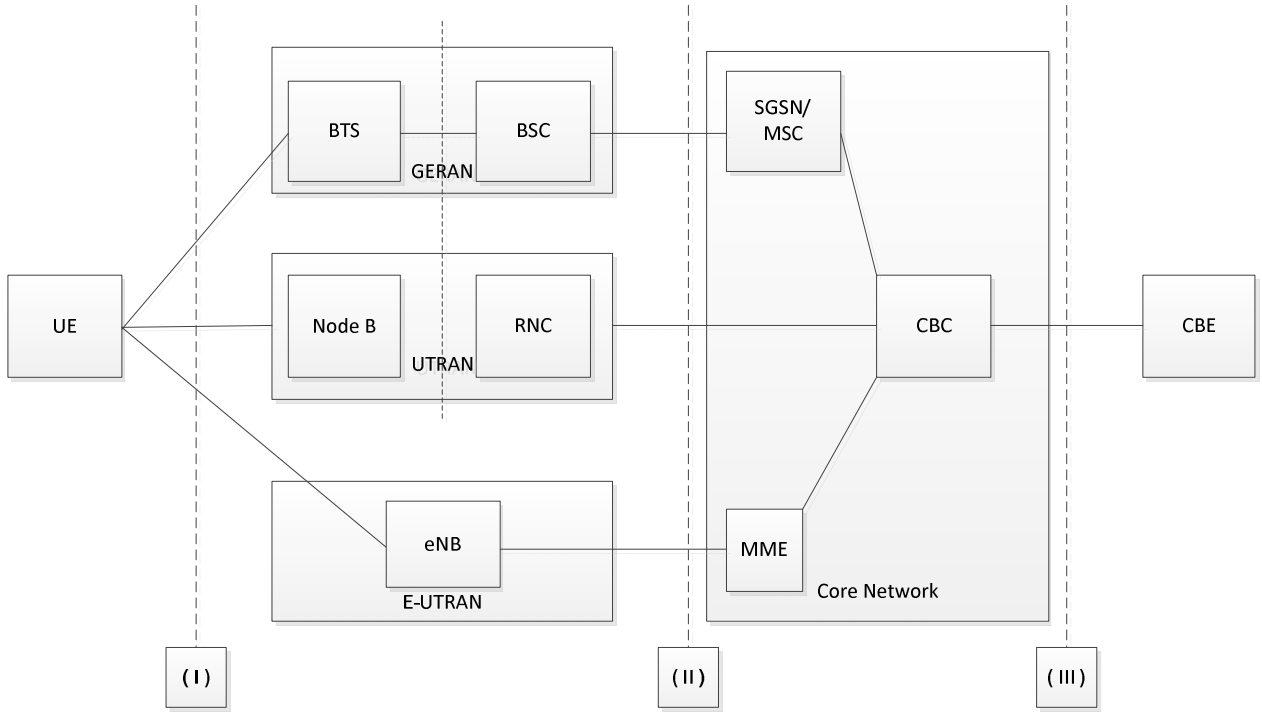

Figure 5.1: PWS system architecture overview

Figure 5.1 gives an overview of the system architecture.

- Air interface between UE and access network needs security protection as PWS Warning Notification messages are broadcast to UE via SYSTEM INFORMATION.

- CBC is part of the core network and connects to the network entity.
  For GERAN, CBC connects with the access network entity BSC.
  For UTRAN, CBC connects with the access network entity RNC.
  For E-UTRAN, CBC connects with the core network entity MME.
  The protocols between the CBC and these network entities are defined in 3GPP TS 48.049 [4], TS 25.419 [5], TS 23.041 [3] and TS 29.168 [25].

- CBE is on a national level and outside of the scope of the 3GPP network.
  It is assumed that the CBE is responsible for all aspects of formatting CBS, including the splitting of a CBS message into a number of pages and the actual signing.

- MSC/SGSN or MME can be used to deliver PWS keys to UEs.

Editor's Note: The security solution should minimize the impact to the current mechanism

Editor's Note: MSC/SGSN may receive PWS keys from CBC, or PWS keys are configured in MSC/SGSN directly. It is for FFS how MSC/SGSN gets the PWS keys in GERAN/UTRAN, and whether a new interface between MSC/SGSN and CBC should be added, e.g. for the synchronization of NSUC.

6 Security features of PWS

6.1 PWS threats and analysis

6.1.1 General

In the following potential threats and attacks are discussed.

The solution ultimately needs to protect against attacks that are in the interface between PLMN and the Warning Notification provider. However, this is outside the scope of 3GPP. The attacks which are within the wired network can effectively be dealt with using NDS methods. So the most crucial threat is the one over the air interface.

Furthermore, the operating conditions and liability that PWS Security can handle need to be clarified. One aspect to consider in particular is the size of the group that **"should"** be protected (individual user, small group of users, large crowd). Other aspects to consider are: the time frame needed by an attacker to prepare and execute an attack, the complexity of the attack (manpower and means), and the size of the geographical area the attacker is able to target.

Finally, assessing the type and amount of damage that can be done by an attack is useful when weighing the potential damage against the cost of additional features introduced for PWS Security.

Editor's Note: It needs to be further clarified what are the relevant threats to PWS.

6.1.2 PWS Security circumvention attack

The possibility to attack an unprotected PWS is the motivation for PWS Security. The attack consists in setting up a false BS and sending false warning messages in order to create panic. The means required for the circumvention attack are largely the same as the ones required for performing the attacks that motivated the need for PWS Security in the first place. Hence, if an attack on unprotected PWS is assumed to be possible then also the circumvention attack described here has to be assumed to be possible. Or, in other words, if it is believed that PWS Security is necessary then it should be accepted that there is a need to prevent the circumvention attack.

First, the assumptions made for the attack are listed and then two attack variants are described.

Assumptions:

- (A1): All networks in country A implement PWS Security, as defined by 3GPP. All UEs that are capable of PWS Security, and whose home network is in country A, have PWS Security enabled.
- (A2): [The network may or may not implement PWS] There is a network VN in country B that does not implement PWS Security; and UEs with home network in country A are configured to display unprotected warning messages while roaming in network VN. VN has roaming agreements with the networks in country A.
- (A3): An attacker is capable of setting up one or more false BS, making a sufficiently large number of users camp on them and sending false warning messages through the false BS(s) to these users resulting in a large-scale panic or other significant damage.
- (A4): The attacker is, in addition to (A3), capable of setting the (MCC, MNC) broadcast by the false BS(s) to that of VN from (A2).
- (A5) (required only for one of the attack variants): UEs with home network in country A are configured to display unprotected warning messages while in Limited Service State (LSS).

NOTE: (A1) can be enforced by a regulator of country A. (A2) and (A5) are compatible with the requirements in TS 22.268 [2] and in clause 4 of the present document. (A3) is the assumption that motivates the need for PWS Security in the first place. (A4) is seen as a trivial step, given (A3).

Attack description:

The attack is easiest over a GERAN access network, but possible for UTRAN and E-UTRAN as well. Note that, in order to circumvent PWS Security, it would be sufficient if the attack worked only for one of the attack variants.

In all cases, the attacker looks for the weakest signal of a network in country A for the chosen access technology and makes his false BS broadcast with a very strong signal on the corresponding frequency. If necessary, the attacker could also jam all other frequencies to make sure that the UE cannot attach to a network broadcasting a weaker signal on another frequency (Note that operators and access technologies are separated by different frequency bands). If the (MCC, MNC) of the false BS is that of a network with which the UE's home network has a roaming agreement (which is true for the network VN in country B according to assumption (A2)) the UE will try to attach to it.

The cases are now described in turn.

Attack over a GERAN access network: The false BS, emulating also the behaviour of an MSC/VLR or SGSN, will reply to the UE's LAU Request or RAU Request with an unencrypted LAU Accept or RAU Accept.
The false BS will not send a Ciphering Mode Command, hence the communication will remain unencrypted. The UE will believe to have successfully registered to network VN. The false BS can then start to broadcast unprotected warning messages, which the UE will display to the user according to assumption (A2).

Attack over any access network technology using Limited Service State (LSS): The false BS, emulating also the behaviour of an MSC/VLR or SGSN, will reply to the UE's LAU Request or RAU Request with an unencrypted LAU Reject or RAU Reject. As the UE does not find any other acceptable network around, the UE will be in LSS. The false BS can then start broadcasting unprotected warning messages, which the UE will display to the user according to assumption (A5).

6.1.3 Spoofing, tampering, and suppressing

For PWS Warning Notification messages, the security threats are similar with ETWS. There may be spoofing attacks, e.g. an attacker may forge and issue PWS Warning Notifications maliciously. The messages sent over the air may introduce spoofing attacks. Another threat may be tamper attacks, e.g. an attacker may record and tamper a PWS Warning Notification message over the air interface.

RAN2 has decided to broadcast PWS Warning Notifications to user via SYSTEM INFORMATION over air interface. However, broadcasts of SYSTEM INFORMATION are not protected. If an attacker can imitate the BS behaviour maliciously and broadcast false PWS Warning Notifications or tamper PWS Warning Notifications coming from CBE, it will cause serious panic among the population.

Suppressing the display of a genuine warning message is another possible attack. It should be noted that jamming the radio interface could help in suppressing warning messages, but not in forging them. The attack is geographically limited.

NOTE: A more detailed risk analysis is missing in the present document and would needs to answer at least the following questions:

- Is it considered a relevant threat to PWS if an attacker can send a forged warning message to an individual user or a small group of users, or only if he can send it to a large crowd?
- Is it considered a relevant threat to PWS if an attacker can suppress the display of a warning message to an individual user or a small group of users, or only if he can suppress it for a large crowd?
- What would be the timeframe for preparing and executing these attacks?
- Would the attacks be geographically confined?

Editor's Note: It is ffs where the attack is originating

6.1.4 Threats to the delivery of the public key

This clause assumes that a digital signature is used to protect a PWS Warning Notification, for details see clause 6.2.

The key for verifying the signature of a PWS Warning Notification is public. However, there is potential to tamper with it if not delivered in a secure way.

An attacker could modify a public key and/or distribute a false public key and is therefore able to send signed faked messages. The UE verifies the message with the false public key. It believes to have received a correct warning message because it has been correctly verified. Thus, the main threat in PWS Security can be seen in compromising a public key.

The public key would be issued by the entity that creates the signature, i.e. the national regulator or the authority to broadcast warning messages. It **"should"** be made available in an authentic way either by the CBE or any entity that is trusted by the CBE.

Options for public key delivery discussed in this study comprise a 3GPP network element, an application server, the distribution during manufacturing time or OTA to the USIM, or the distribution together with the warning message, if a root certificate has been made available to the UE beforehand.

Thus, it needs to be answered who is responsible for the public key delivery and how does the terminal gain root key(s) (in case of certificate usage) or the public key (of a CBE or a signing proxy (in case of many CBEs))?

6.1.5 Location of node protecting the public key delivery in PWS

This clause assumes that a digital signature is used to protect a PWS Warning Notification, for details see clause 6.2.

In case, a 3GPP network element delivers the public key, the operator takes responsibility in the public key delivery, the placement of the node that protects the delivery of the key is an important consideration in the security for PWS. For E-UTRAN and GERAN PS, it is possible to protect the PWS key delivery from the core network node to the UE using legacy security mechanism, while in UMTS and GERAN CS the protection can only be applied from RAN nodes. These RAN nodes (e.g. collapsed NodeBs or HNB in UMTS) may be deployed in location that are at the edge of the network and hence not be in the most secure locations. As a result of this they are significantly more vulnerable to attack than core network nodes.

Suppose that the node is towards the edge of the network is used to protect the delivery of the PWS key to the mobile. Then the compromise of such a node would allow the attacker to send false keys to all the users that are attached to that node. It would be enough to break the secure tunnel between this node and the operator's network by getting the relevant key out of the compromised node. Then a man-in-the-middle could be inserted between the compromised node and the core network that modifies the signalling to send a known PWS key to the users. It would be then easy to fake a warning message that all the users under that node would believe is genuine. A more sophisticated attack would be to use a compromised network element, for example an open HNB, to get keying material in order to establish to establish a false BS from which to launch an attack. If such attacks are deployed at places where large crowds gather, then it could be possible to make a large number of people incorrectly receive a warning message simultaneously.

In case the public key is delivered by an application server, the server needs to be protected such that no tampering can happen. Security protocols are needed for retrieving the public key.

In case of distribution during manufacturing time, the manufacturer **"should"** make sure that there is no possibility to tamper the public key.

In case the public key is broadcast, it should have a certificate attached in order to not tamper with.
If the terminal wants to verify that the public key received is authentic, it **"should"** have the root key to verify the certificate.
If a signing proxy is in place, the public key of the proxy needs to be securely delivered.

6.2 Proposed security features of PWS

6.2.1 General

In order to guarantee the authenticity and integrity of the Warning Notifications, the security requirements specified in 3GPP TS 22.268 [2] are introduced. In order to meet these security requirements, it has been decided that PWS Warning Notifications **"should"** be protected with a signature that is included in the Warning-Security-Information IE in the WRITE-REPLACE Request message. Moreover, some PWS Security features should be considered and defined in details as to solve the remaining security issues listed.

A UE that has the capability to receive PWS message **"should"** support the PWS interface as specified in TS 23.041 [3].
CBE sends Warning Notifications to the user via core network points and access network points. When receiving PWS Warning Notifications with security, the user verifies the signature with the corresponding key and the algorithm.
So it is essential that the user **"should"** be notified which key and algorithm should be used for signature verification. Otherwise, it will cause verification failure.

As mentioned above, it **"should"** ensure the synchronization of signature key and the signature algorithm between UE and the network. In the current specification, it only states PWS Warning Notifications **"should"** be protected. How to verify PWS Warning Notification messages that are integrity protected has not been specified yet. If PWS uses some popular signature algorithm, e.g. RSA (the length of the message signature is at least 1024 bits) the Warning-Security-Information field cannot meet possible length restrictions of the different radio access technologies. So the length of the signature in particular should be considered.

In summary, there are several dimensions for choosing which digital signature algorithms can be used for PWS Warning Notifications protection. Thus several security features are considered for PWS Security as follows.

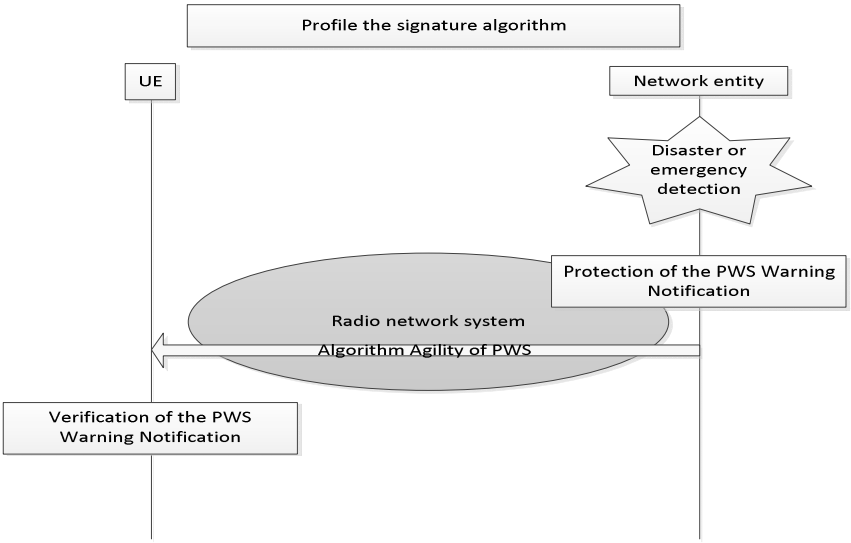

Figure 6.2.1.1: PWS Security features

6.2.2 Restrictions on the PWS message signature length

6.2.2.1 General

PWS is a common system for the distribution of ETWS, CMAS, KPAS and EU-Alert warning messages. All of these warning systems except KPAS **"should"** be supported on GERAN, UTRAN and E-UTRAN radio access technologies. KPAS only needs to be supported in E-UTRAN.

NOTE: TS 22.268 [2] contains no such statement for ETWS, while it explicitly mentions the RANs to be supported for the other three warning systems. However, TS 23.041 [3] and the corresponding RAN and GERAN specifications specify ETWS for all three radio access technologies.

A PWS security solution should support all or a subset of the PWS warning systems.

6.2.2.2 Warning message format in CMAS, KPAS, and EU-Alert

CMAS, KPAS, and EU-Alert warning systems share the same message format. This format is defined in TS 23.041 and differs slightly depending on the access technology but the main content remains the same. It is possible to extend this message with a signature field (and other necessary security parameters) without breaking the length restrictions that the different access technologies put on the message. However, care **"should"** be taken when extending the message so that UEs that support PWS but not PWS security will continue to be able to parse the message.

6.2.2.3 Warning message format in ETWS

Caution is necessary when applying PWS security to ETWS.

ETWS is different from the other warning systems in that it consists of two separate warning types with different delivery requirements.

- The ETWS Primary Notification only contains the most urgent information such as warning type (e.g. Earthquake) and shall be delivered to the UE within 4 seconds (see TS 22.268 [2]).

- The ETWS Secondary Notification contains more detailed textual information such as seismic intensity, epicentre, etc. and does not have the same requirement on the delivery time; but obviously it should be reasonably short.

The ETWS Secondary Notification uses the same message format and delivery mechanism as the other warning systems in GERAN and UTRAN. In E-UTRAN the only difference is that the ETWS Secondary Notification is distributed using SIB 11 while the other warning systems use SIB 12. However, these two SIBs are almost identical in format and have similar length restrictions. It is therefore possible to add a signature field to the ETWS Secondary Notification as well.

The ETWS Primary Notification on the other hand differs significantly from the other warning messages. In order to meet the delivery time requirement, the ETWS Primary Notification uses both a shorter message format and a faster delivery mechanism.

- In GERAN, the ETWS Primary Notification is broadcasted in a cell using one or more paging messages. Due to requirements in earlier versions of ETWS, the Primary Notification in GSM already contains a security information field but unfortunately it is too short for the type of signature and security parameters that are considered here. Increasing the length of the security information is problematic since there is a risk of exceeding the delay limit as more bytes get added to the message. According to the analysis in the GERAN2 LS reply GP-111304, the length of the security information field can be extended from today's 50 bytes to a maximum of 75 bytes within the maximum delay limit of 4 seconds.

- In UTRAN, the ETWS Primary Notification is delivered using the RRC message ETWS PRIMARY NOTIFICATION WITH SECURITY. This message contains an optional security information field with the same length and contents as the one in GERAN. According to the RAN2 LS reply R2-114814, the security information field can be extended to fit a longer signature without violating any size or delay requirements. The LS reply does not, however, provide any information on the maximum size of the security information.

- In E-UTRAN, the ETWS Primary Notification is broadcasted in a cell using SIB 10. This message also contains an optional security information field with the same length and contents as the one in GERAN. According to the RAN2 LS reply R2-114814, the security information field can be extended to fit a longer signature without

violating any size or delay requirements. The maximum length of the security information field is estimated to be 210 bytes.

NOTE: After that RAN2 sent their LS reply the ETWS with security feature has been invalidated in UTRAN and E-UTRAN (see RP-130228). The security information field in ETWS PRIMARY NOTIFICATION WITH SECURITY (UTRAN) and SIB10 (E-UTRAN) was removed and replaced with a dummy field of the same length. Furthermore, in UTRAN the procedure for delivering ETWS PRIMARY NOTIFICATION WITH SECURITY to UEs in idle mode and connected mode URA_PCH, CELL_PCH, and CELL_FACH state was removed. To enable PWS security, both the UTRAN and E-UTRAN security information field and the UTRAN delivery procedure have to be re-introduced.

6.2.2.4 Conclusion on signature length

It is desirable to have common security solution for all warnings systems (ETWS, CMAS, KPAS, and EU-Alert) and all access technologies (GERAN, UTRAN, and E-UTRAN). It is clear from the analysis above that the problematic case is ETWS Primary Notification over GERAN.

If the security solution is going to support ETWS Primary Notifications over GERAN then the total length of the signature and related security parameters cannot exceed 75 bytes. This limit rules out the possibility of including a certificate with the signed warning message, even when the certificate is stripped down to a bare minimum and only includes the subject public key and the issuer signature. However, so called implicit certificates can meet this length restriction at the expense of limiting the security level to 112 bits. Furthermore, the length limit also implies that RSA cannot be used as signature algorithm. Recall that the length of an RSA signature is equal to the length of the RSA key, which at the 128 bit security level is 3072/8=384 bytes long.

If ETWS Primary Notifications only need to be supported in UTRAN and E-UTRAN or not supported at all, then there is significantly more space available for the signature.

NOTE: The profiling of the signature algorithm should take the above limitations into account. Depending on the key distribution method chosen, the profiling may also need to pay attention to the size of the key (which otherwise may induce too much data sent over the air-interface). Further limits may also be identified. The intention is to later ask SAGE for the best algorithm profiling that fulfils these limitations.

Editor's Note: Whether a solution for PWS security is required to support protection of ETWS Primary Notification over GERAN is ffs.

6.2.3 Algorithm agility of PWS

6.2.3.1 General

The network should indicate to UE which algorithm to be used. By this way, UE can obtain signature algorithm and know which signature key should be used to verify the signature of PWS Warning Notifications.

Editor's Note: It should avoid negotiation of security information during PWS warning.

An n-bit identifier is allocated to identify the signature algorithm with the following algorithm defined in table 6.2.3.1.

Table 6.2.3.1 Signature algorithms

Value	Signature algorithm
0	128-ECDSA
1	128-DSA
2-2^n	For further use

It has been agreed to limit the number of standardized algorithms to at most the two algorithms listed above.
If companies or governments wants to use the "For further use"-range, the registration of new signature algorithms **"should"** be handled and approved by 3GPP.

Editor's Note: The number of bits in the signature algorithm identifier is FFS.

Editor's Note: It is FFS is the number of standardized algorithms should be narrowed down to only a single algorithm.

6.2.3.2 ECDSA domain parameters

DSA and ECDSA private/public key pairs are generated with respect to a particular set of domain parameters.
NIST [10] states that although domain parameters may be common to a group of users and may be public information, they **"should"** be managed so that the correct correspondence between a given key pair and its set of domain parameters is maintained for all parties that use the key pair. A set of domain parameters may remain fixed for an extended time period. The goal of this clause is to estimate the minimum number of bits needed to transfer domain parameters. ECDSA uses significantly smaller public key sizes compared to DSA.

ECDSA is defined for two arithmetic fields: prime and binary field (representations). In the following only details of ECDSA domain parameters are investigated.

NOTE 1: Similar calculations could be done for DSA and ECDSA over binary fields

Domain parameters for ECDSA in the general case (see [10], clause 6.1)) are of the form *(q, FR, a, b {, domain_parameter_seed}, G, n, h)* where *q* is the field size; *FR* is an indication of the representation used (prime or binary); *a* and *b* are two field elements that define the equation of the curve; *domain_parameter_seed* is the domain parameter seed and is an optional bit string; *G* is a base point of prime order on the curve (i.e., $G = (G_x, G_y)$); *n* is a prime number and the order of the point *G*, and *h* is the cofactor (which is equal to the order of the curve divided by *n*).

The optional *domain_parameter_seed* is needed to validate that the primes were generated correctly. For the purpose of public key distribution in PWS, it is assumed here that the UE does not need to validate these primes. Therefore, it is assumed that, in prime representation, for the verification of the signature the mandatory six domain parameters

$$(q, a, b, G, n, h)$$

need to be known by the terminal. In the following, those are further analysed based on NIST recommended elliptic curves in [10], Annex D.1.

NOTE 2: These are examples; the elliptic curves selected for PWS Security by SAGE may be different.

q is equal to the length of *p* (prime number). NIST lists for each prime *p*, a pseudo-random curve of prime order *n*.
For these curves, the cofactor is always $h = 1$. NIST makes for reasons of efficiency the selection $a \equiv -3$.
Thus, if the same values are used, *h* and *a* can be fixed and do not need to be provisioned to the terminal.
For the field *GF(p)*, the security strength is dependent on the length of the binary expansion of *p*, which is 256 as stated

in [10, Table D-1]. *n* can then have a length between 256 and 383, this is the range recommended for 128 bit security strength as specified in [10, Table 1]. For the purpose of calculating the minimal bit length necessary to provision to the terminal, the minimum for *n*, i.e. 256 bit, is assumed.

If *h* and *a* are fixed, only *q*, *b*, *G*, *and n* of the set *(q, a, b, G, n, h)* need to be provisioned, i.e., 256 bits for *p*, 256 bits for *b*, 256 bits for G_x and 1bit for G_y (assuming that the terminal could calculate *y* fast enough itself for determining $G = (G_x,G_y)$), and n = 256 bits.

Thus, with the above assumptions a minimum 1025 bits would be needed to transfer domain parameters for a 128 bit strength valid set of ECDSA domain parameters.

In summary, from a pure message size point of view it seems possible that a domain parameter set would be distributed with the public key. Even in 2G NAS messages (except CS) it should be possible to send one domain parameter set with the public key. However, pre-provisioning one or several domain parameter sets in the UE when new terminals are rolled out would be a more efficient approach for PWS. In particular, in a situation that needs a fast change of the public key this could be of advantage as, even without the distribution of domain parameters, the extra NAS signalling in case of a public key change could easily lead to an overload of the MME or the radio link.

One could also think about an update mechanism for NAS similar to those for encryption algorithms.
Newly standardized encryption algorithms are not downloaded to terminals but just implemented in new terminals.

Editor's Note: It is FFS if additional parameters, e.g. the so-called domain parameters [10], are necessary to send or negotiate together with the distribution of the public keys in order to allow the UE to verify a signature, or whether these parameters are globally standardised, and, if so, how many sets of such parameters.

Editor's Note: Domain parameter sets pre-provisioned in newly rolled-out terminals is clearly preferable from a protocol point of view. But this needs to be checked with regulators. It needs to be decided whether one or more domain parameter sets need to be provisioned.

Editor's Note: It is for further study how feedback from regulators operating the PWS signing entities is gained and if the standardisation of a limited number of domain parameter sets by 3GPP is acceptable and if yes, how many sets would be needed.

6.2.4 Security level and key length of signature algorithms proposed

The following clause summarizes the recommendations from NIST ([10], [11] and [12]).

The security level or security strength is presented by the number of bits of security, which relates to the number of operations required to break a cryptographic algorithm or system. The recommendations for parameter choices regarding a certain security level are shown in Table 6.2.4.1 (extract from [11], clause 5.6, Table 2) and are combined with information about the security strength time frame (extract from [11], clause 5.6.2, Table 4) and Table 1 of [12].

Column 2 (FFC) indicates the minimum size of the parameters associated with the standards that use finite-field cryptography (FFC). The largest key size approved in [10] is (L = 3072, N = 256).

Column 3 (ECC) indicates the range of f (the size of n, where n is the order of the base point G) for algorithms based on elliptic-curve cryptography (ECC). The value of f is commonly considered to be the key size.

Table 6.2.4.1 Parameter choices for DSA and ECDSA of comparable security strength

Bits of Security (and security strength time frames)	FFC (e.g. DSA) (in bits)	ECC (e.g. ECDSA) (in bits)	Signature lengths (in bits)
80 (applying is deprecated till 2013, afterwards disallowed; processing for legacy use also in 2031 and beyond)	L = 1024 N = 160	f = 160-223	DSA: 320 ECDSA: 320-446
112 (applying is acceptable till 2030, disallowed in 2031 and beyond; processing is acceptable till 2030, for legacy use also in 2031 and beyond)	L = 2048 N = 224, 256	*f* = 224-255	DSA: 448 ECDSA: 448-510
128 (Applying and processing acceptable up to, including and beyond 2031)	L = 3072 N = 256	*f* = 256-383	DSA: 512 ECDSA: 512-766

Hash functions that **"should"** be used for providing 128 bit security strength for the generation of digital signatures for signature algorithms are SHA-256, SHA-512/256, SHA-384, SHA-512 (extracted from [11], clause 5.6.1, Table 3).

DSA

For DSA, the key size is considered to be the size of the modulus p, where L is the bit length of p. The length of the digital signature (s,r) is twice the size of q, where N be the bit length of q. DSA will produce digital signatures of 320, 448, or 512 bits depending on the security level ([11], clause 4.2.4.1). Note, the largest key size approved in [10] is (L = 3072, N = 256) ([11], clause 5.6.1).

NIST specifies choices for the pair L and N. According to the recommendation ([10], clause 4.2), a Federal Government entity other than a Certification Authority (CA) should use only the following (L, N) pairs (1024, 160), (2048, 224) and (2048, 256). A CA **"should"** use an (L, N) pair that is equal to or greater than the (L, N) pairs used by its subscribers.

The security strength associated with the DSA digital signature process is no greater than the minimum of the security strength of the (*L*, *N*) pair and the security strength of the hash function that is employed ([10], clause 4.2).

It is recommended for DSA that the security strength of the (L, N) pair and the security strength of the hash function used for the generation of digital signatures be the same unless an agreement has been made between participating entities to use a stronger hash function ([10], clause 4.2).

ECDSA

ECDSA specifies a minimum key size of 160 bits and produces digital signatures that are twice the length of the key size ([11], clause 4.2.4.3). For elliptic curves, the key size is the length f.

NIST Recommended curves are provided in Appendix D of [10] ([10], clause 6.1.1)

The security strength associated with the ECDSA digital signature process is no greater than the minimum of the security strength associated with the bit length of the key and the security strength of the hash function that is employed ([10], clause 6.4).

It is recommended that the security strength associated with the bit length of *n* and the security strength of the hash function be the same unless an agreement has been made between participating entities to use a stronger hash function ([10], clause 6.4).

Editor's Note: It needs to be verified whether broadcast channels on all radio interfaces envisaged for PWS can cope with a signature length corresponding to this security strength, e.g. with a signature length of 512 bits if the security strength is 128 bits.

6.2.5 Verification of PWS Warning Notification message

6.2.5.1 General

The UE **"should"** support the verification of the signature attached to a PWS Warning Notification in case PWS security is applied in a regulatory domain. How to verify PWS Warning Notifications when integrity protected **"should"** be solved. By this way, UE can verify whether the message comes from an authenticated authorized source and whether the messages have been modified maliciously.

The UE **"should"** support a USIM data file with two settings to disable the PWS functionality as specified in TS 22.268 (for requirements and detailed handling of the USIM data file see clause 4.6.4 on Enabling and disabling of Warning Notifications in TS 22.268 [2]).

Editor's Note: The impacts of sending more than one signature to the UE and if this solves the overload problem is FFS.

If UEs cannot receive public keys from the network through any form of signalling or user plane interaction, e.g. when the UE is in limited service state, the required information for verifying signed warning messages has to be provided by other means, e.g. through various forms of previous interactions between UE and network. The required information would be available at least for the implicit-certificate-based approach (solution 6 in the present document) where root CA public keys are installed in the UE at manufacturing time or when the UE is switched on for the first time, and the CBE public keys are implicitly distributed by broadcast as part of the warning message.

Editor's Note: For other solutions in clause 7 of the present TR, public key distribution in a situation where the UEs cannot receive public keys from the network through any form of signalling or user plane interaction is ffs.

6.2.5.2 Handling of Warning Notifications without signature

As PWS Security is an optional feature and several regions (US, Japan) have made clear that broadcast of signed Warning Notifications are unlikely, PWS Security may be deployed locally but not globally.
A UE supporting PWS Security will likely encounter genuine Warning Notifications without signature from PWS.

As current deployments of PWS (e.g.: CMAS, EU-ALERT and ETWS) do not have any integrity protection on the broadcast Warning Notifications, a UE cannot determine whether an unsigned Warning Notification is genuine or false. In regions where the Warning Notifications are broadcast without integrity protection, a false BS could be setup for all RATs (GERAN, UTRAN, EUTRAN). In addition, in regions where signed Warning Notifications are broadcast but a PWS Security enabled UE is allowed to display also unsigned Warning Notifications in the honest attempt to display possibly genuine Warning Notifications, the false BS scenario is also a threat to those regions. PWS circumvention attacks are possible.

The trivial solution to discard all Warning Notifications without signature (irrespective whether they are genuine or false) is secure and robust, but it is not clear if this would be acceptable from a safety perspective. Other potential solutions include network-independent location verification and the usage of a UE-controlled timer.
For details refer to clause 7.9.

In summary, as the objective of PWS Security is to protect against false BSs, displaying unsigned Warning Notifications (i.e. without signature) irrespective whether they are genuine or false would make PWS Security worthless.

A robust mechanism to securely distinguish between genuine and false Warning Notifications without signature could make PWS Security more acceptable from an availability and safety perspective. Proposed mechanisms are discussed and described in clause 7.9.

6.2.6 Primary and secondary notifications

To achieve immediate distribution of the highest priority information, ETWS specifies delivery of emergency information in two different notifications:

- The primary notification only contains the most urgent information such as warning type (e.g. Earthquake). When receiving a primary notification the UE sounds an alarm sound and displays a pre-determined warning message on screen.

- The secondary notification contains more detailed textual information such as seismic intensity, epicentre, etc. When receiving a secondary notification, the UE simply displays the information on screen.

In case of an earthquake, a UE will typically receive the primary notification several seconds before receiving the secondary notification. When receiving a primary notification, the user has no way of knowing the magnitude of the earthquake, as this information is only included in the secondary notification. And as the magnitude and epicentre is typically not known when sending out the primary notification, users receive primary notifications also for relatively non-serious earthquakes.

An adversary wanting to cause panic might therefore start sending out false notifications. While a false primary notification might only signal "Tsunami", causing people to run to shelter, a false secondary notification might falsify the magnitude of a minor earthquake "Earthquake, Magnitude 9.7" or instruct people to take hazardous or even fatal actions "Drink chlorine bleach to prevent radiation damage". While the above could also be done by e-mail, the impact is likely to be much higher when received through a trusted warning system.

It is therefore important that all notifications carrying warning information are equally protected.

6.2.7 Network sharing impact to PWS Security

6.2.7.1 General

In both GWCN and MOCN configuration for network sharing types, there is no impact to PWS Security in GSM, UMTS and EPS when using the current solutions, i.e. NAS messages to distribute public key and CBC to distribute the signature as long as there is no material in the signature specific to any of the operators.

It **"should"** be ensured that the security constructs in PWS do not cause problems with network sharing, for example, key derivations and signatures should preferably not be dependent on areas, network identities and the like.

6.2.7.2 GWCN configuration

GWCN applies for EPS and UMTS, not for GSM according to TS 23.251 [6] Network Sharing. In EPS and UMTS sharing network, a supporting UE decodes the shared network information and supplies the available core network operator PLMN-ids as candidates to the PLMN selection procedure. The UE performs network selection among available PLMNs. The UE sends an ATTACH REQUEST message to the network entity indicating the chosen core network operator. Then the shared MME/SGSN determines whether the UE is allowed to attach or not and sends the appropriate ACCEPT/REJECT message back to the UE. If successful, a supporting UE has attached to the selected shared network.

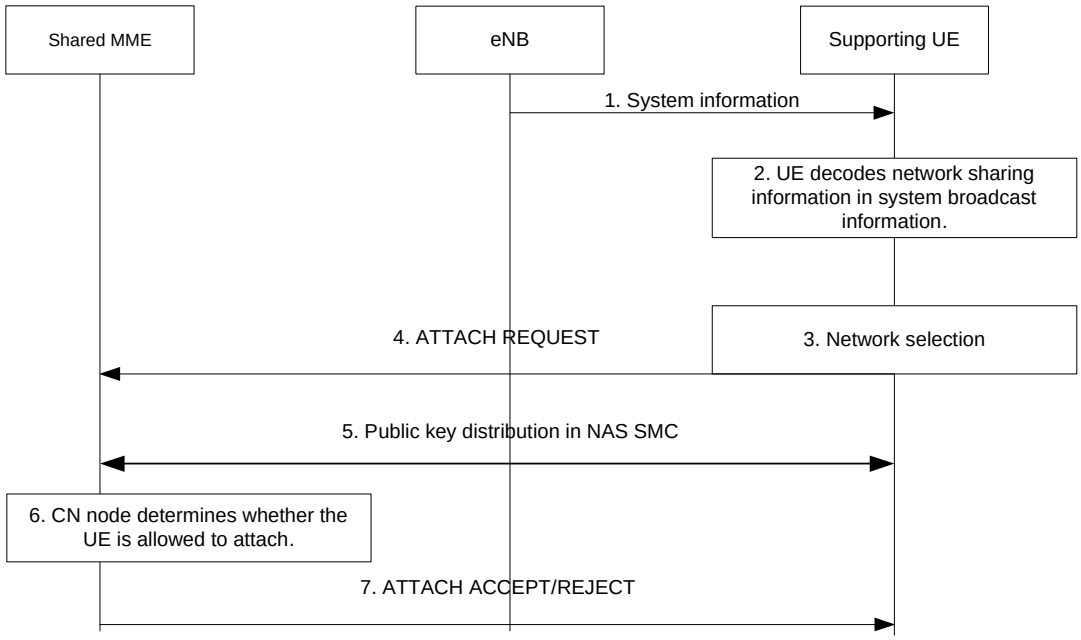

Figure 6.2.7.2.1: Example of network selection in GWCN configuration for a supporting UE in a shared EPS network for PWS public key distribution

Figure 6.2.7.2.1 shows that network selection procedure in GWCN configuration for network sharing has no impact to public key distribution in NAS SMC for PWS. Moreover, network sharing is an agreement between operators and **"should"** be transparent to the user. This implies that a supporting UE needs to be able to discriminate between core network operators available in a shared radio access network and that these operators can be handled in the same way as operators in non-shared networks. This also means that there is no impact for PWS public key distribution, provided there is no operator-specific material for PWS keys that differentiate the sharing operators.

With regard to PWS signature distribution procedure for GWCN configuration, since the pre-condition defined for network sharing in CBS is using only one single common CBC, CBE always contacts this CBC to broadcast warning messages including signature etc. security information. This single common CBC will use "impacted area" information received from CBE to know which core network entity (EPS) or radio network entity (UMTS) to contact. The following procedure is the same as the normal one in non-shared network. So there is no impact for PWS signature distribution in GWCN configuration, provided there is no operator specific material in the signature.

For GWCN configuration, UMTS have the same situation like EPS does. So there is also no impact of network sharing for UMTS PWS services.

6.2.7.3 MOCN configuration

MOCN applies for all the system, i.e. GSM, UMTS and EPS. In sharing network, a supporting UE decodes the shared network information and supplies the available core network operator PLMN-ids as candidates to the PLMN selection procedure. The UE performs network selection among available PLMNs. The UE sends an ATTACH REQUEST message to the network. It also indicates to the radio access node the chosen core network operator.
The eNB/RNC/BSC uses the routing information to determine which core network operator the message should be routed to and the ATTACH REQUEST message is sent to the core network operator chosen by the UE.
The core network determines whether the UE is allowed to attach to the network. The shared core network node sends the appropriate ACCEPT/REJECT message back to the UE. If successful, a supporting UE has attached to the selected shared network.

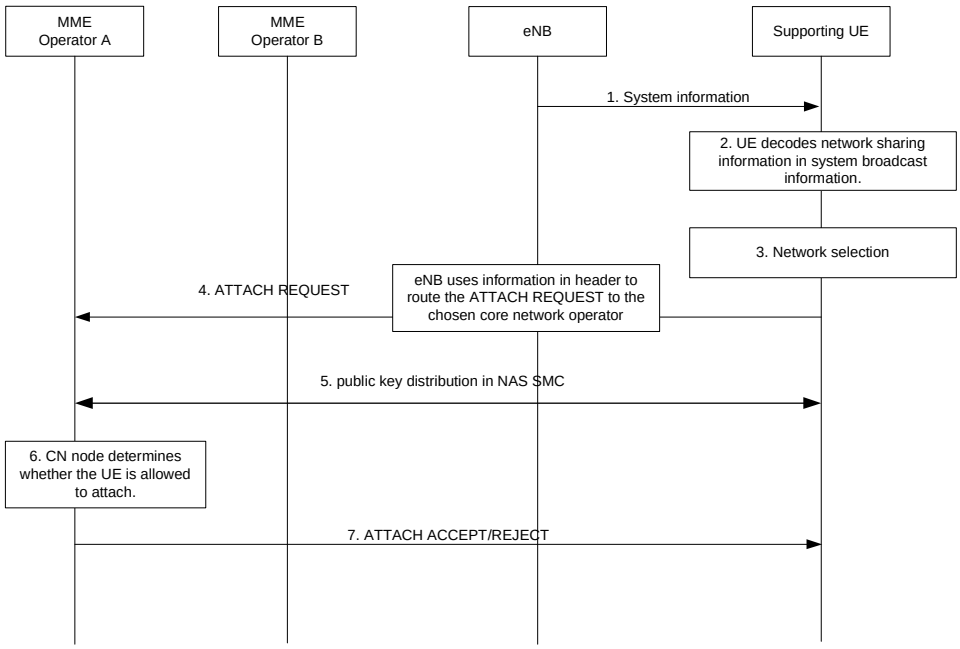

Figure 6.2.7.3.1: Example of network selection in MOCN configuration for a supporting UE in a shared EPS network for PWS public key distribution

Figure 6.2.7.2.1 shows that network selection procedure in MOCN configuration for network sharing has no impact to public key distribution in NAS SMC for PWS. Similar like the analysis for GWCN case, there is also no impact for PWS public key distribution in MOCN case as long as keying material is not specific to any of the sharing operators.

With regard to PWS signature distribution procedure for MOCN configuration, since the pre-condition defined for network sharing in CBS is using only one single common CBC, CBE always contacts this CBC to broadcast warning messages including its signature which is out of operator control. This single common CBC will use "impacted area" information received from CBE to know which core network entity (EPS) or radio network entity (GSM and UMTS) to contact. The following procedure is the same as the normal one in non-shared network. So there is no impact for PWS signature distribution in MOCN configuration, provided there is no operator specific material in the signature.

For MOCN configuration, GSM and UMTS have the same situation like EPS does. So there is also no impact of network sharing for GSM and UMTS PWS services.

6.2.8 Triggering condition for public key update

There are two scenarios for public key update, i.e. when the signing entity is changed or the signing entity provides new public key to the network. The scenario of triggering public key update could be infrequent.

When the signing entity is changed, the public key is changed and it should be updated. It happens when UE roams to a new cell which belongs to another PLMN with a different CBE. The new PLMN can be associated to the same CBE that the old one is and it can also be the different CBE. If this new PLMN is associated with a different CBE, the UE should use the public key provided by the new CBE to verify PWS signature. In this public key updating scenario, both the PLMN serving the UE and the signing entity-CBE have been changed after UE HO to a new cell.

There can be several reasons for signing entity providing new public key to the network, including end of public key lifetime, compromised public key and replay protection, etc. ECDSA/DSA with 128 bit security level is considered secure enough beyond year 2030 according to NIST recommendations. Furthermore, each government will be very careful to send such kind of public warning. So the public keys for PWS signature could be seen as safe for a considerable period of time. However, there remains requirement for PWS public key update for some reasons e.g. private key leakage, key management hole etc.

There may also be policy reasons for changing the key. For example, the key lifetime of the key may expire.
The lifetime of the key may be in the order of months or years.

PWS keys should not be updated immediately after a Warning Notification has been sent. Otherwise there is a risk that secondary warning messages will be rejected either because the UE already has the new key while the warning messages are still signed with the old key, or the UE still has the old key while the warning messages are already signed with the new key.

6.2.9 Roaming impact to PWS Security

A roaming UE can attach to VPLMN network or perform TAU/RAU/LAU/handover from HPLMN to VPLMN network. Since different PLMN may connect to different CBE/CBC, a roaming UE need to initiate PWS key updating to get latest PWS key of the VPLMN. Then UE can use the latest PWS key to verify Warning Notifications broadcasted by the VPLMN.

- Case1: UE attaches to VPLMN network, PWS key can be distributed to UE via SMC or attach accept message by VPLMN.
- Case 2: UE is in idle state and performs TAU/RAU/LAU to VPLMN network, PWS key can be distributed to UE via TAU/RAU/LAU accept message by VPLMN.
- Case 3: UE is in PS-Connected state and performs handover to VPLMN network:
  - When UE handovers to LTE network, UE will initiate TAU just after handover procedure and PWS key can be distributed to UE via TAU accept message by LTE network.
  - When UE handovers to UMTS PS domain, UE will initiate RAU just after handover procedure and PWS key can be distributed to UE via RAU accept message by UMTS network.
  - When UE handovers to GERAN PS domain, UE will initiate RAU just after handover procedure and PWS key can be distributed to UE via RAU accept message by GERAN network.
  - When UE SRVCC handovers to UMTS/GERAN CS domain, then UE has to wait for CS service terminated and then performs LAU at once. Thus UE cannot obtain PWS key in time via LAU accept message since the duration of CS service is uncertain.
- Case 4: UE is in CS-Connected state and performs handover to VPLMN network:
  - When UE handovers to UMTS CS domain, then UE has to wait for CS service terminated and then performs LAU at once. Thus UE cannot obtain PWS key in time via LAU accept message since the duration of CS service is uncertain.
  - When UE handovers to GERAN CS domain, then UE has to wait for CS service terminated and then performs LAU at once. Thus UE cannot obtain PWS key in time via LAU accept message since the duration of CS service is uncertain.
  - When UE rSRVCC handovers to LTE/HSPA PS domain, UE will initiate RAU just after handover procedure and PWS key can be distributed to UE via RAU accept message by LTE/HSPA network.

As mentioned above, UE cannot obtain PWS key in time in the following scenarios:

- Scenario 1: If UE is in CS-Connected state and it handovers to UMTS CS domain, UE has to wait for CS service terminated and then performs LAU at once to update PWS key.
- Scenario 2: If UE is in CS-Connected state and it handovers to GERAN CS domain, UE has to wait for CS service terminated and then performs LAU at once to update PWS key.
- Scenario 3: If UE is in PS-Connected state and it SRVCC handovers to UMTS/GERAN CS domain, UE has to wait for CS service terminated and then performs LAU at once to update PWS key.

In above three scenarios, UE cannot obtain PWS key via LAU accept message in time. Since the duration of CS service is uncertain, if UE receives CBS warning message from VPLMN during the duration, it cannot verify the warning message by using latest PWS key.

But in TS 23.041 [3] clause 2, the following text and table is described.

'Reception of CBS messages for an MS/UE is not a requirement if it is connected in the CS domain. It should be possible for an MS/UE to receive messages if it is connected in the PS domain and no data is currently transmitted.'

CS-Domain	CS-Connected	CS-Idle	CS-Idle
PS-Domain	-	PS-Idle	PS-Connected
Reception of CBS Message	Not possible	Possible	Depends on RRC mode

In GSM system, when UE is in CS-Connected mode and CS service is on going, UE can only receive and decode messages on TCH channel. But CBS message is delivered on CBCH channel. It is not a requirement for UE to listen to both TCH and CBCH in CS-Connected state. So it is not a requirement for UE to receive CBS message in CS-Connected mode.

In UMTS system, when UE is in CS-connected mode and CS service is on going, UE is in CELL-DCH state.
In CELL-DCH state, UE can only receive and decode messages on physical channel DPDCH/DPCCH. But CBS message is delivered on CTCH channel which is mapped to physical channel S-CCPCH. It is not a requirement for UE to listen to both DPDCH/DPCCH and S-CCPCH in CELL-DCH state. So it is not a requirement for UE to receive CBS message in CS-Connected mode.

In case the UE is in PS-Connected mode it depends on the Radio Resource Control State whether reception of CBS messages is possible. If UE is in PS-Connected mode and no data is currently transmitted, UE is in CELL_PCH or URA_PCH state. In CELL_PCH or URA_PCH state, UE can listen to physical channel S-CCPCH and receive CBS message. If UE is in PS-Connected mode and some data is currently transmitted, UE is in CELL_DCH or CELL_FACH state (depends on algorithm of network side). In CELL_DCH state, UE can not listen to physical channel S-CCPCH and can not receive CBS message as described above. In CELL_FACH state, UE can transfer data and receive CBS message.

Based on above analysis, it is not possible for a CS-Connected UE to receive CBS warning message. Even if UE has to wait for CS service terminated and then performs LAU at once to update PWS key, there is no impact on PWS key updating and warning message verification.

In conclusion, there is no new security requirement needed for PWS key distribution and warning message verification in roaming case on both PS domain and CS domain.

6.2.10 Discussion on parameters to be sent when distributing public keys or broadcasting warning messages

6.2.10.1 Public Key Identifier (PKID)

It was suggested in several solutions to send the public key and the next public key. The advantage of keeping two keys of one signing entity at the terminal are: to facilitate the verification of a signed warning message by the UE during the period of a key change and, particularly interesting for the network operator, to reduce overload as this mechanism allows extending the key change over a longer period. If a new public key needs to be distributed, it is not possible for e.g. the MME to do this at once, a longer period is needed in which step by step public keys can get renewed.
Thus, if the terminal already holds two keys, an indicator would be sufficient to notify the terminal to use the next public key. If indicated, the terminal knows it needs to deactivate the old public key, to use the next public key as the active public key and to request within a certain timeframe a new "next public key".

There are two dimensions, on which the number of public keys in the same MME/SGSN/MSC region depends:
The signing entity could decide to supply one or two public keys (active and next) and/or there could be more than one signing entity which wants to broadcast in the same MME/SGSN/MSC region. The latter multiplies the number of public keys by the number of signing entities within one region. Thus, a public key identifier may be needed to indicate in a warning message which public key the terminal should use. However, if it is easier for the UE to figure out itself, which is the active public key, an identifier may not be needed at all in the warning message.

The introduction of a Public Key Identifier (PKID) allows the UE to determine the active public key to be used for verifying a warning message very fast. If different public keys are used among several signing entities as described above, a public key identifier could be useful as well. Furthermore, if the UE wants to indicate at registration time which public keys it has available and which one is the active one, the PKID could be used in the warning message.

Furthermore, for all NAS-based solutions the PKID is needed so that the MME/SGSN/MSC can tell from the PKID whether the UE already has the correct public key(s) or whether the public key(s) need to be distributed to the UE. In the former, only a short ACK to the UE is needed which saves a lot of bandwidth.

What could be an acceptable identifier for a public key? Its usage purpose implies a worldwide unique identification. To achieve this, a structure is needed. A regulator may want to be able to act independently in assigning a part of the PKID within its authority. Thus, one could imagine a representation of a public key identifier that breaks it down to national level, e.g. a PKID could include "country + public key number" or "regulator ID + public key number" or "Region + public key number". E.g. a regulator ID could be worldwide, European, country, state, or region specific. At minimum one authority should be able to assign a public key number without conflicting with other authorities. Furthermore the current active and the next public key need to be identifiable as well. Thus, a PKID could have an additional bit which indicates "active" or "next".

Editor's Note: It is for further study, how the structure of the PKID looks like.

The consideration on PKID is similar for NAS-based and GBA-based solutions. For the IMPCERT solution no identification mechanism is needed since the public key can be reconstructed from the implicit certificate that is always broadcast together with the warning message.

6.2.10.2 Signing entity identifier

A signing entity identifier is an identifier proposed in one of the NAS-based solutions. If the public key has been securely distributed to UEs, a PKID should be enough for authentication. No extra signing entity identifier is needed.

However, in case there is still a need to identify the signing entity, a standardized generic structure how to present this identity should be in place. One could imagine a similar structure as done, e.g., for the PKID.

The signing entity identifier would be used to identify a CBE. In case a signing proxy is in place, it needs to be identified accordingly. In this case, CBE identification would most likely not be needed.

Editor's Note: It is for further study if a signing entity identifier is needed.

6.2.10.3 Signature Algorithm Identifier (SAI)

If more than one signature algorithm is standardized, the Signature Algorithm Identifier (SAI) may be useful to ease the verification mechanism for the UE. An x-bit indicator could tell whether to use ECDSA or DSA or another algorithm,

and which set of domain parameters (see 6.2.3.2). In general, the number of algorithms should be limited as much as possible. If domain parameters are pre-installed, this identifier is not needed since the indication of the set of domain parameters implies which algorithm is to use.

6.2.10.4 Domain parameters

As discussed in clause 6.2.3.2 domain parameters are necessary to indicate the correct parameters, e.g., curve parameters for ECDSA. If not pre-installed, domain parameters would be sent and may be then combined with the indication for the signature algorithm identifier.

6.2.10.5 Domain set indicator

If it was possible to standardize and pre-install a few sets of domain parameters, a domain set indicator should be used to signal with which set a distributed public key is associated, i.e., an x-bit indicator would be needed that would be sent together with the public key.

Editor's Note: It is for further study if a domain set indicator would be acceptable as the most sufficient way of indicating pre-provisioned parameters for signature verification.

6.2.10.6 Hash function indicator

Currently all NIST recommended ECDSA curves use SHA-1 as the hash algorithm. However, in future maybe also other hash algorithms may be in use to avoid collisions. Therefore an indicator for 'which hash function to use' should be available to the UE. If the hash function is known due to the algorithm used, this parameter is not needed as it would be considered to be included in the signature algorithm identity.

6.2.10.7 Network Security Use Counter (NSUC)

The Network Security Use Counter (NSUC) was introduced as a countermeasure to replay attacks. The NSUC is used by three entities: CBE signs the warning message including the NSUC and monotonically increases the value of NSUC each time the public key is used, i.e. a fresh signed warning message is generated by the CBE. Whenever the CBE increases the NSUC the updated value of NSUC, together with the PKID, is communicated to the MME/SGSN/MSC. UE sends the NSUC to the network, such that the MME can decide whether it has to send an updated NSUC to the UE.

This counter or another freshness indicator needs to be used in all proposed public key distribution solutions, but the solutions for the IMPCERT approach and the GBA approach are FFS.

6.2.10.8 Time stamp (Void)

Void

6.2.10.9 CA ID (Void)

Void

6.2.10.10 Conclusion

Above, all parameters included in NAS-based, GBA-based and IMPCERT-based solutions have been discussed. The following can be concluded.

NAS-based

It is suggested that the following parameters are sent to the UE with the public key:

- PKID and if used, next PKID
- NSUC
- Signature algorithm and/or domain set indicator as follows

- Domain set indicator or SAI, if domain parameters are pre-installed (uniquely associated with the algorithm) and there is more than one domain parameter sets, or
- Domain parameters or both Domain parameters and SAI, if domain parameters are not pre-installed in terminal

It is suggested that the following parameters are sent with the broadcast message, i.e. the signed warning message:

- NSUC
- PKID

Parameters that need to be sent by the UE in a NAS message:

- All relevant PKIDs it has available (it may make sense to keep old PKIDs stored while roaming)
- NSUC, if not always sent by the network

GBA-based

Editor's Note: Parameters are FFS.

IMPCERT solution

For the IMPCERT solution the public key can be reconstructed from the implicit certificate, which is sent together with the warning message. Therefore PKID are not needed.

It is suggested that the following parameters are sent with the implicit certificate broadcast message:

- Time stamp or another par. for replay protection should be included before signature generation
- Signature algorithm and/or domain set indicator as above if not limited to one with the implicit certificate approach
- CA-ID

NOTE: In the implicit certificate approach, there is the broadcast channel only, i.e. the UE cannot indicate any parameters.

6.2.11 Considerations on networks in disaster areas

In disaster areas, sometimes 'networks in a box' are deployed to enable local or regional communication when the regular infrastructure does not function any more after serious damage through the disaster. In addition to communication, the networks in a box may broadcast vital information in disaster areas e.g. location of shelter, fresh water, refugee camps, and status of roads. The networks in a box may also provide further warning messages e.g. for flooding or provide "all clear" messages.

It needs further study in how far such networks can support PWS and PWS Security at all. Here some considerations:

- Case 1): The network in a disaster area has connection to a regular CBE.
  - Then the normal PWS and PWS Security procedures could apply.
- Case 2): The network in a disaster area has no connection to a regular CBE.
  - In case 2), some of the typical warning messages like earthquake warning would unlikely to be possible as these types of warnings have to rely on an extensive sensor and processing network for recognising imminent earthquake or tsunami threats. So, some of the warning messages most suitable for creating panic would not be sent in case 2).
  - In order to further reduce the potential for creating panic by unprotected false warning message, the UEs could be configured to selectively display warning messages when it could not verify the PWS Security status of the network.
    This selective display is supported already today by the requirements in clause 4.6.4 of TS 22.268 [2]:
    "*It shall be possible for users to **disable** (e.g., opt-out) presentation of **some or** all of the Warning Notifications, subject to regulatory requirements and/or operator policy. The user shall be able to select PWS-UE enabling/disabling options via the User Interface to **disable, or later enable**, the PWS-UE behaviour in response to **some or** all Warning Notifications*."
- Case 2a): PWS Security can still be applied; and unprotected warning messages could be discarded by the UE, if the network in the disaster area contains a CBE function ('local CBE') able to sign warning messages that can be verified with a public key available to the UE.

Editor's Note: In general, it is ffs how the distribution of the public key of the local CBE would work.

- Case 2b): PWS Security cannot be applied if the network does not contain a 'local CBE' with a public key, or if that public key cannot be distributed. Then there is only the alternative to either discard all warning messages or accept the risk of an attack creating panic through false warning messages (defeating the purpose of PWS Security).

Editor's Note: It needs further study how networks deployed in disaster areas can support PWS and PWS Security.

Editor's Note: Requirements are ffs as it may be difficult to provide a perfect security solution in such cases.

7 Possible Security solutions of PWS

7.0 General

The proposed solutions can be based in groups based on how they securely transfer - from the CBE to the UE - the public key that is needed for the signature verification of warning messages.

- Solution 3 and 5 use NAS for secure transfer, where solution 5 suggests a special mechanism for using NAS over GERAN.
- Solution 4 uses GBA for secure transfer, where NAS based transport is one of several options.
- Solution 6 and 7 use certificate-based approach for secure transfer. Solution 7 discusses general aspects while solution 6 discusses specific aspects of an implicit certificate- based approach.
- Solution 8 uses UICC OTA for secure transfer to the UICC.

- Solution 9 describes a signing proxy solution.

7.1 Void

NOTE: Solution 1 has been archived in Annex A.

7.2 Void

NOTE: Solution 2 has been archived in Annex A.

7.3 Solution 3: NAS based solution

7.3.1 General

This solution is for GSM, UMTS and LTE.

With regard to public key distribution procedure, NAS messages, e.g. TAU/RAU/LAU accept can be used to distribute public key which is also in solution 1 and 2 in Annex A. From previous SA3 meeting discussions, public key update should also be considered. LTE and UMTS can use similar procedures for public key distribution and update. For GSM PWS Security solution, it may be different from previous two systems. The present document gives some solutions and it depends on the SA3 meeting discussions and operators' choice for it.

NOTE: This solution is a merger of solution 1 and 2 which also includes some new improvements.

7.3.2 PWS public key distribution

7.3.2.1 Initial PWS public key distribution

For LTE system, NAS messages, i.e. NAS SMC/Attach accept/TAU accept are used to distribute public key. Specifically, NAS SMC message is used to distribute PWS public key when UE attaches to a PLMN for the first time. In addition, the Network Signature Use Counter (NSUC) which is a monotonic increasing value that can be increased every time the signing key is used.

For UMTS system, NAS and AS messages, i.e. SMC/Attach accept/RAU accept are used to distribute public key. Specifically, SMC message is used to distribute PWS public key when UE attaches to a PLMN for the first time. In addition, NSUC is sent. When UE has inter-PLMN handover, TAU accept/RAU accept are used to distribute new PLMN PWS public key and NSUC.

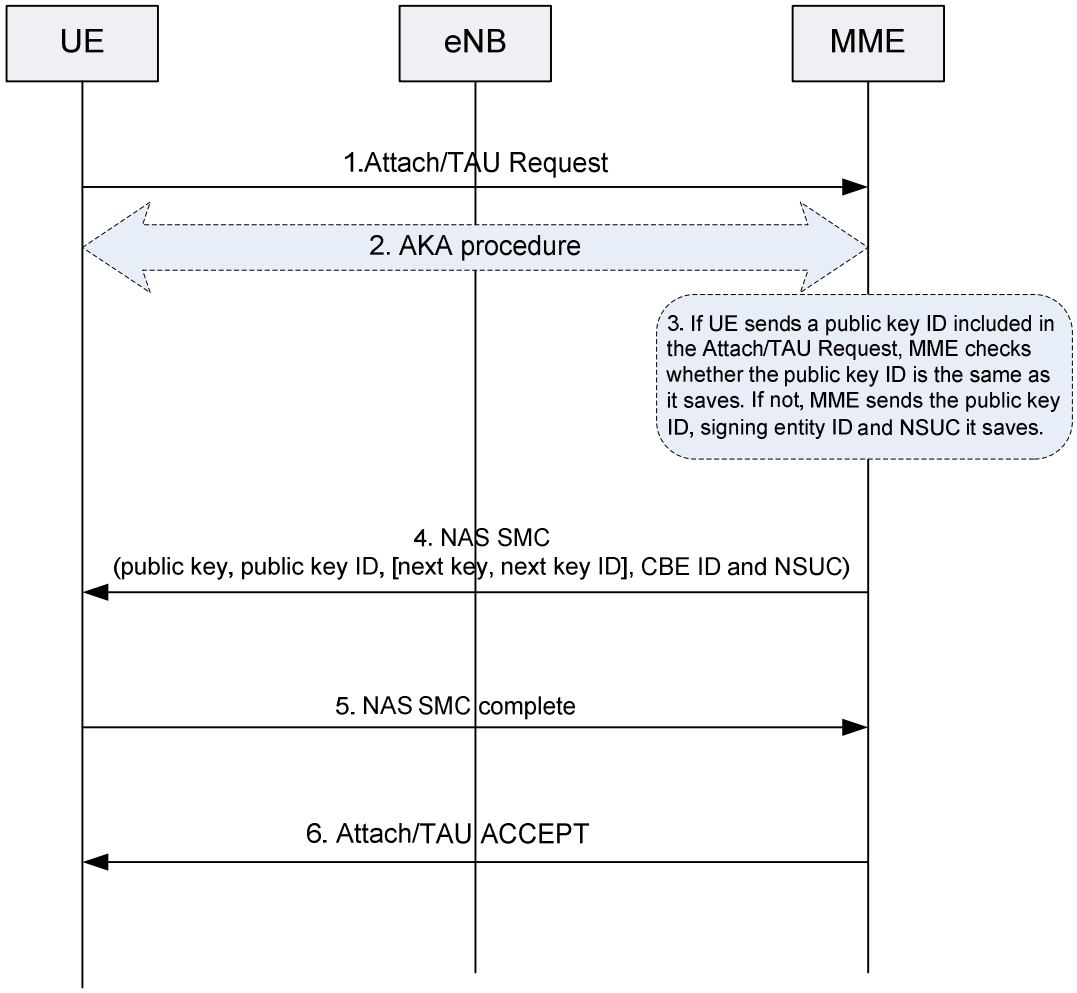

Figure 7.3.2.1.1 Initial distribution of PWS public key in LTE system

When MME receives the initial attach request, MME distributes public key(optional next key) and the corresponding public key ID(s), CBE ID, and NSUC for the current key in NAS SMC messages. UE receives and saves the public key(s), CBE ID(s), NSUC and public key ID(s). When UE receives warning messages, UE verifies the signature of PWS Warning Notification message with public key, NSUC and the signature algorithm. The signature covers emergency warning, and NSUC. UE verifies that NSUC received from the network in the notification is greater or equal to the NSUC stored on the UE. After receiving a warning message, the UE ceases to update its stored NSUC that is associated with this signing key.

NOTE 1: If there are too few bits to actually send the NSUC over the air in the warning message, it could be left out. Then the UE would need to test the potential NSUCs starting from the stored NSUC to the stored NSUC plus a window size. The signature will always have to cover both the warning message and the NSUC.

If UE has attached the network before, UE sends public key ID(s), NSUC, and CBE ID in the attach request/TAU request. When MME receives attach request or TAU request, MME checks whether public key ID(s) and NSUC is the same as the same as it has already saved. If not, MME sends the public key(s), public key ID(s), NSUC and CBE ID it saves.

NOTE 2: [next key, next key ID] means that the next public key and ID is optional to send. It depends on operators' and public key issued entity's policy to use. The procedure of distributing two public keys is the same as distributing one public key. When a new key is put into use, NSUC for this key starts with 0.

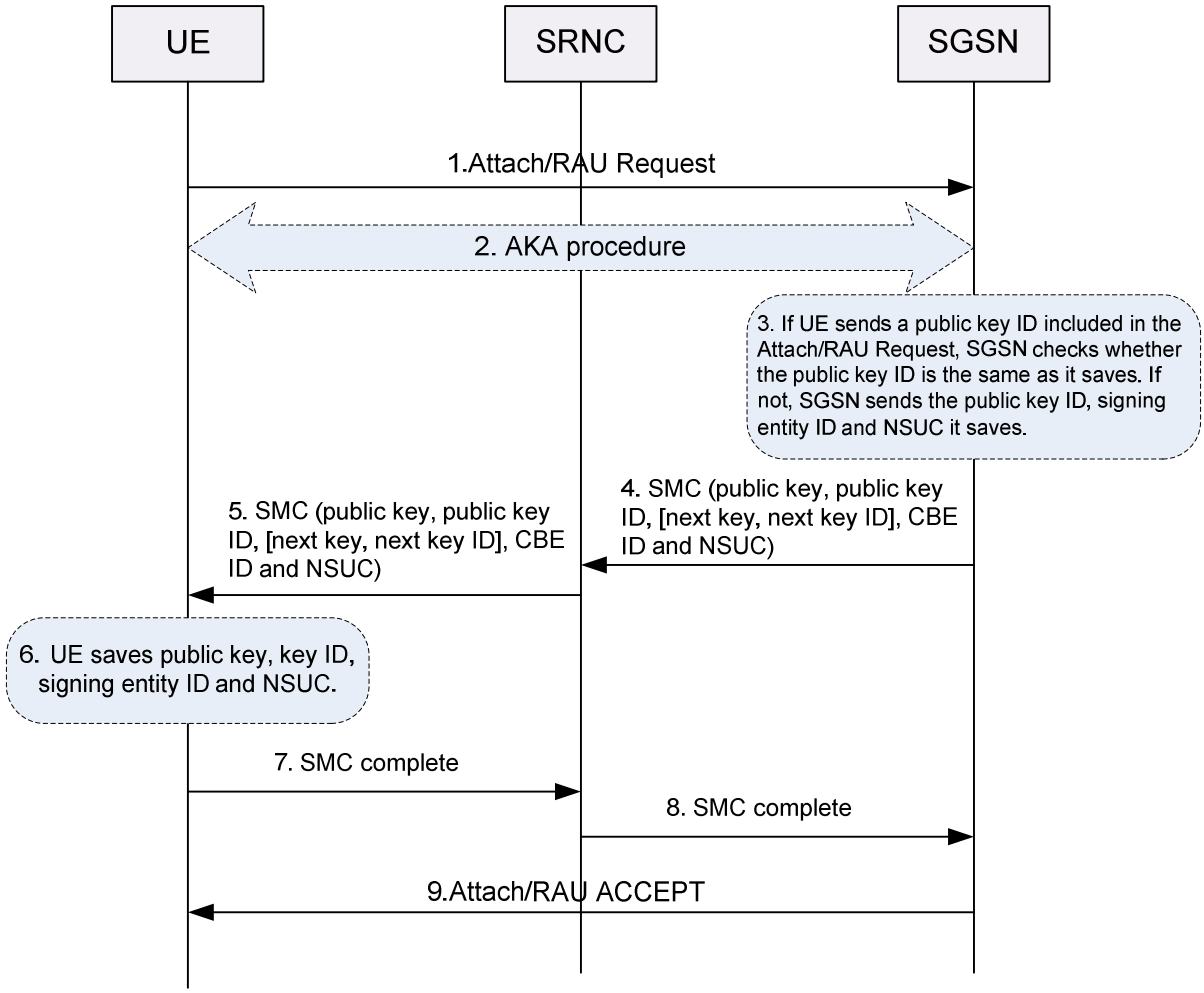

Figure 7.3.2.1.2 Initial distribution of PWS public key in UMTS system

When SGSN receives the initial attach request, SGSN distributes the latest public key (optional next key) and the corresponding public key ID(s), NSUC and CBE ID in SMC messages. UE receives and saves the public key(s), public key ID(s), NSUC and CBE ID and the relationship in all of them. When UE receives warning messages, UE verifies the signature of PWS Warning Notification message with public key and the signature algorithm. The signature covers emergency warning, SAI, and NSUC. UE verifies that NSUC received from the network in the notification is greater or equal to the NSUC stored on the UE. After receiving a warning message, the UE ceases to update its stored NSUC that is associated with this signing key.

NOTE 3: If there are too few bits to actually send the NSUC over the air in the warning message, it could be left out. Then the UE would need to test the potential NSUCs starting from the stored NSUC to the stored NSUC plus a window size. The signature will always have to cover both the warning message and the NSUC.

If UE has attached the network before, UE sends the saved public key ID(s), CBE ID, NSUC and the corresponding PLMN ID in the attach request/RAU request. When SGSN receives attach request or RAU request, SGSN checks whether public key ID(s) and NSUC is the same as it saves. If not, SGSN sends the public key(s), public key ID(s), NSUC and CBE ID it saves.

7.3.2.2 Core network PWS public key configuration

Regarding how the core network elements get public key, NSUC and relative ID, for LTE MME can get public key and relative ID by pre-configuration and also can get them through SBc interface from CBC. For UMTS, SGSN can get public key and relative ID by pre-configuration.

For LTE if MME gets public key(s) and the corresponding public key ID(s) through SBc interface from CBC, CBC uses Write-Replace Warning Request message to send current and next (optional) public key(s), the corresponding public key ID(s) and the signing entity ID to MME. After MME saves all the received public key context, it responds

Write-Replace Warning Confirm message to CBC. This procedure is triggered when public keys on CBC are initially configured or updated.

NOTE 1: [next key, next key ID] means that the next public key and ID is optional to send. It depends on operators' and public key issued entity's policy to use. The procedure of distributing two public keys is the same as distributing one public key.

NOTE 2: NSUC distribution is safe, i.e. if an MME lags behind in being updated, here may be a window of opportunity for replay attacks, however warning messages can still be verified.

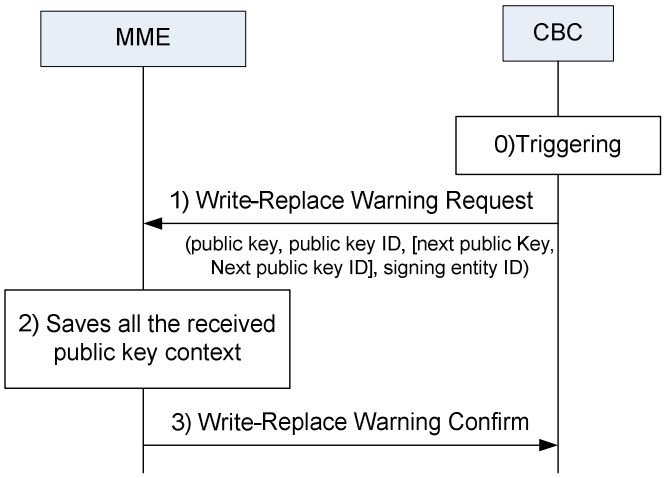

Figure 7.3.2.2.1 MME PWS Key Configuration in LTE system

7.3.2.3 PWS public key update

When the network updates the public key or NSUC, it uses the very next TAU/RAU procedure to distribute the update public key for PWS Security. This TAU/RAU can be normal procedure that UE moves to trigger and also can be periodic TAU/RAU procedure. The network sends the latest public key, public key ID and CBE ID in TAU/RAU accept to UE.

There are two cases for UE to get new public key or NSUC.

1. CBE sends periodic warning message "test" which is signed by the latest public key. The warning type of this warning message is "test". Public key ID, NSUC and CBE-ID should also be included in the test warning message. When UE receives it, UE verifies the signature using the public key it saves. If successful, the public key UE saves is the latest. UE updates NSUC if greater than the NSUC stored on the UE. If signature verification is not successful, UE sends the public key ID and CBE ID in the next TAU/RAU request to the MME/SGSN. The MME/SGSN checks whether public key ID is what it has saved. If not, MME/SGSN sends the update public key, public key ID, NSUC and CBE ID it saves in TAU/RAU accept message. Especially, the CBE sends warning message "test" which is signed by the latest public key to let UE knows in time once the CBE updates the public key.

2. UE sends the public key ID and CBE ID and NSUC in the TAU/RAU request to the MME/SGSN. The MME/SGSN checks whether public key ID is what it has saved. If not, MME/SGSN sends the update public key, public key ID, NSUC and signing entity ID it saves in TAU/RAU accept message.

NOTE 1: There can be some policy for UE to know when to send public key ID, CBE ID, NSUC and the corresponding PLMN-ID in the TAU/RAU request. For example, there can be some pre-configured periodical time for UE to use.

NOTE 2: Next public key and ID is optional to send. It depends on operators' and public key issued entity's policy to use. The procedure of distributing two public keys is the same as distributing one public key.

7.3.3 PWS Warning Notification message

For LTE system, CBE can sign the PWS Warning Notification.

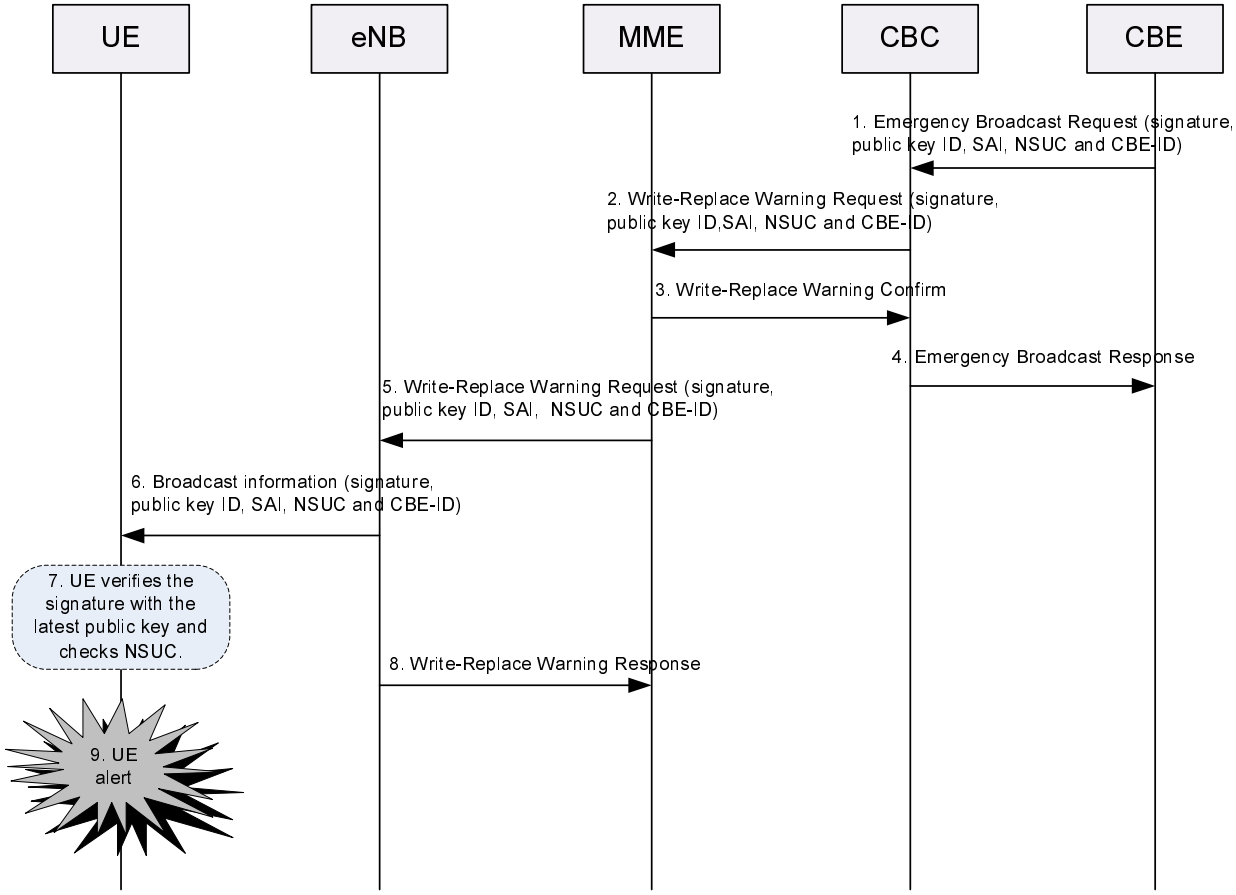

Figure 7.3.3.1 PWS Warning Notification procedure for LTE system

1. CBE sends public key ID, SAI, NSUC, CBE-ID and the signature included in Emergency Broadcast Request to CBC. See clause 6.2.10 for detailed description of the usage of these IDs.

2. CBC sends public key ID, SAI, NSUC, CBE-ID and the signature in Write-Replace Warning Request to MME.

3. MME sends a Write-Replace Warning Confirm message that indicates to the CBC that MME has started to distribute the warning message to eNB.

4. Upon reception of the Write-Replace Confirm messages from MME, the CBC may confirm to the CBE that the PLMN has started to distribute the warning message.

5. When MME receives this request, it sends public key ID, SAI, NSUC, CBE-ID and the signature in the Write-Replace Warning Request to eNB.

6. eNB broadcasts public key ID, SAI, NSUC, CBE-ID and the signature for the network's coverage area to all UEs.

7. At receiving the broadcast information message, UE verifies the signature with the latest public key and signature algorithm. The signature covers emergency warning, SAI, and NSUC. UE verifies that NSUC received from the network in the notification is greater or equal to the NSUC stored on the UE. After receiving a warning message, the UE ceases to update its stored NSUC that is associated with this signing key.
   Note: if there are too few bits to actually send the NSUC over the air in the warning message, it could be left out. Then the UE would need to test the potential NSUCs starting from the stored NSUC to the stored NSUC plus a window size. The signature will always have to cover both the warning message and the NSUC.

If UE failed to verify the signature with current public key, and UE has received the optional next public key and key ID, UE should try to verify the signature with next public key.

8. eNB sends Write-Replace Warning Response message to MME to let MME know it has broadcast the warning messages.

9. UE alerts the user what kind of warning will happen.

For UMTS system, CBE can sign the PWS Warning Notification.

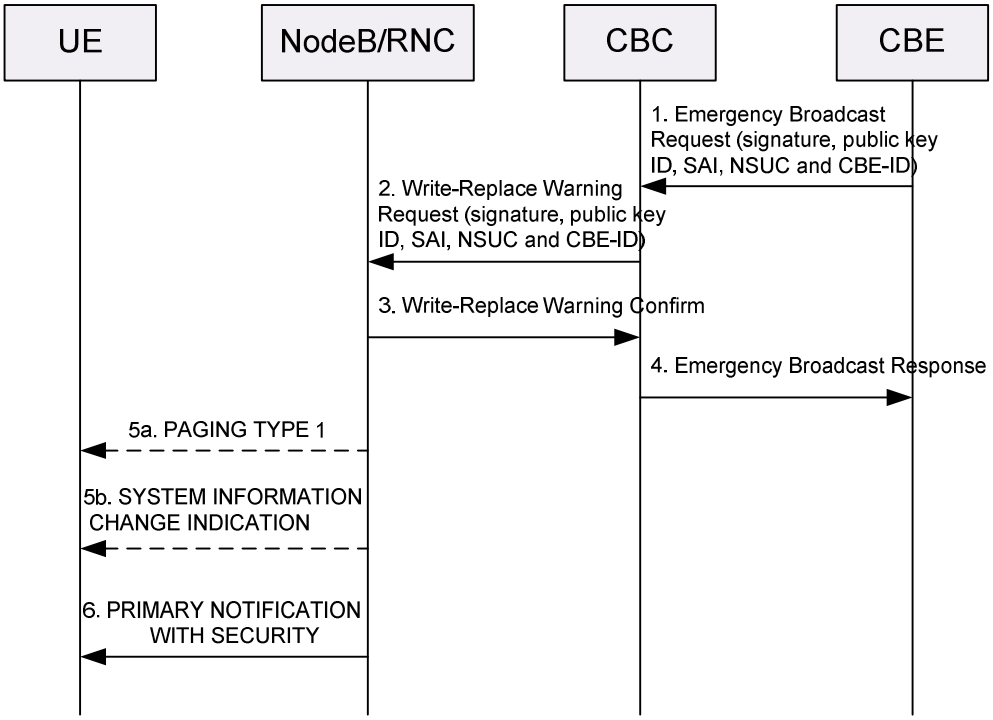

Figure 7.3.3.2 PWS Warning Notification procedure for UMTS system

1. In the Emergency Broadcast Request, CBE provides public key ID, SAI, NSUC, CBE-ID and the signature to CBC.

2. CBC transmits public key ID, SAI, NSUC, CBE-ID and the signature to UTRAN with Write-Replace Warning Request. See clause 6.2.10 for detailed description of the usage of these IDs.

3. UTRAN sends a Write-Replace Warning Confirm message that indicates to the CBC that it has started to distribute the warning message to service area.

4. Upon reception of the Write-Replace Confirm messages from UTRAN, the CBC may confirm to the CBE that the PLMN has started to distribute the warning message.

5. When UTRAN receives this request, it first sends a PAGING TYPE 1 message or a SYSTEM INFORMATION CHANGE INDICATION message, including the warning type.

6. After the reception of the warning type in either the PAGING TYPE 1 or the SYSTEM INFORMATION CHANGE INDICATION message. If RRC is configured from upper layers to receive primary notification with security, UTRAN **"should"** send public key ID included in PRIMARY NOTIFICATION WITH SECURITY and the signature to UEs. And UE verifies the signature of PWS Warning Notification message with the public key and signature algorithm.

Test warning messages:

According to the security requirement in clause 2 of this living document,

"A serving network should periodically send test warning messages on the broadcast channel."

CBE can send periodic warning message "test" which is signed by the latest public key. The warning type of this warning message is "test". Public key ID, NSUC and CBE ID should also be included in the test warning message. When UE receives it, UE verifies the signature using the public key indicated by the public key ID in the "test" message. If successful, the public key UE saves is the current. UE updates NSUC if greater than the NSUC stored on

the UE. If signature verification is not successful, UE sends the saved public key ID, NSUC and CBE ID in the next TAU/RAU request to the MME/SGSN. The MME/SGSN checks whether the received public key ID is what it has saved. If not, MME/SGSN sends the public key (optional next key), public key ID, and CBE ID it saves in TAU/RAU accept message. Especially, the CBE sends warning message "test" which is signed by the latest public key to let UE knows in time once the CBE updates the public key.

7.3.4 Solutions to security issues in GSM/GPRS and with 2G subscribers in UMTS

7.3.4.1 General

Unlike LTE and UMTS, GSM/GPRS security mechanism does not provide integrity protection on the radio interface. So the proposed PWS public key distribution solution based on integrity protection in AS and NAS messages in UMTS and LTE is infeasible.

The case of 2G subscribers with UTRAN access also needs consideration: Integrity in UMTS and LTE protects AS and NAS messages in transfer, but it also implies message origin authentication. The latter is essential in PWS public key distribution as it guarantees the authenticity of the public key to the UE (assuming a genuine network entity only distributes authentic public keys). However, for 2G subscribers with UTRAN access, UMTS integrity cannot guarantee that the UE is connected to a genuine network entity. This is so because it is possible for an attacker to obtain a valid GSM triplet (RAND, Kc andRES), while the 2G subscriber is connected over GERAN, through breaking one of the still widely used weak GSM or GPRS encryption algorithms. Note that there is no strong time constraint on the attacker as such triplets are valid for the lifetime of the SIM, due to the lack of replay protection for RAND. Once the attacker has obtained such a triplet he can feed RAND to the UE in an Authentication Request through a device that emulates a UTRAN BS system combined with an SGSN or VLR and compute the keys CK, IK converted from Kc according to TS 33.102 [8]. The attacker can then set up integrity-protected communication with the UE and send false public keys to the UE.

This clause describes solutions on how to distribute the public key and other security information to the UE in GSM/GPRS and to 2G subscribers over UTRAN. Several possible approaches are suggested:

There are 6 methods listed for GSM/GPRS solutions and with 2G subscribers in UMTS solutions.

- Re-use current GSM/GPRS security mechanism with initiating ciphering with using a UE-controlled timer(clause 7.3.4.2 and clause 7.3.4.6)
- Only cipher LAU/ RAU ACCEPT with UP still remaining unencrypted with using a UE-controlled timer(clause 7.3.4.2 and clause 7.3.4.6)
- Not initiating ciphering in the whole GSM/GPRS system with using a UE-controlled timer(clause 7.3.4.2 and clause 7.3.4.6)
- Enhanced integrity protection mechanism for GSM /GPRS(clause 7.3.4.3)
- Limiting key updates in GSM/GPRS and use periodic PWS warning test message(clause 7.3.4.4 and clause 7.3.4.5)
- Using NAS layer security(clause 7.6)

7.3.4.2 Re-use current GSM/GPRS security mechanism with initiating ciphering

In GSM/GPRS, PWS public key can be ciphered with the key Kc. The solution proposes a public key distribution based on NAS message. Figure 7.3.4.2.1 shows an example that distributes public key in GSM/GPRS.
The RAU/LAU ACCEPT message can also be used.

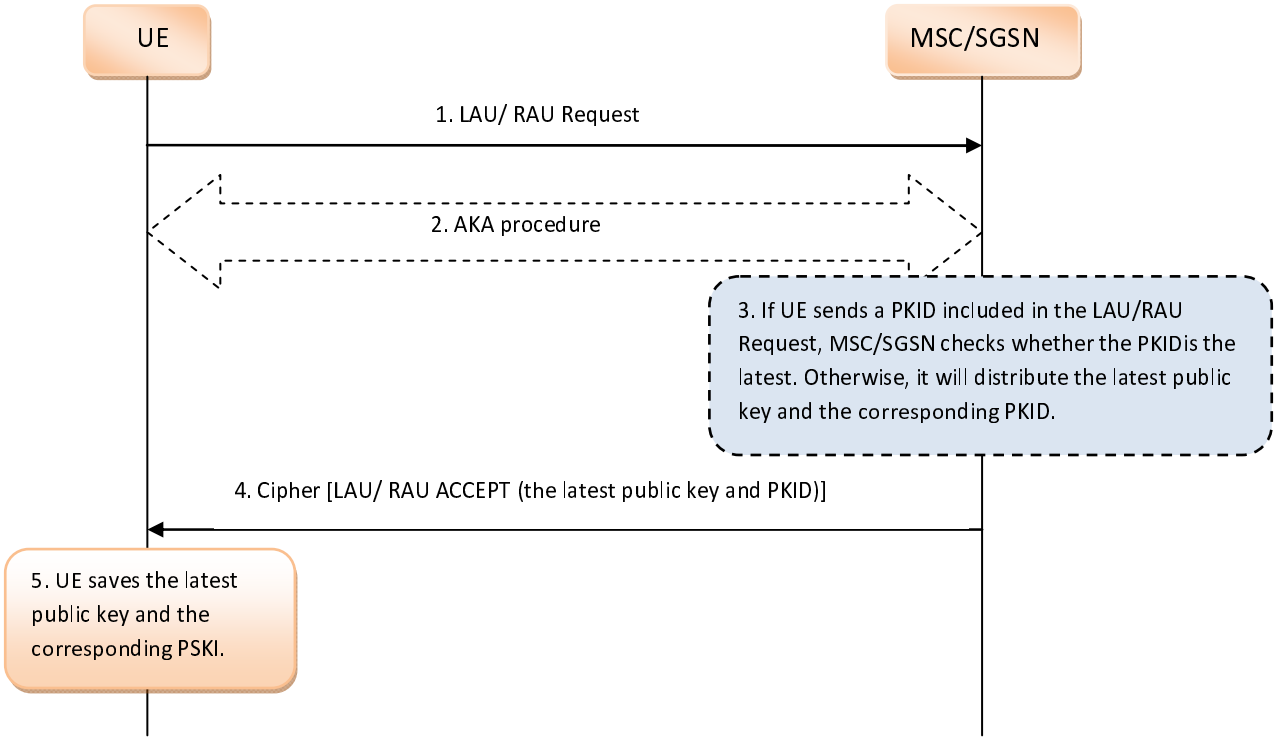

Figure 7.3.4.2.1: Distribution of public key information in GSM/GPRS

In the initial LAU/RAU procedure, UE sends the LAU/RAU request. When MSC/SGSN receives the LAU/RAU request, MSC/SGSN sends LAU/RAU Accept message to UE. In the LAU/RAU Accept, the latest public key and PKIDare included. BTS or SGSN encrypts the LAU/RAU Accept message with Kc. And the PWS public key and PKID are protected. When UE receives LAU/RAU Accept message, it decrypts the LAU/RAU Accept message to obtain the latest public key and PKID, saves the latest public key and PKID.

If UE has attached to the network before, UE can send the public key identifier to the network entity in LAU/RAU. MSC/SGSN checks whether the PKID is the latest. Otherwise, it will distribute the latest public key and the corresponding PKID. When UE receives LAU/ RAU ACCEPT, it saves the latest public key and PKID.

Only cipher LAU/ RAU ACCEPT with UP still remaining unencrypted:

In common views, it cannot only mandate ciphering LAU/RAU one procedure and leave others and UP without ciphering since once ciphering is turned on, it is better not to be turned off for security reasons. If operator does not want to turn on ciphering according to local policy, a possible alternative can be that SGSN/MSC mandates ciphering when performing RAU/LAU procedure for distributing public key. If SGSN/MSC decided to carry PWS public in RAU/LAU accept message, the current GSM security context is used to cipher LAU/RAU accept message carrying with PWS public key. Normally, after that, UE will release RRC connection and be in idle mode. In the next session, UE connects to the network and MSC/SGSN sends cipher mode command with turning off ciphering in Cipher Mode Command setting to the UE when local policy is remaining UP unencrypted. Please note that above solution needs some changes in SGSN/MSC. In addition, there may be a possibility that cipher algorithms are disabled in BSS, i.e. BSS does not support any cipher algorithms, if cipher is not allowed by local policies. If it is the case, BSS should also be modified.

Not initiating ciphering in the whole GSM/GPRS system:

In case that operator will not initiate ciphering anyhow in GSM/GPRS, it is suggested to send PWS public key and identifier directly without ciphering in LAU/ RAU ACCEPT message. To some extent, it can also ensure that UE will get a genuine public key to verify the signature than without doing any security to PWS in GSM/GPRS.

7.3.4.3 Enhanced integrity protection mechanism for GSM /GPRS

- Generate the integrity key based on the current GSM security. Kc is the encrypted key which generate from GSM AKA, Kmac is the integrity key used in PWS generated from Kc. Then Kmac can be used to protect PWS public key. Note that in this solution it is not restricted integrity mechanism only for PWS, it can be used in the whole system if operator want to enhance the security in the whole system.
- Key Derivation method directly:
  - Kmac is derived from Kc. It can be generated as follows:
  - Kmac=KDF [Kc, S], S=Fc||P||L, Fc=0x14, P=UE id, L= the length of UE id
- The configuration of the integrity algorithm.
  - Pre- configured the integrity algorithm in MS and network node.
  - Distribution the integrity algorithm to MS from network. When the integrity key is generated, the integrity algorithm or the algorithm identity indication can be distributed with protection.

In GSM, integrity algorithm and PWS public key can be integrity protected with Cipher mode CMD message. The integrity key Kmac is generated with the method discussed above. The procedure is shown in Figure 7.3.4.3.1.

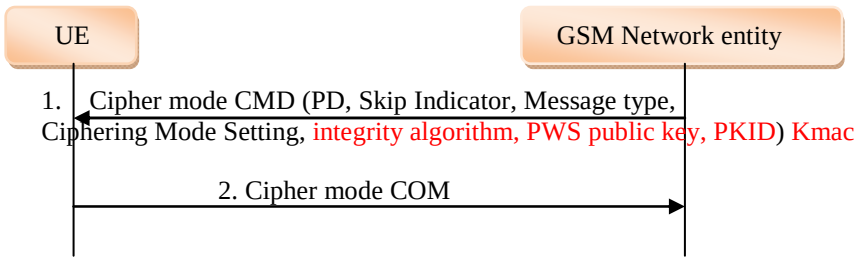

Figure 7.3.4.3.1: Distribution of public key information in GSM

In GPRS, integrity algorithm and PWS public key can be integrity protected within Authentication and Ciphering Request message. The integrity key Kmac is generated with the method discussed above. The procedure is shown in Figure 7.3.4.3.2.

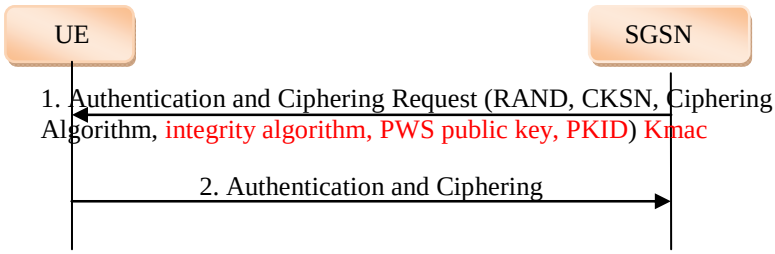

Figure 7.3.4.3.2: Distribution of public key information in GPRS

7.3.4.4 Limiting key updates in GSM/GPRS

If the protection of the key distribution in GSM/GPRS has a lower security level (or no security) than the protection of the key distribution in the other accesses, this lower security level might spread to UMTS or LTE capable UEs if they listen for key distribution messages in GSM/GPRS.

The reason being than an UMTS or LTE capable UEs might attach to a GMS network when there is no UMTS or LTE coverage. Even if the UE is configured to discard messages without a valid signature, an adversary could potentially inject false keys and false warning messages in an attempt to cause panic.

- For GSM only UEs, the only solution is to introduce some kind of enhanced GSM/GPRS security context. Making such a large change to existing GSM/GPRS networks seems unjustified just for PWS.

- For UMTS or LTE capable UEs, the problem could be mitigated by only accept key distribution messages in GSM/GPRS if there is no valid key received from UMTS/LTE. If the same signature key is used in all accesses, improved robustness and coverage could still be achieved by listen for warning messages in GSM networks.

As a subscriber with an UMTS or LTE capable UEs could have GSM only coverage for weeks (e.g. when going on vacation), this puts some extra requirements on the key distribution methods. The lifetime of the signature keys would need to be at least as long as the time an subscriber might have GSM only coverage.

NOTE: Whether to accept key distribution messages in accesses where the user does not have a subscription depends on regulatory requirements.

7.3.4.5 Mechanisms of NAS solution for GSM/GPRS

It is considered to mitigate this security issue through the NAS solution's mechanism. Two possible methods can be used to mitigate the security impacts caused by no integrity protection in GSM/GPRS.

The first one is prior receiving PWS public key from UMTS/LTE network. Nowadays most UEs will support multi-mode and UE can receive PWS public key from different RATs. When the UE and network perform the initial distribution or update of PWS public key, UE receives and stores public keys independently from GERAN/UMTS/LTE. As the same signature key is used in all accesses in one PLMN for the same CBE, UE should prior use the public keys from UMTS/LTE to verify the PWS notification message received from GSM/GPRS network. This can avoid injecting a forge PWS public key to UEs by a fake BS. This method does not work for terminals that only have GSM coverage for an extended period of time or are only GSM capable. The second one is broadcasting periodic PWS testing warning messages by the network. As the PWS public key update procedure described in clause 7.3.1.2, the network periodically broadcasts PWS warning message "test" signed by the latest public key. UE verifies the signature using the public key it stores. If successful, UE determines that the public key it is storing is the latest, and discards the warning test silently. Otherwise, UE determines that the public key it is storing is not correct, and sends the public key ID and signing entity ID in the next RAU/LAU request to the SGSN/MSC. The SGSN/MSC checks whether public key ID is what it has saved. If not, SGSN/MSC sends the update public key, public key ID and signing entity ID it saves in RAU/LAU accept messages. In this way, if UE receives and stores a forge PWS public key from GSM/GPRS network, UE can always identify whether the public key is correct or not by periodic PWS testing warning message, and then UE could fetch a correct PWS public key from the network in time. The test message can be received when the terminal is in PS connected state. It depends on the RRC state.

NOTE: If the time UEs perform next RAU/LAU is distributed in the interval, the risk of DDOS attack can be decreased to minimum. There is no solution can solve key forgery attack completely on account of the inherent security weakness of GSM/GPRS network. Solutions in clause 7.9 can be referred to solve the problem to some extent.

7.3.4.6 Delaying public key update using a UE-controlled timer

This subclause contains a solution for enhancing the security of public key distribution over GERAN for subscribers with a SIM or USIM. The solution also applies to 2G subscribers with access over UTRAN, but it is only described here for GERAN. It is intended to counter attack scenarios, in which an attacker uses a false BS to first distribute a false public key, for which he knows the corresponding private key, over LAU/RAU Accept messages and then broadcast false warning messages to create a panic.

Such a panic is most effectively created in a crowd. It is assumed that such crowds gather for some time and then disperse, or that the members of a crowd are changing over time. It is further assumed that the attacker cannot determine the members of a crowd, and communicate with them, in advance. Consequently, the attacker has to perform

both tasks, distributing the false public key and broadcasting the false warning messages, on site in a relatively short period of time (hours say). The basic idea of this solution is to delay any public key update over GERAN so that the attacker can no longer perform both these tasks while the crowd is present on the site.

The solution works as follows:

When a UE of a subscriber with a SIM or USIM receives a LAU or RAU Accept message over GERAN that indicates a required public key change, or contains a new public key, then the UE does not accept this public key, but starts a timer T associated with this public key. Only when the timer T is up the UE will reach a state where it is ready to accept this public key. The UE will indicate in the next LAU or RAU Request message over GERAN that it is now ready to accept this particular public key and will store this key when receiving it in the response. When this key is not contained in the response the UE will delete any information about this particular public key. The value of T could be randomly selected by the UE e.g. from an interval between x hours and y hours where suitable values for x and y would have to be determined (e.g. x=12 and y= 24). It is important that the network is not allowed to influence the setting of the timer.

When a LAU or RAU Accept message over GERAN is received while the timer is running, and this message confirms (one of) the currently stored public key(s), then the UE stops the timer and deletes any related information about the associated public key. When a LAU or RAU Accept message over GERAN is received while the timer is running, and this message indicates a change to a public key not stored in the UE and different from the one for which the timer is running, then the UE stops the timer, deletes any related information about the associated public key, and starts a timer for the newly received or indicated public key.

Editor's Note: It is ffs whether the previous timer could be kept running in this case. This would help in the case when the previous timer was associated with a genuine new public key.

When the UE moves to UTRAN or E-UTRAN and the subscriber has a USIM then the timer is stopped and any related information about the associated public key is deleted.

Editor's Note: It is ffs whether the timer could be kept running in this case to speed up things when the UE moves back to GERAN. But, in any case, the timer would not apply to USIM subscriber while in UTRAN.

In order to minimize the number of public keys sent by the network in a LAU or RAU Accept message over GERAN, if a timer is running for a particular public key the UE should indicate this fact in any LAU or RAU Request message over GERAN. This would keep the MSC or SGSN from sending this public key in the response to that request in vain. But it would not keep the MSC or SGSN from sending any other public key or key indicator.

The network should continue signing warning messages with the old private key at least for a period as long as the maximum value of the timer. In this way, UEs can verify genuine warning messages using the old public key while a timer is running.

Editor's Note: It is ffs whether this maximum value should be standardised, or guidelines for it should be given.

7.4 Solution 4: GBA based protection

7.4.1 General

Since this is so far just a sketch of how a solution based on GBA could work, it is not comparable one-to-one with the solution based on NAS security. However, it gives a hint about how such a solution could work and some pros/cons can be identified.

A main benefit of using a GBA based approach is that the PWS related problems with GSM access networks, with their lack of integrity protection and network authentication and security termination point far out in the access network, would no longer be a security weak point. A GBA based approach brings the security termination point to a server (NAF) in the core network and offers application layer integrity protection and network authentication.

A second benefit of GBA is that it is not necessary to use the existing NAS messages as transport channels for the delivery of the public keys used to verify the warning messages. Based on the reply LSs from GERAN2 and CT1, the number of available bytes in the GERAN CS NAS and AS messages are limited. As CT1 states that "CT1 needs to have sufficient bytes available for this further protocol evolution" sending information in these messages might be problematic.

A third benefit is that the same protection and transfer mechanism can be used for all RATs (i.e. GERAN, UTRAN, and EUTRAN). In other proposed solutions it has been suggested to use different messages in the three different RATs, add new protection mechanism for GERAN, or to do a lot of special handling in GERAN to minimize the security weaknesses there. By using the same protection and transfer mechanism can be used for all RATs, UE implementation is simplified.

All solutions for PWS Security can be split in two main parts:

1) A protection mechanism for the public keys distribution.

2) A transport mechanism for public keys distribution to the terminal.

A GBA based approach uses GBA as part of the protection mechanism.

7.4.2 GBA based protection mechanism for public key distribution

7.4.2.1 Key establishment

The protection is based on the establishment of a shared symmetric key Ks_(int/ext)_NAF between the UE and a NAF (defined in the GBA architecture, see TS 33.220 [26]). The NAF assumes the task of distributing the PWS public keys and will henceforth be called PWS Key Center.

Editor's note: it is ffs whether the PWS key center is part of CBC, or if it is a standalone entity.

Key establishment with GBA is flexible and can be done in several different ways. Either the UE can initiate the establishment (pull) or the NAF can initiate the establishment (push). GBA specified in TS 33.220 is defined for pull, while GBA-Push specified in TS 33.223 [27] is defined for push. GBA-Push can however be used in a pull like mode by adding a non-GBA message asking the NAF to push out the key establishment information.

Note that GBA-Push requires the UE to hold a USIM or ISIM and is not defined for 2G subscribers.

While the security level of the public keys (used in broadcast) need to be very high, the symmetric integrity protection of the public key distribution (used in unicast) does not require as high security. The lifetime of the key Ks_(int/ext)_NAF can therefore be very long. Both GBA_ME and GBA_U could be used.

GBA_ME does not require UICC support. While GBA_U provides enhanced security, it is not necessarily required. However, if the UICC needs to be changed anyway, to provide support for the PWS elementary file EF_{PWS}, then consideration should be given to using UICC's supporting GBA_U for additional security.

A. Key establishment using GBA: This method (see Figure 7.4.2.1.1) uses the GBA procedures specified in TS 33.220 and on a high level it works as follows:

A1) After the UE has registered with the network, the UE performs a GBA bootstrap with the BSF and derives Ks_(int/ext)_NAF.

A2) If the UE doesn't have the current public key, it requests the current public key from the PWS Key Center.

A3) The PWS key center communicates with the BSF and derives Ks_(int/ext)_NAF

A4) The PWS key center distributes the current public key to the UE.

The GBA procedures (A1 and A3) can be reused for several key distributions.

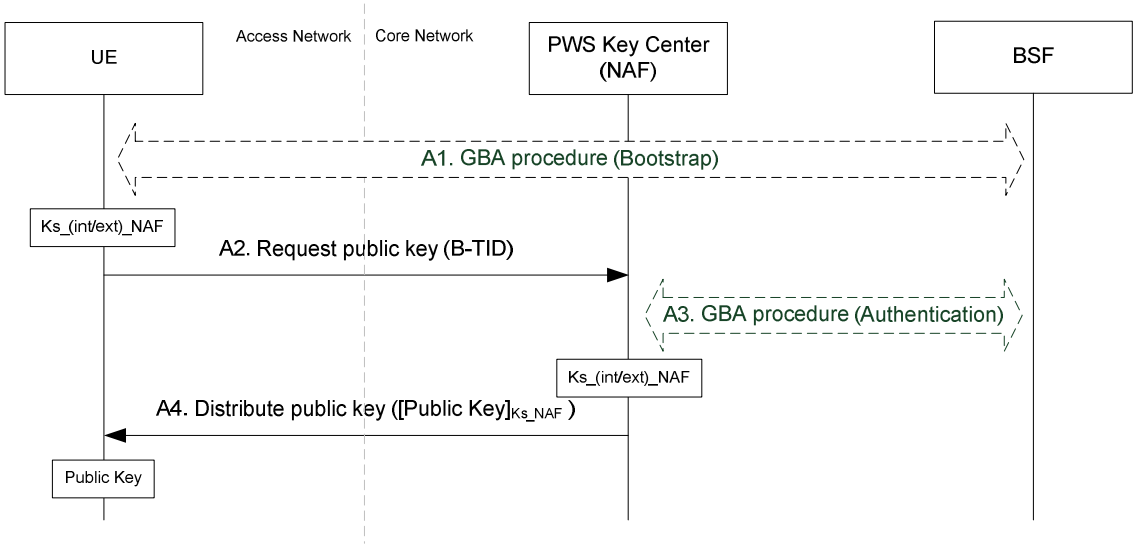

Figure 7.4.2.1.1: Key establishment using GBA

B. Key establishment using GBA-Push: This method (see Figure 7.4.2.1.2) uses the GBA Push procedures specified in TS 33.223 [27] and on a high level it works as follows:

B1) If the UE doesn't have the current public key, it requests the current public key from the PWS Key Center.

B2) The PWS key center communicates with the BSF and derives Ks_(int/ext)_NAF

B3) The PWS key center distributes the current public key to the UE.

The GBA push procedure (B2) can be reused for several key distributions.

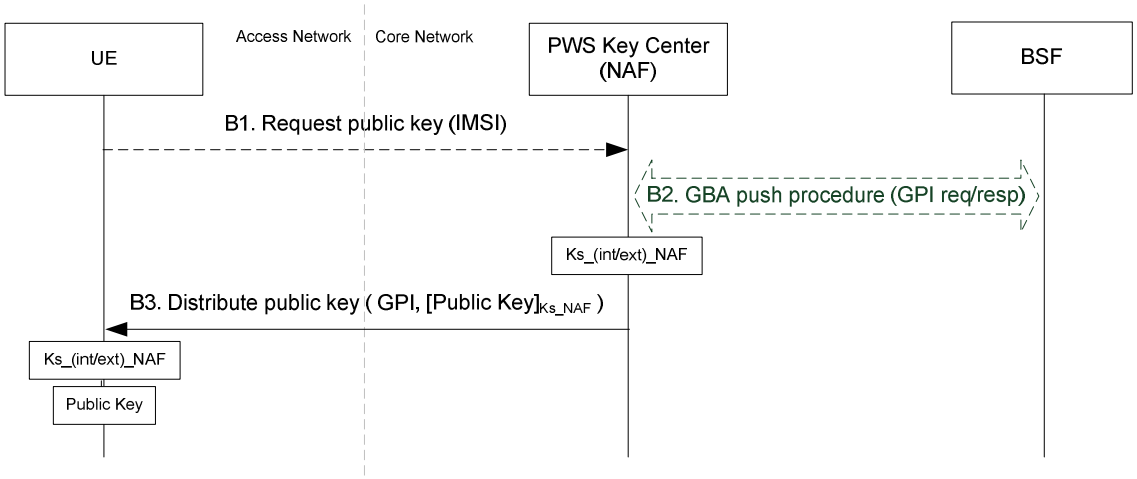

Figure 7.4.2.1.2: Key establishment using GBA-Push

After the initial key establishment, the UE and PWS Key Center has a shared symmetric key (Ks_(int/ext)_NAF) and public key updates can be distributed via either pull or push without any new GBA procedures.

As can be seen, using GBA-Push has several advantages over GBA (TS 33.220 [26]):

- Both Pull and Push can be supported. Pull can be under normal circumstances and push can be used for quick update of the public key. The advantage of push is that the PWS Key Center has control over how many public keys updates are done per second. Therefore there is no risk for overload.

- The amount of traffic over the access network is smaller than in TS 33.220 as the GBA procedures are sent only in the core network.

7.4.2.2 Security protocol

The security protocol needs to provide integrity protection. The solution also needs some form of replay protection, but this does not necessarily need to be on the transport layer.

If GBA (TS 33.220 [26]) is used for key establishment, HTTPS is the default standard for secure communication with a NAF. In HTTPS, TLS provides both integrity and replay protection, but requires handshake consisting of two roundtrips to set up.

If GBA-Push (TS 33.223 [27]) is used for key establishment, the lightweight GBA Push Layer (GPL), see TS 33.224 [28], is a perfect candidate. GPL does not require any setup roundtrips, it is lightweight, it integrates with GBA-Push, and it provides integrity protection and replay protection. GPL can also be used with 33.220 [26].

7.4.3 Transport mechanisms

7.4.3.1 Transport mechanisms for establishment of GBA keys

For UE-initiated GBA, as defined in TS 33.220 [26], the Ub interface is IP-based, hence a packet switched data bearer is required. One could, of course, think of tunnelling IP in some other protocol, but this would require some significant additional specification work and may be undesirable from a protocol point of view.

For GBA push, as defined in TS 33.223 [27], the transport over the Upa interface could for example be over a packet switched data bearer, SMS, USSD, or even using NAS messages.

7.4.3.2 Transport mechanisms for public key distribution

If the protection of the public keys is provided by an application layer mechanism such as GBA and GPL, there are several options for transporting the public keys to the terminals.

- **Existing NAS procedure:** the integrity protected public keys might be included in existing NAS messages. This does not provide any savings in terms of NAS message space, but the integrity protection can be terminated in the core network. For example, the PWS key center can integrity protect the keys and deliver them to the SGSN/MSC/MME, or a separate protection function can be included in the SGSN/MSC/MME that is provided with the keys for the PWS key center. As sufficient bytes need to be available for protocol evolution, this approach cannot be used for GERAN CS.

  - Can be used in all accesses.

- **New NAS procedure:** Instead of using the existing NAS procedure a new NAS procedure for PWS key distribution can be created. The NAS procedure can work similarly in GERAN, UTRAN, and EUTRAN as well as in CS and PS. Another benefit is that while the current NAS procedures are not logically associated to PWS Security at all, this new procedure would be. Having an independent NAS procedure would also make potential overload problems easier to handle as the whole message can be routed to a separate function or node. The overload problem for the signalling system would still exist in times of key change for all users.

Editor's note: New NAS procedure needs further study.

- **New NAS protocol:** Instead of creating a new procedure in the mobility management and session management protocols a new NAS protocol specifically for PWS Security can be created. The new protocol would be assigned a new protocol discriminator. The NAS procedure can work similarly in GERAN, UTRAN, and EUTRAN as well as in CS and PS. This has similar benefits as creating a new NAS procedure, it would require more specification work but routing and potential overload problems would be even easier to handle. The overload problem for the signalling system would still exist in times of key change for all users.

Editor's note: New NAS protocol needs further study.

- **PS bearer:** UE can set up a PS bearer and use IP to connect to the PWS key center. This work for all three RATs but does not work for terminals that only support the CS domain and would hence have to be combined with one of the other options below to cater for all types of terminals.

  - Cannot be used in CS

- **Short Message Service (SMS):** Probably only practical for use in a push fashion. An SMS can carry 140 octets and if one or two 256-bit public ECDSA keys are sent that is plenty of space. Sending the corresponding security level DSA keys would not be possible in a single SMS. In that case chained SMSes would have to be used.

  - Can be used in all accesses.

- **Unstructured Supplementary Service Data (USSD);** TS 23.090 [29]: USSD is commonly implemented and provides data traffic in both uplink and downlink. A USSD message can carry 160 octets, and if one or two 256-bit public ECDSA keys are sent that is plenty of space. Sending the corresponding security level DSA keys would not be possible in a single SMS. In that case chained SMSes would have to be used. USSD support both pull and push. It is present in UTRAN, GERAN and over IMS, so it can be used for any of the 3GPP accesses.

  - Can be used in all accesses.

- **Circuit switched data (CSD);** TS 23.202 [30]: CSD can be only be used over GERAN CS and UTRAN CS, but could be combined with a PS bearer in LTE. CSD however seems to be mainly intended for UE to UE communication and may hence not be appropriate to use as a bearer between a terminal and a server in the network.

    - Cannot be used in LTE.

A robust solution may require the possibility for the UE to be able to pull keys from the network. Note that to support pull, all of the options except the NAS based, requires a separate CS or PS bearer to be set up.

PS bearer and CSD also have limitations and cannot be used in all accesses. Of the suggested options only New NAS procedure, New NAS protocol, and USSD can be used in all accesses. There are therefore seen as the main transport mechanisms candidates.

7.4.4 Analysis

7.4.4.1 Pros

A GBA based solution would not suffer from the weaknesses related to false or hacked BSs (e.g., BSS in GERAN and home (e)NBs in UTRAN/E-UTRAN) that applies to the NAS based solution. The reason is that a GBA based solution protects the distribution of the key all the way from the PWS key center located in the core network.

Most protocols for key establishment and protection of the public key delivery are actually in place already: GBA and/or GBA Push for key establishment, GPL and HTTPS to protect the key delivery to the terminal (push and pull respectively).

The signing/verification, display of warning to the user and possible public key infrastructure are mainly application layer functions. Therefore it would be preferable from a design cleanness perspective to also do the key distribution on the application layer (compared to mixing application layer and the radio layer as is the case for the NAS based solution). Layer violations (or cross layer optimizations as they are sometimes called) usually lead to complexity.

If a new NAS procedure or protocol is used as transfer protocol, the GBA approach would be more connected to the radio layer, but the termination point for the key distribution messages and key distribution protection point would be firmly on the application layer.

7.4.4.2 Cons

If a new NAS procedure or protocol is used as transfer protocol, the GBA approach has some but not all of the cons as the NAS solution.

For 2G subscribers, only (UE-initiated) 2G GBA, as defined in TS 33.220 [26], Annex I, is available. 2G GBA requires a TLS server certificate for the BSF and the installation of the corresponding public verification key in the UE. But if procedures for generating server certificates and for installing root keys in the UE are available anyhow then it may be easier to select solution 7 because, once a root key is installed in the terminal then, for warning messages in the home area, no further protocol steps would be required, and, even more importantly, no BSF and NAF entities would be required. For the roaming case, only a one-time cross-certification would be required instead of repeated interactions between a visited NAF and the BSF.

For 2G GBA, the bandwidth requirements would be relatively high compared to GBA using a USIM or ISIM due to the need for TLS. On the other hand, the Ub run would not have to occur very often, and it would not have to happen at the same time as a PWS public key update.

This PWS Security solution requires GBA support in the network and in the terminal.

At the moment not many cons have been identified, but clearly cons to some degree would be discovered once more details would be examined.

7.4.4.3 Cost

BSF and NAF can be achieved by deploying independent physical entities or be achieved on existing function node. NAF could also be achieved on other application servers. The deployment of GBA requires physical entities or addition of new function on the existing network elements for the purpose of PWS Security alone will increase the cost of deployment.

The NAF needs to be dimensioned to handle a huge number of simultaneous requests for the current key in the same way the MSC/SGSN/MME would have to be dimensioned to handle the distribution of public keys in NAS messages. But the cost of dimensioning a single function is of course lower than the cost of dimensioning three separate functions.

The home network needs to deploy a BSF and a PWS key center (NAF). There could be a cost to re-dimensioning the home BSF/HSS.

GBA software needs to be set or upgraded in ME to support GBA function for PWS Security, if it is not available for other purposes. If GBA_U is used, then it requires UICC support.

7.4.4.4 Comparison to other solutions

Editor's Note: Comparison to the other solutions is needed.

7.5 Solution 5: using NAS layer security

7.5.1 High level solution discussion

This solution is for GSM and UMTS.

This solution is for both UTRAN and GERAN and is motivated by the desire to protect the delivery of PWS keys from the core network to reduce the risk of compromised RAN nodes. The proposed solution is designed to only require changes in the ME and core network node of the serving network. This ensures that if serving network wants to use secured PWS messages, then any roaming users with handsets that support the functionality will be able to receive the warning messages.

The proposed solution uses part of solution 2 in the UTRAN KH (see TR 33.859 [7]) to generate a root key that remains in the core network (note: the rest of that solution, e.g. providing a fresh key at idle to active is not needed here). From the key and the COUNT values used to ensure a fresh key, the UE and core network node can generate a key when needed to protect the delivery of the PWS key. This is enhanced to allow the calculation of K_{ASMEG} from Kc after a run of GSM AKA which is used with COUNT to generate a fresh key to protect the delivery of the PWS key.

In addition the solution for the USIM case provides a mechanism to ensure that the root key (or keys that could derive the root key) never leave the core network. This is provided by the following three bits of functionality. Firstly, the UE with a USIM will never accept a PWS key protected by a security context generated by a run of GSM AKA. In general, this could happen with a pre-Rel 99 VLR/SGSN or an ME that does not support the ME to USIM interface (see TS 33.102 [2]). Both these cases can be ruled out for PWS Security and hence there is no need to fall back to GSM AKA for the protection of PWS keys. This prevents an attacker forcing a UE to fall back to GSM AKA in order to be able to deliver a false PWS key.

Secondly, in response to a challenge requesting it to establish the enhanced security context, the UE does not respond with RES but rather with an enhanced response derived from CK and IK. When the core network receives such a response, it will not release CK and IK to the RAN nodes. Similarly, the UE will only accept PWS keys when protected using a security context where it returned the enhanced response. This means that an attacker that breaks into a RAN node would not be able to get CK and IK unless RES can be provided to the core network. An attacker deploying a false GSM BS could get 32 bit of information about RES due to the way that the GSM response is calculated from the UMTS response (X) RES (see TS 33.102 [8]). Hence the level of security provided by this is 32 bits less the length of (X)RES. Milenage (see TS 35.206 [9]) uses a 64 bit (X)RES and hence a USIM using standard Milenage would get 32 bit of security from this.

Thirdly, an AMF bit is assigned and when that bit is set a UE will **not** use the security context derived from such an AV to protect the delivery of PWS keys. A core network node that wishes to send PWS keys to a UE **"should"** inform the HSS that it intends to use the AV to generate a security context that will protect the delivery of PWS keys. This means that a HSS can control which core network can receives AVs that can be used to provide protection of PWS keys. If the HSS sends an AV with the AMF bit set to all other nodes, this means that it is not possible for an attacker to get CK and IK unless it breaks into a core network node. This functionality would be optional to implement in the HSS. It should be noted that by having the AMF bit set to mean that the AV can be used for PWS Security would require a change in the home network before a roaming UE could receive PWS keys and would be against the design principle of the solution.

In summary the above solution for the USIM case provides a NAS level solution to protecting the delivery where the home operator has control over the level of security that is provided to each UE.

For the SIM, case, only the second part of the above will be used, that is the modification of the authentication response. This ensures the UE and network agree that an enhanced authentication has been run.

Editor's note: Some analysis of the security in the SIM case is needed

7.5.2 Solution details

7.5.2.1 General

The solution is made of three components: firstly changes to the mobility message, secondly enhancement to the authentication procedure and finally changes to the context transfer messages between core network nodes. Each of these changes will be detailed in turn.

7.5.2.2 Changes in the mobility messages from the UE

In the Request mobility message (e.g. Attach, RAUs and LAUs), the UE needs to send the relevant information to the network that it is capability of receiving PWS keys and for the network to decide if the UE has the correct PWS key. The UE **"should"** also include the COUNT value (as described in solution 2 of TR 33.859 [7]) if it has an enhanced security context.

In the Response message, if the UE needs a new key, the network needs to include the PWS key identity, PWS key and a MAC calculated over the transmitted key and its identity using the $K_{PWS\text{-}int}$ as the key and over the transmitted key. $K_{PWS\text{-}int}$ is calculated as follows:

$$K_{PWS\text{-}int} = KDF(K_{ASMEU}/K_{ASMEG}, COUNT)$$

7.5.2.3 Changes to the authentication procedure

The authentication procedures is enhanced to provide a key, K_{ASMEU} that does not leave the core network as in solution 2 in the UTRAN KH specification (see clause 5.2 of TR 33.859 [7]). In order to achieve this, the core network node signals that it want to run an enhanced authentication to the UE in the message carrying the authentication challenge. This results in both the UE and core network node calculating K_{ASMEU}/K_{ASMEG} depending on whether a UMTS or GSM AKA was run and setting the COUNT value associated with it to zero. On receiving an authentication challenge for an enhanced AKA, the UE checks that the relevant AMF bit is not set. It also calculates an enhanced authentication response (EAR) as follows:

$$EAR = KDF\ (K_{ASMEU}/K_{ASMEG}, \text{other parameters})$$

Editor's note: The other parameters in the above calculation need to be defined.

The core network node will check that the received EAR is correct.

When requesting AVs in order to be able to send PWS keys, the core network node will inform the HSS that it wants AV for protecting PWS keys. If the subscription has a USIM, an HSS that support this feature will send AVs with the relevant AMF bit set to all core network nodes that do not require an AV for protecting PWS keys and AVs without the relevant bit set to core network nodes that require AVs to protect PWS keys.

7.5.2.4 Changes to context transfers between core network nodes

When transferring the UE's context from one core network node to another, the old core network node **"should"** include K_{ASMEU}/K_{ASMEG} as appropriate and COUNT.

7.5.3 Comparison with other solutions

In some ways solution 4 using GBA and solution 5 are similar in that they both use a run of the relevant 3GPP authentication algorithm to generate keying material to protect the deliver y of PWS keys to the subscriber.
The solutions differ in that solution 4 proposes to deliver the keys from an application outside the 3GPP access networks whereas solution 5 proposes that the PWS keys are delivered by the core network nodes in the 3GPP access networks. Solution 4 proposes to use one of the existing GBA mechanisms, whereas solution 5 proposes new changes to the AKA protocol in UTRAN/GERAN. When delivering the keys directly from the UTRAN/GERAN core network nodes, there seemed no value in replicating the full GBA architecture as this would require the relevant core network node to act as a BSF and support the Zn interface when keying material for the protection of PWS keys can easily be derived directly from the authentication between the UE and core network node. Furthermore a GBA based solution requires the home network to deploy a BSF whereas with solution 5 only the core network node in the serving network and the UE need to be changed. For 2G subscribers, the GBA based solution provides a higher level of security due to the certificate based network authentication.

When comparing against solution 3, solution 5 provides a higher level of security for the delivery of PWS keys over UTRAN/GERAN due to the fact the PWS keys is protected from the core network by keys that do not leave the core network. This is particularly true for subscriptions with USIM where the authentication can not be replayed. The cost of this gain is some additional complexity in the NAS signalling.

7.6 Solution 6: implicit certificate PKI based PWS solution

7.6.1 General

This solution is access independent.

An overview of the implicit certificate based approach is shown in figure 7.6.1-1. UE firmware is provisioned with public keys of several CAs. The message signer periodically obtains an implicit certificate from a CA which can be included as part of the security portion of a PWS transmission. The implicit certificate combined with the CA's public key results in the message signer's public key allowing the UE to verify the signature.

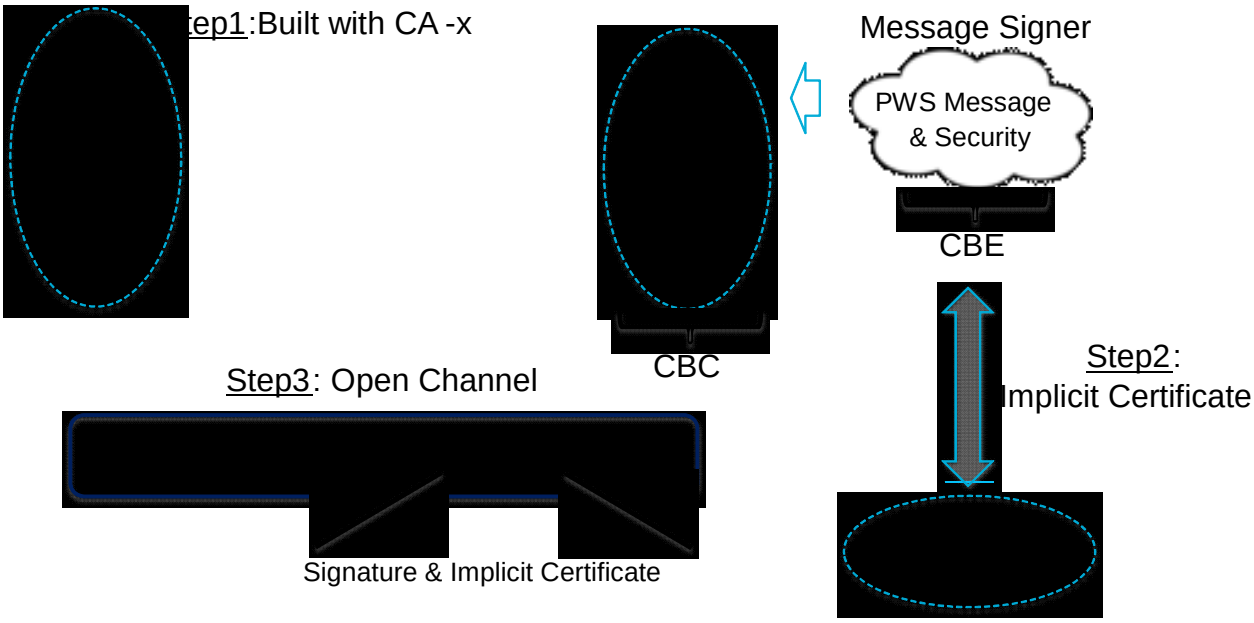

Figure 7.6.1-1: Overview of implicit certificate approach

An advantage of this system is its scalability. That is, multiple CBEs can share the same set of CAs. Simply put, if a national authority requires the addition of a new CBE, the CBE need only obtain an implicit certificate from one of the available CAs without the need of signalling new keying material to UEs or an operator's network except for testing purposes.

Although CAs are assumed to be long lived entities (~20 years), allowance **"should"** be made for changing the set of CAs and their public keys. While this would most likely be a planned event, in the rare occurrence a CA or CBE is compromised or potentially a UE is reset, such an update might be necessary.

Two potential approaches to updating the list of CA public keys can be considered, 1) Using periodic test messages to carry update information and 2) a push mechanism such as (U)SIM Application Toolkit.

7.6.1.1 CA updating via PWS test messaging

Updates to the list of CAs and associated public keys stored by a UE could be achieved through a new PWS message type. This message can be signalled as a PWS CA update message by using the existing Message Identifier parameter [1] but could otherwise be transmitted to UEs in the same manner as warning related PWS messages.

Figure 7.6.1.1-2 shows the contents of the PWS CA update message could contain the identifier of the CA (CA-ID), its new public key and an indicator to signal whether the update message is adding or revoking a CA.

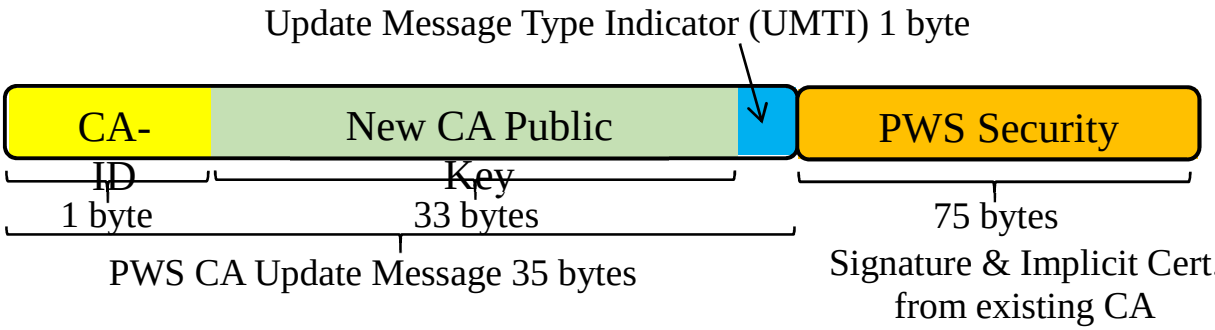

Figure 7.6.1.1-2: PWS CA update message format

When an update message is received the Update Message Type Indicator (UMTI) is first checked.

If the UMTI indicates the addition of a CA then the UE checks if the CA-ID is already stored. If it is not then the CA-ID and associated public key are added, otherwise the update message is discarded.

If the UMTI indicates the removal of a CA then the UE checks if the CA-ID is already stored. If it is then the CA-ID and associated public key are removed, otherwise the update message is discarded.

In addition to facilitating the update of the CA list, the UMTI could potentially also be used as the basis for revoking an individual PWS message signer. In this case the signalled public key is of the PWS message signer to be revoked. If the UMTI indicates the revocation of a PWS message signer, then the UE checks if the signalled public key has been stored previously. If so the update message is discarded, if not then the public key is added to a list of revoked PWS message signers.

Since the contents of the PWS CA update message are critical to the functioning of the system, to ensure the UE can trust the message contents UEs should be required to receive at least two update messages containing where the implicit certificate used in each message is from a different existing CA. However without an additional safeguard a UE could be locked out of accepting further CA updates if the number of CAs in its list is reduced to one. This can be addressed by requiring there always be at least 2 CAs. This means if UEs have 2 CAs a third "**should**" be added before revoking one becomes possible.

In the case more than one PWS message signer is supported in a region each message should also be from a different existing PWS message signer.

The case of only one PWS message signer could also be accommodated for example by requiring Implicit Certificates to be obtained from a CA who issues short-lived implicit certificates for the purpose of PWS CA update messages. Procedural steps following this example could be:

- Step 1: Receive and validate PWS CA update message;
- Step 2: Check previously validated PWS CA update messages;
- Step 3: If a PWS CA update message is stored in the ME and has been validated using a different existing CA and either the current or existing different CA is used for the purpose of PWS CA update messages then update the current CA list in the ME with the PWS CA update message contents;
- Step 4: If a CA update message stored in the ME according to Step 3 is not found then store the current CA update message in the ME with its validating CA but without updating the contents of the CA list.

7.6.1.2 CA updating via (U)SIM

A UEs list of allowed CA list could also be updated by relying on the USIM application. This would require the addition of a file under the USIM containing the list of allowed CAs and that would be read by the ME during the initialisation or the refresh of the USIM (this would require therefore the addition of a new procedure in CT6 TS 31.102 [22] clause 5). The update of this new file would be made possible by using Remote File Management as defined in clause 5.2 of TS 31.116 [23].

However, as this updating mechanism is tied to the network, operators would bear greater responsibility and cost with this approach.

7.6.2 Certificate authorities

7.6.2.1 General

CAs act as the trust anchors for PKIs. It is essential for a functioning PKI to have at least one universally accepted CA. However, in systems like PWS that span multiple government and regulatory authorities, agreement on a sole trust anchor is encumbered. There are a few working models in similar fields that are worth consideration such as:

- Advanced Access Content System used in Blu-ray;
- Zigbee Smart Energy uses a single commercial CA vendor that issues certificates to devices that are certified at an approved testing lab;
- CA Browser Forum (CAB) used in support of web browsers;
- WiMax uses two CAs, Verisign and Motorola that are approved to service the community.

Most of these examples are focused on issuing certificates to a large number of devices so that they can securely operate in an ecosystem. However, the PWS situation requires a large number of devices to be able to authenticate messages from a relatively few entities, in this aspect it is perhaps most similar in use as example 3 (many browsers compared to TLS servers).

Here UE firmware is provisioned with public keys of several CAs much in the same way as for CAs used with browsers today.

As responsibility for security in the implicit certificate approach rests at the national level, creating requirements on CAs UE vendors **"should"** support as well as upkeep of these CAs rests at the national level and not with operators. Operator responsibility in this regard is simply to pass requirements necessitating support of CA public keys mandated by government agencies to UE vendors.

As shown in Figure 7.6.2.1-1, CBEs from different regions need not necessarily share the same set of CAs. There may be some overlap and indeed agreement between CBEs from different countries to share the same CAs is possible; however no such requirement need be exist within 3GPP. Moreover the responsibility for root management concerns such as the provisioning of CAs, overlap in usage of CAs or indeed cross certification of CAs would be decided and enforced at the national level.

As an example consider Figure 7.6.2.1-1. The government in region A may decide that UEs sold in its country should only be pre-provisioned with CA1 and CA2. In such cases, UEs from region A whether in their own region or visiting others, will not process PWS warning messages signed by CBE-B or CBE-C as these use untrusted CAs.

However, UEs from region B visiting region C will receive PWS warning messages with implicit certificates from CA5 since the government in region B would mandate UEs sold in its region be pre-provisioned with CA5. The risk government B has taken is a compromised CBE in region C can be used to broadcast false PWS warning messages in government B's own region since it now shares at least one CA.

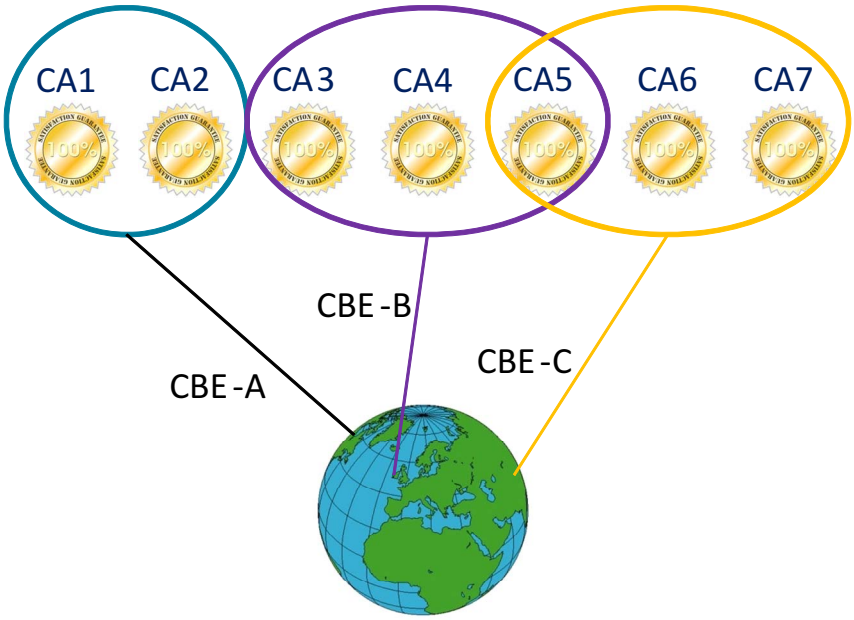

Figure 7.6.2.1-1: Certificate Authorities (CAs) mandated at the national level in various regions

A consequence of this approach is UEs using a CAs public key shared by CBEs outside its own region will accept any PWS message signed by those CBEs. Therefore it is the responsibility of the UEs national government to establish confidence in those CBEs outside it region before allowing public keys from such shared CAs to be pre-provisioned on UEs sold within its boundaries.

7.6.2.2 UE provisioning [public key] and [CA-ID] updating of home network

To simplify the manufacturing process it can be assumed UEs are provisioned with public keys of all CAs globally. In this way UEs will be capable of displaying secured PWS messages even when in limited service state.

However, just as CBEs in a particular region should supported by one group of CAs similarly a UE in that region should only display warning messages verified by a public key from the same group of CAs once it enters service.

Several approaches can be considered in identifying the relevant group of CAs.

Approach 1: User controlled CA list

An option is available for the user to select the location of his home network. This could be used to select CAs allowed by the user's home government from the current global list of CAs. CAs outside this selection can be marked as inactive.

While the option of the user setting his home network location would always be available as an option, a user could be explicitly prompted for this information when a new UICC is used.

Approach 2: PWS test message type

As previously described a new type of PWS message can be used to modify available CA information. Shown in figure 7.6.2.2-1, this message could additionally contain a field indicating the set of CAs used in the home region of the PWS broadcast.

Figure 7.6.2.2-1: PWS CA key update test message

Once a UE has received such a message it can use the CA Set-ID to select CAs allowed by the user's home government from the global list of CAs. CAs outside this selection can be marked as inactive.

To allow for a change in a UEs home location, if a UE receives a test message with PWS Security but indicating an inactive CA-ID the UE could be allowed to verify the signature using the inactive CAs public key. While no-action on key updating would be taken unless verification occurs with active CAs, in the event several test messages are verified over an extended period of time using inactive CAs, the user could be prompted to confirm his home region based on the CA Set-ID via the user controlled CA list approach described in approach 1.

Approach 3: USIM Reading

While UEs can be provisioned at manufacturing with the CAs in use globally as a step in satisfying limited service state requirements, once a USIM is inserted into the UE the ME could read the list of allowed CAs mandated by the government of its home network from the UICC.

This can be enabled by asking CT6 to create an additional file in TS 31.102 [22] containing the list of CAs and their public keys. Additionally, CT1 and CT6 could be requested to create an update mechanism along the lines of (U)SIM Toolkit to securely update this CA list.

While offering a clear unambiguous solution to updating the CA list in case of change in the home network, this approach does place a clear responsibility on the operator in maintaining the active list of CAs mandated by the regional government that the other two approaches do not.

Approach 4: USIM triggering

Similar to approach 3, UEs are provisioned at manufacturing with the CAs in use globally as a step in satisfying limited service state requirements. However, in this case once an USIM is inserted, the regional CAs associated with the UE's home country are identified based on the UE's home network.

7.6.2.3 Roaming considerations

Roaming is of concern to all approaches to PWS Security. That is when a UE whose home network supports PWS Security roams onto a network supporting PWS but without security and in particular one that does not authenticate itself to the UE, then the UE **"should"** reject all PWS messages or leave itself open to attack.

Two niche cases more highlighting security arrangements at the national and operator level may however be worth discussing.

In the first case two operators in different regions have no security arrangements between each other while the governments have agreed to obtain implicit certificates from the same set of CAs. In this case as governments bear the responsibility of bearing roaming agreements, PWS messages with security can be received in both regions by both roaming and non-roaming UEs.

In the second case, governments' in two regions requiring PWS with security have decided to obtain implicit certificates from CAs not provisioned in UEs in each others regions and operators in these regions do have security agreements in place between their networks. While not currently proposed in the implicit certificate approach it is possible in principle for the CAs a UE uses for verification to be dependent on the network it is authentically attached to while roaming. However to date this proposal wasn’t made as in essence the home government now needs to trust security agreements operators in its region made with operators outside its region thereby placing greater liability on operators.

7.6.3 Implicit certificates

7.6.3.1 High level view of an implicit certificate approach from the UE perspective

Implicit certificates are a well-known approach used in cryptography and can be used to reduce the amount of storage and computation in public key systems. Instead of a CA generating a signed certificate in order to certify a signer's explicitly embedded public key, the signer's public key is computed by the UE using the certificate in combination with a CA's public key.

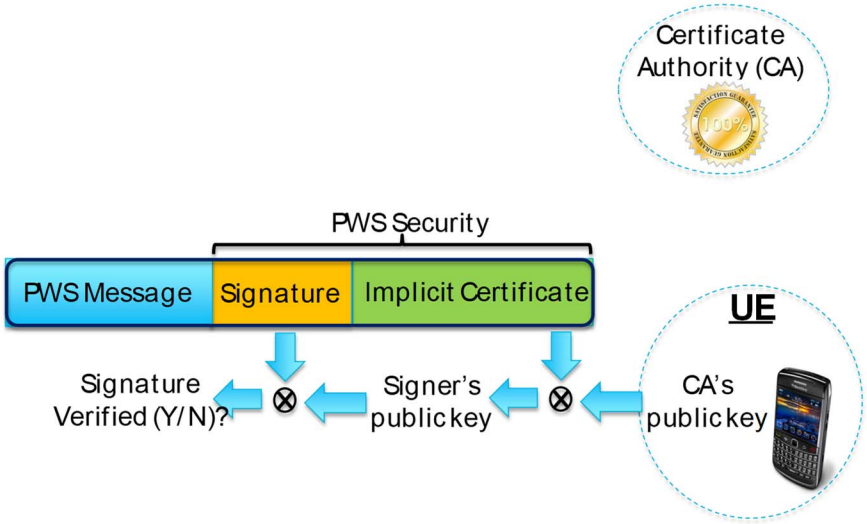

Figure 7.6.3.1-1: UE perspective of implicit certificate in PWS

A high level view of an implicit certificate approach from the UE perspective is shown in figure 7.6.3.1-1.
The UE derives the signer's public key using the received implicit certificate and the CA's public key.
The UE then verifies the signature using the derived signer's public key. The authenticity of the signer (and indeed the derived public key) is implied by proof of possession of the associated private key of the signed message.

7.6.3.2 Generation of implicit certificate

As shown in figure 7.6.3.2-1, the PWS message signer contacts the CA with a random number "α" whenever a new implicit certificate is desired. This could be once a week, month or year; depending on how long the signer wants the public key derived from the implicit certificate to be valid for. However, long the implicit certificate is valid for, it is independent of the PWS message and can be used in regenerating the same PWS message signer's public key for multiple warning messages.

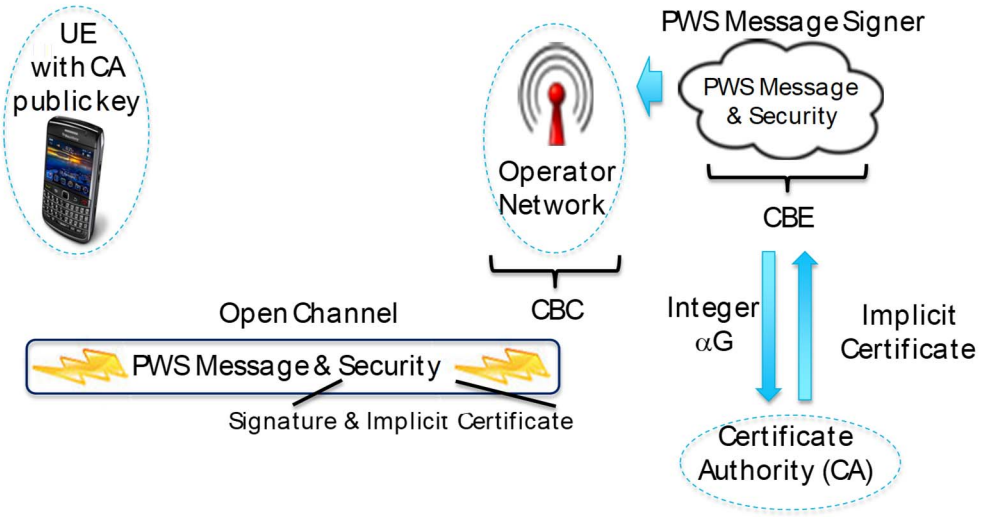

Figure 7.6.3.2-1: Implicit certificate in PWS

On receiving the integer "α", the CA then generates the implicit certificate and returns it to the PWS message signer.

Formal steps in this process taken by the CA for the ECQV implicit process are as follows:

Let G be a generating point of order n.
Certificate Authority (CA) has private key c and public key $Q_{CA} = cG$.

The PWS Message Signer requests the implicit certificate IC_A from the CA.

1) PWS Message Signer generates a random integer α, computes αG and sends that to the CA.
2) CA Select a random integer k from $[1, n-1]$ and computes kG.
3) CA computes $P = \alpha G + kG$.
4) CA forms the implicit certificate $IC_A = (ID_A, P)$ where ID_A is NWMS's identifying information.
5) CA computes $e = H(IC_A)$, where H is a cryptographic hash function.
6) CA computes $s = ek + c \pmod{n}$
7) CA sends (s, IC_A) to the PWS Message signer

The PWS Message Signer's private key is $a = e\alpha + s \pmod{n}$

The PWS Message Signer's public key is $Q_A = eP + Q_{CA}$

Given IC_A and Q_{CA}, UEs can generate Q_A.

7.6.3.3 PWS Security contents

Implicit certificates are versatile and can be used with a variety of signature approaches including DSA and ECDSA, however the approach considered here due to efficiency in size is a Keyed-MAC signature scheme.

When operating at 112-bit security level, using a 112-bit MAC and assuming an ECQV certificate structure, 14-bytes, 28-bytes and 31-bytes are required to encode the values MAC, *s* and IC_A respectively.

The 31-byte length for IC_A assumes a certificate structure containing a 225 bit public key reconstruction value, a 15 bit certificate timestamp and an 8 bit CA_ID value. The certificate timestamp can provide one approach to protection in case a key is compromised at the message signer. The validity period of the certificate and therefore the frequency at which a message signer obtains new certificates from the CA would be decided at the national level and need not be the responsibility of operators.

In total the signature and implicit certificate occupy 73-bytes leaving 2 additional bytes that can be used for a PWS message timestamp. This timestamp would be provided and signed by the PWS message signer and indicates the validity period for the PWS warning message.

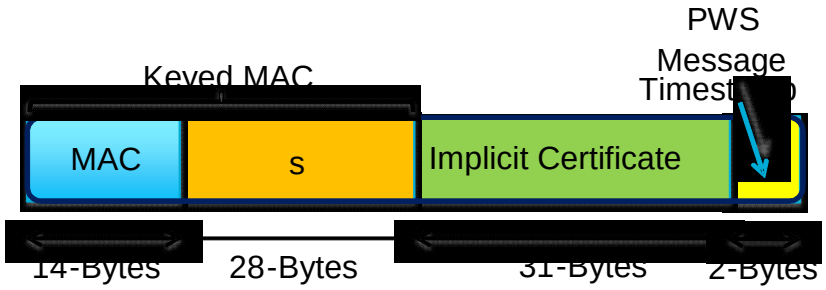

Figure 7.6.3.3-1: PWS Security content

The final two bytes of the security contents consist of a timestamp provided by the message signer and indicating the validity period for the PWS warning message for the purposes of replay protection.

This can take the form of a traditional timestamp or as a message counter.

Timestamp

For the timestamp to function correctly, some degree of synchronization is necessary between the UE and the PWS message signer.

Although an automatic network time such as NITZ would provide a means of time synchronization between the UE and network, whether such a mechanism is securely available is network dependent and is not assumed.

In one approach, the UE could indicate the receipt of a PWS message with an expired certificate if one is received and present the user with the current time understood by the UE and the option of proceeding or discarding the message.

Alternatively a PWS timer could be provisioned in UEs at manufacture with a conservative time. This time could then be adjusted in the normal course of operations either by a PWS timestamp update message similar in concept to the PWS CA update message, or by an additional timestamp field in the PWS CA update message itself. Such an update timestamp would detail the current time of the PWS message signer to all receiving UEs.

Message counter

In the case a message counter is used in order to avoid the need for co-ordination between message signers, a message signer identifier should be included as part of the implicit certificate. As shown in Figure 7.6.3.3-1 this can be accommodated by reducing the implicit certificate timestamp from 15 bits to 7 bits allowing an 1 byte field for a message signer identifier (PKID).

In order to protect out of date UEs (e.g.: those who miss PWS warning messages and the resulting increments to a message signer's counter) from replay attack, a PWS counter update message similar in concept to the PWS CA

update message could be used or alternatively if there are only a few PWS message signers, an extra field could be included in the PWS CA update message itself. Such a field could contain 3 bytes, the message signer's identity (PKID) of 1 byte and current counter value (NSUC) of 2 bytes, for each message signer signalled.

Whether the PWS message timestamp takes the form of an actual timestamp or a message signer counter, the 2 bytes in the PWS Security content should be included in the computation of the keyed MAC signature.

Using ECQV, the UE **"should"** compute the Message Signers Public key using the implicit certificate in addition to verifying the PWS signature.

Considering available cryptographic signature benchmarks from eBATS and assuming the ARM EABI platform running at 1782MHz and 128-bit level security, the full implicit certificate based approach will takes roughly 6.5ms and not more than 7.4ms. This is compared with 3.7ms for ECDSA and 18ms for DSA signature verification indicating comparable complexity to other signature schemes.

The complexity time estimates of the implicit certificate based approach are approximate and were made by considering the steps 3 and 4 of signature verification and comparing with similar steps in algorithms benchmarked in eBATS.

Steps both in encoding (at the PWS message signer) and verification (at the UE) of the Keyed-MAC can be as follows:

Keyed-MAC Signature Generation

INPUT: PWS Message Signer's private key d_A, and associated ECQV certificate structure IC_A, and a message to be signed M.

OUTPUT: A signed message M, with associated security information MAC; s; IC_A.

1) Generate ephemeral key pair *(d,Q)*.
2) Construct MAC key *k = KDF(Q)*, where KDF is a key derivation function that takes as input a point, and possibly other information, and generates an encryption key.
3) Compute *MAC = MACAlgorithm(M,k)*.
4) Compute *h = Hash(MAC||M)*, where *Hash* is a suitable hash function, that takes as input additional information including a possible identity string.
5) Convert *h* to an integer *e*.
6) Calculate *s = e _ dA+d (mod n)*.

Output *s,MAC*, along with input value IC_A as the associated security data for *M*.

Keyed-MAC Signature Verification

INPUT: Signed message *M*, with security information *s*, *MAC*, IC_A, and the CA's public key Q_{CA}.

OUTPUT: VALID, or INVALID.

1) Compute $h = Hash(MAC||M)$, with the same hash function used in the signature generation scheme, and the additional input information.
2) Convert *h* to an integer *e*.
3) Recover the PWS message signer's public key from the certificate, $Q_A = ECQVPublicKeyReconstruction(Cert_A, Q_{CA})$.
4) Compute $Q' = sG - eQ_A$.
5) Compute $k' = KDF(Q')$, using the same key derivation function used in the signature generation algorithm, including the same additional information.
6) Compute MAC' = MACAlgorithm(M,k').

If MAC' = MAC then return VALID, else return INVALID.

During this process the UE combines information contained within the implicit certificate with the public key of the appropriate CA to produce the message signer's public key. As several CAs may and indeed should be supported, a means is needed to distinguish which public key is used.

This can be achieved through use of the one byte CA-ID field described in clause 7.7.3.3. Each CA public key would be assigned a CA-ID value which the UE can read from the implicit certificate. Using the CA-ID the UE can look up the CA public key tied to that CA-ID in its provisioned list of CAs.

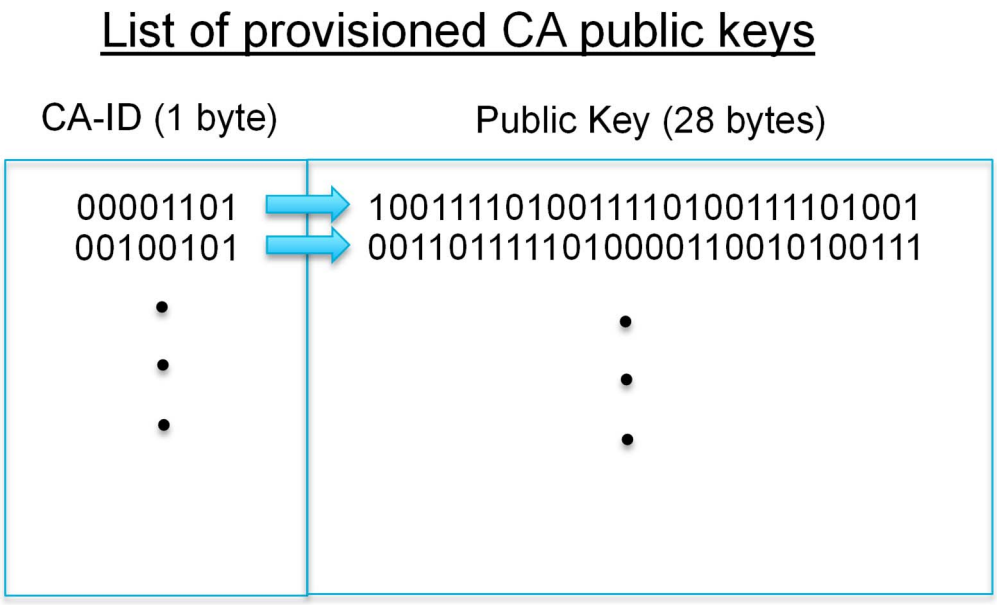

Figure 7.6.3.3-2: Example list provisioned CA public keys with associated CA-IDs

7.6.4 Properties of solution

The implicit approach is the only proposed PKI approach for PWS public key distribution. Compared to other approaches it offers significant advantages in the areas of network resource consumption, CBE scalability, operator liability and operator cost.

- **Network resource consumption**: Link and core network resource usage is less than with other approaches and is consumed only when a PWS message is sent. No additional resources are expended either for roaming UEs or during an update of a CBEs public key/implicit certificate.

- **CBE scalability**: If multiple CBEs share the same CA, the system is easily scalable to support the additional CBE. Simply put, if a national authority requires the addition of a new CBE, the CBE need only obtain an implicit certificate from one of the available CAs without the need of signalling new, per CBE, keying material to UEs or an operator's network except for testing purposes.

- **Several OEMs share same CAs:** Details are ffs

- **Operator liability**: Operator liability is kept to a minimum. Responsibility for key management issues such as setting up, functioning and upkeep of the CAs is at the national level and not the responsibility of the operator. However the operator may have to assist with UE provisioning.

- **Operator cost**: Compared to other approaches the implicit certificate approach has minimal impact on an operator's network. The only known impacts at this time are:

  1) Upgrading the PWS Security field from 50 bytes to 75 bytes;

  2) Installation of the CBC-CBE interface.

  On the other hand there are disadvantages to be considered with the implicit certificate approach, at least in the areas of CA setup/operation and overhead in the PWS Security field.

- **CA setup/operation**: A major cost in any PKI system is setup and operation of the CA. However, since this is done at the national level, costs could be borne by the national government or alternatively by operators.

- **PWS Security field overhead**: While efficient in size, implicit certificates do occupy space and are a source of additional overhead in the PWS Security field resulting in a security level of 112-bits.

As with all PWS Security proposals the implicit certificate approach has an impact on the UE. This includes, provisioning a list of CA public keys, enabling implicit certificate and signature algorithm and support of ancillary functionality necessary for PWS Security such as key update mechanisms.

7.7 Solution 7: generalized certificate-based approach for PWS

7.7.1 Introduction

All three major schemes for PWS Security described in the present document, so far, NAS-based, GBA-based, and implicit-certificate-based, have shown serious problems.

- The scheme laid out in most detail so far, the NAS -based scheme, suffers from the problems inherited from weak 2G security: it seems almost impossible to provide strong PWS Security to SIM users in this way, and, for USIM users, strong PWS Security over GERAN access could only be provided if significant changes were made to the 2G security architecture, which may be commercially unviable for this legacy technology. On the other hand, for the NAS-based scheme, strong PWS Security over GERAN seems essential as an attacker could always force users, even selectively per user, to camp on its false GSM BS by emitting a strong signal, bypassing strong PWS Security over UTRAN or E-UTRAN in this way.

- For the GBA-based scheme, there are also problems inherited from 2G security (cf. clause 7.4): there is no 2G variant for GBA push, and for (UE-initiated) 2G GBA, as in TS 33.220 [26], Annex I, a TLS server certificate with a corresponding public key in the UE is required. The latter fact begs the question what the advantage of 2G GBA over a purely certificate-based approach would be. Furthermore, the GBA-based scheme has not been investigated in much detail. While this disadvantage seems to go away with the working assumption that users with a 2G SIM need not be supported for PWS security, it is still true that the GBA-based scheme currently is the least developed of all and would require much further study.

- The scheme based on implicit certificates (called 'IMPCERT' henceforth) assumes a set of root CAs whose public keys are pre-installed in the terminal. One of these root CAs issues an implicit certificate to a particular signing entity. This implicit certificate is then sent together with the warning message to the ME over a broadcast channel. Concerns have been raised regarding the need for some sort of global coordination regarding the distribution of public root keys, cf. more on this below.

This solution presents a certificate-based approach, which is more general than IMPCERT. This may help with the acceptability of a certificate-based approach and the gradual introduction of PWS Security around the globe.

The PWS Security solution based on implicit certificates is generalized in the following respects:

- Structure of Certificate Authorities (CAs);

- Distribution of public root keys;

- Certificate format.

Public root key is used to authenticate CBE certificates. If it succeeds, UE can verify the warning message afterwards by using the public keys for PWS carried in CBE certificates.

After terminal is configured with CBE certificates, it can verify warning messages sent from CBEs. Since UE may receive warning messages signed by different CBEs, which CBE certificate should be used is a problem. A parameter combined with the warning message is needed to indicate which CBE certificate to be used. This parameter is the CBE identifier, which suffices for this purpose, as the UE should have only one certificate for one CBE at a given point in time.

7.7.2 Structure of CAs

7.7.2.1 Top-down approach to CAs

This approach is similar to what is proposed in the IMPCERT scheme:

- There is a number of top-level CAs;
- Each PWS signing entity obtains a certificate from one these top-level CAs;
- The public root keys of all top-level CAs are available in all terminals.

As shown in clause 7.6.2 of the present document, there are real-world examples following this kind of approach. One example is the collection of root keys in the key store of a browser. Of course, PWS is different in that the top-level CAs and the signing entities would be under the responsibility of regulators, and not an industry sector.

It is not required for this approach that the top-level CAs mutually trust each other, or cross-certify each other, if it can be ensured that the usability of a root key of a CA is somehow limited to the jurisdiction of the regulator owning this CA. (This is still to be shown, cf. Editor's note in clause 7.6.3.3.). But, as a minimum, some sort of global repository for root keys, from where terminal manufacturers could obtain the collection of root keys in an authentic way before manufacturing a terminal, would be required. This global root key repository would not have to be a CA itself, or know any private keys, but it would have to have trust relationships, and communication channels guaranteeing integrity (not necessarily through cryptography), with all regulators owning the top-level CAs as well as with all terminal manufacturers. It is not clear who could play the role of providing such a trusted global root key repository. Without such a trusted global root key repository, the reliable provisioning to the terminals cannot be assured.

7.7.2.2 Bottom-up approach to CAs

This approach starts from the observation that it is questionable whether PWS Security will be introduced in all, or even a large number of, countries within the same time frame as its introduction depends very much on national regulations. (Actually, not a single regulator is known to have set requirements mandating the introduction of PWS Security.) This is one reason why, at least in the initial phases of the global roll-out of PWS Security, the provision of a global root key repository may meet with difficulties. The bottom-up approach would, in contrast, allow one country - or one group of countries agreeing on a common regulation - to go ahead without being dependent on the rest of the world.

The bottom-up approach for PWS is similar to the approach various 3GPP specifications have taken, cf. below, when they assume the use of 3GPP server certificates and corresponding public verification keys in terminals. The approach works as follows:

A regulator who decides to introduce PWS Security sets up a CA that issues certificates for the signing entities responsible for signing warning messages in this regulator's area of responsibility. The public root key would be implemented in the terminal typically after manufacturing time, see clause 7.7.3.1. The terminals could then verify warning messages in the area of that regulator. For other areas, terminals could, in the initial phase, either accept unprotected warning messages, or not accept warning messages at all, according to the preferences set in the USIM, cf. TS 22.268 [2], cf. also clause 7.9 on circumvention attacks.

Right from the start (or after some time when PWS Security has gained increased acceptance around the world) the regulator owning the CA could cross-certify the public root keys of CAs of other regulators responsible for areas that are most frequently visited by users in his own area. The cross-certificates could become part of roaming agreements. In order for the cross-certification to remain manageable, the number of partners, with which cross-certificates are exchanged, would have to be somehow limited. But it is believed, based on typical roaming patterns, that agreements with only a quite limited number of roaming partners would suffice to ensure that most users would be present in their home area or one of the partner areas most of their time.

When the number of partners, with which cross-certificates would have to be exchanged, would grow too large to be manageable this would be an indication that PWS Security is gaining traction around the globe, and it would be time to set up a number of root CAs according to the top-down approach; but this may be quite some time from now.

7.7.2.3 More complex CA structures

It may be desirable in certain situations to have intermediate CAs below a root CA where the intermediate CAs would provide the certificate for the PWS signing entity, e.g. when the root CA would be at a regional level (e.g. European

Union) while the intermediate CA would be at a national level. Then not only would the public root key have to be available in the terminal, but also the certificate of the intermediate CA. This seems more easily compatible with the bottom-up approach and a distribution of certificates and root key via configuration, cf. clause 7.7.3.2, than with an approach, as in IMPCERT, where the certificate is distributed on the cell broadcast channel, together with a warning message.

7.7.2.4 Comparison with server certificates in other 3GPP specifications

The following 3GPP network elements are assumed to have server certificates and the UEs are assumed to have the corresponding public root keys installed:

- *IMS*: the P-CSCF has a TLS server certificate when the IMS access signalling is secured by means of TLS as in TS 33.203 [31];
- *3G-WLAN interworking*: the PDG, acting as an IKEv2 responder in 3GPP IP access, has a certificate, cf. TS 33.234 [32];
- *Non-3GPP access to the EPC*: the ePDG in untrusted access, and the PDN GW, acting as a Home Agent for DSMIPv6, both have certificates for their roles as IKEv2 responders, cf. TS 33.402 [35];
- *GBA*: the BSF has a TLS server certificate for GBA_Digest and 2G GBA, cf. TS 33.220 [26], Annexes I and M;
- *GAN*: the GANC-SEGW, acting as an IKEv2 responder in GAN access, has a certificate, cf. TS 43.318 [36] (which points to TS 33.234 [32] for security).

Note that a P-CSCF, a PDN GW, a PDG, or an ePDG can reside in a visited network, according to TS 23.228 [37], TS 23.234 [38] and TS 23.402 [39]. If this is the case then a UE will need a cross-certificate for being able to verify the server certificate, or have the root key for the visited network stored.

7.7.3 Distribution of public root keys

7.7.3.1 Pre-installation in terminals at manufacturing time

This approach was already discussed in the context of the top-down approach to CAs above. The terminal manufacturers would obtain the public root keys from a trusted global root key repository.

Open questions include how new root keys could be added to or removed from the key store during the lifetime of the terminal, e.g. due to lifetime expiry or revocation of keys, or due to new regions introducing PWS Security and setting up new CAs. An approach to a solution is outlined in clause 7.6.1 of the present document.

7.7.3.2 Configuration when terminal is first taken into use

Clearly, pre-installation in terminals at manufacturing time is not well compatible with the bottom-up approach to CAs as this would involve producing country-specific versions of terminals, which is seen as quite problematic by terminal manufacturers. Therefore, a different approach is needed that would allow for incremental growth of PWS Security:

A public root key valid in the home area of a UE (defined by its USIM) could be loaded into a terminal, when the terminal is first switched on, in a way similar to how the terminal is configured with other parameters today, e.g. email access points etc. This could be done e.g. via OMA DM, using SMS, etc. Alternatively, the public root key could be stored on the USIM when the USIM is issued, or securely downloaded to the USIM OTA.

Editor's Note: It is ffs whether particular security measures would be needed for configuring the root key in the terminal when it is first switched on, or whether the highly distributed nature of this configuration process would be sufficient to prevent the relevant attacks against PWS. (This depends, of course, on the attack model, cf. clauses 6.1.1 and 8).

When such a terminal, configured with a home root key, is roaming, and a cross-certificate is available for the visited area, this cross-certificate could be distributed to the terminal together with other information that is sent to the terminal by the visited operator anyhow. E.g., a roaming UE typically receives one or several welcome SMSs when first registering in the visited network; the cross-certificate could become part of such an SMS.

7.7.3.3 Public key update and revocation

Regular public key update (due to key lifetime expiry) is to be handled on the basis of the solution outlined in clause 7.6.1 of the present document. If one takes into account that the standard certificate revocation mechanisms CRL and OCSP work only under the assumption that there is a signing key for the CRLs or OCSP responses that is not compromised, and in particular, that the root CA is not compromised, then the approach in clause 7.6.1 may also be usable for certificate revocation.

7.7.3.4 Comparison with server certificates in other 3GPP specifications

The other 3GPP specifications mentioned in clause 7.7.2.4 of the present contribution do not tell how the public root keys would be installed in terminals; TS 33.220 [26] explicitly states that this is out of scope.

Furthermore, none of these other 3GPP specifications explicitly mentions the need for cross-certificates, let alone makes explicit reference to TS 33.310 [40] where a 3GPP CA structure for cross-certificates is described.
When these other specifications reference TS 33.310 [40], they do so only for IKE or TLS profiles.

Regular public key updates (due to key lifetime expiry) are not addressed by these 3GPP specifications either.
They do, however, specify mechanisms for certificate revocation, either CRLs (e.g. optional implementation and use of CRLs in TS 33.203 [31], 33.220 [26] and 33.234 [32]) or OCSP (e.g. optional use and mandatory client implementation in TS 33.234 [32]).

7.7.4 Certificate format and distribution of certificates

Implicit certificates are attractive because they are so short that they could be distributed together with the signed warning message. However, when root keys are distributed over a different channel anyhow, as in the configuration-approach using UICC OTA in clause 7.8.2.2, then it would also become possible to use different certificate formats, e.g. X.509 certificates, that would be longer (although they should, of course, not become arbitrarily long), assuming that also the certificate are distributed via that channel.

Alternatively, certificates may be distributed over a broadcast channel. There are two possible options:

a) when no length restrictions on warning messages apply then the certificates can be distributed together with the signed warning message;

b) otherwise, it is preferable to distribute certificates over test messages. Test messages are assumed to be sent regularly in PWS anyhow (cf. e.g. clauses 7.3.4.1, 7.6.1.1, 8.3.5.5) and do not suffer from any time constraints and length restrictions. (Note that it was the time constraints on ETWS primary notifications that led to the length restrictions summarised in clause 6.2.7.)

When using approach b), it needs to be ensured that test messages including certificates need to be broadcast sufficiently often over an extended period of time so that a vast majority of UEs has the opportunity to pick them up from the broadcast channel.

It is also possible to mix the two ways of distributing certificates: One can distribute certificates together with the root key OTA to the UE and distribute only certificate updates and revocations over a broadcast channel.

7.7.5 Considerations on pre-provisioned CAs public keys shared by CBEs

The approach of UEs pre-provisioning with a CAs public key shared by CBEs outside its own region, as described in the previous clauses, has two unwanted consequences:

1) a national government has to establish confidence in CBEs outside its region, which may be difficult or impossible;

2) if a national government of a country cannot establish confidence in CBEs in some regions of another country, the UEs sold within that government's region cannot use PWS Security in those other regions. In other words, a global solution becomes impossible.

To avoid that impacts of a compromise of a CBE or CA in one region spread around the world, or at least the region of mutually established confidence, the scope of a root CA public key or a CBE certificate to certain regulatory domains or geographical areas, e.g. one country or one larger region, e.g. European Union, or China, or USA, needs to be limited. The scope is the area or domain where the key is authorized to be used and it should be securely associated with CA and/or CBE.

Depending on the regulatory structure of a country or region the useful definitions of 'scope' may vary considerably. E.g. in USA thousands of CBEs could exist, while in other countries the number of CBEs is very restricted. A CBE could be responsible for just one warning type or for several warning types covering one district (mapped to a small cell area of the mobile network operator) or many different districts (mapped to the complete network of the mobile operator) as illustrated in figure 7.7.5.

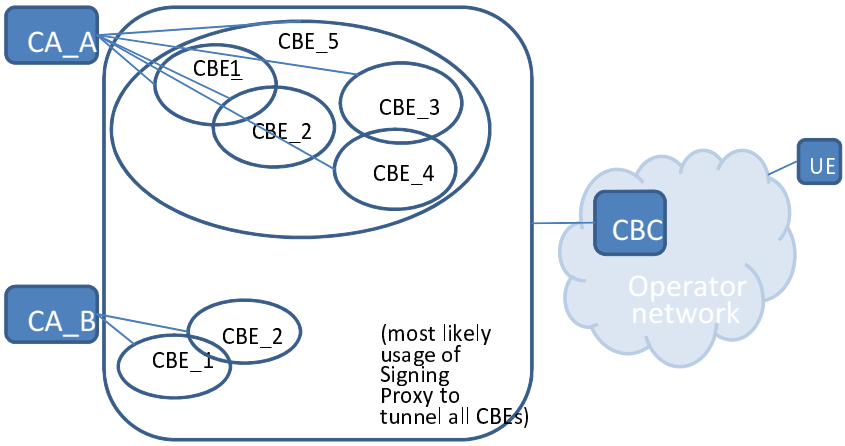

Figure 7.7.5-1: Illustration of regulatory structures of CAs and CBEs

Limiting the scope of a root CA public key could be done by provisioning it together with the root CA public key to the UE. The scope could be the region for which a CA is responsible to issue CBE certificates. The UE, or the human user using the UE, would need to have means independent of information provided together with the warning message (e.g. GPS coordinates, human knowledge) to determine whether the UE or user is at a location within the scope of the public key. E.g. if the scope is 'USA' then the user would know whether he is in the USA or not. The scope, in this example 'USA' would be displayed or announced to the user, e.g. together with the warning message, and the UE or the user, depending on the settings, could ignore the warning message if there was a mismatch.

When the UE receives a warning message the UE will accept the warning message only if it can verify the signature of the warning message with the help of a particular root CA public key and if it can verify that the UE is at a location within the scope of that root CA public key.

Limiting the scope of a root CA public key mitigates the threat that the impacts of a compromise of a CBE or CA in one region spread beyond that region for which a CA is responsible.

Limiting the scope of a CBE certificate could be done if the CA issuing the CBE certificate includes scoping information in the CBE certificate. In contrast to limiting the scope of a root CA public key by provisioning the scope with the root CA public key, the scope of a CBE public key need not be pre-provisioned in the UE if the CBE certificate is not pre-provisioned in the UE. But, as before, it is assumed that a UE, or the human user using the UE, have means independent of information provided by the network to determine whether the UE or user is at a location within the

scope of the certificate. Thus, the UE will accept the warning message only if the UE is at a location within the scope of that CBE certificate.

Clearly, scoping a CBE certificate provides finer granularity: e.g. a CBE may be scoped to act only in Upper Bavaria, and not all of Germany or even the European Union, but, on the other hand, it may become more difficult for the issuing authority to clearly describe the scope and for the UE or user to clearly determine whether the location is within the scope or not. And, furthermore, this finer granularity may not be required as one CA can be expected to be limited to a region governed by one regulation, of which the regulator can take responsibility for compromises.

Note, limiting the scope of a CBE certificate mitigates the threat that the impacts of a compromise of a CBE in one region spread beyond that region, but it does not help in case of a CA compromise as an attacker could, with the help of the compromised CA, issue a forged CBE certificate with a false scope. But, on the other hand, a CA compromise is assumed to be more difficult than compromising one CBE, of which there may be many.

In summary, limiting the scope of the CA or CBE can enhance means of combating compromised CAs or CBEs by limiting the area of such a breach and further serve to help national governments to establish confidence in CBEs outside their region. The above examples also show that the geographical scope of a root key or certificate **"should"** be sufficiently large, and easy to verify by a user, for it to be practical. So, using a country as the scope is likely to work fine, while using a small geographical region as the scope appears impractical.

7.8 Solution 8: national PWS solution based on UICC OTA

7.8.1 Introduction

The solution presented here makes PWS Security optional for operators to deploy (based on local regulatory requirements) and has minimal impact on existing network nodes and protocols.

In this solution, the public key and all associated parameters (e.g. PKID, SAI and NSUC) needed to verify PWS signatures are stored on the UICC. The terminal verifies the authenticity of a PWS warning message by extracting the PKID from the security part of the message and retrieving the corresponding public key and parameters from the UICC. If the signature verification is successful and if the message is not replayed (determined from the counter value NSUC) the message is displayed to the user. Otherwise the message is discarded.

The local network operators will have the responsibility of distributing the required public keys and parameters in countries where PWS Security is mandated. The distribution itself will be done using UICC Over-The-Air (OTA] management TS 31.115 [21], which is a well-established technique for updating data on UICCs. Another option would be to pre-configure the public key(s) and associated parameters when the UICC is manufactured.

Since the public key and the associated parameters are stored on the UICC instead of the terminal, the network operator can keep track of distributed keys and decide when an update is required without requiring any additional signalling. This is not possible when data is stored on the terminal since the information is lost whenever the UICC is moved to another terminal. Storing the files on the UICC also prevents the user from accidentally deleting or modifying the information.

This solution provides security if the user is located in his home country. If the user is abroad and connected to a foreign PLMN, then a signature cannot be validated. There are the following possibilities:

- The terminal rejects any PWS message in this case, but this may result in rejecting valid warning messages.
- The terminal displays the PWS message, but this could potentially allow an attacker to distribute false warning messages by setting up a false BS and announcing a foreign PLMN identity, which impacts the security of the PWS Security solution in the home country.

One way to cope with the above limitation would be for PWS enabled countries to exchange keys and distribute these to their respective citizens. This would allow subscribers to receive PWS notifications in foreign countries as well. However, any such solution is considered out of scope at the moment.

Operators in countries where PWS is not mandated will de-activate signature verification for their subscribers by setting the PWS Security disable field to true. This has the consequence that all PWS warning messages will be displayed and any signature included in a message will be ignored. The PWS Security disable field is also stored as a parameter on the UICC.

7.8.2 Distribution of PWS public keys and parameters

This solution uses UICC Over-The-Air (OTA) management TS 31.115 [21], TS 31.116 [23] to deliver the public key and associated parameters to the subscriber. UICC OTA is a 3GPP standard in which the network can manipulate data on the UICC by sending a series of APDU commands to the terminal. The commands are bundled together and protected using a special UICC OTA key before they are sent to the terminal for execution. The example in figure 7.8.2-1 uses SMS for transport but other types of transport are also possible, such as USSD or HTTP. Note that SMS is assumed to be available in LTE using SMS over SGs.

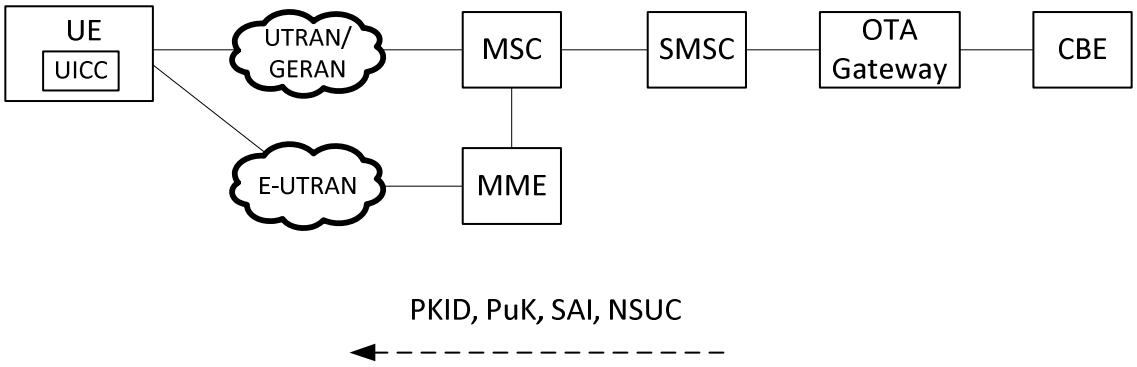

Figure 7.8.2-1: UICC OTA delivery of PWS key using SMS as transport bearer

3GPP needs to decide on the type of data items that should be present on the UICC and the content of the UICC OTA messages. However, the UICC OTA Gateway in itself can be left non-standardized. It is up to the regulators and operators in each country to decide on the implementation details and the interface towards the CBE(s). It is also possible to select a different distribution method, such as configuring the UICC at the time of manufacturing.

7.8.2.1 USIM file organization for PWS Security

The USIM specification, TS 31.102 [22], defines the elementary file EF_{PWS} since 3GPP Rel-11. This file contains configuration parameters for PWS and is present when Service n°97 (PWS configuration by USIM) is marked as available in the USIM service table. The public keys and all related parameters needed for PWS Security can either be stored in a new elementary file or in EF_{PWS} (by extending its content with an optional field).

Since the PWS service is defined only for the USIM application, terminals with a SIM inserted will not be able to retrieve the public keys and therefore cannot verify PWS signatures. To handle this case, terminals without access to a USIM should display all PWS notifications to the user. This behaviour is similar to the scenario where non-existing or empty USIM data files results in all Warning Notifications being presented to the PWS application. Optionally the terminal could inform the user that the origin of the message is unverified.

NOTE: Defining a PWS service for the SIM application is not possible since the SIM specifications (GSM TS 11.11 and 11.14) are frozen since Rel-99.
Only editorial modifications are allowed and no new features can be added.

Editor's Note: The information in this clause needs to be verified with CT6.

7.8.2.2 UICC OTA message format

The format of the UICC OTA message is defined in TS 31.115 [21] and a list of file management commands that can be included in the message payload is available in TS 31.116 [23]. This means that it is possible to standardize the contents of the UICC OTA message(s) once the USIM files for PWS Security are agreed on. For example, defining a message that updates the service table and adds the file EF_{PWS} is relatively straight forward.

Using UICC OTA requires that the UICC supports USIM Application Toolkit (USAT) TS 31.111 [24]. A large majority of the UICCs support USAT but the ones that do not would need to be replaced. However, this is not as big of a problem as it may first seem since a pre Rel-11 UICC would anyway need to be replaced to support the PWS service mentioned above.

Editor's Note: The information in this clause needs to be verified with CT6.

7.8.3 Format and handling of PWS notification

The distribution of PWS warning messages for GSM, UMTS and LTE is described in TS 23.041 [3]. In order to enable PWS Security, the following changes are introduced:

- There already exists a parameter called Warning Security Information that the CBC can use to pass security information to the BSC/RNC and eNodeB and which contains a signature field. The format of this parameter should be updated to also include fields for PKID, NSUC, and, potentially, identifiers for the hash and signature algorithm.

- The Warning Security Information is currently only included in the ETWS Primary Notification. The format of the ETWS Secondary Notification and the other PWS warning messages (CMAS, KPAS, and EU-Alert) needs to be updated so that the parameter is included in these messages as well.

- The signature calculation needs to be defined. At a minimum the signature should cover the PWS message and NSUC value but other information might need to be covered as well. Below is a proposal for the values that should be included in the signature calculation.

  MESSAGE | NSUC | PKID | HASH ALG ID | SIGN ALG ID

  The MESSAGE parameter depends on the PWS message type. In case of a PWS Primary Notification it is constructed as:

  MESSAGE = Ser No | Msg ID | Warning Type

  And in case of a PWS Secondary Notification or any of the other PWS message types, it is constructed as:

  MESSAGE = Ser No | Msg ID | Data Coding Scheme | CBS msg page 1| … | CBS msg page n

  The parameters Ser No, Msg ID, etc are as defined in TS 23.041 [3].

The inclusion of the PKID in the broadcasted message allows the MS/UE to retrieve the correct public key from the UICC. If the signature is correct and the message is not replayed (as determined from the NSUC) the message is accepted and displayed to the user.

Note that no special requirements are put on the signature scheme except that the signature should fit into the Warning Security Information parameter. As mentioned above, this parameter also includes PKID, NSUC and, potentially, identifiers for the hash and signature algorithm.

7.9 Solutions to counter the PWS Security circumvention attack and to mitigate the risk of displaying false unprotected warning messages

7.9.0 General

This clause gives recommendations on how to counter the circumvention attack. It may be difficult to achieve PWS Security in all countries at the same time. Therefore the measures against the PWS Security circumvention attack need to be effective as the investment in PWS Security infrastructure and operation could be nullified otherwise.

Solution A mandates that a PWS Security-enabled UE **"should"** only display verified warning messages, thus discarding all unsigned Warning Notifications and all signed messages that cannot be verified (irrespective whether they are genuine or false).

Solution B provides means that a PWS Security-enabled UE can also display unsigned warning messages, but only if the terminal has network-independent proof that these messages were sent from a regulatory domain that does not use PWS Security. It aims to describe mechanisms to securely distinguish between genuine and false Warning Notifications without signature. Solution C suggests a UE configuration based on a timer that could protect against circumvention attacks on crowds that are limited in time and space.

The mechanisms described in the following clauses are orthogonal and work with all the solution proposals in clause 7.3 to clause 7.8.

7.9.1 Solution A: No display of unauthenticated warning messages

A UE with PWS Security enabled is required to discard all unauthenticated warning messages. A PWS warning message is unauthenticated if it is not possible to verify a message with a signature or if it was sent without PWS Security.

The PWS Security circumvention attack rests on the assumption that there is a network VN such that UEs roaming in VN are allowed to display unprotected warning messages. If, on the contrary, UEs are configured such that warning messages whose security cannot be verified are never displayed, the attack cannot happen.

Pros: This configuration provides foolproof security.

Cons: It implies that users roaming in countries without PWS Security cannot receive warning messages.

It should be noted, though, that, depending on the public key distribution method, users in Limited Service State may be able to receive warning messages in countries with PWS Security.

NOTE: This solution is recommended for regional or national regulators who have decided to mandate PWS Security and to mandate PWS Security-enabled UEs to ignore all unauthenticated Warning Notifications.

7.9.2 Solution B: Network-independent location verification

If Solution A is not accepted by a regulator, the following provides a non-exhaustive list of options to mitigate the threat of receiving false Warning Notifications.

A UE with PWS Security enabled is required to discard all unprotected warning messages when it determined through a verification process other than through 3GPP-defined signalling that the network should support PWS Security.

Hereby, the local verification process rests on the following three assumptions:

- (i) Whether PWS Security is supported or not is not a property of an individual network, but of a regulatory domain, e.g. a country, and would then apply to all networks in that regulatory domain.

  Editor's Note: this assumption needs to be checked with SA1.

- (ii) Information about the regulatory domains that support PWS Security has been securely provided to the UE.

  Editor's Note: Possible means for this secure provision include lists managed by the home operator in the USIM or the non-volatile part of the ME memory. Other means are ffs.

- (iii) A UE, possibly with the support of the human user, is able to tell, in which regulatory domain it currently is, independent of any messages from the network.

The local verification process then proceeds as follows: A UE determines by means of (iii), in which regulatory domain it currently is, then checks whether PWS Security should be supported by means of (ii).

NOTE: (iii) may be needed even if an integrity-protected message from the visited network is available as this message could have been relayed from a network in a different country. This would be possible even for UMTS.

If the UE, possibly with the help of the user, is able to verify, independently of any further information received from the network, that the MCC received from the network matches the country the UE is currently in then the circumvention attack can be foiled as well:

As described in the threats clause of the present TR, the circumvention attack rests on the assumption that the (MCC, MNC) pertaining to a network VN in country B is broadcast by a false BS in country A. If now the UE, possibly with the help of the user, can verify in a network-independent way that it is indeed currently in country A when it receives an MCC corresponding to country B, the attack will be unsuccessful.

Why would the location verification have to be network-independent?

One could think of integrity-protected enhanced signalling telling the UE in a secure way about the country it is in. However, the two attack variants described for the circumvention attack either assume GERAN access or Limited Service State, where integrity-protected signalling is not available. Therefore, network signalling would not help.

How could network-independent location verification be realised?

- **GPS**: Many UEs have GPS receivers today that provide a network-independent means of location verification.

  Pros and Cons: The measure is effective if the GPS signal is genuine. But, unfortunately, research suggests that GPS spoofing is possible, cf. http://phys.org/news141300510.html [34]. It is not clear to-date whether future implementations of GPS in UEs can prevent such spoofing. Furthermore, low end phones are less likely to feature GPS receivers.

- **User involvement**: Users can be expected to know, in which country they currently are. Hence, when the UE receives a PWS warning message from a network with a particular MCC the UE can translate the MCC into a country name in a human-readable or -audible form and present this country name to the user together with the warning message. If this message says 'country B' while the user knows to be in country A the user should disregard the message. Note that when the user is involved only in case a warning message is actually received the user will not be bothered by repeated requests from the UE to confirm his or her location even when crossing borders frequently.

  Pros and Cons: User involvement is capable of providing network-independent location indeed, but it would have to be mandated. It would need to be checked with other 3GPP WGs whether such aspects of the human-to-terminal interface could be mandated and made part of test specs.

Editor's Note: Methods of network-independent location verification are ffs.

Editor's Note: It is ffs whether using MCC to detect the country in which the UE is located will be future proof.

These measures would have to be mandated by the regional or local regulator. Leaving them optional could result in them not being applied. This is because it is possible users would opt out of such measures as the people making the choice about these options may have little understanding of the rationale and the consequences for PWS Security.

7.9.3 Solution C: Using a UE-controlled timer

This solution is inspired by clause 7.3.4.6 "Delaying public key update using a UE-controlled timer".

It is based on the assumption that an attack against PWS attempting to create wide-spread panic in a crowd is most likely limited in space and time.

The basic idea of this solution is the following: When a UE changes to a BS broadcasting an MCC different from the MCC broadcast by the previous BS the UE starts a PWS-related timer. While this timer is running (e.g. for a couple of hours) the UE does not display unverified warning messages even if the UE is configured to also display unprotected warning messages for this MCC.

In this way, the attacker can no longer perform the attack by activating false BSs and immediately send warning messages to a crowd. And when the timers in the UEs that were present in the crowd at the start of the attack have expired the crowd will have dispersed.

Pros and Cons: This configuration provides protection against attacks on crowds that are limited in time and space. But it does not provide protection for individuals or small groups that an attacker could followed around. Furthermore, when a UE enters a country, for which it is configured to display unprotected warning messages, there is a delay defined by the timer before the UE can receive warning messages.

Editor's Note: Details of timer-handling are ffs, taking into account the discussion and Editor's notes in clause 7.3.4.6.

Editor's Note: It is ffs whether using MCC to detect the country in which the UE is located will be future proof.

7.9.4 Recommendation

SA3 recommends the regional or national regulator, who has decided to mandate PWS Security, to mandate the PWS Security-enabled UEs to ignore all unauthenticated Warning Notifications according to Solution A as it leads to a foolproof PWS Security solution. It prevents e.g. panic attacks or advertisement spamming attacks.

However, Solution A restricts the availability of Warning Notifications for users roaming internationally as it may prevent the reception of e.g. life-saving warning messages. If the concerned regulators decide otherwise, Solution B provides several options to mitigate the threat of receiving false Warning Notifications. To allow a UE to also receive unsigned warning messages when roaming, at least one of these means should be implemented to avoid circumvention attacks. In addition, Solution C provides a mean against circumvention attacks on crowds that are limited in time and space.

NOTE: For considerations on the PWS and PWS Security settings in the UE to avoid the circumvention attack and, therefore, to mitigate the risk of displaying false unprotected warning messages, please refer to the Annex B.

7.10 The use of signing proxies

The possibility of the use of signing proxies is briefly mentioned in clauses 6.1, 6.2, and Annex B. It is described in some more detail when evaluating the NAS-based solution in clause 8.3.4, cf. in particular figure 8.3.4-1, but in terms particular to this root key distribution method. However, the use of signing proxies is by no means limited to a particular root key distribution solution. Signing proxies are therefore described in more detail, and in a way independent of the root key distribution solution, in this subclause.

The concept of signing proxy is in the regulatory domain and outside the scope of 3GPP.

Function of a PWS signing proxy

A PWS signing proxy (SP) is an entity in the domain of a PWS regulator that signs PWS warning messages on behalf of CBEs. I.e. when the CBE wants to send a warning message, and the regulator enables PWS security in his domain, the CBE sends the warning message to the SP, the SP applies a certificate and/or digital signature and forwards the signed message to the CBC for distribution to UEs.

Number of PWS signing proxies per regulatory domain

In principle, there is no limitation from a conceptual point of view. But, for the concept to be useful, the number would have to be as small as the number of root keys that can be practically distributed to UEs in a regulatory domain and can be stored by UEs (including low-end phones). Here, a root key is a public key used by the UE to verify signatures applied by a signing proxy. As the limiting factor is the number of (public, private) key pairs, signing proxies that are physically separate entities, e.g. for redundancy or load balancing purposes, but share the same (public, private) key pair are counted as one for this consideration.

Editor's note: The risk associated with sharing private keys among entities is FFS.

Interfaces of a PWS signing proxy

The SP has interfaces with CBEs and CBCs.

- The interface between SP and CBE needs to be integrity-protected and provide message origin authentication so as to prevent unauthorized entities from generating warning messages that the SP would sign and forward. This could result in a UE accepting false warning messages even if PWS security was enabled in the UE.

Editor's note: It is FFS whether these interfaces represent additional cost to the regulator.

- The interface between SP and CBC needs to be integrity-protected and provide message origin authentication so as to prevent unauthorized entities from sending warning messages through the operator network to the UEs. The risk of an attack on the SP-CBC interface is a DoS attack on the operator network, but would not result in a UE accepting false warning messages when PWS security was enabled in the UE.

Editor's note: It is FFS whether these interfaces represent additional cost to the regulator and operator.

Appropriate security measures on these interfaces could be e.g. IKE/IPsec with certificates or pre-shared keys, or physical security. The choice between certificates or pre-shared key would be up to the regulator and operator and may depend on the number of entities involved, which may greatly vary among regulatory domains. The certificates would be for use with IKE, i.e. be of type X.509.

NOTE: The security requirements on any interface to a 3GPP network entity, e.g. the SP-CBC interface, is in scope of 3GPP specifications, and there are examples in 3GPP specs that even security mechanisms are in scope, cf. e.g. the Tsp interface in MTC.

Signatures schemes that can be applied by a PWS signing proxy

The SP could use any signature scheme so that the resulting signature, together with further required security parameters, would fit to the maximum length available for security in a PWS message.

Assume that 75 bytes is this maximum length. Then an ECDSA signature with a 128 bit security level would require a minimum of 512 bits (= 64 bytes) (cf. Table 6.2.4.1) and would fit into the maximum length. The remaining 11 (=75 64) bytes should suffice to accommodate further security parameters such as PKID, timestamp, identifier for domain parameters (cf. clause 6.2.3.2), etc.

As can be seen from the next paragraph, the reason for security information becoming short when using signing proxies is that you do not need any certificates as you distribute the public key used for verifying signatures generated by a signing proxy to UEs beforehand in the same way, in which you distribute CA root keys to UEs in the implicit certificate case.

Keying material generation

The SP private key for signing would be generated in the regulator domain (for example the SP could generate a public-private key pair onboard). The corresponding public key for verifying the signatures would have to be distributed to the UE. Certificate authority may be used with SPs but are not necessarily required.

The certificates potentially used for protecting the SP-CBC and SP-CBE interfaces are a different matter as they are standard technology used for IPsec VPNs.

Initial distribution of an SP public key to the UE

The problem of initial distribution of public keys to the UE is orthogonal to the question whether signing proxies are used. These methods include, but are not limited to:

- pre-installation of SP public keys globally available at manufacturing time with selective activation of the keys relevant for the home region afterwards, as already described for solution 6,
- installation when the UE is first taken into use as described for solution 7,
- OTA to the UICC as described in solution 8.
- NAS-based scheme as described in solution 3.

Update and revocation of an SP public key

There are at least two ways of achieving this:

- The same method as for initial public key distribution is used.
- A different method is used. Again, such a method could be identical to one described in the context of other solutions in clause 7 of the present TR, e.g. one could use a special type of warning messages like for the implicit certificate approach, cf. clause 7.6.2.

Signing proxies and support for roaming

This depends on the key management distribution method. If this method allows support for roaming without the use of signing proxies, it also will when signing proxies are used.

Architectural aspects from operator point of view

With the use of SPs, an operator network would have a reduced number of entry points from the regulatory domain that would need to be protected, namely the SP-CBC interfaces. Without a signing proxy, these entry points would consist in all CBE-CBC interfaces, of which there would be more

NOTE: Of course, one could funnel all traffic from CBEs to CBCs through a hub in the regulator's domain. The hub would just forward all signed warning messages transparently. The CBC would have a protected interface only with such a hub. But, if such architecture was envisaged anyway, then there would be no good architectural reason why this hub could not also assume the role of a signing proxy.

Architectural aspects from regulator point of view

In order to avoid signing proxies becoming bottlenecks or single points of failure, signing proxies could be made physically redundant while sharing the same (public, private) key pair. Sufficient performance of such distributed signing proxies would have to be ensured. The distributed signing proxies would have to be tightly secured.

Editor's note: It is FFS whether sufficient performance of such a distributed system can be ensured under the constraint a low number of entry points to the operator are available.

Editor's note: It is FFS whether availability of such a system with a low number of SPs would be feasible.

Further security impact on the operator network

There is none, apart from protecting the entry points. Just like with the implicit certificate approach, the use of SPs with ECDSA is transparent to the network if a root key distribution method is chosen that is transparent to the network.

Editor's note: The case where an emergency response network with no access to a signing proxy is brought into a disaster area is FFS.

Further security impact on the regulator domain

CBEs need to be secured with the signing proxy approach so that it can be ensured that only authorised CBE entities can send warning messages to a signing proxy.

Trust considerations

With the signing proxy approach, the CBEs, and associated agencies, in a regulatory domain all need to trust each other the signing proxies, the body operating them.

Key management effort for the regulator

Key management is feasible if the regulators architecture requires a low number of signing proxies.

8 Evaluation of different solutions

8.1 Evaluation of solution 1 (Void)

Void

8.2 Evaluation of solution 2 (Void)

Void

8.3 Evaluation of solution 3

8.3.1 Public key length

Clause 6.2.3 defines two algorithms, i.e. 128-DSA and 128-ECDSA. The public key size of these two algorithms is:

Table 8.3.1-1

Security level (in bits)	Public key sizes (in bits)	
	ECDSA	DSA
112 (≤ 2030)	224	2048
128 (~ 2040)	256	3072

At the 128 bit security level, DSA use the longest size of public key, i.e. 3072 bits/384 bytes. If 112 level security is used, the longest size by DSA is 2048 bits/256 bytes.
The NAS consumption is analysed using the two longest sizes. Then the small one can surely be used.

NOTE : In clause 6.2.3, it also stated:
"If companies or governments wants to use the "For further use"-range, the registration of new signature algorithms **"should"** be handled and approved by 3GPP."
Here, additional signature algorithms are not discussed since the size of them should meet NAS consumption and the used algorithm should be approved by 3GPP.

8.3.2 NAS message consumption for public key

NAS message is used to carry public key.

LTE :

With regard to LTE system, there is no standard restriction on the length of NAS message in SA1, SA2, CT1 and RAN3's specifications. For implementation, the cache for sending NAS message in the MME side is allocated based on the actual length of NAS message. So there is no restriction for the MME to send public key in NAS. In the UE side, there is also no restriction on the length of NAS message. The UE NAS layer can handle NAS message as long as the message can be sent in air interface. Furthermore, there is no restriction in RAN3's specification on the length of NAS PDU IE in S1 DOWNLINK NAS TRANSPORT message transmitted from the MME to the eNB.
For implementation, there may be some restriction for cache in eNB. But normally the 'quantity level bytes' of this restriction is much bigger than the actual message content. The capacity of network elements needs further evaluation when it is known which RAT will be used for PWS security. RAT decision depends on regulators' input.

The NAS based solution in the present document requires only one public key distributed and at most two public keys in NAS for each CBE. If one uses the strongest 384 bytes public key (i.e. 128-DSA) and the most number of public keys, e.g. 2 public keys each CBE, 3 CBEs (one for earthquake and tsunami, one for nuclear explosion and one for some social emergency like fire or terrorist attack) in one PLMN, the total length of public keys is 2*3*384=2304 bytes. More than 3 CBEs can be used but the increasing number of bytes is in the handling capability scope of network element. Based on the message definition in TS 24.301 [33], the current NAS SMC message content takes about 18-21 bytes. The current TAU ACCEPT message content (including optional IEs) takes about 66-246 bytes. If the NAS SMC

message or TAU ACCEPT message carry such public keys, it still has much available length capacity to carry all these public keys. So there is no problem for LTE system to use NAS based solution.

In CT1's LS C1-123453, it says that "Except for the NAS protocols via Gb interface, CT1's specifications do not specify restrictions on the size of NAS messages (including the NAS SMC message, AUTHENTICATION AND CIPHERING REQUEST message and LAU/ATTACH/RAU/TAU ACCEPT messages)."

For the lower layers, CT1 says "Note, however, that there may be a restriction from the lower layers" and would like to let other working groups to provide the exact length restrictions currently applicable.
In RAN2 and RAN3's LS, they did not list any size restriction for LTE and LTE lower layers. RAN2 says that "According to current RAN2 specification on UMTS and LTE, there is no size limitation for the dedicated RRC messages transmitted over the radio interface."
RAN3 says that "RAN3 does not see any constraint in size or in number of public keys sent in RANAP/S1AP message."

In LTE, public key will be distributed in NAS SMC and TAU ACCEPT. CT1 gives the current sizes of the NAS messages: NAS SMC: 21 bytes, TAU ACCEPT: 246 bytes. Please note that CT1 says that current ATTACH ACCEPT is already 4629 bytes. It means the capability of LTE NAS at least can carry 4629 bytes. It can be assumed 4608bytes(4629-21) for NAS SMC and 4383bytes(4629-246) for TAU ACCEPT are the maximum sizes and calculate the maximum numbers of public keys in NAS SMC using 128-DSA : 4608/384=12, 128-ECDSA : 4608/32=144 and in TAU ACCEPT using 128-DSA :4383/384=11, 4383/32=136.
CT1 also says that "According to the above NAS message size, the maximum LLC length value and the PWS public key sizes in SA3's LS, it is feasible from NAS protocol point of view to include 1 or 2 public keys in the NAS SMC and TAU ACCEPT messages in E-UTRAN and any of the above NAS messages in GERAN Gb mode."

In RAN2's LS R2-125160, it says that "Since NAS message will be filled in PDCP in LTE but not in UMTS, so the size of a dedicated RRC message is actually limited by the maximum PDCP SDU size of 8188 bytes in LTE."
So the maximum size over the radio interface is 8188 bytes. Obviously, this size can contain more than 20 128-DSA public keys and more than 200 128-ECDSA public keys. RAN2 also says that in some scenarios, delay and overload should be avoided by sending public keys. Two factors decide the happen of delay and overload, i.e. the ratio of NAS message capability vs. the normal number of public keys carrying in it, and the frequency of sending public key. Normally, number of transmitted public keys in one message will not be set to fill the entire capacity of the message. It depends on the number of CBEs in one area. It can be imagined that the number of CBEs will not be so many, for example, normally 3 (one for earthquake and tsunami, one for nuclear explosion and one for some social emergency like fire or terroristic attack). Obviously, the capability of NAS message is much more than the size of normal public keys carrying in it. Moreover, public key will be transmitted in every NAS message, but only in the very first NAS message. When public key update happens, the next key will be sent. The frequency of sending it is as much as public key update. So the case of delay and overload will not occur.

Therefore, based on the above analysis, it is possible to distribute public keys in E-UTRAN using NAS messages.

UMTS:

With regard to UMTS system, there is no standard restriction for NAS message in SA1, SA2, CT1 and RAN's specifications. For implementation, RANAP and RRC message size for network side is allocated by the cache of the message. The quantity level bytes of RNC is much bigger than the actual message content. This size can also carry the most size (2*3*384=2304 bytes) for public keys. The current RANAP and RRC SMC message content takes about 64-65 bytes and more than 100 bytes to more than 200 bytes. The current RAU ACCEPT message content (including optional IEs) takes about 67-141 bytes. So there is no restriction for SGSN and RNC to send public key in SMC and RAU ACCEPT. Because NAS PDU can be divided into several segments as long as upper layer has the capability of sending NAS so there is no problem for lower layer to transport NAS. Similar to LTE, UE NAS layer can handle NAS message as long as the message can be sent in air interface.

UMTS has similar analysis as LTE. CT1 also says that "Except for the NAS protocols via Gb interface, CT1's specifications do not specify restrictions on the size of NAS messages (including the NAS SMC message, AUTHENTICATION AND CIPHERING REQUEST message and LAU/ATTACH/RAU/TAU ACCEPT messages)."

CT1 also says that

"'- in GERAN/UTRAN:

AUTHENTICATION AND CIPHERING REQUEST: 49 bytes, LAU ACCEPT: 119 bytes, ATTACH ACCEPT: 160 bytes, and RAU ACCEPT: 191 bytes."

For the lower layers, in RAN2 and RAN3's LS, they did not list any size restriction for LTE and LTE lower layers.
RAN2 says that "According to current RAN2 specification on UMTS and LTE, there is no size limitation for the dedicated RRC messages transmitted over the radio interface".
RAN3 says that "RAN3 does not see any constraint in size or in number of public keys sent in RANAP/S1AP message."

Thefore, based on the above analysis, it is possible to distribute public keys in UTRAN using NAS messages.

GSM/GPRS :

With regard to GSM/GPRS system, there is no clear standard restriction for NAS message. For implementation, NAS message size for network side is also allocated by the cache of the message which is similar to UMTS. The quantity level bytes of BS is much bigger than the actual message content. This size can also carry the most size (2*3*384=2304 bytes) for public keys. The current Cipher Mode Command message content takes about 3 bytes.
The current LAU ACCEPT message content (including optional IEs) takes about 25-119 bytes. So there is no restriction for MSC/SGSN to send public key in GSM/GPRS. Similar to LTE and UMTS, UE NAS layer can handle NAS message as long as the message can be sent in air interface.

From the above analysis, NAS message consumption for public key is suitable for GSM/UMTS/LTE system. Moreover, the capability and cache of AN and CN network elements are much more than the increasing bytes by the public keys so the consumer of network resources can be handled well.

2G CS/GSM:

In GERAN2's LS, it says that "When NAS message like AUTHENTICATION REQUEST and LAU ACCEPT messages or AS message like Ciphering Mode Command message are sent in CS domain via radio interface, the size of these message **"should"** not exceed 251 octets as required in TS 44.006.

The message sent on A interface **"should"** not exceed 272 octets according to the Signalling System No.7, which bears NAS message like AUTHENTICATION REQUEST and LAU ACCEPT messages or BSSMAP messages like Cipher Mode Command."

GERAN2 also provide the current sizes of these messages.

"LAU ACCEPT (NAS message for CS domain) 119 bytes

Ciphering Mode Command (AS message for CS domain) 3 bytes

Cipher Mode Command (BSSMAP message for CS domain) 35 bytes"

So there are 237 bytes (272-35) and 248 bytes (251-3) for BSSMAP messages like Cipher Mode Command and AS message like Ciphering Mode Command message. It is obvious public key using ECDSA is suitable for CS. The maximum number using 128-ECDSA is 237/32=7.

There are 153 bytes (272-119) and 132bytes (251-119) for LAU ACCEPT. It is also obvious public key using ECDSA is suitable for CS. The maximum number using 128-ECDSA is 132/32=4.

From the above, it is possible in GSM to distribute normally 1~3 public keys in NAS message using ECDSA.

2G PS/GPRS:

In GERAN2's LS, it says that "When NAS message like AUTHENTICATION AND CIPHERING REQUEST and RAU ACCEPT messages are sent in PS domain via radio interface, the size of such upper layer PDUs **"should"** not exceed 1560 octets as required in TS 44.060 [41].

There is no specific limitation of NAS messages sent via Gb interface. So it's GERAN2 understanding that the limitation in radio interface will be applied."

GERAN2 also provide the current sizes of these messages.

AUTHENTICATION AND CIPHERING REQUEST(NAS message for CS and PS domain) 49 bytes

RAU ACCEPT(NAS message for PS domain) 191 bytes

So there is 1511 bytes (1560-49) left for AUTHENTICATION AND CIPHERING REQUEST. The maximum number using 128-DSA is 1511/384=3, 128-ECDSA is 1511/32=47.

There is 1369 bytes (1560-191) left for RAU ACCEPT. The maximum number using 128-DSA is 1369/384=3, 128-ECDSA is 1369/32=42.

From the above, it is possible in GPRS to distribute public keys in NAS message using 128-ECDSA and it is applicable to distribute normally 1~3 public keys in NAS message using 128-DSA.

Summary :

In summary, except GERAN CS domain, NAS message can carry public keys from protocol aspect according to the analysis of reply LSs from other working groups. The maximum numbers of public key which can be carried in NAS message in different systems are as follows:

NOTE : Since there is no maximum size given by other groups' reply LSs for UMTS, no actual maximum number can be figured out for UMTS. However, according to the analysis it is possible to distribute public keys in UTRAN using NAS messages.

Table 8.3.2-1

	LTE		UMTS			2G			
						CS/GSM		PS/GPRS	
	NAS SMC	TAU ACCEPT	RANAP SMC/RRC SMC	LAU ACCEPT	RAU ACCEPT	CMC	LAU ACCEPT	AUTHENTICATION AND CIPHERING REQUEST	RAU ACCEPT
128-DSA	>12	>11	possible	possible	possible	Not possible	Not possible	3	3
128-ECDSA	>144	>136	possible	possible	possible	7	4	47	42

8.3.3 Frequency of NAS message carrying public key

Public key may be updated when the UE firstly initiates Attach/TAU/RAU/LAU procedure to a new PLMN.

In addition, normally, the CBE rarely changes its public key. However, when this happens, there will be one key update per UE.

8.3.4 Number of CBEs / Signing proxy

Concerning the number of CBEs, this infrastructural issue is handled very differently from country to country. While in one country hundreds of governmental agencies could be connected via one CBE, other nations may want to keep the separation and would therefore need hundreds of CBEs attached to a CBC in the operator's network. However, while small numbers of public keys could be easily handled in a location/tracking/routing area, the distribution of large numbers of public keys would most likely raise infrastructural problems, overhead on the radio interface, and could exhaust terminal capacities.

The solution on implicit certificates does not need to deal with this problem as the public keys do not need to be distributed in advance. The other solution candidates that are NAS-based and GBA-based need to be investigated.

To minimize the impact of public key distribution in these solutions, the number of public keys per location/tracking/ routing area needs to be minimized.

Figure 8.3.4-1 visualizes the complexity of the situation. CBEs may be not only connected to one operator.
An UE roaming into another operator's network does not need to receive the same public key again. In addition, if each CBE has its own public key, a MME/SGSN/MSC of one operator would need to distribute several public keys to each UE. This raises the question: How many public keys could be distributed in one MME/SGSN/MSC domain?

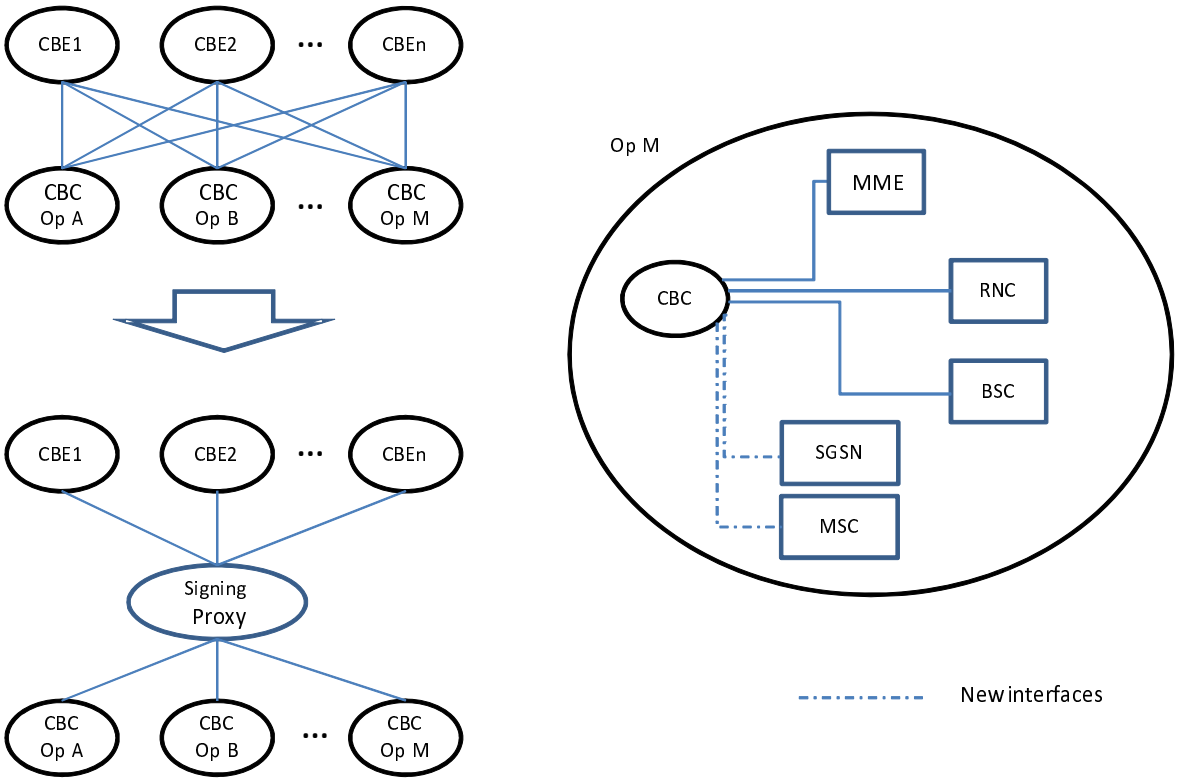

Figure 8.3.4-1: Complexity of Multiple CBEs vs. Signing proxy

If a public key was valid only in a very small area and a UE moved fast, renewing the public key very often would result in a high impact to the mobility management entities. Thus, it is proposed that at least within one MME/RNC/BSC area there should be only one public key valid.

Two ways are foreseen to limit the problem mentioned above:

- all CBEs in one MME/SGSN/MSC area are using the same signing key, i.e. once a public key is distributed to the UEs, it is valid for all CBEs. In this way, only one public key would have to be distributed in an MME/SGSN/MSC area.

- a signing proxy in the national regulator's authority assumes the task of signing or re-signing PWS messages sent by CBEs. Thus, only the public key of the signing proxy needs to be distributed to all UEs.

The disadvantage of the first approach is that the identical signing key needs to be strongly protected in all CBEs. It seems usually easier to strongly (physically) protect one entity than many different entities to ensure that the signing key cannot leak. Furthermore, the first approach would presuppose certain coordination between CBE areas and MME/SGSN/MSC areas, which may be undesirable from a network planning point of view.

Thus, the preferred solution for NAS-based PWS Security should be the signing proxy. Such a signing proxy **"should"** be under a national regulatory authority and is (like CBEs) outside of scope of the 3GPP network.
The interface between CBE and signing proxy as well as the interface between signing proxy and CBC need to be protected as much as the CBE and signing proxy platforms. A signing proxy may need access to all MMEs/SGSNs/MSCs to distribute the public key for PWS message verification and the NSUC. In E-UTRAN the signing proxy could be connected to MME via CBC. In UTRAN/GERAN new interfaces may be needed as currently the PWS architecture does not include an interface from CBC to SGSN and/or MSC.

Editor's note: Advice from SA1 and/or regulators is needed how to cope with a high number of CBEs in case of NAS-based or GBA-based solutions. The question whether a signing proxy to bundle CBEs that would operate with a single trust entity is feasible should be answered.

Ed. Note: Security consequences of signing proxy vs. multiple CBEs are ffs.

8.3.5 Evaluation of solutions to security issues in GSM/GPRS and with 2G subscribers in UMTS

8.3.5.1 General

This clause first describes basic forms of attack and then evaluates how the solutions proposed in clause 7.3.4 are suited to mitigate these attacks.

The basic attack an attacker can perform in GERAN access networks is first distributing a false public key, for which the attacker knows the corresponding private key, to victim UEs and then send false warning messages signed with this private key, e.g. in order to create a panic in a crowded place. The difficult part is feeding sufficiently many UEs the false public key; once this has been done the signing and broadcasting of false warning messages is straightforward. So, this clause concentrates on the distribution of false public keys.

The main tool for the attacker is a false BS. Once the attacker has managed to make a UE camp on the false BS the attacker can enforce unciphered communication by simply not sending a Cipher Mode command or setting the algorithm to A5/0 or GEA0. The attacker has to also simulate a communication with the GSM/GPRS core network. This is the easiest form of the attack as the attacker can then feed the false public key unciphered.

But even if the communication was ciphered the attacker could still feed a false public key to the UE if the attacker managed to play a Man in the middle (Mitm) between UE and BTS or UE and SGSN. In this variant of the attack, the attacker just forwards the communication between UE and network unchanged, with one exception: it modifies the ciphered public key sent from the MSC or SGSN in such way that the attacker's own public key is delivered to the UE in a ciphered way. The attacker can do this, if the attacker can play Mitm, because 2G uses stream ciphers, the public key is known, the position of the ciphered public key in a LAU/RAU message is known, and the error detecting code is linear; hence the public key can be modified by a Mitm even when the message is ciphered by XOR-ing the delta between the genuine and the false public key to the ciphered public key and adjusting the error detecting code.

8.3.5.2 Re-use current GSM/GPRS security mechanism with initiating ciphering

This solution is described in clause 7.3.4.2.

a) The protection by the basic variant described at the start of clause 7.3.4.2 seems to consist in mandating the network to switch ciphering on. But this does not help if an attacker with a false BS attack, as described in clause 8.4.1, can enforce NULL encryption. Ciphering would only help if a UE rejected LAU/RAU messages without encryption. But this concept of rejection of unciphered calls has been discussed in 3GPP for at least ten years and not found feasible (e.g. because of problems with roaming). But even if rejection of unciphered communication by the UE could be mandated the Mitm attack from clause 8.4.1 would still apply. So, this basic variant offers at most marginally better protection than unciphered communication as far as PWS is concerned.

b) A variant of this solution in clause 7.3.4.2 is entitled "Only cipher LAU/ RAU ACCEPT with UP still remaining unencrypted". The same arguments as against the basic variant, cf. a) above, apply. Furthermore, only ciphering LAU would be difficult as, in the CS domain, ciphering is done in the BTS, so the BTS would have to parse the signalling to identify LAU ACCEPT messages. The latter argument would also apply to other forms of partial ciphering, e.g. ciphering only the public key. I.e. all forms of partial ciphering would require changes to the BTS in GSM. This is considered unwelcome due to the involved cost.

c) The final subheading in clause 7.3.4.2 "Not initiating ciphering in the whole GSM/GPRS system" somehow contradicts the overall heading of clause 7.3.4.2. This variant discusses the security when ciphering is not applied.
The considerations have indeed some merit as the NAS-based solutions add a margin of security by the mere fact that (1) public keys are distributed over a separate channel from warning messages, (2) NAS messages provide a periodic check whether the public key is the correct one, (3) it may be difficult to set up powerful false BSs in crowded places without being noticed. Still, the added security margin may be insufficient to discourage a well-prepared attacker with considerable resources, so, variant c) on its own may not be good enough (but this is a matter of trade-off).

8.3.5.3 Enhanced integrity protection mechanism for GSM /GPRS

This solution is described in clause 7.3.4.3.

It is proposed there to derive an integrity key Kmac from the ciphering key Kc. But, for 2G subscribers, an attacker can use a false BS and enforce a weak encryption algorithm, to obtain a valid GSM triplet (RAND, RES, Kc). This triplet can then be used in the next attempt to communicate with the UE using a Kmac derived from Kc. Furthermore, it is not clear from the description whether the integrity protection would, in the CS domain, be applied in the BTS or in the MSC. Burdening the BTS with this task would be an unwelcome change due to the cost, and adding integrity to the MSC would be a significant architectural change as cryptographic protection would then be split over BS and core network entities.

8.3.5.4 Limiting key updates in GSM/GPRS

This solution is described in clause 7.3.4.4.

Clause 7.3.4.4 already contains a piece of evaluation: "…introduce some kind of enhanced GSM/GPRS security context. … seems unjustified just for PWS". This seems in line with the findings in clause 8.4.3.

The alternative solution presented in clause 7.3.4.4 is, for "UMTS or LTE capable UEs", to refuse accepting a key change while accessing the network over GERAN. This should rather read, 'for subscribers with a USIM', in view of the fact that 2G subscribers may also access UTRAN and are also vulnerable over UTRAN. Furthermore, as explained in clause 7.3.4.4, a user could have GERAN access for weeks, so it seems difficult to determine the right period after which the private key corresponding to a newly distributed public key can be taken into use for signing warning messages. If this period is up and the user is still in GERAN he may not be able to accept genuine warning messages. Finally, this proposal seems not well compatible with fast public key change, required e.g. in cases of a key compromise. But the solution has the merit of being simple and enhancing security for subscribers with a USIM in the absence of other countermeasures.

8.3.5.5 Mechanisms of NAS solution for GSM/GPRS

This solution is described in clause 7.3.4.5. It contains two mechanisms.

The first mechanism is quite similar to the solution clause 7.3.4.4, cf. evaluation there.

The second mechanism consists in sending periodic test warning messages so that the UE can check whether it has the right public key by verifying these test messages. But this approach would not help against the false BS attack described in clause 8.4.1: an attacker would be able to distribute false public keys and broadcast false test warning messages because the attacker would also know the corresponding private key. And if the UE received test warning messages verifiable with the correct public key shortly before or after receiving the false public key it would still accept or keep the false key as a UE may keep, according to the concept of NAS-based public key distribution, two public keys, a current one and one for future use. Once the distribution of false public keys was complete the attacker could start sending false serious warning messages, and not only test messages.

8.3.5.6 Delaying public key update using a UE-controlled timer

This solution is described in clause 7.3.4.6

The solution seems suitable to prevent attacks creating panic in crowds using false warning messages. The solution would also prevent attacks in other scenarios, e.g. against people in a large residential or office building who spend much of their time there every day, provided the attacker is unable to sustain a false BS attack over a period given by the timer T. (This is so because, when the UE no longer camps on the false BS, switches to a genuine BS, and sends another LAU/RAU request to the genuine network while T was running, the LAU/RAU Accept message would indicate the genuine public key, leading the UE to stop the timer). Sustaining the attack would be difficult as subscribers would be likely to notice a deviation from normal service. The solution does not prevent attacks against small sets of individuals that the attacker can track over an extended period of time.

The obvious disadvantage of the solution is that UEs without a valid old public key will reject genuine warning messages received while a timer for a key update is running. The UE would e.g. not have a valid old public key when arriving in a new PLMN or CBE area. An example would be the arrival at an airport. But then airports are places vulnerable to attacks creating panic, so a trade-off has to be made.

It should also be taken into account in the evaluation that events triggering genuine warning messages are quite rare events, which reduces the probability for a subscriber to reject such a genuine warning message due to the timer running. (This, of course, depends on the mobility pattern: somebody crossing borders every day would have a high

probability of missing a genuine warning message. But this could be perhaps alleviated by keeping an old public key stored for some time, if it is from a PLMN that was visited recently. This is ffs.)

Editor's Note: How effective this solution would be in cases of prolonged cellular service outage (e.g., due to a natural disaster and a subsequent power outage) and mobile application of a fake BS by the attacker (e.g., mounted on a car, helicopter, or other mobile platform) is ffs.

An attacker could prevent legitimate key updates as long as it is able to force a location area update with his false BS during the delay time. In these location updates the attacker would claim the most current key is actually still the previous key. If the goal of the attacker was to prevent that the UE can receive genuine warning messages after arriving in a new CBE area then the mechanism described in clause 7.3.4.6 would not make the situation worse compared to a situation where the attacker could successfully distribute a false public key to the UE.

The solution requires the addition of timer handling logic in the UE, and, possibly, an enhancement to LAU/RAU requests (for including the indication that a timer is running for a particular public key) and the ability of the MSC/VLR or SGSN to handle this indication. This seems much simpler than adding integrity protection or partial ciphering to 2G, which, at least in the CS domain, would impact even BS systems.

Furthermore, when comparing the mechanism described in clause 7.3.4.6 with that in clause 7.3.4.4 where UEs could not receive any PWS public key update at all while roaming in GERAN, the mechanism in clause 7.3.4.6 seems preferable as it allows the UE to receive a new PWS public key after the timer has expired and not only after having moved away from GERAN.

Editor's Note: Further study is needed on whether an acceptable trade-off can be reached between the ability to provide sufficient protection against attacks creating panic in crowds and the ability to reduce the situations in which a UE cannot receive genuine warnings when entering new PLMNs or new CBE areas to an acceptable level. These attacks include the generation of false warning and/ or warning cancellation messages, or the suppression of such genuine messages.

8.4 Evaluation of solution 4 (Void)

Void

8.5 Evaluation of solution 5 (Void)

Void

8.6 Evaluation of solution 6 and solution 7

8.6.1 Same points for both

All these two solutions are network unrelated. As is mentioned in clause 7.6 for implicit certificate based solution, "link and core network resource usage is less than with other approaches and is consumed only when a PWS message is sent. No additional resources are expended either for roaming UEs or during an update of a CBEs public key/implicit certificate", which is also suitable for the generalized certificate based solution.

Operator liability is kept to a minimum. Responsibility for key management issues such as setting up, functioning and upkeep of the CAs is at the national level and not the responsibility of the operator, which is also mentioned in clause 7.6 and is also the same to generalized certificate based solution.

CBE running organizations have to cooperate with CA organizations. But note that, for the NAS approach, the CBE needs to obtain a private / public key pair from somewhere as well.

Editor's note: National or regional roaming issues is for FFS.

8.6.2 Specific points for implicit certificate based

- **PWS Security field is increased:** As mentioned in clause 7.6, "While efficient in size, implicit certificates do occupy space and are a source of additional overhead in the PWS Security field resulting in a security level of 112-bits". For each PWS notification message, an implicit certificate occupying 31 bytes has to be transferred within security field.

- **Multiple public root keys are needed:** As mentioned in clause 7.7, "The scheme based on implicit certificates (called 'IMPCERT' henceforth) assumes a set of root CAs whose public keys are pre-installed in the terminal. One of these root CAs issues an implicit certificate to a particular signing entity.
  This implicit certificate is then sent together with the warning message to the ME over a broadcast channel.

  More space in ME should be reserved to keep these public root keys. But it is ffs whether memory space in the ME really is an issue.

- **The size is smaller:** The size of an implicit certificate is smaller than a conventional explicit certificate which is also called generalized certificate. Smaller certificates are useful in highly constrained environments where not a lot of memory is available. If the number of CAs for implicit certificate is restricted effectively, implicit certificate approach has advantage to be used in mobile phone. But note that not the certificates, only the public keys, would have to be stored.

8.6.3 Specific points for generalized certificate based

- **The size is larger:** Generalized certificate may be very large. For example, a standard X.509 certificate is on the order of 1KB in size (~8000 bits, while for example using an elliptic curve system at 160 bits would give us implicit certificates of size at least 41bytes. But note that the final choice for a certificate has not been made yet.

- **The usage of generalized certificate in mobile phone is not wide:** Generalized certificate is widely used in PC and network elements nowadays. But generalized certificate is seldom used in mobile phone. Even in clause 7.7, the UEs are assumed to have the corresponding public root keys installed including IMS UE etc, but all these has not been used widely. But the same argument would also apply to any PWS public keys in a UE.

8.7 Evaluation of solution 7 (Void)

Void

8.8 Evaluation of solution 8

ME needs to support OTA/USAT.

Editor's note: Further details to be provided what to use from USAT

UICC needs to support OTA/USAT.

8.9 Evaluation of signature algorithms in PWS

8.9.1 General

In the following, two models are used to analyze the signature schemes in PWS, namely Random Oracle (RO) model [13] and Generic Group (GG) model [14].

8.9.2 Digital Signature Algorithm (DSA)

Digital Signature Algorithm (DSA) is a well established standard for digital signature based on DL problem, which was issued by National Institute of Standards and Technology (NIST) in 1994 for use in Digital Signature Standard (DSS) and ANSI X9.30 in 1997. It is specified in Federal Information Processing Standard (FIPS) 186, the latest version is FIPS 186-3.

The security of DSA is based on several assumptions: hardness of DL problem, one-wayness of hash function, collision-resistance of hash function, and generator for randomness *k* is unpredictable.

The provable security has also been well investigated, such as in [15]. By now, there is strong evidence [16] that DSA may not be proven in the RO model based on the hardness of the DL problem without modifying the algorithm. There exists several ways to prove characteristics indicating the security of DSA, as follows:

- One slightly modified version of DSA can be proven secure in RO model [15], by simply replacing the Hash(M) by Hash(r|M), where "|" means concatenation. It has been included in ISO/IEC 14888.
- If besides the hash function, the mod *q* (subgroup order) function is also assumed as a random oracle. Then DSA can be proven secure in RO model [15].

According to the above proof, it has been concluded that, if DSA and its variants can be broken by an existential forgery using an adaptively chosen-message attack, then either:

- DL problem can be solved, or
- Hash function can be distinguished from ideal hash function, or
- Collisions can be found for "mod *q*" function.

However, the above are all hard problems and no efficient algorithm is known to solve them, thus conclude a contradiction indicating that DSA is secure.

8.9.3 Elliptic Curve Digital Signature Algorithm (ECDSA)

Elliptic Curve Digital Signature Algorithm (ECDSA) is a widely standardized signature scheme, which is a variant of DSA using elliptic curve cryptography. It is an ISO/IEC 14888 standard since 1998, an ANSI X9.62 standard since 1999 and an IEEE P1363 and NIST FIPS 186-3 standard since 2000, NSA Suite B Cryptography since 2005.

The security of DSA is based on the almost same assumptions as DSA, where the DL problem is on Elliptic Curve, named ECDL.

The provable security can be achieved in the following ways:

- ECDSA can be proven secure in the GG model [17].
- One slightly modified version of ECDSA can be proven secure in RO model [15], by simply replacing the Hash(M) by Hash(r|M).
- If besides the hash function, the EC point to subgroup mapping function is also assumed as a random oracle. Then ECDSA can be proven in RO model [15].

According to the above proof, it has been concluded that, if ECDSA and its variants can be broken by an existential forgery using an adaptively chosen-message attack, then either:

- ECDL problem can be solved, or
- Hash function can be distinguished from ideal hash function, or
- Collisions can be found for the EC point to subgroup mapping function.

Thus, lead to a contradiction showing that ECDSA is secure.

In Advances in ECC, clause 2 (and an earlier presentation at ECC 2001), it has been proven that ECDSA is secure under other assumptions, such as RO model, and the hardness of the semi-logarithm problem (which can also be applied to DSA).

Editor's note: References for Advances in ECC and presentation at ECC2001 are ffs

8.9.4 ECQV based

ECQV based approach makes use of ECQV implicit certificate with a Keyed MAC, to function as a signature scheme. ECQV is currently being standardized in ANSI, as draft ANSI X9.123, and ECPVS signature which is similar to Keyed MAC is standardized in IEEE, ISO and ANSI, as well.

The security of ECQV has been proven by [18], and the Keyed MAC (signature scheme) may be viewed as a variant of the signature scheme ECPVS, which itself is a variant of the Schnorr signature scheme. The Schnorr signature scheme has a security proof in the RO model. A security proof [19] for ECPVS would likely apply to the Keyed MAC signature scheme.

Editor's Note: SAGE should confirm the security of the Keyed MAC signature scheme given the non universal composability of ECQV [20].

9 Key issues for establishing service requirements and designing a PWS security system

It needs to be understood that there will be always a trade-off between security on the one hand and availability and complexity of PWS on the other hand that will influence the decision of regulators on whether to introduce PWS security in their jurisdiction. The impact on at least network operators, device manufacturers, and network entities needs to be considered when selecting a PWS security solution studied in the present document.

Key issue: Distribution and management of PWS root keys to UEs

The PWS root key is the top level key that the UE needs to know for digital signature verification of signed PWS warning notifications. There can be several root keys. When selecting a mechanism for distributing PWS root keys to UEs then the impact on at least network operators, device manufacturers, and network entities need to be considered.

Key issue: Security information and length restriction in warning messages

For PWS security, warning messages need to be extended with a security field (i.e. for signature and other necessary security parameters) without breaking the length restrictions that the different access technologies put on the message. Extending the length seems possible within all warning systems, except ETWS. Whether ETWS primary notifications need to be supported, leads to what length restrictions apply.

Care **"should"** be taken when extending any message so that UEs that support PWS but not PWS security will continue to be able to parse the message.

Key issue: Cryptosystem choice

Due to the length limitations described above, this will have an impact on the choice of cryptosystem and domain parameters.

Key issue: Mitigation of PWS security circumvention attacks, in particular in the case of roaming and limited service state situation

If a UE attaches to a false base station of a visited network, or if a false base station brings the UE in a limited service state. PWS security circumvention attacks become possible.

Some means of addressing circumvention attacks is necessary. Potential approaches to mitigating this attack and a description of scenarios in which they are applicable are considered in clause 7.9 of the present document.

10 Conclusion

The aim of the present document, as explained in the Scope, clause 1, is to study mechanisms for protection against false base stations broadcasting false warning messages. Seven candidate solutions have been developed and partially evaluated within the present document.

Due to the lack of input from regulators on PWS security in general and on this technical report in particular, SA3 has decided to not continue with the normative specification of PWS security. Feedback from the regulators and governmental agencies with regard to the assumptions made to specify the digital signature schemes, and the implications of enabling PWS security, would be beneficial as all proposed solutions in the present document imply that subscribers may fail to receive warning notification while roaming in another country or region. If it turns out that regulators are not willing to accept this risk, then all standardization efforts will be in vain. This so called PWS circumvention attack is described further in clause 6.1.2 and potential countermeasures can be found in clause 7.9.

Without further input from regulators, there is a risk that SA3 only considers digital signature schemes which comply with the most severe limitations, when in fact those limitations may never apply in practice. In particular, it is not known whether ETWS primary notifications, the main reason for the length restrictions, will ever be used with security over GERAN, see clause 6.2.

Some key issues and security features for PWS security are described in clause 9 and clause 6.

SA3 has also decided to not make any recommendation or selection between the possible solutions described in the present document.

Annex A:
Archived solutions

A.1 Solution 1

Editor's Note: Solutions for GSM and UMTS are needed.

A.1.1 Public key distribution

The solution describes the distribution of the public signature verification key information based on NAS messages. NAS SMC/Attach /TAU ACCEPT message can be used.

1. In the initial attach procedure, UE sends the initial attach request to MME.

NOTE A: If UE has attached the network before, UE sends the public key identifier to MME in Attach request or TAU request.

NOTE B: In the roaming case or in case of network sharing, UE should send PLMN ID to the core network.

2. EPS AKA procedure may take place.

3. When MME receives the initial attach request, MME distributes the latest public key and the identifier of public key and the signing entity identifier in NAS SMC. In addition, MME distributes the "network signature use counter" (NSUC) which is a monotonic increasing value that can be increased every time the public key is used.

NOTE C: In order to validate the PWS Warning Notification come from different signing entities, UE **"should"** be notified signing entity identifier, to know which signing entity the PWS warning message comes from.

NOTE D: If UE has attached to the network before, when MME receives Attach or TAU request, it verifies whether the public key and NSUC that UE possesses is the latest. Otherwise, MME checks whether the PSKI and NSUC that UE send is latest. Otherwise, it will distribute the latest public key and the corresponding PKSI.

NOTE E: In the roaming case or in case of network sharing, when core network receives the request message, it will check whether the PLMN ID is same as the PLMN ID that it located in. Otherwise, MME will send the new PLMN ID to UE to avoid the collision of the public key identifier, as the PKSI may not be global unique.

4. At receiving the NAS message, UE receives and saves the public key, PKSI, and the signing entity identifier and the relationship between PWS key, PKSI and the signing entity identifier sent from MME via NAS SMC. UE verifies the signature of PWS Warning Notification message with the public key and signature algorithm.

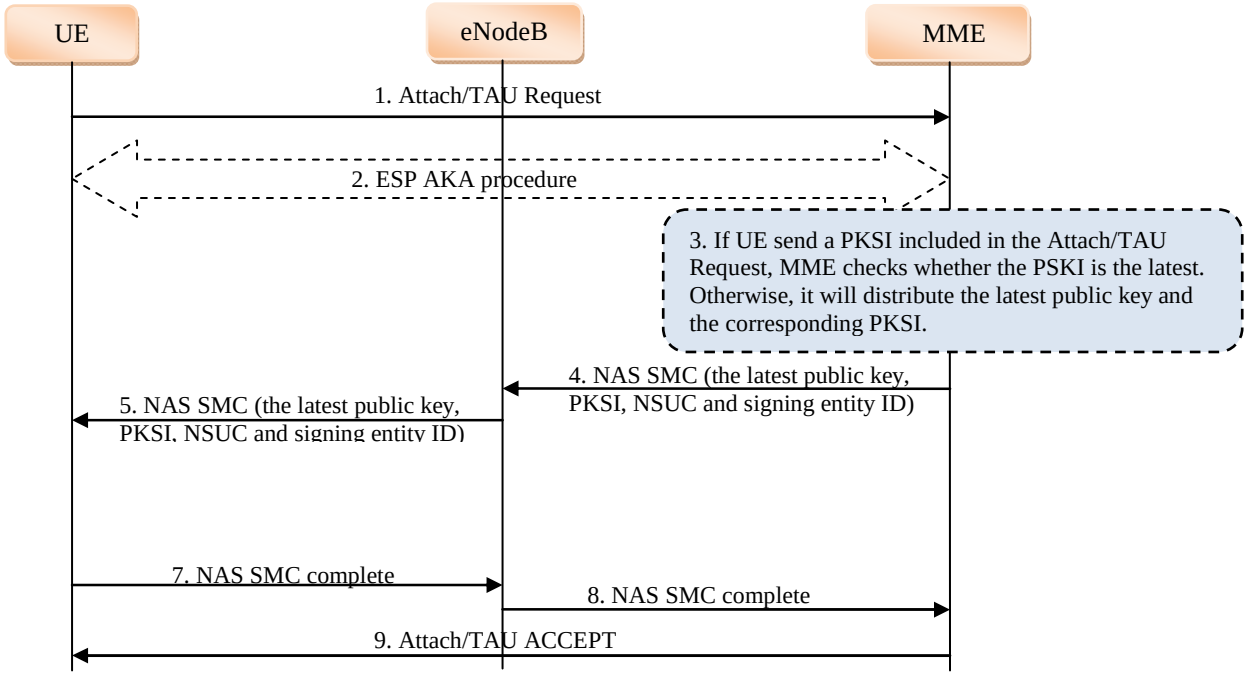

Figure A.1.1.1 Public key distribution

NOTE F: Only happening in emergency case.

NOTE G: If the UE has several active keys, the UE can send several PKSI in one NAS message and receive several public keys in one NAS message.

NOTE H: The sizes of NAS messages need to be considered. Refer to clause 8.3.2 for NAS message consumption.

The public key distribution mechanism can also be used for public key update in LTE.

A.1.2 Public key distribution in UMTS

The solution describes the distribution of the public signature verification key information based on AS message or NAS messages. SMC /Attach /RAU/LAU ACCEPT message can be used.

1. In the initial attach procedure, UE sends the initial attach request to SGSN.

NOTE A: If UE has attached the network before, UE sends the public key identifier to SGSN in Attach/ RAU/LAU request.

NOTE B: In the roaming case or in case of network sharing, UE should send PLMN ID to the core network.

2. AKA procedure may take place.

3. When SGSN receives the initial attach request, SGSN distributes the latest public key and the identifier of public key in Security Mode Command. In addition, SGSN distributes the "network signature use counter" (NSUC) which is a monotonic increasing value that can be increased every time the public key is sent.

NOTE C: If UE has attached the network before, when SGSN receives Attach/RAU/LAU request, it verifies whether the public key that UE possesses is the latest. Otherwise, SGSN checks whether the PSKI that UE send is latest. Otherwise, it will distribute the latest public key and the corresponding PKSI.

NOTE D: In the roaming case or in case of network sharing, when core network receives the request message, it will check whether the PLMN ID is same as the PLMN ID that it located in. Otherwise, SGSN will send the new PLMN ID to UE to avoid the collision of the public key identifier, as the PKSI may not be global unique.

4. At receiving the Security Mode Command message, RNC transmits this message to UE.

5. When receiving the Security Mode Command message, UE receives and saves the public key sent from RNC via Security Mode Command. UE verifies the signature of PWS Warning Notification message with the public key, NSUC and signature algorithm.

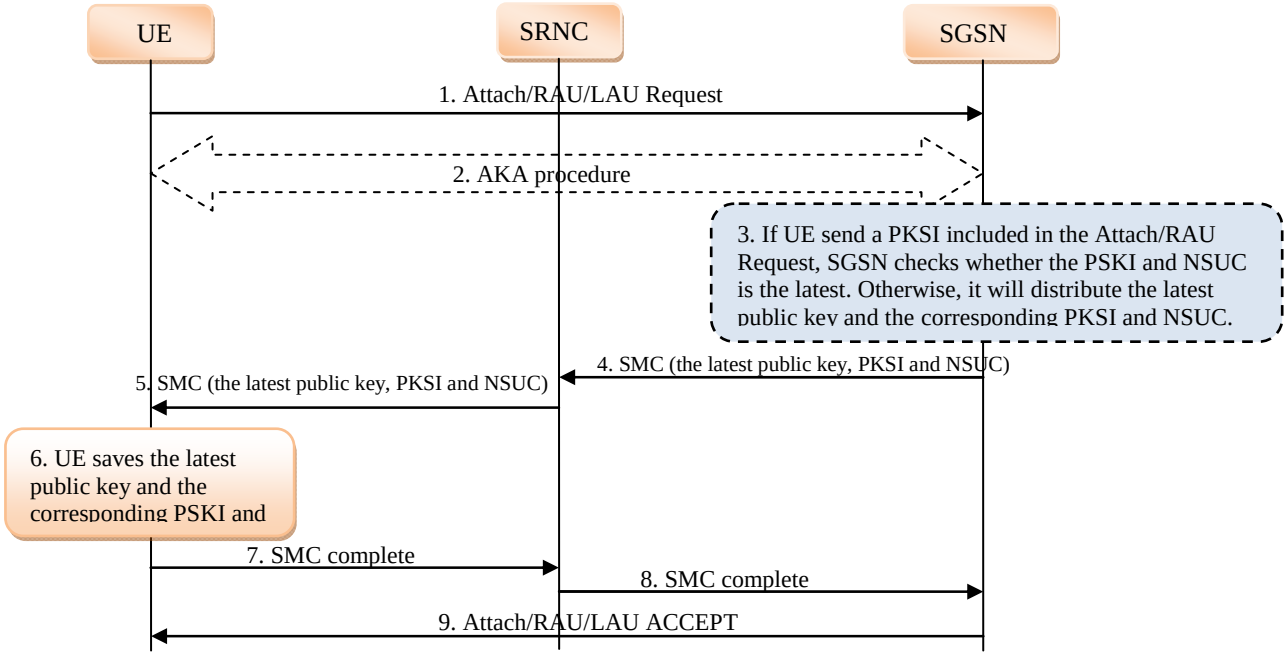

Figure A.1.2.1 Distribution of public key information in UMTS

The public key distribution mechanism can also be used for public key update in UMTS.

A.1.3 Signature algorithm agility

This solution describes the distribution of the signature algorithm identifier based on Warning Notification messages and broadcast message. CBE signs the PWS Warning Notification. Figure A.1.311 gives an example.

NOTE: SAI: Signature Algorithm Identifier

1. In the Emergency Broadcast Request, CBE provides the signature algorithm identifier to CBC.
2. CBC transmits the signature algorithm identifier to MME with Write-Replace Warning Request.
3. The MME sends a Write-Replace Warning Confirm message that indicates to the CBC that MME has started to distribute the warning message to eNB.
4. Upon reception of the Write-Replace Confirm messages from MME, the CBC may confirm to the CBE that the PLMN has started to distribute the warning message.
5. When MME receives this request, it transmits the signature algorithm identifier with Write-Replace Warning Request to eNB.
6. eNB broadcasts the signature algorithm identifier for the network's coverage area to all UEs. And UE verifies the signature of PWS Warning Notification message with the public key and signature algorithm.

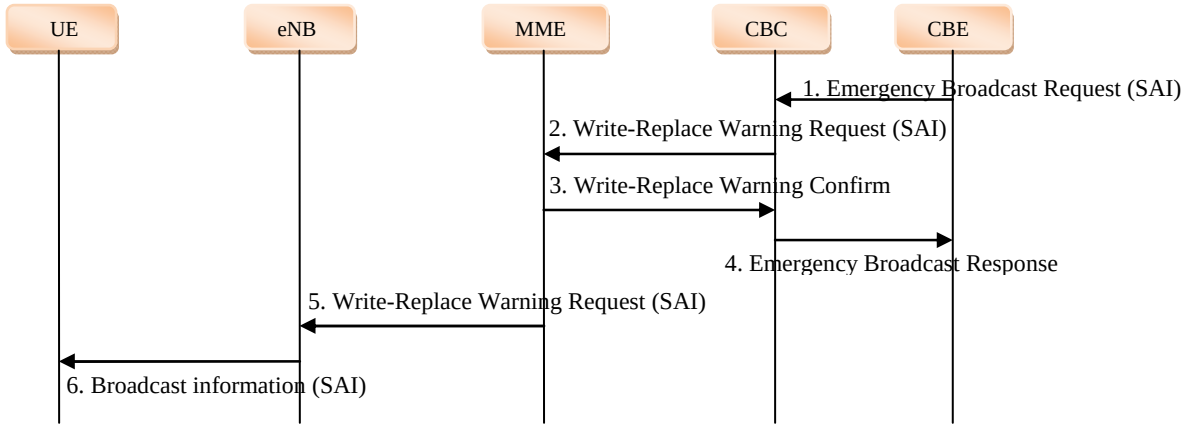

Figure A.1.3-1: Distribution of signature algorithm identifier

PWS signature algorithm identifier is set in the IE: Warning-Security-Information of PWS Warning Notification.

The signature algorithm identifier can be set in the Warning-Security-Information IE in WRITE-REPLACE Request/Indication. Then the corresponding message over air interface will have no impact. If this approach is introduced, it will not increase the overload for network entity,

A.1.4 Distribution of signature algorithm identifier in UMTS

This solution describes the distribution of the signature algorithm identifier in the ETWS PRIMARY NOTIFICATION WITH SECURITY or in the Warning Security Information of WRITE-REPLACE Request message.

NOTE: SAI: Signature Algorithm Identifier

1. In the Emergency Broadcast Request, CBE provides the signature algorithm identifier to CBC. CBC transmits the signature algorithm identifier to UTRAN with Write-Replace Warning Request.
2. UTRAN sends a Write-Replace Warning Confirm message that indicates to the CBC that it has started to distribute the warning message to service area.
3. Upon reception of the Write-Replace Confirm messages from CBC, the CBC may confirm to the CBE that the PLMN has started to distribute the warning message.
4. When UTRAN receives this request, it first sends a PAGING TYPE 1 message or a SYSTEM INFORMATION CHANGE INDICATION message, including the IE "ETWS information".
5. After the reception of the IE "ETWS information" in either the PAGING TYPE 1 or the SYSTEM INFORMATION CHANGE INDICATION message. If RRC is configured from upper layers to receive the ETWS primary notification with security, UTRAN **"should"** send SAI included in ETWS PRIMARY NOTIFICATION WITH SECURITY to UEs. And UE verifies the signature of PWS Warning Notification message with the public key and signature algorithm.

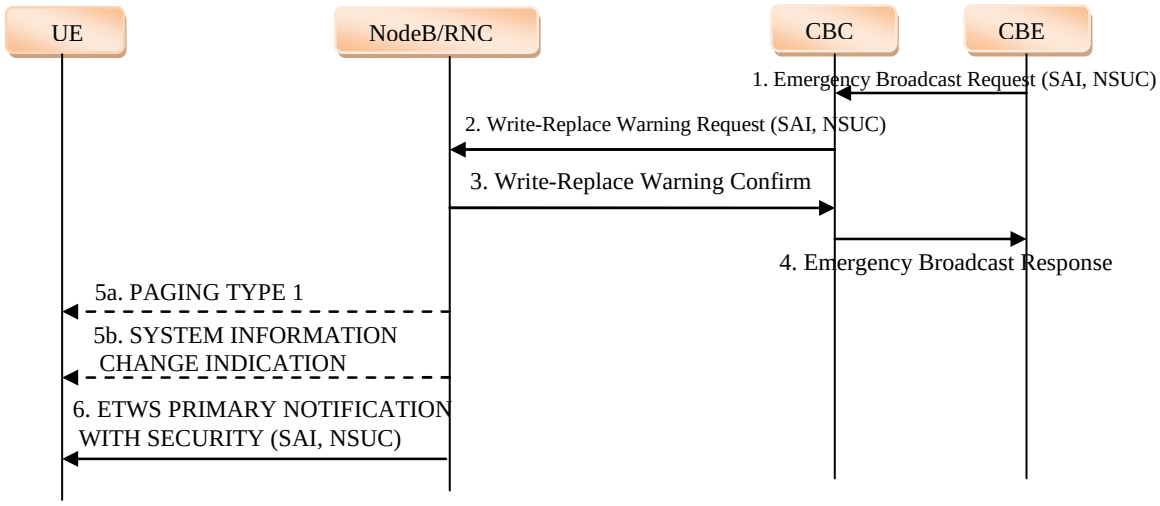

Figure A.1.4.1: Distribution of signature algorithm identifier in UMTS

A.1.5 Verification of PWS Warning Notification message

This clause describes the solution that UE verifies the signature of PWS Warning Notification message with the saved public signature key and signature algorithm Figure A.1.5-1 gives an example to show the solution with CBE as the signature entity.

1. CBE sends SAI, NSUC and the signature included in Emergency Broadcast Request to CBC. The signature also covers the NSUC.
2. CBC sends SAI, NSUC and the signature in Write-Replace Warning Request to MME.
3. MME sends a Write-Replace Warning Confirm message that indicates to the CBC that MME has started to distribute the warning message to eNB.
4. Upon reception of the Write-Replace Confirm messages from MME, the CBC may confirm to the CBE that the PLMN has started to distribute the warning message.
5. When MME receives this request, it sends SAI, NSUC and the signature in the Write-Replace Warning Request to eNB.
6. eNB broadcasts SAI, NSUC and the signature for the network's coverage area to all UEs.
7. At receiving the broadcast information message, UE verifies the signature with the latest public key. The signature covers emergency warning, SAI, and NSUC. UE verifies that NSUC received from the network in the notification is greater or equal to the NSUC stored on the UE. After receiving a warning message, the UE ceases to update its stored NSUC that is associated with this signing key.

NOTE: If there are too few bits to actually send the NSUC over the air in the warning message, it could be left out. Then the UE would need to test the potential NSUCs starting from the stored NSUC to the stored NSUC plus a window size. The signature will always have to cover both the warning message and the NSUC.

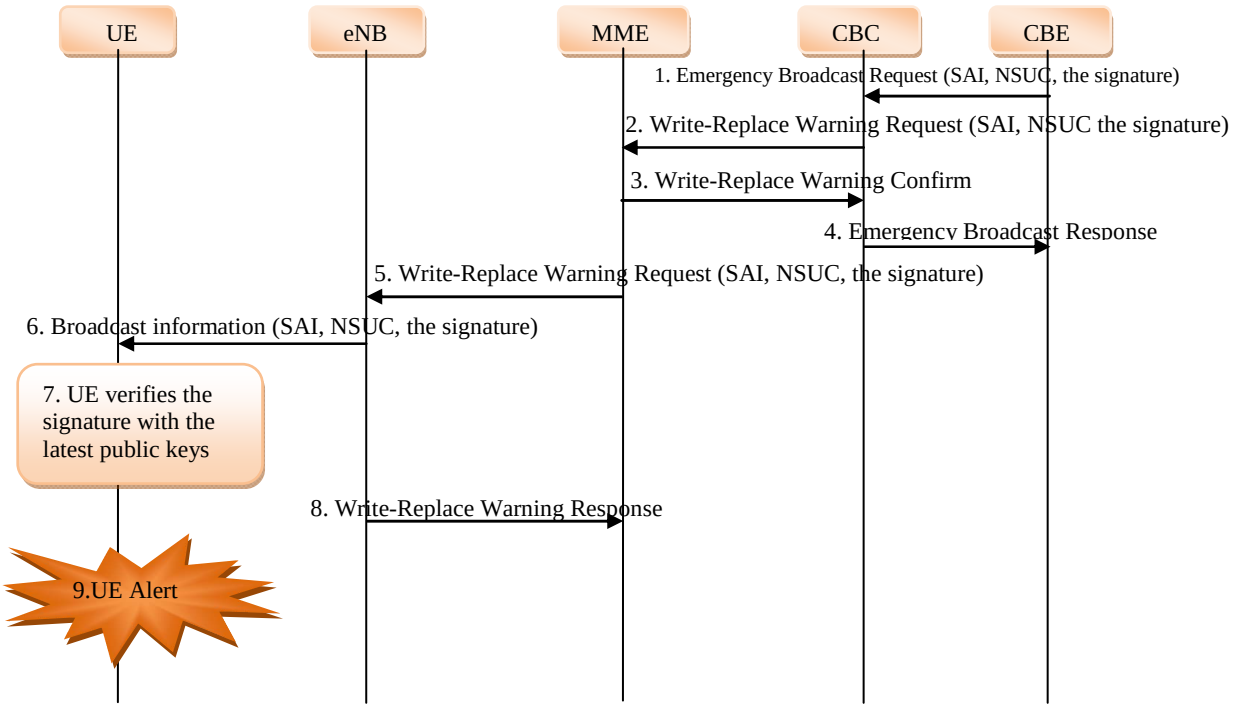

Figure A.1.5-1: Verification of PWS message

A.2 Solution 2

A.2.1 General

In this solution, a secure point-to-point channel is used to distribute PWS keys to UE registered to the network. Two aspects are included: the one is the network entity (MME/SGSN) distribute PWS key to UE (the blue line as showed in figure A.2.1-1 below); the other is network entity (MME/SGSN) get PWS key from CBC/CBE (the red line as showed in figure A.2.1-1 below).

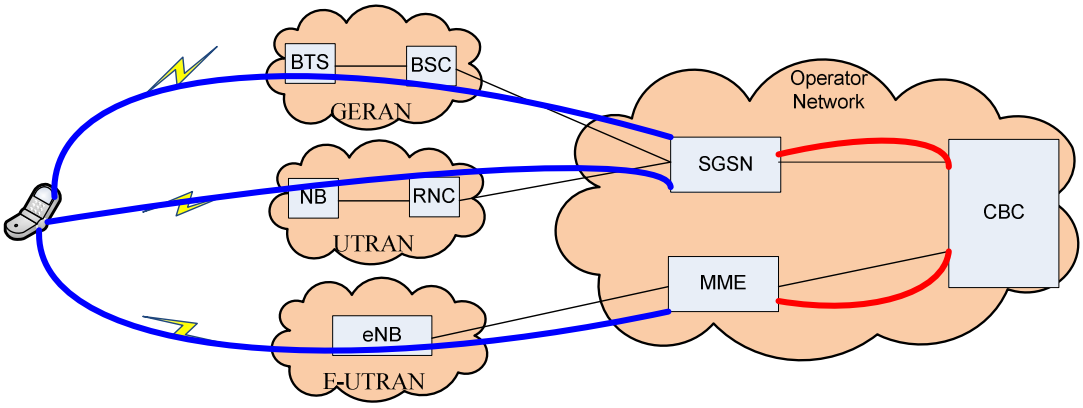

Figure A.2.1-1: PWS key distribution path

A.2.2 Initial PWS key distribution

Initial PWS keys should be ready just after UE has registered to the network immediately. In this way by anytime there is a PWS warning message sent by the network, UE can verify it with the PWS key it has stored. So a solution is proposed that the initial PWS keys are distributed in attach procedure.

- LTE: Two PWS keys and the corresponding public key identifiers (PKIDs) are sent to UE in Attach Accept message by MME, which is integrity protected.
- UMTS: Two PWS keys and the corresponding PKIDs are sent to UE in Attach Accept message by SGSN, which is integrity protected.
- GSM: Two PWS keys and the corresponding PKIDs are sent to UE in RAU Accept message during attach procedure by SGSN.

NOTE: Whether security enhancement is needed for GSM/GPRS is FFS.

The above two PWS keys are defined to be the current PWS key and the next PWS key. The current PWS key is the currently activated key which is used to sign the PWS notification message. The next PWS key is activated and becomes the current key after the old current one is deactivated.

A.2.3 Network PWS key configuration

Since PWS keys are sent to UE in L3 signalling, the network entity (MME/SGSN) should be configured with the PWS keys when PWS service is determined to be provided to UE by the network. Thus there is a requirement that CBC and MME/SGSN **"should"** have an interface to distribute the PWS keys.

When the network determines to provide PWS service to UEs, CBC **"should"** send PWS keys to MME/SGSN. After PWS keys are configured in MME/SGSN, once there is a UE registered to the network, MME/SGSN should distribute the PWS keys to the UE in the attach procedure.

When CBC/CBE updates the PWS keys of a specified notification area, CBC **"should"** send the updated PWS keys to the network entities (MME/SGSN) which have connections with the affected RAN.

Editor's Note: It is FFS if the working assumption on a national root of trust determines CBC/CBE.

A.2.4 PWS key update

Even if the frequency of PWS key update is rather low, it should also provide a mechanism to permit PWS key to be updated. This solution uses a point-to-point secure channel to update PWS keys.

The network activates and updates PWS keys as follows:

- Two PWS keys are used: the current PWS key and the next PWS key. The current one is the activated one which is used to verify the current PWS notification; the next PWS key is used to verify the PWS notification after it has been activated when the current one is deactivated.
- When CBC determines to change the next PWS key to current PWS keys, it **"should"** also update the next PWS key with a new one. And CBC **"should"** send the updated next PWS key and its identifier to the network entities (MME/SGSN), together with the current PWS key identifier.

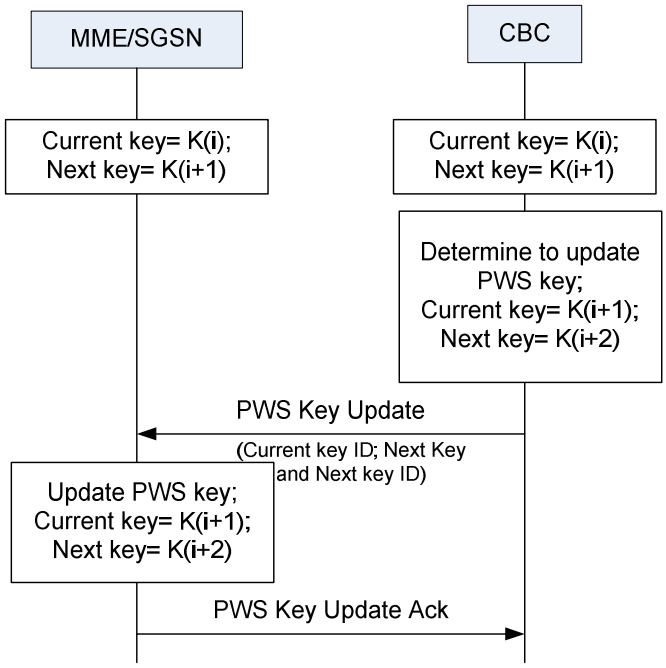

Figure A.2.4-1: PWS key activates and updates by network side

UE activates and updates PWS keys as follows:

- The serving network always broadcasts the current PWS key identifier and the next PWS key identifier. The network entities notify the corresponding RAN to broadcast the just activated current PWS key identifier and the updated next PWS key identifier.

Editor's Note: the broadcast is not authenticated. Therefore, an attacker can do two things: 1. deny reception of legitimate PWS warning by broadcasting next key ID as current key ID; 2. induce UE key update by broadcasting different PWS key identifier. The severity of these attacks is for further study, as it constitutes a reflector attack with amplification.

- UE activates the stored next PWS key to the current key as the serving network indicates.
- Once a UE notices that at least one of the broadcasting PWS key identifiers is different from the one it stores, UE will performs PWS key update till the next normal TA/RA/LA update procedure.
- In response to each successful tracking area, routing area or location area update, the network entity provides the PWS key requested by UE.
- UE stores the received new PWS key as the next one.

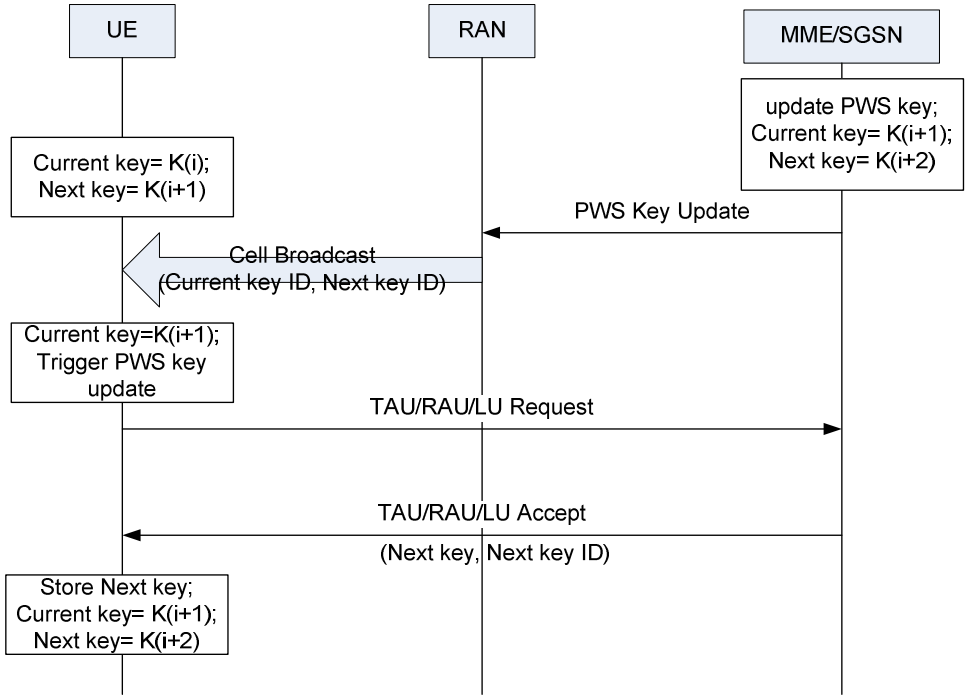

Figure A.2.4-2: PWS key activates and updates by UE side

Editor's Note: PWS Key Update in Figure A.2.4.2 needs more detail.

Editor's Note: Lifetime of the key should be longer than the TAU/RAU/LU lifetime.

A.2.5 Delivery of PWS Warning Notification message

When network nodes distribute PWS Warning Notification message to UE, the public key identifier (PKID) should be included in the message which is identified the public key used to sign the message. This method can avoid UEs trying each public key to verify the signature effectively. Figure A.2.5.1 gives an example to show the solution used in LTE network.

1. CBC/CBE sends a Write-Replace Warning Request message to MME. The message **"should"** include a"PKID" to identify the public key used to sign the message, as well as the "the signature".
2. MME sends a Write-Replace Warning Confirm message to the CBC/CBE.
3. When MME receives this request, it sends a Write-Replace Warning Request message ("PKID", and "the signature") to eNB.
4. When eNB receives this request, it broadcasts PWS warning message ("PKID", and "the signature") to all UEs in the network's coverage area.
5. At receiving the broadcast information message, UE verifies the signature with the public key identified by PKID which is received in the broadcast message.
6. The eNB sends a Write-Replace Warning response message to MME to confirm the request.
7. The UE alerts the user.

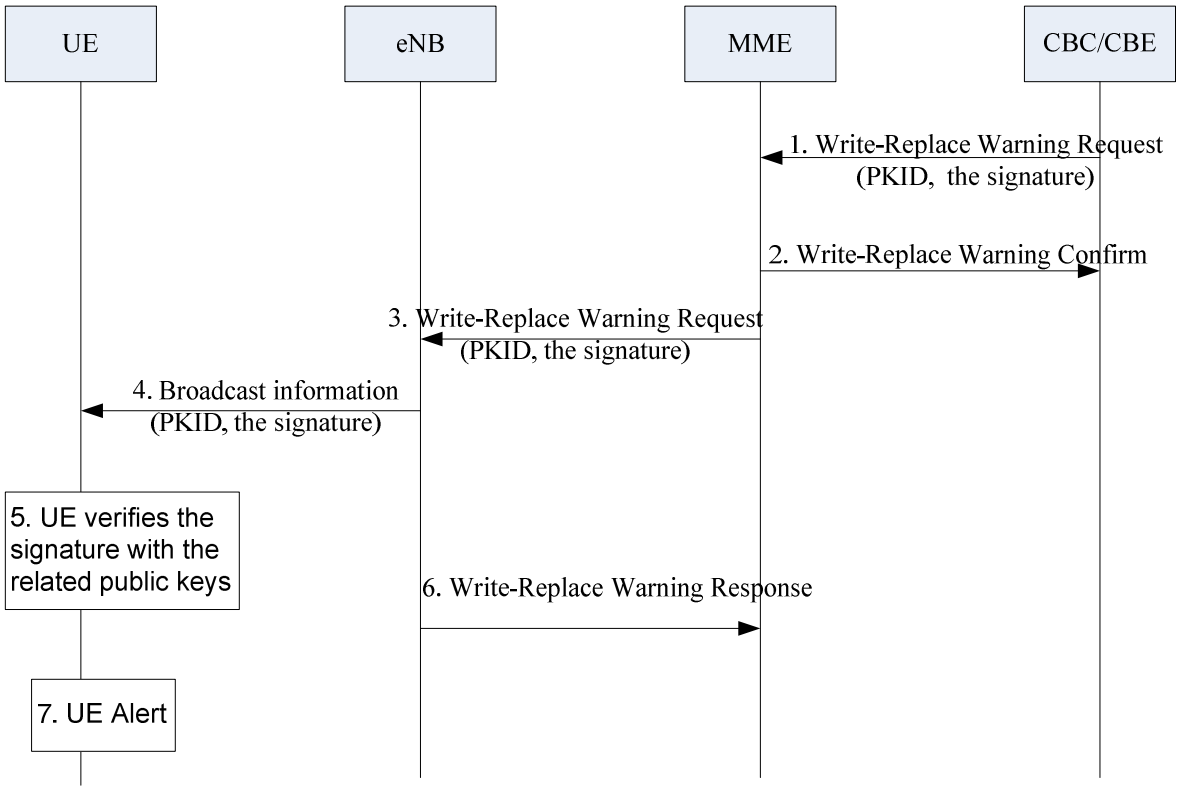

Figure A.2.5-1: Verification of PWS message

NOTE: Step 6 happens after step 3, otherwise there are no other sequence requirements.

Annex B:
Threat discussion depending on the PWS settings in the UE relating to roaming

Depending on SA3 decisions whether the option of displaying unverified messages in the users "PWS Security mandated home network" is considered (at all, in the first step, or later), this clause points out possible threats due to settings in the UE, i.e. a user staying in his home network may be the victim of a circumvention attack, in which case roaming inside its own country becomes true.

Thus, roaming impact need to be considered if PWS Security **"should"** be effectively applied and the display of warning messages without security in another country that does NOT have deployed PWS Security is wanted by the home operator, i.e. setting the PWS USIM flag to "process PWS messages in VPLMNs". Otherwise PWS Security in the home country can be circumvented. (See clause 6.1.2 for a more detailed description of the circumvention attack).

The following are different cases with respect to roamers and PWS Security settings in the UE:

- roamers within a network that sends warning messages without PWS Security and whose home operator has also not implemented PWS Security: this is the starting point of discussion for introducing PWS Security, i.e. before Rel-12;
- roamers within a network that sends warning messages without PWS Security, but whose home operator has implemented PWS Security;
- roamers within a network that sends signed warning messages, but whose home operator has not implemented PWS Security;
- roamers within a network that sends signed warning messages and whose home operator has also implemented PWS Security.

In Rel-11 a PWS USIM flag was introduced as a new UE setting regarding PWS (TS 31.102 [22], service 96), which allows the home operator to configure the UE behaviour to process or ignore PWS warning messages in the home and/or visited network. When introducing PWS Security, the setting of this PWS USIM flag to "process" is assumed to be the precondition for the usage of PWS Security.

The introduction of PWS Security will most likely happen only step by step and on a national basis.
Thus, operators could consider configuring the UE behaviour depending on the introduction of PWS Security in their own country and/or other countries in a similar way as the PWS USIM flag would need to be configured. E.g. a flag to enable or disable PWS Security could be additionally introduced as new USIM setting regarding PWS Security, called "PWS Security flag". It will reside on the USIM.

Precondition for checking the PWS Security flag would be that the operator has set the PWS USIM flag to process PWS warning messages. Then, the UE settings regarding the PWS Security flag would tell the UE whether PWS warning messages **"should"** be always displayed or be displayed only after they could be verified. In the former case, the UE would NOT need to verify a signature even if a signed warning message was broadcast.

Considerations on the different options for UE settings are summarized in the following tables. For completeness, table B-1 is included, which covers the PWS USIM flag to process or ignore warning messages that is supposed to be present in any Rel-11 UE.

Table B-1: Potential UE behaviour with regard to PWS in Rel-11

PWS Security support		Possible PWS USIM flag settings: ignore/process warning messages		If PWS USIM flag is set to process, how does the UE behave wrt display of warning messages (could be called PWS Security flag)		Comments on Rel-11 UEs
in HN	in VN	process PWS in HN	process PWS in VN	Display in HN only if verified	Display in VN only if verified	
N	N	Y/N	Y/N	n/a	n/a	Rel-11 has no PWS Security. In order for the operator to be able to influence, whether a UE is allowed to receive warning messages, e.g. when roaming, a PWS USIM flag was introduced: Operator decides on behalf of its users whether warning messages are ignored (N) or processed (Y) in HPLMN (home network, HN) and/or VPLMN (visited network, VN), depending on the trust in other operator networks

The analysis in table B-2 covers Rel-12 and beyond. The following assumption is taken: if PWS Security is in use in the home network (HN), the UE processes warning messages by default (i.e. PWS USIM flag set to "Y") and displays only verified messages in its HN. Therefore, some Y/N combinations are not considered in the following table.
For PWS Security in any visited network (VN), this assumption is not taken due to the fact that verification of a signed warning message in a visited network depends on the availability of the public key and may be less easy to provide as in the home network.

NOTE: The verification of warning messages in visited networks also implies the obligation to the home network to provide the keys needed for verification, else the home operator or national authority may be held responsible for damage and loss of life caused by non-displayed messages.

Table B-2: Potential UE behaviour with regard to PWS and PWS Security in Rel-12

PWS Security support		Possible PWS USIM flag settings: ignore/process warning messages		If PWS USIM flag is set to process, how does the UE behave wrt display of warning messages (a.k.a PWS Security flag)		Comments on Rel-12 UEs
in HN	in VN	process PWS in HN	process PWS in VN	Display in HN only if verified	Display in VN only if verified	
Y	N	Y	Y	Y	N i.e. display all messages (also without verification)	PWS Security is only in HN available, UE is supposed to process PWS everywhere. The operator restricts the UE to display only verified warning messages in the HN, but allows display of any warning message while roaming. However, in terms of security, this is not a PWS Security-sound UE configuration: if unsigned warning messages can be displayed, PWS is easy to attack (circumvention attack).
Y	N	Y	Y	Y	Y	PWS Security-sound UE configuration avoiding attacks: allow the display of verified messages only (in any network)
Y	N	Y	N	Y	n/a	PWS Security-sound: only process warning messages in HN but do not process (and display) while roaming
N	Y	Y	Y	N i.e. display all messages	N i.e. display all messages	PWS Security-sound: if in HN no PWS Security is implemented, then UE ignores any signature and displays warning messages unverified at home and visited networks without any considerations on PWS Security used in a VN
N	Y	Y	Y	N i.e. display all messages	Y	This would be PWS Security-sound too: where PWS Security is available, use it. Note, root key / CBE key of the VN **"should"** be available
N	Y	Y	N	N i.e. display all messages	n/a	It is possible to configure UEs like this but if the HN operator could distinguish VNs and would trust the security in one VN, why not to at least allow processing of warning messages in those cases, when the HN trusts a VN?
N	Y	N	Y/N	n/a	Y/N	This would be PWS Security-sound but there is no need to restrict HN in message processing. If HN is not using security, it should be able to display any message.
Y	Y	Y	Y	Y	N i.e. display all messages	How to deal with signed warning messages, if no root key or CBE key of the roamed-in VN is available in the UE? This configuration could lead to the circumvention attack!
Y	Y	Y	Y	Y	Y	PWS Security-sound: only display a warning message if it is verifiable. Possible in solutions that provide the public key of the roamed-in VN beforehand or together with the warning message, in case root key/CBE key are known. Note this may result in suppression of real warning messages and the responsible entity (e.g. the home operator or national authority) could be held responsible, e.g. if the flag is not set correctly/timely for this country or the keys are not provisioned correctly.

The following summarizes the conclusions from the table, starting with the case that a HN does not provide PWS Security:

If the HN is not using PWS Security, i.e. UE displays warning messages without verification in HN, there are two possible UE behaviours for the roaming case: do not care how VN is configured with respect to PWS Security, i.e. display in VN what is broadcasted there as well, or: ignore warning messages in VNs, because HN operator wants to protect subscribers from fraud outside its own domain. According to TS 22.268 [2], this instruction to ignore warning messages in VNs can apply to all VNs, or only to selected VNs that may be known to the HN to have security problems. Both settings may be sensible.

If the HN uses PWS Security and UE is allowed to display messages in visited networks without verifying the signature, circumvention attacks are possible. Thus, if operators do not want to nullify the usage of PWS Security at all, they either have to forbid receiving warning messages in case their UEs are roaming or to allow the display, but only after their UEs could verify the signature. Consequently, if the home network uses PWS Security, one valid configuration setup for a UE is to ignore warning messages in any VN that does not use PWS Security. From security point of view, this is a PWS Security-sound approach, but it may result in suppressing real warning messages, if the keys are not available (for whatever reason) or if the flag for this visited network is not set timely or correctly. This implies a great responsibility for the home operator. (Clause 7.9.1 provides more details on this type of counter-measure).

Another valid set up for a UE, where the home network uses PWS Security, is to also accept warning messages in a VN that does not use PWS Security. However this works only in a PWS Security-sound way, if network-independent location verification is in place. Note that the home operator may have configured the UE with regard to PWS and PWS

Security, but the user may want to have a different setting with his personal preference, i.e. depending on what is important to the user: risking that real warning messages may be suppressed or false warning messages may be received. (Clause 7.9.2 provides more details on this type of counter-measure).

Note, current solution 8 would not allow any message display in VN if the message could not be verified. However, if a VN uses PWS Security and the UE would have the knowledge about this VN behaviour and even could verify signatures on warning messages in this VN, it should be preferred to display the message. Thus, a more subtle differentiation of VNs may be appropriate.

Depending on the solution selected during this study phase, the UE would either receive the valid public key of the CBE or a signing proxy on the fly (while the HN operator may have already given the necessary root key) as in solution 6/7, receive it beforehand by OTA as in solution 8, or gains it via NAS or by GBA as in 3/4. If the roaming case should be supported and home and visited network support PWS Security, the home network operator needs to assure that the credentials for verifying a message while roaming are available.

In summary, as long as not all operators introduce PWS Security (which is likely to happen as regulators of some countries do prefer warning message broadcast without security), on one hand circumvention attacks are possible and should be mitigated. On the other hand, the safety of the user while roaming may need to be considered, wherever he is.

In addition to the ignore/process PWS USIM flag, additional PWS Security flags may be needed in the USIM indicating the security status (PWS Security on/off) of home or visited countries and/or networks. This could be introduced as part of the solution, together with e.g. network-independent location verification.

Editor's Note: It is ffs if several PWS Security flags are needed.

Annex C:
Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2014-09	SP-65	SP-140583			Presented for information and approval	0.8.0	1.0.0
					Upgrade to Rel-12 version	1.0.0	12.0.0
2016-01	SP-70				Upgrade to Rel-13 (MCC)	12.0.0	13.0.0

Change history							
Date	**Meeting**	**TDoc**	**CR**	**Rev**	**Cat**	**Subject/Comment**	**New version**
2017-03	SA#75					Promotion to Release 14 without technical change	14.0.0
2018-06	-	-	-	-	-	Update to Rel-15 version (MCC)	**15.0.0**
2020-07	-	-	-	-	-	Update to Rel-16 version (MCC)	**16.0.0**

History

Document history		
V16.0.0	August 2020	Publication