3GPP TR 33.820 V8.3.0 (2009-12)

Technical Report

3rd Generation Partnership Project;
Technical Specification Group Service and System Aspects;
Security of H(e)NB;
(Release 8)

Keywords
UMTS, GSM, security, architecture, H(e)NB

3GPP

Postal address

3GPP support office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet
http://www.3gpp.org

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.

Introduction

H(e)NB is able to provide new services with higher data rate in a low cost. Operators have already indicates their interest in this area. Study of H(e)NB has already started in 3GPP in order to investigate the feasibility of developing a standard solution for H(e)NB. Security is an critical aspect of H(e)NB, so it is necessary to investigate security issues of H(e)NB.

1 Scope

The present document identifies special security threats of H(e)NB and study the countermeasures to these threats.

The study should include, but not be limited to, threat analysis of H(e)NB, mutual authentication and security protection between H(e)NB and rest of network, maintenance of the security context between H(e)NB and rest of network, security requirements on the H(e)NB, provisioning of security credentials on the H(e)NB, security solution for verifying the location of the H(e)NB etc.

With regard to security protection between the H(e)NB and the rest of the network, bandwidth efficiency should be taken into consideration.

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 22.011: "Service Accessibility".

[3] R3-080021, "Reply LS on Home NodeB/eNodeB regarding localization/authorization", RAN3#59 (February 2008)

[4] R3-081121, "HNBs Location Certification" 3GPP TSG RAN WG3 Meeting #60 Kansas City, USA, 5th – 9th May 2008

[5] ETSI ES 282 004 V1.1.1 Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); NGN functional architecture; Network Attachment Sub-System(NASS)[S]. 2006.

[6] ETSI ES 283 035 V1.1.1 Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); NASS; e2 interface based on the DIAMETER protocol.[S]. 2006.

[7] Void.

[8] Void.

[9] Void.

[10] 3GPP TS 33.234: "Wireless Local Area Network (WLAN) interworking security".

[11] IETF RFC 4306: "Internet Key Exchange (IKEv2) Protocol".

[12] 3GPP TS 33.310: "Network Domain Security (NDS); Authentication Framework (AF)".

[13] IETF RFC 4739: "Multiple Authentication Exchanges in the Internet Key Exchange (IKEv2) Protocol".

[14] ETSI TS 102.310: "Smart Cards; Extensible Authentication Protocol support in the UICC"

[15] 3GPP TS 33.401: "3GPP System Architecture Evolution (SAE) Security Architecture"

[16] 3GPP TS 25.467: "UTRAN architecture for 3G Home NodeB; Stage 2"

[17] IETF RFC 4945: "The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2, and PKIX", Aug 2007

[18] 3GPP TS 33.210: "Network Domain Security (NDS); IP network layer security (IP)".

[19] 3GPP 24.008: "Mobile radio interface Layer 3 specification; Core network protocols"

[20] 3GPP 24.301: "Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS)"

[21] 3GPP 32.581: "Concepts and Requirements for Type 1 interface HNB to HNB Management System (HMS)"

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Access point Home Register A database that holds subscription records and relevant service attributes of the H(e)NB.
CSG A closed subscriber group identifies subscribers of an operator who are permitted to access one or more cells of the PLMN of but having restricted access ("CSG cells")
H(e)NB device identity server core network function which holds the information of valid H(e)NB device identities.
Hosting party the party hosting the H(e)NB and having a contract with the PLMN operator.
Hosting Party Module a module holding the credentials for authentication of the hosting party.
Security Gateway Element at the edge of the core network terminating security association(s) for the backhaul link between H(e)NB and core network.
Subscriber the user of a UE with subscription to PLMN operator, may be camping on the H(e)NB.

3.2 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].

AAA Authentication, Authorization and Accounting
ACL Access Control lists
AHR Access point Home Register
AKA Authentication and key agreement
ARP Address Resolution Protocol
CA Certification Authority
CSG Closed Subscriber Group
(D)DoS (Distributed) Denial of Service
eNB Evolved Node-B
EAP Extensible Authentication Protocol
ESP Encapsulating Security Payload
EPS Evolved Packet System
E-UTRAN Evolved UTRAN
FQDN Fully Qualified Domain Name
GSM Global System for Mobile communications
HNB Home Node-B
HNB GW 3G HNB Gateway
HeNB Home eNode-B
HeNB GW Home eNode-B Gateway
HSS Home Subscriber Sever
HLR Home Location Register
IGMP Internet Group Management Protocol
IKE Internet Key Exchange
IMSI International Mobile Subscriber Identity
LTE Long Term Evolution

MME	Mobility Management Entity
NAS	Non-Access Stratum
PKI	Public Key Infrastructure
PPPoE	Point-to-Point over Ethernet
SeGW	Security Gateway
SIM	(GSM) Subscriber Identity Module
TCP	Transmission Control Protocol
TrE	Trusted Environment
UDP	User Datagram Protocol UMTS Universal Mobile Telecommunication System
UICC	Universal Integrated Circuit Card
UP	User plane
USIM	Universal Subscriber Identity Module
UTRAN	Universal Terrestrial Radio Access Network

4 System architecture

Editor's Note: Several 3GPP working groups are conducting work on the system architecture. Related work on other working groups should be taken into account.

4.1 General

On the architecture we assume that

a) A kind of access concentrator function (e.g. Gateway) maybe the first contact in the core network (i.e. within a secured domain) for the H(e)NB.

b) Home access point (like H(e)NB are normally connected to the Internet via some access device (e.g. ADSL, cable modem). In these cases, such access device could be integrated with the H(e)NB, or be in a separate box.

c) A software distribution centre or O&M centre is supposed to be located in a secured domain.

H(e)NB terminology:

Regarding the UP encryption, three cases have to be differentiated:

a) HNB: UMTS case where UP encryption does not terminate in HNB

b) HNB: UMTS case where UP encryption terminates inside HNB

c) HeNB: LTE case

Where applicable the difference in consequences will be described.

Authentication scheme and terminology

Different solutions are possible for authentication of H(e)NB towards the core network. We distinguish these solutions by

a) the device authentication scheme

b) type of secure credential storage.

This results in considering following cases:

Table 1: Different authentication token variants

secure credential storage	device authentication scheme	
	shared key	Certificates
Irremovable	Case 1	Case 2
Removable	Case 3	Case 4

NOTE 1: This does not exclude combinations of the above solutions: Example, a removable token combined with an onboard certificate.

NOTE 2: The threats section uses the term 'authentication token' to denote the collection of the above cases. Where needed a certain property of the authentication token (e.g. row, column above) may be under attack in the threat analysis.

4.2 System architecture of HNB

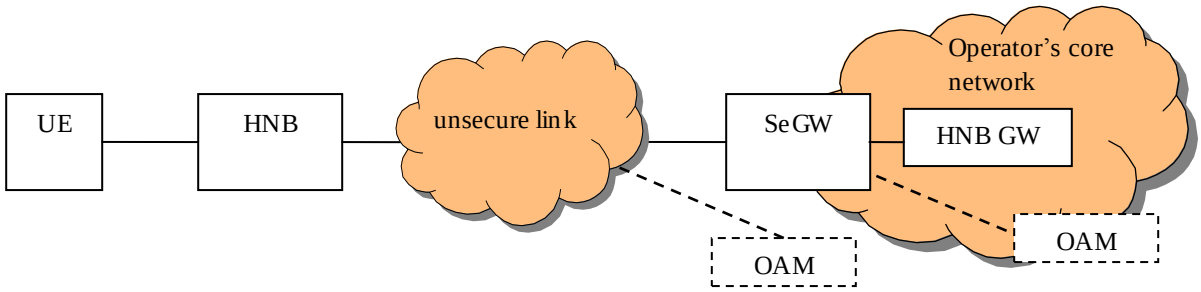


Figure 1: System Architecture of HNB

Description of proposed system architecture:

- Air interface between UE and HNB should be backwards compatible air interface in UTRAN;
- HNB access operator's core network via a Security Gateway. The backhaul between HNB and SeGW may be insecure.
- Security Gateway represent operator's core network to perform mutual authentication with HNB. Mutual authentication may need support of authentication server or PKI. SeGW and HNB GW are logically separate entities within operator's network.
- Security tunnel is established between HNB and Security Gateway to protect information transmitted in backhaul link.
- HNB-GW performs the access control for the non-CSG capable UE attempting to access the HNB. SeGW may be integrated into HNB GW. If the SeGW and the HNB GW are not integrated, then the interface between the HNB-GW and the SeGW may be protected using NDS/IP [18].
- Secure communication is required to Operation, Administration and Maintenance (OAM). This becomes even more important if OAM is placed outside the operator's network.

Editor's Note: The security implications of collapsing certain Core networks related functionality (e.g. SGSN or GGSN) in the HNB should be studied

NOTE: There may be a Home Gateway in the architecture at the customer premise. If such a Home Gateway is a physically and logically separate entity than the HNB, such a Home Gateway should not be present in the architecture since the security of the HNB should not rely on the security of the Home Gateway. However, if such a Home Gateway is physically or logically integrated with a HNB, it should be studied if security aspects (e.g. device security) of the Home Gateway may impact that of the HNB. In addition, the existence of any Home Gateway (integrated or separated) may imply restriction on the selection of backhaul security solutions, e.g. to allow NAT traversal.

4.3 System architecture of HeNB

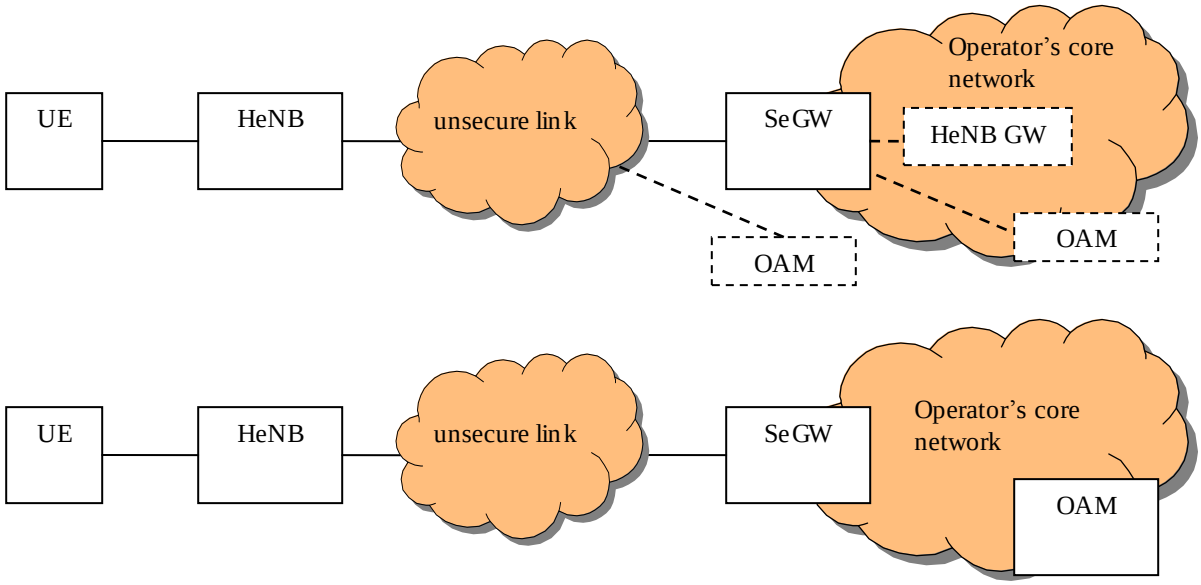


Figure 2: System Architecture of HeNB

Description of proposed system architecture:

- Air interface between UE and HeNB should be backwards compatible with air interface in E-UTRAN;
- HeNB access operator's core network via a Security Gateway. The backhaul between HeNB and SeGW may be insecure.
- Security Gateway represent operator's core network to perform mutual authentication with HeNB. Mutual authentication may need support of authentication server or PKI.
- Security tunnel is established between HeNB and Security Gateway to protect information transmitted in backhaul link.
- HeNB GW is optional to deploy. If HeNB is deployed, then SeGW may be integrated into HeNB GW. If the SeGW and the HeNB GW are not integrated, then the interface between the HeNB-GW and the SeGW may be protected using NDS/IP [18].
- Secure communication is required to Operation, Administration and Maintenance (OAM). This becomes even more important if OAM is placed outside the operator's network.

Editor's Note: The security implications of collapsing certain Core networks related functionality (e.g. Serving GW) in the HeNB should be studied

NOTE: There may be a Home Gateway in the architecture at the customer premise. If such a Home Gateway is a physically and logically separate entity than the HNB, such a Home Gateway should not be present in the architecture since the security of the HNB should not rely on the security of the Home Gateway. However, if such a Home Gateway is physically or logically integrated with a HNB, it should be studied if security aspects (e.g. device security) of the Home Gateway may impact that of the HNB. In addition, the existence of any Home Gateway (integrated or separated) may imply restriction on the selection of backhaul security solutions, e.g. to allow NAT traversal.

4.4 Overview of Security Architecture

Figure 3 gives an overview of the complete security architecture of H(e)NB.

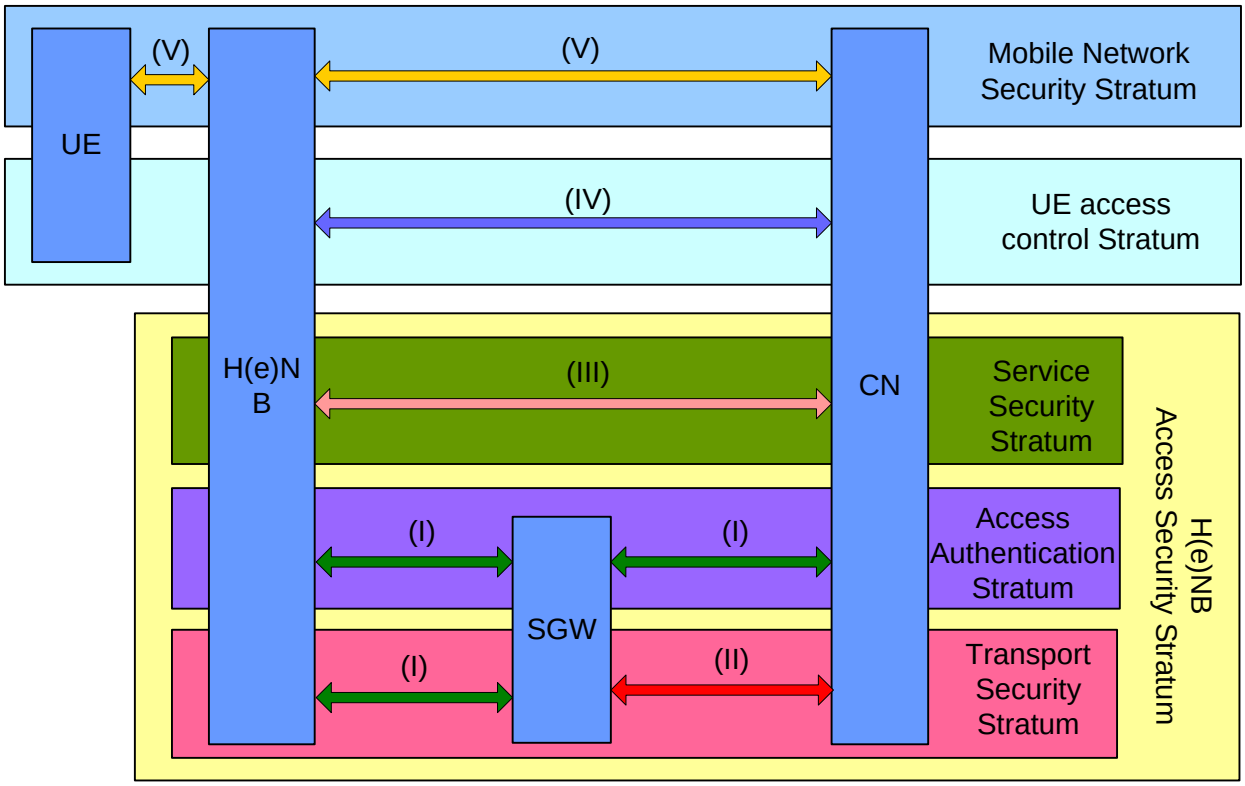


Figure 3: Overview of the H(e)NB security architecture

Five security feature groups are defined. Each of these feature groups meets certain threats and accomplishes certain security objectives:

- **H(e)NB access security (I):** the set of security features which include the mutual authentication between H(e)NB and network, security tunnel establishment between H(e)NB and SeGW, authorisation mechanisms and location locking mechanisms of H(e)NB. SeGW should interact with entities (AAA/HLR or H(e)NB device identity server) located in CN for performing mutual authentication and authorization.
- **Network domain security (II):** the set of security features which include security communication and security communication between SeGW and CN.
- **H(e)NB service domain security (III):** the set of security features which include security communication between H(e)NB and entities located in CN. For working properly, H(e)NB should interact with an OAM server or a device management server located in CN to download software or update configuration data. Communication between H(e)NB and these entities should be secured.
- **UE access control domain security (IV):** the set of security features which include UE access control mechanisms. These security features only apply to legacy UEs. For Rel-8 compliant UEs, the access control of the UE is based on the allowed CSG list and accomplished on the terminal and CN, H(e)NB will not perform access control for the Rel-8 compliant UEs .
- **UE access security domain (V):** the set of security features that provide UEs with secure access to mobile communication system. Since the introduction of the H(e)NB should have no influence on the UE, the security features of this domain is as same as the security features defined in the corresponding mobile communication system specifications and consequently out of scope of current specification.

Editor's Note: It is ffs whether the stratum represented in the figure has values. Need to revisit this sub-section once the TR is complete to have clear/improved overview of the security architecture.

5 Threats analysis

NOTE 1: A reference to certain implementation platform mentioned in this TR is for illustrative purposes only. Such examples are by no means exhaustive and are not to be construed as threat-mitigating solutions.

Editor's Note: It has to be checked whether there is any bias in the threat formulation with respect to the implementation in the future (cfr. mentioned examples).

5.1 Common threats to H(e)NB

In this section threats common to HNB and HeNB are presented. The section starts with a list of threats that are then grouped in different categories. Details of each threat is also given in this section together with the impact of each threat on different assets and the risk level they belong to.

5.1.1 Threats List

Threats identified in this TR are listed below. These threats are detailed in Section 5.1.3.

1) Compromise of H(e)NB authentication token by a brute force attack via a weak authentication algorithm.

2) Compromise of H(e)NB authentication token by local physical intrusion.

3) Inserting valid authentication token into a manipulated H(e)NB.

4) User cloning the H(e)NB authentication Token.

5) Man-in-the-middle attacks on H(e)NB first network access.

6) Booting H(e)NB with fraudulent software ("re-flashing").

7) Fraudulent software update / configuration changes.

8) Physical tampering with H(e)NB.

9) Eavesdropping of the other user's UTRAN or E-UTRAN user data.

10) Masquerade as other users.

11) Changing of the H(e)NB location without reporting.

12) Software simulation of H(e)NB.

13) Traffic tunnelling between H(e)NBs.

14) Misconfiguration of the firewall in the modem/router.

15) Denial of service attacks against H(e)NB.

16) Denial of service attacks against core network.

17) Compromise of an H(e)NB by exploiting weaknesses of active network services

18) User's network ID revealed to H(e)NodeB owner

19) Mis-configuration of H(e)NB

20) Mis-configuration of access control list (ACL) or compromise of the access control list

21) Radio resource management tampering

22) Masquerade as a valid H(e)NB

23) Provide radio access service over a CSG

24) H(e)NB announcing incorrect location to the network

25) Manipulation of external time source

26) Environmental/side channel attacks against H(e)NB

27) Attack on OAM and its traffic

28) Threat of H(e)NB connectivity to network access

29) Handover to CSG H(e)NB.

5.1.2 Grouping of Threats

The threats of Section 5.1.1 can be grouped in 6 different categories as given in this below.

The above threat maybe grouped together as the following:

Compromise of H(e)NB Credentials

1) Compromise of H(e)NB authentication token by a brute force attack via a weak authentication algorithm.

2) Compromise of H(e)NB authentication token by local physical intrusion.

4) User cloning the H(e)NB authentication Token.

Physical attacks on a H(e)NB

3) Inserting valid authentication token into a manipulated H(e)NB.

6) Booting H(e)NB with fraudulent software ("re-flashing").

8) Physical tampering with H(e)NB.

26) Environmental/side channel attacks against H(e)NB

Configuration attacks on a H(e)NB

7) Fraudulent software update / configuration changes.

19) Mis-configuration of H(e)NB

20) Mis-configuration of access control list (ACL) or compromise of the access control list

Protocol attacks on a H(e)NB

5) Man-in-the-middle attacks on H(e)NB first network access.

15) Denial of service attacks against H(e)NB.

17) Compromise of an H(e)NB by exploiting weaknesses of active network services

25) Manipulation of external time source

27) Attack on OAM and its traffic

28) Threat of H(e)NB network access

Attacks on the core network, including H(e)NB location-based attacks

11) Changing of the H(e)NB location without reporting.

12) Software simulation of H(e)NB.

13) Traffic tunnelling between H(e)NBs.

14) Misconfiguration of the firewall in the modem/router.

16) Denial of service attacks against core network.

24) H(e)NB announcing incorrect location to the network

User Data and identity privacy attacks

9) Eavesdropping of the other user's UTRAN or E-UTRAN user data.

10) Masquerade as other users.

18) User's network ID revealed to Home (e)NodeB owner

22) Masquerade as a valid H(e)NB

23) Provide radio access service over a CSG

Attacks on Radio resources and management

21) Radio resource management tampering

5.1.3 Threats

Threats listed in Section 5.1.1 are detailed in the following. Each threat starts with a title same as that given in the list of Section 5.1.1 followed by prerequisites to perform the attack, a description of the threat, probability of the threat, extent of impact the threat can have, assets that are affected by the threat and potential means to mitigate the threat.

1) Compromise of H(e)NB authentication token by a brute force attack via a weak authentication algorithm.

Prerequisites: Token with weak authentication algorithm is used for H(e)NB authentication to the operator's network. This threat refers to a specific usage of shared secrets for H(e)NB authentication i.e. the cases 1 and 3 of Table 1: Different authentication token variants.

Description: An example for a token using a weak authentication algorithm is GSM SIM with COMP128-1, which is known to be possible to crack by brute force. In an H(e)NB setting such attacks could be launched from spoofed network access concentrator on internet if initial communication with access concentrator is not adequately secured.

Probability: Possible.

Impact: Harmful, but only if combined with other attacks.

Threats to assets:

1) H(e)NB: An attacker gain unauthorized access to H(e)NB with above mentioned weak token
2) User: Compromised token can be used to masquerade H(e)NB to User and mount further attacks towards user.
3) Operators Network: An attacker could use the obtained authorization to try to mount further attacks towards the core network.

Mitigation: Any authentication token with a weak algorithm like GSM SIM with COMP128-1 should not be used for H(e)NB authentication. Backhaul link protection mechanism should be strong enough.

NOTE 1: In S3-070614 SA3 answers suggests that for initial authentication S1-based authentication should be used. *"Authentication of Home NodeB to the Serving Network, as well as Serving Network to the Home NodeB is needed and required to ensure overall security of the 3GPP system. As far as authentication when first connected, the security will need to be maintained, perhaps by maintaining a security context between Home NodeB and rest of network. SA3 is currently specifying security mechanisms for S1 interface, which may be applicable to Home NodeB. However, SA3 would also like to add that these answers are not limited to LTE-based Home NodeB's."*

NOTE 2: SA3 have decided to use certificates based authentication on S1 and X2 interfaces in the case of macro eNB.

2) Compromise of H(e)NB authentication token by local physical intrusion

Description: An attacker reads authentication credentials from the wires of the H(e)NB and takes a copy. After that, any other device can use it and impersonate the H(e)NB.

Probability: Depends on the implementation. If the H(e)NB authentication data is not stored in a protected domain, such as a TPM module or a UICC, the probability of such compromise is high. Otherwise, low.

Impact: Harmful. Threats assets are the same as in the previous case.

Mitigation: Authentication credentials of the H(e)NB shall be stored inside a secure domain i.e. from which outsider cannot retrieve the credentials.

3) Inserting a valid authentication token into a manipulated H(e)NB.

Prerequisites: H(e)NB authenticates to the network with a removable token (e.g. a UICC) or an embedded UICC or TPM that can be physically removed (i.e. case 3 and 4).

Description: User inserts/installs valid authentication token into a fake H(e)NB.

Impact: A device (manipulated H(e)NB) with some other functionality (re-flashed H(e)NB, or an H(e)eNB from another, incompatible manufacturer), can identify itself to the operator using a valid credential, and proceed with any kind of security violation. The consequences on the unknowing user are due to manipulations of the H(e)NB.

Threats to assets:

1) Threats to H(e)NB: Introduce malicious configuration changes
2) Threats to user: eavesdropping, impersonation of legitimate user due to H(e)NB manipulation.
3) Threats to operator: Attacks to the infrastructure (radio, core), misuse of user channels, changed signalling.

Mitigation: A non-removable authentication token is helpful to mitigate the risk. Also new users could be required to explicitly confirm their acceptance before being joined to an H(e)NB. This way an H(e)NB owner could only perform eavesdropping/masquerade attacks against those who join the H(e)NB. This approach relies on additional access control being enforced in core network, not just only at the H(e)NB.

It is possible that introducing device authentication or binding removable token to certain H(e)NB can also mitigate the risk, which may need a combination of a removable token and an onboard token.

4) User cloning the H(e)NB authentication Token.

Prerequisites: The token used to authenticate H(e)NB can be cloned and is inserted in a genuine H(e)NB.

Description: Attacker clones authentication credentials of legitimate H(e)NB and installs credentials into another H(e)NB. The cloned H(e)NB is activated near the legitimate H(e)NB. The difference to Threat 3 is that the attack is mounted using an unmodified, legal H(e)NB.

Impact: very harmful.

Threats to assets:

1) Threats to H(e)NB: --
2) Threats to user: Ability to eavesdrop/spoof GSM/3G/LTE calls would have serious and wide-ranging impacts. If the H(e)NB works in an open mode and UP ciphering terminates inside H(e)NB, the impact of the attack is worse since the attacker could eavesdrop or spoof any mobile terminal, not just those authorized to use the cloned H(e)NB.
3) Threats to operator: Issues appear in case a bill would be related to the H(e)NB owner based on H(e)NB identity. H(e)NB owner may be billed for attacker's calls which is routed by cloned H(e)NB.

Mitigation: the authentication credentials of the H(e)NB should be difficult to clone. Also new users could be required to explicitly confirm their acceptance before being joined to an H(e)NB. This way an H(e)NB owner can only perform eavesdropping/masquerade attacks against those who join the H(e)NB. This approach relies on additional access control being enforced in core network, not just at the H(e)NB. Multiple instances of the same H(e)NB should not be allowed simultaneous access to the core network. Some forms of location locking (e.g. to DSL line) may also help to mitigate this threat.

5) Man-in-the-middle attacks on H(e)NB first network access

Prerequisites: H(e)NB does not have unique authentication credentials, pre-installed at the factory or inserted into the H(e)NB.

Description: H(e)NB makes a first contact to the operator's network. During this contact, operator's endpoint cannot reliably identify the peer. An attacker on the internet can intercept all traffic from H(e)NB and later get access to all private information, impersonate the H(e)NB and so on. If the authentication data is not unique to the H(e)NB, a replay attack can be possible.

Probability: Possible.

Impact: Very Harmful.

Threats to assets:

1) Threats to H(e)NB: --
2) Threats to user: Such attack allows for eavesdropping of all the data, passing between the H(e)NB and the network, and also for sending any data on behalf of any party.
3) Threats to operator: If the attacker get in the possession of non-unique initial contact credentials then an attacker may try to obtains network access for whatever H(e)NBs..

Mitigation: H(e)NB shall have authentication credentials already during the very first contact with the network. These credentials shall be recognized at the operator's side. Un-authenticated traffic should not be accepted even at the "first-contact" phase. Either USIM on a UICC, or vendor certificates could be used for this. The logistical consequences could be different. UICC could be inserted in the H(e)NB by the point of sales or customer. Vendor certificate has to be inserted in the H(e)NB at stage of manufacture.

For certificate based solution, mutual authentication is performed between first contact node (i.e. Security GW) and H(e)NB.

For UICC-based solutions, mutual authentication is between HSS and UICC. Certificate of first contact node (i.e. security GW) may be used to authenticate itself toward H(e)NB if necessary.

6) Booting H(e)NB with fraudulent software ("re-flashing")

Description: Boot software at the H(e)NB is modified by the attacker.

Probability: Very likely if a user-accessible boot code update method is used. For example, re-flashing of mobile phones to avoid various restrictions is a common practice in some parts of the world.

Impact: up to disastrous. Possibility to use any software can mean any violation of the security:

Threats to assets:

1) Threats to H(e)NB: Adding non-official software may cause non-optimized functioning of the H(e)NB.
2) Threats to user: eavesdropping on communication, impersonation towards the network.
3) Threats to operator: attack on the radio interface (jamming), denial of service possibilities.

Mitigation: Booting process shall be secured by the cryptographic means, for example using a TPM module. Additional security measures may be needed in case of USIM-based H(e)NB authentication towards the network.

7) Fraudulent software update / configuration changes

Description: H(e)NB should naturally accept software updates from the network. If the software distribution center is compromised, a huge number of access points may receive and install malicious software.

Probability: Possible. A compromise of the SW distribution center / O&M facility is required first. The software distribution centre / O&M facility is supposed to be located in a secured network domain. However possibility of a malicious insider / disgruntled employee should not be discounted.

Impact: Extremely harmful. Possibility of very powerful distributed attacks if many H(e)NB are impacted.

Threats to assets:

1) Threats to H(e)NB: Adding non-official software may cause non-optimized functioning of the H(e)NB.
2) Threats to user: eavesdropping, impersonation
3) Threats to operator: attacks on the radio interface, service costs: all compromised access points must be manually re-flashed. Denial of service attacks to the network could mounted.

Mitigation: All software updates and configuration changes shall be cryptographically signed, and H(e)NB shall have means to verify the signature.

8) Physical tampering with H(e)NB

Description: H(e)NB components could be modified or replaced.

Probability: Possible. A user (attacker) could change components in his H(e)NB, e.g. to extend coverage

Impact: Harmful.

Threats to assets:

1) Threats to H(e)NB: Physical tampering may introduce some degradation of H(e)NB lifetime.
2) Threats to users of H(e)NB: Malicious HW configuration may imply health risks. Modified RF components may interfere with other wireless devices in the environment of the user and cause them to malfunction.
3) Threats to operator: an H(e)NB with modified RF components could have adverse affects on surrounding macro network.

Mitigation: H(e)NB shall be physically secured to a moderate extent to prevent easy replacement of components. Trusted computing techniques could be used to detect when critical components are modified or replaced..

9) Eavesdropping of the other user's UTRAN or E-UTRAN user data

Prerequisites: H(e)NB leaves user traffic unprotected in some part of the H(e)NB; this refers in particular to the HeNB and HNB where UP ciphering terminates inside HNB.

Description: an attacker purchases H(e)NB, installs it, and configures to the ***open access*** mode. Data, which is neither available unprotected on air-interface, nor with IP-interface security, is read (for example, by inserting a card in the bus of the H(e)NB, where that data flows). Victim is using normal air interface, but camps to this H(e)NB without knowledge. All data, flowing between the victim and the network, could be read.

Probability: Possible. First, reading data from wires (e.g. memory bus) is still difficult. Second, manufacturers are strongly recommended (or even requested) to run the processing inside one chip. If a manufacturer cannot provide this, then at least some obfuscation or encryption with a secret key would be applied to the open data.

Impact: (very) harmful, dependent on sensitivity and value of communicated data.

Threats to assets:

1) Threats to H(e)NB: The threats of physical tampering are described in Threat 8.
2) Threats to users of H(e)NB: Privacy of users can be seriously harmed without them ever knowing about it. Such H(e)NB can be used as a "general air interface sniffing device", unless users, concerned about their privacy and suspecting that they are eavesdropped, choose to select network manually on their devices. If the H(e)NB works in an open mode, the impact of the attack is worse since the attacker could eavesdrop any mobile terminal, not just those authorized to use the H(e)NB.
3) Threats to operator: --.

Mitigation: Unprotected user data should never leave a secure domain inside H(e)NB. The user could be notified when the UE camps on a closed or open type H(e)NB. User could be notified (or give his/her explicit acceptance) when he/she is added to the access list of a closed type H(e)NB.

NOTE 1: The H(e)NB can work in open access mode, closed access mode and hybrid access mode.

NOTE 2: The threat not only applies to open mode, but to closed mode as well. See following scenario: Suppose members of the same family, who once added their numbers to the access list. Later, Marc installs a sniffing device, and records everything what Bernhard is talking with his friends. This is not acceptable. And explicit adding does not help: Bernhard still expects that his calls are private.

10) Masquerade as other users

Prerequisites: H(e)NB leaves user traffic unprotected in some part of the H(e)NB; this refers in particular to the HeNB and HNB where UP ciphering terminates inside HNB.

Description: an attacker purchases H(e)NB, installs it, and configures it to the ***open access*** mode. Victim is using normal air interface network, but camps to this H(e)NB without knowledge. All data, flowing between the victim and the network, could be read. The difference with Threat 9 is that that in 9 the 'attacker' only listens, while in threat 10 attacker also injects spoofed traffic.

Threats to assets:

1) Threats to H(e)NB: The threats of physical tampering are described in Threat 8.
2) Threats to user: Attacker can eavesdrop the victim's data or spoof calls from H(e)NB towards core network masquerading as victim without his/her knowledge. In LTE spoofing calls might be difficult due to NAS security between UE and MME, but spoofed calls would be possible in 3G if encryption function has been collapsed into HBTS/HNB. Even if spoofed connection set ups are not possible in LTE, then packet injection type attacks would still be possible even with NAS security in place.
3) Threats to operator: --.

Probability: Possible, but probably more difficult than eavesdropping threat.

Impact: (very) harmful. Ability to spoof 3G/LTE calls would have serious and wide-ranging impacts. If the H(e)NB works in an open mode, the impact of the attack is worse since the attacker could eavesdrop any mobile terminal, not just those authorized to use the H(e)NB.

Mitigation: Unprotected user data should never leave a secure domain inside H(e)NB. The user could be notified when the UE camps on a closed or open type H(e)NB. User could be notified (or give his/her explicit acceptance) when he/she is added to the access list of a closed H(e)NB.

NOTE: The H(e)NB can work in open access mode, closed access mode and hybrid access mode.

11) Changing of the H(e)NB location without reporting

Description: Customers may relocate the H(e)NB and make the provisioned location information invalid.

Probability: Very likely.

Impact: Harmful.

Threats to assets:

1) Threats to H(e)NB: None
2) Threats to user: Emergency call from such H(e)NB cannot be reliably located, or routed to correct emergency centre. This also violates governmental requirements in some counties.
3) Threats to operator:
   - Frequency planning of other operators may be affected in the new place. In some countries, operators are mandated to report all emitters at certain frequencies to authorities.
   - Lawful interception position reporting becomes impossible.
   - Revenue leakage as customer may get preferential call rates even when outside their authorized home/office zone. The would especially be a problem if H(e)NB is taken to another country.

Mitigation: Location locking mechanism shall be designed and implemented. If a removable token-based approach is used for authenticating the H(e)NB (case 3 or 4), it may be easier for an attacker to benefit from a weak or non-existent location locking mechanism.

12) Software simulation of H(e)NB

Description: The communication of the H(e)NB with the core network is simulated by a software application running on a computer connected to the home network, with or without the user's consent.

Probability: Probably low, depending on the strength of the authentication of the H(e)NB with the Core network and on the measures to prevent removal/cloning of the authentication token, but if the token is removable, even by hardware manipulation, a legitimate H(e)NB owner could deliberately perform this attack.

Impact: Very harmful.

Threats to assets:

1) Threats to H(e)NB: Operator could bar misbehaving simulator potentially also affecting the genuine H(e)NB.
2) Threats to user: If H(e)NB simulation software runs without the users' consent, the internet connection of the user could maliciously abused by an attacker.
3) Threats to operator: (if fraudulent user runs the simulation intentionally)
    - Simulated H(e)NBs can easily be cloned or carried to other locations. Lawful interception position reporting becomes impossible.
    - Revenue leakage as customer may get preferential call rates even when outside their authorized home/office zone.
    - Denial of service attacks could be carried out.

Mitigation: As software simulation cannot be prevented, is it necessary to enforce strong H(e)NB access authentication and to prevent removal/cloning of the authentication token..

13) Traffic tunnelling between H(e)NBs

Description: A H(e)NB is used at a legal location but with (additional) traffic from one ore more different, not legal locations. The illegal additional traffic is tunnelled via internet to the legal H(e)NB.

Probability: Unclear.

Impact: Very harmful.

Threats to assets:

1) Threats to H(e)NB: Overload conditions may appear
2) Threats to user: If traffic tunnelling takes place without the users' consent, the H(e)NB of the user could be maliciously abused by an attacker.
3) Threats to operator:
    - Calls or data traffic can originate from any location. Lawful interception position reporting becomes impossible.
    - Revenue leakage as customer may get preferential call rates even when outside their authorized home/office zone.

Mitigation: H(e)NB should be able to detect traffic that does not originate from locally connected UE. One countermeasure is to enforce that only authenticated UE is allowed to be used with the H(e)NB.

14) Misconfiguration of the firewall in the modem

Description: Home access point (like H(e)NB) are normally connected to the Internet via some wired access (e.g. ADSL, cable modem). In these cases, a modem/router could be integrated with the H(e)NB, or be in a separate box. Firewall in the modem/router normally is controlled by the user via some web interface. But the H(e)NB requires defined network services (such as TCP or UDP ports) to communicate with a GW of the core network. These services being closed prevent the H(e)NB from connecting to the operator's network. If the modem is not integrated with the H(e)NB, user shall configure it properly, which is error-prone.

Probability: Possible.

Impact: Annoying, mainly service reliability and usability degradation.

Threats to assets:

1) Threats to H(e)NB: --
2) Threats to user: Denial of service. If emergency calls are prohibited, the impact could be life-threatening.
3) Threats to operator: --

Mitigation: In case when the modem/router is integrated with the H(e)NB, it shall have pre-defined and not changeable configuration of the H(e)NB access channel. In case when the modem is a separate box, its correct configuration shall be enforced. One possible approach may be using uPnP mechanism. An additional firewall within the H(e)NB would also be useful.

NOTE: It should be clarified under which conditions emergency calls are allowed via close/open H(e)NBs (SA1).

15) Denial of service attacks against H(e)NB

Description: attacker organizes (probably distributed) denial of service attacks against H(e)NB.

These attacks can fall into three categories:

1) Layer 1-3 attacks (e.g. ARP, IP related)
2) Layer 4 attacks (e.g. TCP, IGMP, UDP)
3) Layer 5-7 attacks (e.g. Any application layer protocol supported by the H(e)NB).

Probability: Possible.

Impact: Annoying. H(e)NB is not vulnerable to denial of service attacks more than any IP device on the Internet. When the IP-level cryptographic protection of the S1/Iu-link is used, DoS traffic (which is assumed to be unauthenticated) is filtered out already at the authentication phase.

Threats to assets:

1) Threats to H(e)NB: ---
2) Threats to user: denial of service
3) Threats to Operator: ---

Mitigation: H(e)NB is partially relieved from the processing load if a firewall at the modem is present, and configured to pass only IKE negotiations and ESP-encrypted traffic to the H(e)NB. We note that IKEv2 (when used on e.g. S1 or X2) is more robust against DoS attacks than IKEv1.

16) Denial of service attacks against core network

Description: attacker organizes (probably distributed) attacks against elements in the core network from (multiple) H(e)NB(s) or from the backhaul link. The types of threats at all layers are described in threat #15 above. In addition, there are following two categories of threats that can be directed to the core network that would not get directed at the H(e)NB:

1. IKEv2 attacks that can be mounted against initial establishment of the IKEv2 tunnel between the H(e)NB and the Security Gateway. These types of attacks can include:
    - IKE_SA_INIT flood attack
    - IKE_AUTH attack
    - Flood of legitimate tunnels attack (exhausting resources on the Security Gateway)
    - Malformed IKE_SA packets
    - Malformed authentication credentials

2. Layer 5-7 volume attacks and IKEv2 volume attacks in situations during which a high volume of signaling traffic or IKEv2 tunnel setup traffic overwhelms the infrastructure within the H(e)NB network. Some of the different events that may cause these spikes in traffic volume include:
   - power outages and brownouts
   - misconfigurations of core layer 2 and 3 network devices
   - mass calling events as a result of activities such as interactive Media Events, or natural disasters
   - H(e)NB software upgrades that contained signaling bugs such as more frequent registrations or additional security tunnel setup attempts (even a small percentage of H(e)NB software upgrades with bugs could affect an entire H(e)NB population)

   These types of legitimate traffic spikes could induce the following resultant behavior (dependent on particular solution which is chosen finally):
   - IPsec tunnel terminator signaling overload: too high rate of IKEv2 signaling packets
   - AAA server overload: too high rate of requests from the IPsec tunnel terminator in case USIM based H(e)NB authentication would be chosen.

Probability: Possible: Very likely for a compromised H(e)NB, unlikely otherwise.

Impact: From annoying to extremely harmful. The operator's service can be disrupted across a large number of H(e)NBs. Note that when the IP-level cryptographic protection of the S1-link is used, DoS traffic from unauthenticated hosts is filtered out already after the authentication phase. Only compromised H(e)NBs with valid authentication credentials can start acting as DDoS bots.

Threats to assets:

1) Threats to H(e)NB: --
2) Threats to user: DoS as consequences of operators networks DoS
3) Threats to operator: denial of service and loss of revenue

Mitigation: Core network elements that shall be secured include:
- Security gateway as first context point in the core network (assume that HNB gateway for Iu concentration architecture coincides cfr RAN3)

The core network elements shall be protected against mentioned security threats.

- For layer 3-7 volume attacks, the Security Gateway shall be remain available in the event that a high rate of IPsec IKEv2 signaling messages are handled by the Security Gateway. The Security Gateway shall protect the upstream network from overload and overflow conditions.

17) Compromise of an H(e)NB by exploiting weaknesses of active network services.

Description: H(e)NB will usually have several network services (protocol handlers) listening on its network interface(s). These services may be required for operation (e.g. DHCP, IKE, IPsec, PPPoE), or they may be listening due to the device's design (e.g. RPC portmapper). Specifically crafted attack traffic injected via the backhaul network or the local connection may cause protocol handlers to fail, and subsequently compromise the whole H(e)NB.

Probability: Possible. This is the most prevalent type of remote attack in IP networks.

Impact: Extremely harmful. Possibility of very powerful distributed attacks if many H(e)NB are impacted.

Threats to assets:

1) Threats to H(e)NB: Adding non-official software may cause non-optimized functioning of the H(e)NB.
2) Threats to user: eavesdropping on communication, impersonation towards the network.
3) Threats to operator: attack on the radio interface (jamming), denial of service possibilities. Attacks directed against the Core Network or Management Centres.

Mitigation: Minimised network services (disabled or firewalled), robustness testing for functional protocol handlers, intrusion detection looking for abnormal H(e)NB behaviour, regular reset to a securely verified system state.

18) User's network ID revealed to H (e)NB owner

Prerequisites: The owner of a H(e)NB is able to add / delete users to / from the to the H(e)NB related Closed Subscriber Group (CSG).

Description: IMSI may be revealed to the owner of the H(e)NB during CSG management.

Probability: High

Impact: Breaking users privacy

Threats to assets:

1) H(e)NB: none
2) Users: Privacy issue
3) Operator network: none (tracking of subscribers may be possible)

Mitigation: A link between IMSI and owner given user ID is stored in the network or secure stored in H(e)NB.

Editor's Note: The users privacy solutions should not interfere with the identity confidentiality mechanisms provided by the core network.

19) Mis-configuration of H(e)NB

Prerequisites: The attacker has access to the H(e)NB configuration. Access can be both wired or wireless.

Description: Having access to the H(e)NB configuration the attacker can either get hold of the complete H(e)NB or can make some configuration changes that will impact the service being provided by the H(e)NB. Possible attacks and their impact are dependent on the amount of configuration possible at the H(e)NB thus many things are possible, e.g., traffic forwarding.

Probability: Depending on implementation and deployment

Impact: Irritating to harmful

Threats to assets:

1) H(e)NB: Modification of the configuration leading to different issues including malfunctioning and denial of service.
2) Users: From privacy and confidentiality issues to DoS attacks
3) Operator network: If the attacker succeeds in traffic forwarding then it could potentially also cause some form of DoS attack on the network.

Mitigation: Secure access to configuration of H(e)NB is needed.

20) Mis-configuration of access control list (ACL) or compromise of the access control list

Prerequisites: The attacker has access the ACL (which includes CSG list) . This can be either by knowing the administrators password or by physical access to the H(e)NB.

Description: The attacker modifies the ACL thus allowing devices that should not have access to the network. Attacker could also remove devices that should have access and possibly change the level of access for different devices.

Probability: Depending on implementation and deployment

Impact: Irritating to harmful

Threats to assets:

1) H(e)NB: Modification of the ACL..
2) Users: Potential DoS attack or change in access rights
3) Operator network: Free service could be provided to some users if the billing is H(e)NB based.

Mitigation: Secure means of creation, maintenance and storage of ACL is required.

21) Radio resource management tampering

Prerequisites: The attacker has access to the H(e)NB and can modify the resource management aspects of the H(e)NB, at least the attacker should be able to tamper with the power control part of the H(e)NB. Changes could be made by configuration of the H(e)NB or by external means, e.g., increasing the interference or noise.

Description: The H(e)NB gives radio resource information that is incorrect thus leading to issues like increased handover, handover of all mobiles in the vicinity to the H(e)NB or forced handover of all devices from H(e)NB to other (e)NBs. The radio resource information could be simply in the form of the transmit power level. The attacker could perform simple modification like range extension adding signal booster to antennas leading to increased interference, increase in range in which cheap rate applies etc.

Probability: Possible

Impact: Potentially harmful

Threats to assets:

1) H(e)NB: Modification in H(e)NB radio behaviour
2) User: Potential denial of service
3) Operator network: Could lead to frequent handover (ping-pong). Provisioning of service increased area than planned leading to monetary loss. Potential disruption of H(e)NB services.

Mitigation: There should be no means to control the radio resource related parameters by a user. The configuration interface of the H(e)NB must have adequate security. It will be difficult to provide protection against range extension.

22) Masquerade as a valid H(e)NB

Prerequisites: The attacker should have a H(e)NB and be able to configure the H(e)NB such that users of a given CSG will join it.

Description: The attacker buys a H(e)NB and configures it similar to that of a H(e)NB of a CSG. Having done that the attacker (1) changes the setting in the H(e)NB to no encryption and integrity level or (2) has access to the user keys in the H(e)NB. The attacker can do this by connecting the H(e)NB to the wired backbone of the H(e)NB provisioning company or use multi-hop solution to connect the H(e)NB to the valid one connected to the wired network.

Probability: Depending on implementation and deployment

Impact: Very harmful

Threats to assets:

1) H(e)NB: none
2) User: Privacy issues, confidentiality issues, monetary issues and DoS
3) Operator network: Having the user keys the attacker can perform different attacks one of them could lead to mis-charging of the user.

Mitigation: CSG setting and other configuration should be hidden. There should be binding between H(e)NBs and the users it can serve that should also be known by the network. The H(e)NB must be authenticated by the network. The case of key leakage requires that the keys in a H(e)NB is stored in a secure location.

23) Provide radio access service over a CSG

Prerequisites: The attacker has a H(e)NB and valid connectivity to a CSG.

Description: There can be different ways in which the attacker can work (1) connect the H(e)NB to one of the H(e)NB in the CSG using Ethernet cable (2) the attacker has a UE (mobile or data card) connected to its H(e)NB belonging to the CSG that by some means is connected to the attackers H(e)NB (or other radio like 802.11 access point). This can be easily achieved by the attacker connecting a UE and an access point to a laptop. The attacker can then do several attacks some of them similar to that described in attack “Masquerade as a valid H(e)NB” and other being the provisioning of free service over the H(e)NB belonging to a CSG.

Probability: Depending on implementation and deployment

Impact: Depending on implementation and deployment

Threats to assets: Same as "Masquerade as a valid H(e)NB".

Mitigation: Radio layer forwarding is difficult to mitigate. They might require RF fingerprinting. Network layer forwarding attacks require similar mitigation as the following threat.

24) H(e)NB announcing incorrect location to the network

Prerequisites: The intruder is in position to modify the H(e)NB or to mis-inform the H(e)NB regarding its location. Further the H(e)NB is expected to work only at a given location.

Description: The attacker either changes the location information of a H(e)NB or is in position to mis-inform H(e)NB regarding its location. Thus a stolen H(e)NB could be used in unwanted place.

Probability: Possible

Impact: Harmful especially for emergency call services.

Threats to assets'

1) H(e)NB: Manipulation in the form of mis-informing the location
2) User: Users might have no service in primarily expected location. Emergency calls might be routed to the wrong location.
3) Operator network: Provisioning of services meant for different location with potential impact on revenue.

Mitigation: Secure location solution is needed.

Requirement: It should not possible to manipulate location information of a H(e)NB. Secure location functions which are supported in the H(e)NB could be preserved by the Trusted Environment.

25) Manipulation of external time source

Prerequisites: H(e)NB shall perform time synchronization based on an external time source. The time source is either a surrounding macro cell from the same or alternative trusted network and/or a clock server located in an independent network and accessed via the Security Gateway. It should be noticed that a clock server located in an independent trusted network is needed anyway since the H(e)NB may be deployed outside of a macro cell coverage area.

Description: An attacker can tamper with the procedures for time synchronization of the H(e)NB in order to make the H(e)NB perform incorrectly. An attacker can install a false macro cell near the victim H(e)NB and force it to perform time synchronization based on the false macro cell. The attacker can also perform an attack on the insecure link between the H(e)NB and the clock server located in the fixed network.

Attacker can mount an attack on clock function in the H(e)NB directly or indirectly via insecure link between H(e)NB and clock server. The effect of the attack is prevention of timing functions from performing correctly and mis-synchronization that may in turn cause other ill effects.

Probability: Unlikely

Impact: Harmful

Threats to assets:

1) Threats to H(e)NB: H(e)NB can not work without clock information. Wrong clock information will incorrectly set the timing of the H(e)NB and which may force it to perform operations, e.g. handover operations or use of expired/revoked digital certificates used for authentication..

2) Threats to user: UE camped on H(e)NB with wrong clock information will experience a low quality of service. e.g. timing synchronization or handover operations.

3) Threats to operator: Low quality service is provided to the user. A clock server suffering attack will affect macro cells or H(e)NBs which perform time synchronization based on it.

Mitigation: H(e)NB should be notified about information of macro cells from which the H(e)NB can obtain clock information so that it can perform time synchronization based on particular macro cell. A trusted clock server should be located behind the security gateway and communication between the clock server and H(e)NB should have adequate protection. Secure clock synchronization and maintenance functions which are supported in the H(e)NB could be executed within the Trusted Environment.

26) Environmental/side channel attacks against H(e)NB

Prerequisites: The attacker is able to change environmental influences like power supply, temperature or communication link of a H(e)NB.

Description: H(e)NB security mechanism may be circumvented or security lowered

Probability: Possible

Impact: harmful

Threats to assets:

4) H(e)NB: Environmental attacks may introduce some degradation of H(e)NB lifetime
5) Users: Confidentiality and privacy issues
6) Operator network: Integrity and confidentiality issues

Mitigation: Environmental attacks robust Implementation; monitoring of power supply, temperature, data connection

27) Threat: Attack on OAM and its traffic

Prerequisites: The intruder has access to the OAM – H(e)NB communication link.

Description: The operator can decide to connect the OAM to the H(e)NB via the SeGW or directly.

If OAM is inside the operator network then the issues and solutions for the link between H(e)NB and SeGW will be the same as for any communication and is already discussed in this TR. There could be other threats instead (a) there would be possibility of insider attacks on the path from the SeGW to OAM, where management protocols are unprotected and (b) here we have a protocol implementation related issue: OAM interfaces usually do not rely on a single function. They usually bring 4-10 different protocols inside the box: for fault management, command line, web GUI, configuration management, firmware download, SW license checking, some 3rd party interfaces. Even if all of them would be cryptographically secure, there would still be the issue of implementation robustness. Even (cryptographically) "secure" protocols will have flaws that can compromise the system. The more of them are accessible (aka "open ports") via the backhaul network, the higher the risk.

When the H(e)NB is directly connected to the OAM then the intruder can have access to the communication link between the OAM and H(e)NB thus it can perform different attacks like (a) sniffing the traffic, (b) man-in-the-middle attack (c) mis-configuration of the H(e)NB etc.

Impact: very harmful.

Threats to assets:

1) Threats to H(e)NB: Potential denial of service or modification of configuration
2) Threats to user: Depending on attack on the H(e)NB itself, different threats are possible on the user.
3) Threats to operator: OAM could be attacked by the intruder that itself could be a major issue. H(e)NB service failure is also a threat for the operator.

Mitigation: The communication between the H(e)NB and the OAM should be secured.

28) Threat of H(e)NB network access

Prerequisites:

The H(e)NB SeGW or other network entity in the core network has no or can't obtain the profile information, e.g. access control information of the service domain for H(e)NB, or the state information of the H(e)NB, to check whether the H(e)NB can access the network.

Description:

Whether a H(e)NB can access the network will depend on the acquired status information of enable or disable the H(e)NB from the network entity (e.g OAM Server). But for a rouge H(e)NB, it can attempt connect to the network even if the status information of the H(e)NB is set to disable. If there is no such information (e.g. access control information of the service domain for H(e)NB, or the state information of the H(e)NB) in H(e)NB GW or other network entity to check the access right of the H(e)NB, the rouge H(e)NB can gain the network accessibility.

Impact: Harmful

Threats to assets:

1) Threats to H(e)NB: --
2) Threats to user:
   - the attacker could eavesdrop or spoof any mobile terminal that camped on the H(e)NB.

Editor's note: The threat to user needs to be verified.

3) Threats to operator:
   - Free service could be provided to the users camped on the H(e)NB if the billing is H(e)NB based.
   - An attacker could use the obtained authorization to try to mount further attacks towards the core network.

Mitigation: H(e)NB SeGW or other network entity in CN should have or can obtain the related profile information, e.g. access control information for H(e)NB, or the status information of the H(e)NB, to check whether a H(e)NB can access the network when it attempts to access the network.

Editor note: Detail of mitigation should be added.

29) Handover to CSG H(e)NBs

Prerequisites:

User can change the Allowed CSG List stored in the UE.

Description:

Handover decision is taken by the radio network while the Allowed CSG List is stored in the UE and access control is done in the core network or the H(e)NB Gateway. Thus it is possible for a rogue UE to perform handover to a H(e)NB with a given CSG ID, to which it does not belong to, by simply modifying the Allowed CSG list. This can be an issue particularly for the case where the handover has to happen for an on-going session because for such case access control might not be performed.

Probability: High

Impact: Harmful.

Threats to assets:

1) Threats to H(e)NB: None
2) Threats to user: The H(e)NB owner might end-up paying the charges for the rogue user.
3) Threats to operator: Such usage could mean that the operator cannot charge anyone for the service used.

Mitigation: Even on handover the network should check whether the given UE is allowed to access the target H(e)NB.

5.1.4 Threats Impact Overview

In this section we present two tables. The first table shows the asset that is impacted by a given threat and the second table shows the risk level of a given threat. The risk level is given by multiplying the probability of a given threat by the impact a given threat can have. Both probability and impact are divided in four levels and scored as 0.25, 0.5, 0.75 and 1.

Table 1 maps threats to assets.

Table 1: Threats/Asset Correspondence

Threat/Asset correspondence	H(e)NB	User	Operator
Threat-1	X	X	X
Threat-2	X	X	X
Threat-3	X	X	X
Threat-4	--	X	X
Threat-5	--	X	X
Threat-6	X	X	X
Threat-7	X	X	X
Threat-8	X	X	X
Threat-9	X	X	--
Threat-10	X	X	--
Threat-11	X	X	X
Threat-12	X	X	X
Threat-13	--	X	X
Threat-14	--	X	--
Threat-15	--	X	--
Threat-16	--	X	X
Threat-17	X	X	X
Threat-18	--	X	--
Threat-19	X	X	X
Threat-20	X	X	X
Threat-21	X	X	X
Threat-22	--	X	X
Threat-23	--	X	X
Threat-24	X	X	X
Threat-25	X	X	X
Threat-26	X	X	X
Threat-27	X	X	X
Threat-28	--	X	X
Threat-29	--	X	X

Table 2 normalizes threats in matrix format.

Table 2: Threat Matrix

Threat	Threat Likelihood probability	Impact	Risk-Level	Comments
1	Possible (0.25)	Medium (0.25)	0.0625; Low	
2	Unlikely-Very Likely (0.1 – 1.0)	Medium (0.25)	0.025 – 0.25; Low-Medium	
3	Possible (0.25)	Medium (0.25)	0.0625; Low	
4	Possible (0.25)	High (0.5)	0.125; Medium	
5	Possible (0.25)	High (0.5)	0.125; Medium	
6	Very Likely (1.0)	Very High (1.0)	1.0; High	High
7	Possible (0.25)	Very High (1.0)	0.25; Medium	Medium
8	Possible (0.25)	Medium (0.25)	0.0625; Low	
9	Possible (0.25)	Medium-High (0.25-0.5)	0.0625-0.125; Low-Medium	
10	Possible (0.25)	Medium-High (0.25-0.5)	0.0625-0.125; Low-Medium	
11	Very Likely (1.0)	Medium (0.25)	0.25; Medium	Medium
12	Unlikely (0.1)	High (1.0)	0.1; Low	
13	Unlikely(0.1)	High (1.0)	0.1; Low	
14	Possible (0.25)	Low (0.1)	0.025; Low	
15	Possible (0.25)	Low (0.1)	0.025; Low	
16	Possible (0.25)	Low-Very High (0.1-1.0)	0.025-0.25; Low-Medium	
17	Possible (0.25)	Very High (1.0)	0.25; Medium	Medium
18	Likely (0.5)	Medium (0.25)	0.125; Medium	
19	Possible (0.25)	Low-Medium (0.1-0.25)	0.025-0.0625; Low	
20	Possible (0.25)	Low-Medium (0.1-0.25)	0.025-0.0625; Low	
21	Possible (0.25)	Low-Medium (0.1-0.25)	0.025-0.0625; Low	
22	Possible (0.25)	High (0.5)	0.125; Medium	
23	Possible (0.25)	Medium (0.25)	0.0625; Low	
24	Possible (0.25)	Medium (0.25)	0.0625; Low	
25	Unlikely (0.1)	Medium (0.25)	0.025; Low	
26	Possible (0.25)	Medium (0.25)	0.0625; Low	
27	Likely (0.5)	High (0.5)	0.25; Medium	
28	Likely (0.5)	High (0.5)	0.25; Medium	
29	Likely (0.5)	Medium (0.25)	0.125; Medium	

The above table contains a threat matrix.

Further work is needed to validate the assignment of threat likelihood probabilities and impact levels to the threats. Even after this validation has been performed, the threat matrix provides indicative results only, and shall not be the only method used to prioritize the threats.

5.2 Specific HNB Threats

Editor's Note: This section analyses the threats caused by introducing HNB to UMTS network. Possible solutions to these threats are listed in chapter 7.1.

5.3 Specific HeNB Threats

Editor's Note: This section analyses the threats caused by introducing HeNB to EPS network. Possible solutions to these threats are listed in chapter 7.2.

6 Security Requirements

6.1 Common Requirements for H(e)NB

Based on this threat analysis, the security requirements for H(e)NB can be summarized as follows:

1) Only strong authentication algorithms shall be used for (Threats 1, 12).
2) Link protection mechanism between the Security Gateway and the H(e)NB shall be of adequate cryptographic strength. All traffic shall be integrity protected and should be confidentiality protected. (Threat 1, 5).
3) H(e)NB authentication credentials shall be stored inside a secure domain i.e. from which outsider cannot retrieve or clone the credentials (Threats 2, 3, 4, 12).
4) The UE should indicate to the user when it camps on H(e)NB. User should be notified (or give his/her explicit acceptance) when he/she is added to the access list of a closed H(e)NB (Threats 3, 4, 9, 10).
5) H(e)NB and the Security Gateway shall mutually authenticate each other, including the first initial contact (Threat 1, 5, 12).
6) The booting process of the H(e)NB shall be additionally secured by cryptographic means (Threat 6).
7) Software updates and configuration changes for the H(e)NB shall be cryptographically signed (by operator or H(e)NB supplier) and verified configuration changes shall be authorized by H(e)NB operator or supplier (Threat 7).
8) Unprotected sensitive data should never leave a secure domain inside H(e)NB (Threats 8, 9, 10).
9) It shall be possible for the operator to lock the H(e)NB service to a specific geographical location. It shall be possible to disable the H(e)NB if it has been detected to be located at an unauthorized location. (Threat 4, 11)

Editors Note: The above requirement might be of SA1 relevance and should be reviewed by SA1: TS 22.011.

10) UE's shall, unless performing an emergency call, be authenticated and authorized by the user home network before receiving service from the H(e)NB (Threat 5, 13).
11) The security solution shall be compatible with common network address and port translation variations, as well as support firewall traversal (Threat 14).
12) Unauthorized traffic shall be filtered out on the links between the Security Gateway and the H(e)NB (Threats 15, 16).
13) H(e)NB should be run with minimised network services (disabled or firewalled), and test regular for a securely verifiable system state (Threat 17)
14) Access to H(e)NB remote management interface by the operator, shall require authentication and authorization and shall not allow modification to user controlled information unless the user gives their permission (Threat 19).
15) ACL (Access Control lists) should be created and modified by authorized party only (Threat 20).
16) The operator shall have means to control the CSG configuration (Threat 22).
17) It shall not be possible to override the operator's policy at a H(e)NB (Threat 23)
18) It shall not be possible to manipulate location information of a H(e)NB (Threat 24).
19) The authentication credential(s) of each H(e)NB shall be unique (Threat 5).
20) A mechanism shall be provided to restrict the number of simultaneous connections between a specific H(e)NB identity and the H(e)NB home Network. (Threat 4)
21) Only authorized end-users shall be able to request modifications to membership of the Closed Subscriber Group. Operator checks those requests and implements changes if accepted. Only the H(e)NB operator shall be able to enable "open mode" (if supported). (Threat 3, 4, 9, 10)
22) Enforcement of H(e)NB access to Closed Subscriber Group members shall not rely solely on access control methods implemented within the H(e)NB itself. Instead the core network shall be able to check that only mobile users in the relevant Closed Subscriber Group can access services via a specific H(e)NB. (Threat 12)
23) Access to H(e)NB local management interface by the H(e)NB owner if allowed by the operator, shall require authentication and authorization and shall not allow modification to operator controlled information, e.g. H(e)NB licensed radio interface parameters. If the operator allows local management access by the H(e)NB owner, The H(e)NB owner shall be able to select the authorization password. (Threat 6, 7, 21)

Editors Note: The above requirement might be of SA1 relevance and should be reviewed by SA1: TS 22.011. The study/need of audit logs may influence this requirement.

24) H(e)NB enclosure should provide indication of physical tampering (e.g. visual or audible). (Threat 8)
25) IMSI of users connected to H(e)NB connected users must not be revealed to the Hosting party of the H(e)NB (Threat 18)

26) a. Communication between time server and H(e)NB should be provided adequate protection. (Threat 25)

b. The TrE should be able to verify both freshness and integrity of time information from the network. (Threat 25)

Editors Note: Addition of requirement 26b is FFS. This requirement needs to be revisited once the TrE definition is agreed.

27) The implementation of a H(e)NB must be robust against Environmental attacks (Threat 26)

28) Confidentiality and integrity protection shall be provided to OAM traffic between H(e)NB and the OAM Server in the operator network (Threat 27).

29) OAM server and/or operator network should be able to assess the trustworthiness of the H(e)NB's state and its capabilities for secure communication with OAM (Threat 27).

30) IMSI request over the air in clear (without encryption) should only be performed when no other means are available to fetch UE identity (Threat 18).

31) The H(e)NB SeGW or other network entity in CN should obtain the related profile information to check whether the H(e)NB can access the network. (Threat 28)

32) Access control should be performed even during handover. (Threat 29)

6.2 Specific Requirements for HeNB

3GPP TS 33.401[15] introduces in clause 5.3 general security requirements for all types of eNBs. These are basic requirements which shall be fulfilled by all types of eNBs. Thus this document has to consider all requirements given in that clause and more detailed in clauses 11, 12 and 13 of [15] for eNB security.

[15] leaves it explicitly to other documents to specify more stringent requirements, if seen appropriate there. Thus this reference to [15] does not restrict the current document, as long as all requirements of [15] are still kept.

NOTE: To avoid duplication of text from [15] in this document, the detailed requirements of [15] are not repeated here.

Editor's Note: it is ffs whether possible usage of TLS towards OAM has an impact on this clause.

Editor's Note: it has to be clarified, if some requirements of [15] should not be applied to HeNB, and if consequently parts of this clause and of clauses in [15] have to be adapted.

6.3 Countermeasures for H(e)NB

Based on these requirements, the countermeasures can fulfil the requirements can be summarized as follows:

1) Mutual authentication and Security tunnel establishment mechanisms
2) TrE of H(e)NB
3) Access Control mechanisms
4) Location Locking mechanisms
5) Clock Synchronization Security mechanisms
6) Security mechanisms for OAM
7) Protections mechanisms for Environmental Security of H(e)NB
8) User authentication mechanism
9) HPM authentication (If used)

Table 3 shows matrix of requirements and countermeasures mapping.

Table 3: Requirements and countermeasures mapping

SECURITY REQUIREMENT	COUNTERMEASURES	HOW TO FULFILL	THE SECTION BE REFERRED
1. Only strong authentication algorithms shall be used.	Countermeasure 1	Certificate-based authentication and EAP-AKA-based authentication can provide this.	7.5 Authentication Implementation Options If the description in the TR is not enough, some text should be added to this section.
2. Link protection mechanism between the Security Gateway and the H(e)NB shall be of adequate cryptographic strength. All traffic shall be integrity protected and should be confidentiality protected.	Countermeasure 1	The IPsec tunnel between the SeGW and the H(e)NB provide this.	7.6.2 Backhaul Traffic Protection for H(e)NB
3. H(e)NB authentication credentials shall be stored inside a secure domain i.e. from which outsider cannot retrieve or clone the credentials.	Countermeasure 2	Both HPM and TrE are secure domain in H(e)NB.	7.2 Secure Storage and Execution
4. The UE should indicate to the user when it camps on H(e)NB. User should be notified (or give his/her explicit acceptance) when he/she is added to the access list of a closed H(e)NB.	Countermeasure 3	This can be provided by the Access Control mechanisms.	The countermeasure not described in the TR and should be included in an appropriate section o f chapter 7.
5. H(e)NB and the Security Gateway shall mutually authenticate each other, including the first initial contact.	Countermeasure 1	Certificate-based authentication and EAP-AKA-based authentication provide this. For EAP-AKA, authentication is based on an appropriate AKA credential for H(e)NB and network certificate for the SeGW. For Certificate-based authentication, authentication is based on device certificate for H(e)NB and network certificate for the SeGW.	7.5 Authentication Implementation Options
6. The booting process of the H(e)NB shall be additionally secured by cryptographic means.	Countermeasure 2	The boot software can be stored in a TrE in the H(e)NB.	7.2 Secure Storage and Execution
7. Software updates and configuration changes for the H(e)NB shall be cryptographically signed (by operator or H(e)NB supplier) and verified configuration changes shall be	Countermeasure 6	All software updates and configuration changes should be cryptographically signed, and H(e)NB should have means to verify the	7.10 Security Mechanism for OAM.

authorized by H(e)NB operator or supplier.		signature.	
8. Unprotected sensitive data should never leave a secure domain inside H(e)NB.	Countermeasure 2	The sensitive data can be stored in a TrE in the H(e)NB.	7.2 Secure Storage and Execution
9. It shall be possible for the operator to lock the H(e)NB service to a specific geographical location. It shall be possible to disable the H(e)NB if it has been detected to be located at an unauthorized location.	Countermeasure 4	The operator can lock the H(e)NB service to a specific geographical location.	7.7 Location Locking mechanisms
10. UE's shall, unless performing an emergency call, be authenticated and authorized by the user home network before receiving service from the H(e)NB.	Countermeasure 8	UEs with valid subscriptions accessing the operator's network via H(e)NB are authenticated with their own credentials by the network (e.g. USIM contained in UE).	7.9 Access Control Mechanisms for H(e)NB
11. The security solution shall be compatible with common network address and port translation variations, as well as support firewall traversal.	Countermeasure 1	The IPsec tunnel between the SeGW and the H(e)NB is setup based on IKEv2 mechanisms which support network address and port translation.	7.6.2 Backhaul Traffic Protection for H(e)NB
12. Unauthorized traffic shall be filtered out on the links between the Security Gateway and the H(e)NB.	Countermeasure 1	The IPsec tunnel between the SeGW and the H(e)NB provide this. Editor's Note: This may not be a full mitigation depending on the interpretation of the requirement	7.6.2 Backhaul Traffic Protection for H(e)NB
13. H(e)NB should be run with minimised network services (disabled or firewalled), and test regular for a securely verifiable system state.	Countermeasure 1	H(e)NB can run minimized network services.	It depends on the configuration of H(e)NB and out of scope of the TR.
14. Access to H(e)NB remote management interface by the operator, shall require authentication and authorization and shall not allow modification to user controlled information unless the user gives their permission.	Countermeasure 2	The H(e)NB configuration information can be stored in a TrE in the H(e)NB, the TrE can provide the secure access to configuration of H(e)NB.	7.2 Secure Storage and Execution
15. ACL (Access Control lists) should be created and modified by authorized party only.	Countermeasure 3	This can be provided by the Access Control mechanisms.	7.9.1 ACL for pre-R8 UE accessing HNB

16. The operator shall have means to control the CSG configuration.	Countermeasure 3	This can be provided by the Access Control mechanisms.	7.9.2 CSG for H(e)NB
17. It shall not be possible to override the operator's policy at a H(e)NB.	Countermeasure 2	The implementation and deployment information can be stored in a TrE in the H(e)NB, the attacker can't get the information.	7.2 Secure Storage and Execution
18. It shall not be possible to manipulate location information of a H(e)NB.	Countermeasure 2	The location information can be stored in a TrE in the H(e)NB.	7.2 Secure Storage and Execution
19. The authentication credential(s) of each H(e)NB shall be unique.	Countermeasure 1	Both AKA credentials and vendor certificates could be used for this and these credentials shall be recognized at the operator's side.	7.5 Authentication Implementation Options
20. A mechanism shall be provided to restrict the number of simultaneous connections between a specific H(e)NB identity and the H(e)NB home Network.	Countermeasure 1	A specific H(e)NB identity is bound to a specific device authentication credentials. Both certificate-based device authentication and EAP-AKA-based authentication can authenticate the specific H(e)NB identity. The authentication credential in the H(e)NB is unique, TrE can ensure the credential can't be cloned	7.5 Authentication Implementation Options
21. Only authorized end-users shall be able to request modifications to membership of the Closed Subscriber Group. Operator checks those requests and implements changes if accepted. Only the H(e)NB operator shall be able to enable "open mode" (if supported).	Countermeasure 3	This can be provided by the Access Control mechanisms.	7.9.2 CSG for H(e)NB
22. Enforcement of H(e)NB access to Closed Subscriber Group members shall not rely solely on access control methods implemented within the H(e)NB itself. Instead the core network shall be able to check that only mobile users in the relevant Closed Subscriber Group can access services via a specific H(e)NB.	Countermeasure 3	This can be provided by the Access Control mechanisms.	7.9.2 CSG for H(e)NB
23. Access to H(e)NB local management interface by the	Countermeasure 2	The radio resource related parameters can be stored in a	7.2 Secure Storage and

H(e)NB owner if allowed by the operator, shall require authentication and authorization and shall not allow modification to operator controlled information, e.g. H(e)NB licensed radio interface parameters. If the operator allows local management access by the H(e)NB owner, The H(e)NB owner shall be able to select the authorization password.		TrE in the H(e)NB..	Execution
24. H(e)NB enclosure should provide indication of physical tampering (e.g. visual or audible).	Out of scope	It depends on the manufacture of H(e)NB, H(e)NB can provide indication of physical tampering.	It depends on the manufacture of H(e)NB and out of scope of the TR.
25. IMSI of users connected to H(e)NB connected users must not be revealed to the Hosting party of the H(e)NB.	Countermeasure 2	The IMSI of users can be stored in a TrE in the H(e)NB.	7.2 Secure Storage and Execution
26. a. Communication between time server and H(e)NB should be provided adequate protection. b. The TrE should be able to verify both freshness and integrity of time information from the network.	Countermeasure 5	Clock synchronization messages can be protected by IPsec tunnel between the SeGW and the H(e)NB. The built-in security protocols of the Clock Synchronization Protocols also can be used to protect the communication between time server and H(e)NB.	7.11 Clock Synchronization Security Mechanisms for H(e)NB
27. The implementation of a H(e)NB must be robust against Environmental attacks.	Countermeasure 7	It depends on the manufacture of H(e)NB, e.g. the protections mechanisms to monitor power supply and temperature can be provided to H(e)NB when H(e)NB is manufactured.	It depends on the manufacture of H(e)NB and out of scope of the TR.
28. Confidentiality and integrity protection shall be provided to OAM traffic between H(e)NB and the OAM Server in the operator network.	Countermeasure 6	OAM traffic can be protected by IPsec tunnel between the SeGW and the H(e)NB or TLS between OAM Server and H(e)NB.	7.10 Security Mechanisms for OAM.
29. OAM server and/or operator network should be able to assess the trustworthiness of the H(e)NB's state and its capabilities for secure communication with OAM.	Countermeasure 6	OAM traffic can be protected by IPsec tunnel between the SeGW and the H(e)NB or TLS between OAM Server and H(e)NB.	7.10 Security Mechanisms for OAM.
30. IMSI request over the air in clear (without encryption) should only be performed when no other means are available to fetch UE	Countermeasure 3	This requirement may happen when the pre-rel-8 UEs access to a H(e)NB.	7.9.1 ACL for pre-R8 UE accessing HNB

identity.			
31. The H(e)NB SeGW or other network entity in CN should obtain the related profile information to check whether the H(e)NB can access the network. (Threat 28)	No Countermeasure	The H(e)NB GW or other network entity in CN obtains the related profile information to check whether the H(e)NB can access the network.	The countermeasure not described in the TR
32. Access control should be performed even during handover. (Threat 29)	Coutermeasure 3	Perform access control also during handover.	Not discussed in current TR.

7. Common Security mechanisms solutions for H(e)NB

7.1 H(e)NB Authentication Principle

The following authentications are necessary for H(e)NB authentication:

a) Mutual authentication of H(e)NB device and the operator's network. Authentication algorithms using the credentials stored in the Trusted Environment (TrE) should be executed inside of the TrE.
This authentication is mandatory.
This mutual authentication shall include (or be tightly bound to) a validation of the platform integrity (i.e. TrE properties).
The two parts of the mutual authentication have the following properties:

a1) The identity of the H(e)NB is authenticated by the network. The credentials for this authentication shall be stored in a TrE in the H(e)NB.

a2) The identity of the operator's network (e.g. represented by Security Gateway – SeGW) is authenticated by the H(e)NB. This authentication may either authenticate the operator's network in general, or the particular SeGW contacted by the H(e)NB.

b) Authentication of the hosting party by the operator's network: The identity of the hosting party is authenticated by the operator's network.
This authentication is optional.
This authentication may be performed in two ways:

b1) The authentication of the hosting party is based on credentials contained in a separate Hosting Party Module (HPM) in H(e)NB. This authentication is performed as additional step over authentication according to a) above.

NOTE 1: Binding of authentication according to b1) to authentication according to a) is handled in section 7.6 together with the respective authentication methods..

b2) The authentication of the hosting party is bundled with the device authentication, i.e. there is no additional authentication step after authentication according to a) above.

NOTE 2: The method of binding of identity of hosting party to identity of H(e)NB is a management topic in operator network and thus out of scope of this document.

If no hosting party is existing (e.g. the operator itself provides the H(e)NB), then authentication b) may not be relevant.

NOTE 3: The authentications described above refer to the authentication of the H(e)NB itself and the hosting party. UEs with valid subscriptions accessing the operator's network via H(e)NB are authenticated with their own credentials by the network (e.g. USIM contained in UE).

7.2 Secure Storage and Execution

7.2.1 Hosting Party Module

Editor's Note: Other HPM implementations, e.g. with binding of HPM to the device, are FFS and may be added later.

The (optional) Hosting Party authentication is based on a Hosting Party Module. The Hosting Party Module (HPM) is a physical entity distinct from the H(e)NB physical equipment, dedicated to the identification and authentication of the Hosting Party towards the MNO. Since HPM contains the credentials used to authenticate the Hosting Party, it should be a tamper resistant environment. The HPM is bound to the Hosting Party and provided by the MNO to the Hosting Party. HPM is removable from the H(e)NB and, in particular, it should be possible for a Hosting Party to change the H(e)NB device by inserting the HPM in the new H(e)NB.

Physical security of interfaces to the HPM has to be considered.

NOTE 1: MNO can prevent unauthorized change of H(e)NB based on the (mandatory) Device Authentication

NOTE 2: HPM allows an optional customization of the H(e)NB based on Hosting Party identity, without impacts on the manufacturers.

Editor's Note: Clarity about the physical entity need to be added later.

7.2.2 Trusted Environment (TrE)

7.2.2.1 General

A Trusted Environment (TrE) is a logically separate entity and set of functions and resources within a H(e)NB. The TrE is a trustworthy environment for the execution of software and the storage of sensitive data, as well as for the protection of particular hardware functions, where needed.

The TrE should provide isolation of the TrE *versus* surrounding. Software executables and data to be secured in the TrE are functionally *and* informationally separated from the H(e)NB as a whole and protected from unauthorized access and tampering. Moreover, data produced through execution of functions within the TrE should be practically unknowable to external entities. The security of the TrE should be assured by physical security of appropriate component(s) and storage that protects data it holds from unauthorized access and tampering.

A TrE should be protected by a secure start-up process, where the TrE is locally ensured to reach a determined, trustworthy state in a normal start-up or boot process. Secure start-up may extend further to the operating system and other secure programs at operation time.

A TrE should securely store the HPM identity TrE.

Editor's Note: A TrE should have its own, unique identity (TrE_ID) that is bound to the identity of the H(e)NB itself. For simplicity, the TrE_ID stored within the TrE may be used interchangeably with the H(e)NB_EI.

A TrE should provide protected functions needed to perform H(e)NB device authentication with a SeGW.

The TrE should be pre-provisioned with any required security-sensitive functions, cryptographic keys and other credentials that relate to the H(e)NB's identity using a secure, out-of-band process. The TrE should be capable of securely authenticating its identity to authorised external entities using standardised protocols. These entities can validate a TrE_ID as being that of a valid, issued, TrE and hence H(e)NB.

A TrE should have cryptographic capabilities needed to perform device authentication and other security-sensitive functions. Examples of such capabilities may include symmetric and asymmetric encryption and decryption, hash-value generation and verification, random number generation, and digital signature creation and verification. A TrE may be able to set up and use secure channels with other parts of the H(e)NB.

Since a H(e)NB is a network element, third-party evaluation of the sensitive TrE may be requested by an operator.

Note: Whether and how the TrE should be evaluated is out of the scope of this TR. A guideline on this may need to be provided in the future. Such a guideline may include recommendations on use of evaluation methodologies accepted widely industry-wide.

Protective measures that may be applied include support and enforcement of a security policy for the TrE and the ability to convey assertions about the trustworthiness of the TrE to an authorized external verifier.

Editor's Note: A more advanced TrE could provide additional security functions and algorithms that associate the TrE to the HPM or data the HPM holds. More clarification is needed.

NOTE: A more advanced TrE may also provide isolation of multiple functions inside it, with different software executables, data and possibly hardware functions which may be separated from each other. Additionally, secondary identities for these functions may be embedded, based upon prior authentication with the entity which can verify the TrE through standardised secure protocols. Additional functions are typically provisioned by download after the H(e)NB is deployed.

7.2.2.2 TrE Interfaces

7.2.2.2.1 General

The TrE within a H(e)NB needs to interact with several H(e)NB functional building blocks to securely perform the desired functions such as device authentication and H(e)NB validation. To establish the necessary connections, the TrE must have access to various interfaces to such functions and resources within the H(e)NB. These interfaces of the TrE are generally functions of the TrE, are initialized in the secure start-up process of the TrE, and are thus assumed to operate correctly.

Under these premises, the TrE can be analysed with regard of the security properties of its interfaces to the rest of the H(e)NB's functional building blocks, in order to establish a secure and efficient design of the H(e)NB.

7.2.2.2.2 TrE Interface Categories

There are two broad security categories of TrE interfaces:

1. Unprotected interfaces. These interfaces facilitate communication between the TrE and general resources of the H(e)NB which is not assumed to be secured against tampering and/or eavesdropping. It should be noted that unprotected interfaces can nevertheless give access to data which is cryptographically protected by the TrE, for instance when the TrE is in possession of pertinent key material and cryptographically secures data stored in unsecure memory. Even unprotected interfaces may also benefit from other security measures such as making the interface available only after the TrE checks the code of its counter-part resource across the interface, for example during a secure boot-up of the H(e)NB.

2. Protected interfaces: These interfaces provide either protection of the integrity and/or confidentiality of the data carried across the interfaces. These interfaces use either security protocols which provide encrypted communication or hardware interfaces. If security protocols are used, they may also provide other security-wise beneficial measures such as authentication of the entity with which the TrE communicates with, and message authentication and/or confidentiality.

In the design of a H(e)NB various aspects are relevant for the choice of a particular TrE interface configuration. Unprotected interfaces may be chosen, when the communicating entity does not provide protection of the communicated data. Protected interfaces may be chosen when there is a need to provide protection of data integrity and/or, confidentiality between the TrE and another resource on the H(e)NB that the TrE needs to communicate with.

When an interface needs to be protected, which type of protection mechanism (a security protocol or dedicated hardware interface) needs to be provided and what type of data protection (integrity, confidentiality, or both) is needed depend on the security requirements of the manufacturer and their customers.

7.2.2.3 H(e)NB Authentication

H(e)NB authentication consists of:

a) H(e)NB identity authentication refers to device authentication as described in section 7.6.2;
b) TrE identity authentication;

c) H(e)NB device identity and TrE identity binding;
d) The H(e)NB integrity verification
This refers to the H(e)NB validation, more specifically, it refers to either verification of the signaling message that H(e)NB may send to the SeGW regarding the outcome or an aspect of one of the validation methods it has performed, as described in section 7.5.2

NOTE 1: the H(e)NB identity and the TrE identity can be used interchangeable when needed.

NOTE 2: the authentication we described here is for some specific scenarios when TrE has a separate identity. Whether it can be optimized depends on the specific policy of the operator.

There could be two broad categories of H(e)NB authentication scenarios:

- Initial authentication
- Re-authentication.

During the H(e)NB initial authentication, the H(e)NB identity and TrE identity authentication, the integrity verification, the H(e)NB device identity and TrE identity binding relationship authentication, and some H(e)NB related authentication/verification(such as HPM authentication, H(e)NB Location verification and UE authentication) will be executed. When the H(e)NB related authentication is successfully, the authentication data will be downloaded and stored in the TrE securely.

During the H(e)NB re-authentication, the authentication process is similar to the initial authentication except that the H(e)NB related authentications performed by the TrE against the authentication data it stored as a result of the initial authentication.

Editor's Note: Inclusion of this section depends on definition of TrE identity. Definition of all authentication terms needs to be clarified. Use of initial and re-authentication needs to be clarified. Title will need to be changed later.

7.3 Comparison of H(e)NB Authentication Methods

Editor's Note: The table of "Comparison of H(e)NB authentication Methods" should be added to this section.

7.4 Authentication Method Selection

7.4.1 Authentication Methods

It is agreed that in H(e)NB we will have mandatory device authentication and optional hosting party authentication. Authentication can either be done by certificates based solution or EAP-AKA. This brings a few combinations of authentication methods:

1. Device authentication with certificates, without HP authentication
2. Device authentication with EAP-AKA, without HP authentication
3. Device authentication with certificates, and with HP authentication using certificates
4. Device authentication with EAP-AKA, and with HP authentication using certificates
5. Device authentication with certificates, and with HP authentication using EAP-AKA
6. Device authentication with EAP-AKA, and with HP authentication using EAP-AKA

EAP-AKA is expected to be the solution of choice for HP authentication thus 3rd and 4th authentication combinations given above are not considered in this document.

7.4.2 Authentication Type Identification and Enforcement

In this section, means to identify and enforcing the authentication combination type is explained.

1. Method to check whether there will be only device authentication or both device and HP authentication:

(a) If the *IKE_SA_INIT response* from SeGW includes MULTIPLE_AUTH_SUPPORTED notification then it is clear to H(e)NB that the SeGW will support both Device and HP Authentication using multiple authentication . Otherwise it is clear to H(e)NB that the SeGW can support only Device Authentication.

(b) If the *IKE_AUTH request* from the H(e)NB contains MULTIPLE_AUTH_SUPPORTED notification, then it is clear to SeGW that the H(e)NB can support both Device and HP Authentication using multiple authentication. Otherwise it is clear to SeGW that the H(e)NB can support only Device Authentication.

2. Method to check the type of device authentication (certificate based or EAP-AKA based):

(a) The CERTREQ payload in *IKE_SA_INIT response* from SeGW to H(e)NB indicates that the SeGW supports certificate based device authentication and it may also support EAP-AKA based authentication. Otherwise it is implied that SeGW only supports EAP-AKA based authentication.

(b) The lack of the AUTH payload in the *IKE_AUTH request* from H(e)NB to SeGW indicates to the SeGW that the H(e)NB intends to perform will be EAP-AKA based authentication. In this case, the H(e)NB should not send the CERT payload to the SeGW.

3. HP authentication always uses EAP-AKA

(a) The ANOTHER_AUTH_FOLLOWS notification in the last IKE_AUTH request of Device Authentication from the H(e)NB indicates to the SeGW that the H(e)NB wishes to perform a second authentication to authenticate the hosting party.

(b) The lack of the AUTH payload in the first IKE_AUTH request from H(e)NB of hosting party authentication of the H(E)NB indicates to the SeGW that the H(e)NB intends to perform EAP-AKA based authentication.

4. Authentication Policy Selection and Enforcement

(a) It is the SeGW's responsibility to enforce the operator's authentication policy by rejecting authentication requests it does not like.

(b) The H(e)NB sends its identity in the IDi payload of the first IKE_AUTH request message to the SeGW. The SeGW may use this identity to retrieve the appropriate security profile for the H(e)NB from its own DB or some external source (e.g. a AAA server or OAM).

(c) The security profile may contain the appropriate authentication choices for the H(e)NB.

(d) The SeGW may also be equipped with a separate, generic security profile that is not dependent on the H(e)NB ID. This type of security profile may specify, for example, a 'preferred' authentication type for all H(e)NBs that may attempt to authenticate to the SeGW.

(e) The SeGW should enforce the authentication choices imposed by the security profile by rejecting inappropriate requests from the H(e)NB. For example:

- If the IKE_SA_INIT response from SeGW does not include MULTIPLE_AUTH_SUPPORTED notification but the H(e)NB proceeds to perform both device and HP authentication, it is up to the SeGW (and its security profile) to decide how to handle the case. For example, the SeGW may accept only the Device Authentication part of the multiple authentication, or, alternatively, it may reject both of the authentication attempts from the H(e)NB.

- If the SeGW is capable of and indicates supporting multiple authentication but during looking up the H(e)NB's policy it decides to prefer Device Authentication only, and if the H(e)NB proceeds to initiate HP authentication, it is up to the SeGW to decide and enforce how to handle the H(e)NB's method of authentication.

The following message flow show the typical use of the above IKEv2 message options for the four authentication cases indicated above.

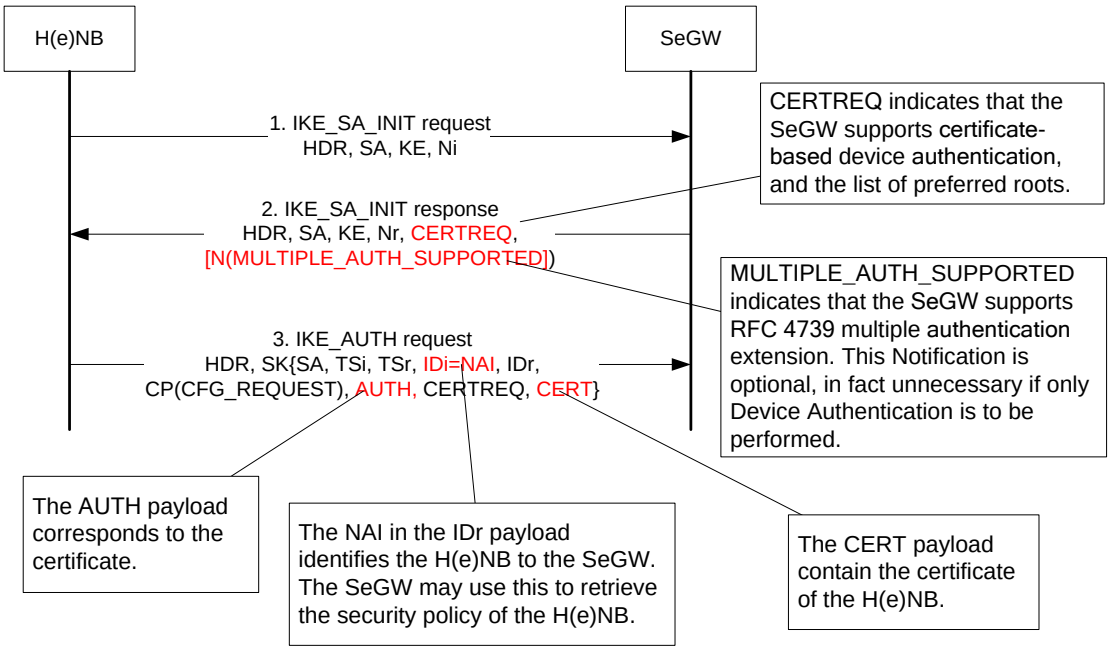


Figure 4: Certificate based Device Authentication.

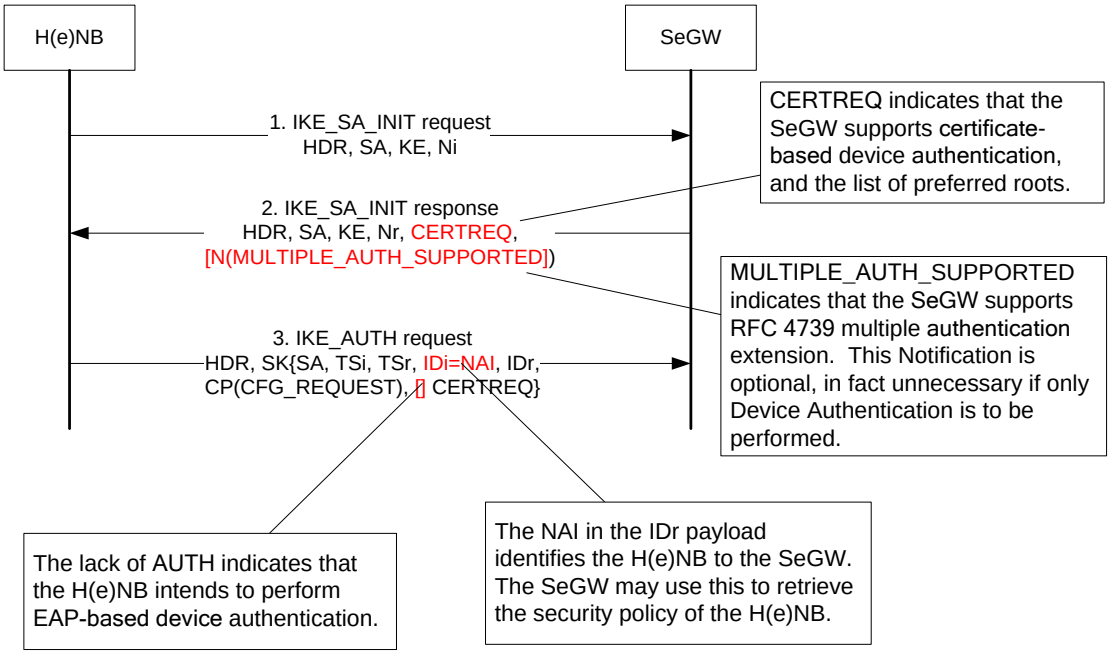


Figure 5: EAP-AKA based Device Authentication

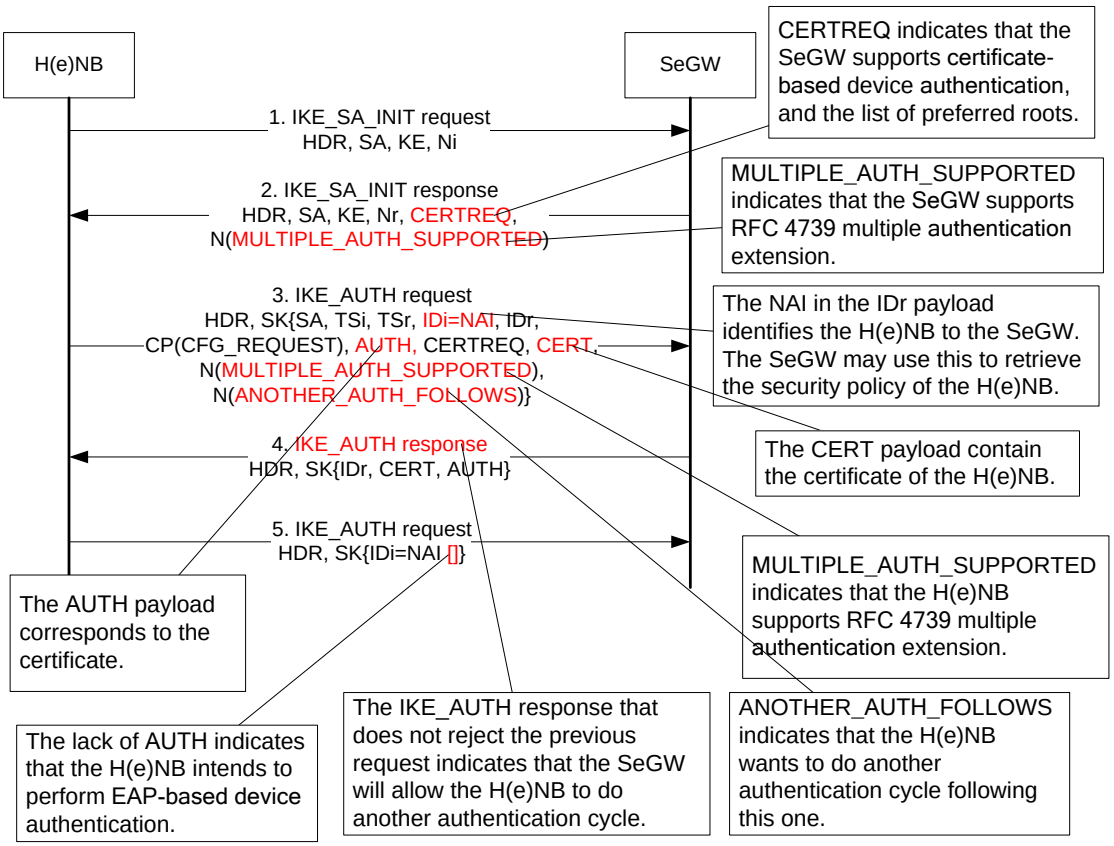


Figure 6: Certificate based Device Authentication followed by EAP-AKA HP Authentication

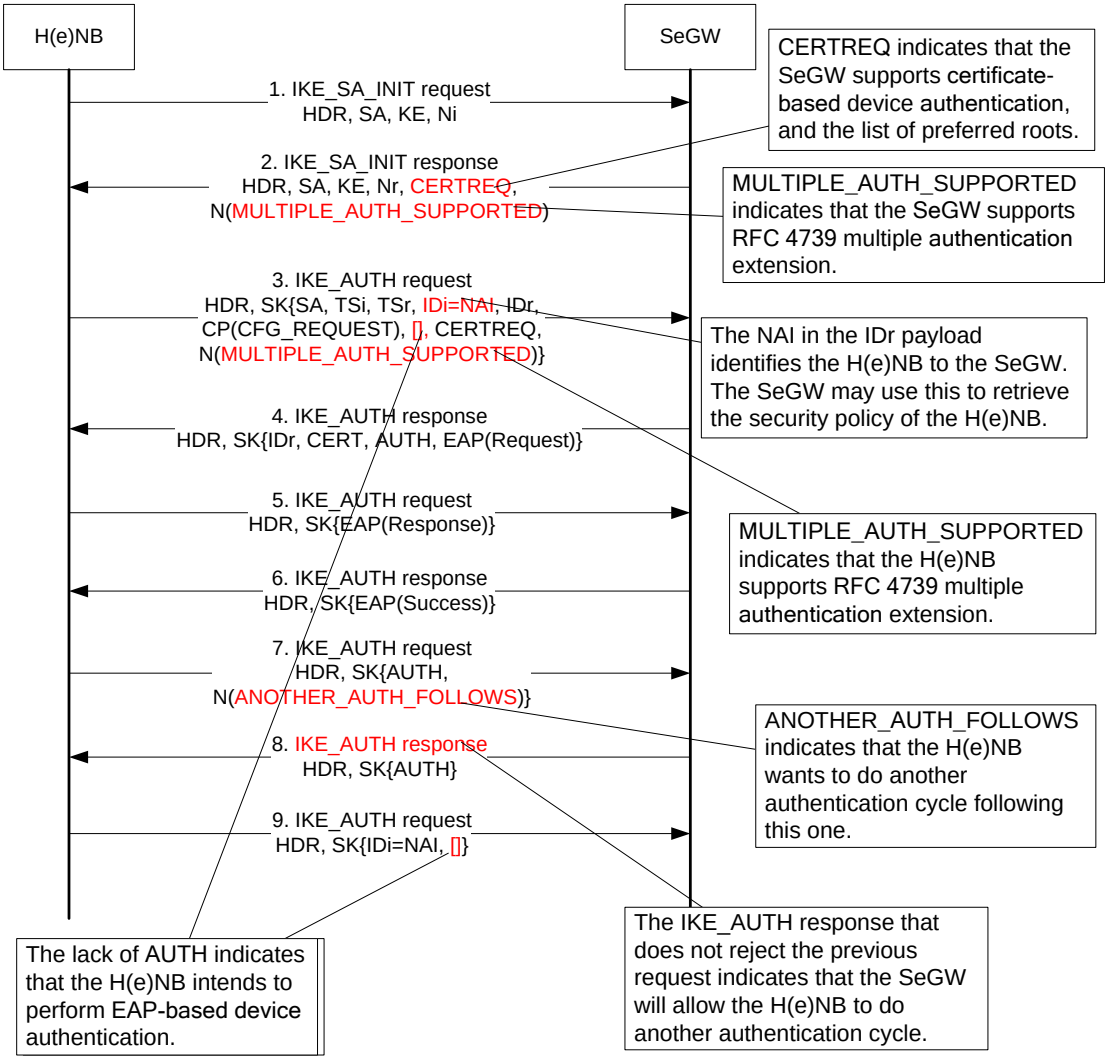


Figure 7: EAP-AKA based Device Authentication followed by EAP-AKA HP Authentication

7.5 Device Integrity Check

7.5.1 General

EAP-AKA authentication only validates the AKA credentials (contained in a TrE). This does not by itself address device authentication or validation and/or possible binding of hosting party authentication to device authentication (cf. sub-section 7.1). In addition a binding between validated device and EAP-AKA based authentication has to be performed. In case of EAP-AKA authentication, two ways for achieving this are known:

1) Logical binding of the TrE holding the AKA credentials (e.g. UICC holding the USIM application) to the H(e)NB. During the EAP-AKA authentication the integrity of the device platform must be validated.

NOTE: There is no standard specifying such check. Also previous attempts of such check have been circumvented quickly ("cracked SIM-lock").

2) Physically binding the TrE holding the AKA credentials to the H(e)NB. During the EAP-AKA authentication the integrity of the device platform must be validated.

In both cases above the actual integrity validation (for HW and SW) has to be performed by a hardware security component securely embedded into the H(e)NB. Note that normally credentials appropriate for EAP-AKA authentication and the related application stored in a physically bound TrE are not designed for the purpose of validating the binding of a removable hardware component to a hosting device.

7.5.2 H(e)NB Validation

7.5.2.1 General

There are four possibilities for H(e)NB validation:

1. Autonomous Validation
2. Remote Validation(RV)
3. Semi-autonomous Validation (SAV)
4. Hybrid Validation (HV)

AuV is an autonomous validation comprises of procedure whereby the H(e)NB's validity is assessed within the H(e)NB itself without depending on external network entities. The success of the validation procedure is implicit in the fact that the H(e)NB successfully performs an authentication process with the network. If the integrity check of any component of the H(e)NB fails, the TrE within the H(e)NB will not give access to the sensitive functions and the private key which are used in the authentication process.

RV is a procedure whereby an external network entity, a Platform Validation Entity (PVE), assesses the validity of the H(e)NB after it receives comprehensive evidence for the integrity check results generated by the H(e)NB's TrE. Since the SeGW is the secure end-point of the core network for the H(e)NB, and since remote validation should take place with an entity that can control access of the H(e)NB further into the network pending the result of the remote validation, the SeGW should act as an enforcement proxy for PVE. The AAA may provide the PVE functionality or a separate network component could also be considered.

In SAV, the H(e)NB's validity is first assessed internally by the TrE without depending on external entities. The TrE first assesses core components of the H(e)NB. If any core component fails the integrity check tests, then the H(e)NB will not attempt to engage in the authentication procedure with the network. If the core components pass the integrity check tests, the H(e)NB then assesses additional components and engages in an authentication process with the network. Information on H(e)NB functions that depend on the modules which failed to pass the integrity check test are signaled securely to the PVE, using the IKEv2 authentication messages. The PVE can then make its own decisions based on that message and validate the H(e)NB.

In HV, validation is performed for different stages of the boot process for secure boot and trusted boot. Validation is performed locally during secure boot. During the trusted boot, integrity measurements collected for local validation are further validated locally and the results of which along with integrity measurements collected for network validation are sent to the network for further validation and verification. What measurements should be checked locally and which of these measurements should be sent to the network are based on decision of a policy in the H(e)NB. The core network makes a H(e)NB access control decision according to the validation results.

> Editor's Note: How these validation techniques could be applied to communication with the HMS via the public internet is FFS.

Validation of the H(e)NB platform should preferably take place before device authentication, although validation after authentication should also be allowed. .

7.5.2.2 Autonomous Validation

If the TrE performs autonomous validation, the following steps could apply:

The H(e)NB may perform an AuV of the integrity of the H(e)NB. Integrity of all code, components and configuration data inside the H(e)NB are checked in a chain of trust based on the TrE and its RoT. The following steps can apply:

1. In stage 1, the RoT checks if the components of the TrE have achieved a predefined state of secure start-up and if they have been successfully integrity checked, loads them.
2. In stage 2, the TrE checks if a pre-defined portion of the rest of the H(e)NB (i.e. components that are pre-defined as part of the secure start-up) have achieved a successful integrity check. The integrity checked code in this stage consists of e.g. basic OS and basic communications to SeGW.
3. If stage 1 and/or stage 2 checks fail then the TrE blocks further stages of integrity checking and authentication from proceeding.

4. If stages 1 and 2 are successful, then stage 3 proceeds, i.e. the remaining H(e)NB modules of code (including e.g. radio access code) are integrity checked.

5. If stage 3 integrity check is successful, the code is loaded and authentication proceeds.

6. If stage 3 integrity check fails, the TrE blocks further stages of integrity checking and authentication from proceeding.

NOTE: stages 1, 2 and 3, as described above, may be combined into fewer stages, according to the implementation of the H(e)NB.

The network becomes indirectly aware of the fact that the H(e)NB has passed an autonomous validation test. For example, when the H(e)NB successfully completes device authentication procedures, the network can know that the H(e)NB ought to have passed its autonomous validation test. This requires binding of the authentication to the successful internal autonomous validation of the device. This can be accomplished if e.g. the private key for certificate based device authentication is stored securely within the device, and is given access to only after successful internal validation. Then the success of authentication proves the successfully passed validation to the SeGW.

NOTE: As the autonomous validation is an internal method for device integrity verification, the binding of a secret or a protected parameter to a successful validation seems to be the only option to indicate a successful autonomous validation.

7.5.2.3 Remote Validation

If the H(e)NB's validity is remotely validated, the following scenario could apply.

1. The H(e)NB starts up to a pre-defined secure state. This step may comprise of the step 1 or steps 1 and 2 of the autonomous validation process described in section 7.3.1.2.

2. The H(e)NB requests the TrE to generate evidence of the platform validity for the H(e)NB.

3. The TrE collects material to be used to produce such evidence from the rest of the H(e)NB. Such material could, for example, critical codes of the H(e)NB, credentials for the H(e)NB's OS, equipment IDs, etc. The TrE generates the evidence for the validate the H(e)NB, and cryptographically protect it (e.g. encrypt for integrity and/or confidentiality).

4. The TrE passes the protected evidence to the H(e)NB,

5. The H(e)NB forwards the protected evidence to the PVE, via SeGW.

6. The PVE evaluates the evidence and determines if the H(e)NB is trustworthy enough to allow it to continue on to perform device authentication. In case such evaluation is done at a PVE that is not the AAA, the PVE should forward the validation evidence it receives from the H(e)NB to the HLR/AAA-server. The PVE forwards its judgement to HLR/AAA, and also informs the H(e)NB to go on with device authentication.

Steps 4 to 6 above could be performed using the same IKEv2 session as is used for device authentication.

NOTE: Whether validation steps described here, if performed before steps for device authentication could introduce delays, and if so, what the impact would be, may need to be studied.

7.5.2.4 Semi-Autonomous Validation

The purpose of semi-autonomous validation (SAV) is to provide a method whereby the PVE has enough evidence to make policy-based decisions, but with a lower messaging overhead than remote validation, whilst providing a higher level of trust than autonomous validation.

The H(e)NB may perform SAV of the integrity of the H(e)NB. Integrity of all code, components and configuration data inside the H(e)NB are checked in a chain of trust based on the TrE and its RoT. The integrity checking process compares the component-wise integrity measurements against their corresponding trusted reference values. The following steps can apply:

1. In stage 1, the RoT checks if the components of the TrE have achieved a predefined state of secure start-up and if they have been successfully integrity checked, loads them. If stage 1 checks fail then the H(e)NB is blocked

from performing further stages of integrity checking, contacting the SeGW, or performing authentication procedures.

2. In stage 2, the TrE checks if a pre-defined portion of the rest of the H(e)NB (i.e. components that are pre-defined as part of the secure start-up) have achieved a successful integrity check. The integrity checked code in this stage consists of e.g. basic OS, basic communications to SeGW and the code which formats SAV reporting messages.

3. If stage 2 checks fail then the TrE blocks further stages of integrity checking, contacting the SeGW, or performing authentication procedures.

4. If stages 1 and 2 are successful, then stage 3 proceeds, i.e. the remaining H(e)NB components (including e.g. radio access code) are integrity checked.

5. If all components of the stage 3 integrity check are successful, the component is loaded and authentication proceeds. The Notify payload field indicates that all stage 3 components have passed the integrity checks.

6. If one or more of the stage 3 integrity check components fail, then these components are not loaded. All components which pass the integrity check are loaded. The TrE prepares a list of the functionalities of the H(e)NB that correspond to any of the components that have failed integrity checks, and sends the list to the SeGW using the Notify payload of IKE_AUTH request.

NOTE 1: In step 5 or 6 above, the data may be signed by the TrE's signing key, to provide authenticity and integrity of that data, over and above the overall message protection provided by the IKE security association.

7. The SeGW authenticates the H(e)NB device using the device certificate. If such authentication is successful, then the SeGW forwards information from the Notify payload to the PVE for a decision on validation. If the PVE decides that the H(e)NB is validated, then the rest of the authentication process, including the authentication of the SeGW, and subsequent establishment of a secure tunnel can proceed.

NOTE 2: The TrE may also put a time-stamp on messages to ensure freshness.. An alternative to time-stamping is that after the protocol for network access starts and then a nonce is supplied by the network to be used by the TrE for combining with the aforesaid message. That may also be a feature of binding the device authentication to the validation.

NOTE 3: stages 1 and 2, as described above, may be combined into a single stage, according to the implementation of the H(e)NB.

NOTE 4: For interoperability and manageability of the SAV, a standardized list of functions must be specified.

Editor's note: Compatibility of this requirement with implementation independence and free interface specification for vendor-independent interoperation is FFS.

Editor's note: the required signalling between the PVE (the policy decision point) and the SeGW (the policy enforcement point) is FFS.

The PVE may use the information forwarded from the SeGW to make fine-grained access control decisions such as:

a. Grant full network access to the H(e)NB if no functional failures are reported.

b. Grant network access to the H(e)NB, with support for some basic functionality, if non critical functional failures are reported. The PVE can alert an H(e)MS to schedule a remote software update for the corresponding components that may have failed integrity checking.

c. Block network access to the H(e)NB if a configuration setting failure has been reported. The PVE can alert an H(e)MS to schedule a configuration update.

d. Block network access to the H(e)NB if a critical functional failure has been reported. The PVE can alert an H(e)MS to perform a remote software update for the corresponding components that failed integrity check.

Editor's note: (a) through (d) above are intended to provide additional security, compared to autonomous validation. The effectiveness of that is FFS.

NOTE 5: The operations above are possible in SAV only by sending interpretable information to the PVE.

7.5.2.5 Policy for H(e)NB Validation

This section describes a mechanism that takes advantage of both trusted start-up and secure start-up with adequate consideration on how the operator's policy applies. Note that this is not a separate method of performing the H(e)NB validation.

Before possible introduction of a policy based mechanism the following topics may have to be clarified:

- Certain measurements/validation values/boot sequences may be existent only in H(e)NBs of some vendors, dependent on selected implementation. Also dependencies between such values and measurements may vary, e.g. which checks are necessary to achieve a certain security level. Such dependencies may even be hidden, as they may result from vendor-specific implementation details. Thus certain measurements may lead to high security level for one implementation, but to a lower level on others. This implementation dependence could lead to proprietary solutions, which do not need standardization, but are then also not interoperable between different vendor's products.

- Making some measurements or checks configurable by policy, all possible ways must be implemented first. Thus either the implementation is minimal (which does not need policies), or a multitude of procedures and configurability has to be provided, which may be complex and thus be counterproductive for an inexpensive customer premise device.

- Policies may need to be expressed in a formalized way and be flexible to accommodate all expected variations of implementations. New formalisms may be necessary which have to be standardized for interoperability. Also the dependencies and achievable security levels have to be formally described, to allow operators an easy statement of policies, while assuring the required security level of the selected policy.

The policy may be divided into two separate parts:

1. Part 1, the locally stored policy in HeNB that rules what measurements should be checked locally by the TrE in the H(e)NB and which of these measurements or data that captures the integrity check made by the TrE about these measurements should be sent to the network;
2. Part 2, the network stored policy that describes how to use the validation results (locally or remotely) and how to make access control decisions, e.g. re-boot, limit access, network isolation or repair online.

The operator is responsible for the policy generation, configuration, update and distribution. Usually the original copy of the policy is stored and maintained in the core network.

The network verifier should be the PVE (which could be implemented in the SeGW or AAA server depending on the operator's implementation). If the SeGW receives the HeNB validation messages, it can handle them itself or forward the messages to the entity that has the ability to verify the integrity of H(e)NB (i.e. PVE or AAA server).

The process of performing validation based on policy is described below:

- Stage 1. Execute a secure boot process STEP by STEP according to the locally pre-configured policy provided by the network.
  E.g. the critical core code needs to be checked with the expected value. Only those definitely necessary checks take place at this time, e.g. BIOS, OS loader,(except for those influence the flexibility).

- Stage 2. H(e)NB executes the local part of trust boot process. Additional components are loaded one by one according to the locally pre-configured policy provided by the network.

- Stage 3. Network executes remote validation to complete the trust boot process.

  - At the end of boot procedure, the TrE sends a signed status message to the network, including the Type 1 local validation result and Type 2 measurements.

  - The network verifier analyzes and assesses the two parts . As to the actual measurement values, it compares them with the expected check-values stored in the network and obtained a remote validation result;

  - Network makes a H(e)NB access control decision according to the validation results (locally and remotely) and the network stored policy. Based on the two assess results, the core network determines whether H(e)NB is allowed to continue the access procedure, or whether H(e)NB is compromised and needs to be isolated or needs to be repaired by OAM.

The policy update and re-validation process may also happen periodically or event-initiated when the H(e)NB has been in active status. Also, OAM and software update may also apply in this stage. A re-validation may require a reboot of the H(e)NB to perform the necessary steps as required by the updated policy.

7.5.2.6 Device Revalidation

Revalidation of the device based on network-initiated reboot can be a routine part of the operational environment of the H(e)NB device. Periodic revalidation will enable the network to have confidence that the device is working in a defined state with reduced risk of rogue code executing. The revalidation will also enable the authentication procedure to be initiated.

Method already available from SA5 specifications (e.g. as based on mechanisms derived from the requirement REQ-OAMP_CM-CON-006 in 32.581 [21], clause 5.1.1) can be used to perform a network initiated reboot of the H(e)NB and thus initiate revalidation of the H(e)NB.

7.5.2.7 Hybrid validation

This section describes a mechanism that takes advantage of both trusted boot and secure boot with adequate consideration on how the operator's rule applies.

The network verifier should be the PVE (which could be implemented in the SeGW or AAA server depending on the operator's implementation). If the SeGW receives the HeNB validation messages, it can handle them itself or forward the messages to the entity that has the ability to verify the integrity of H(e)NB (i.e. PVE or AAA server)

In the Hybrid Validation method (HV), the reference metrics used in the integrity validation process are divided into three categories, and are stored in three locations respectively:

a) Inside the TrE in the H(e)NB protected by higher security level mechanism(high security core), such as hardware storage;

b) Inside the TrE in the H(e)NB but protected by cryptographical methods (i.e. in an ordinary security environment), such as software storage.

c) Inside the PVE, protected by secure methods chosen by the operator.

The reference metrics are all produced by the vendor.

As for a), they are provided before the H(e)NB is delivered to the operator, and the operator does not need to manage or update them.

As for b), they are provided before the H(e)NB is delivered to the operator. But they could and should be updated by the CN, possibly by OAM operation.

As for c), they are provided when the H(e)NB is delivered to the operator and before the H(e)NB is installed.

The reference metrics held in the PVE are provided by the H(e)NB vendor and they should have already been configured before the H(e)NB powers on . So when the H(e)NB initiates network-access , the reference metrics do not need to be conveyed to the PVE to verify the data or modules that may need to be upgraded potentially, such as OS, upper layer software, or configuration data. If and when these components are upgraded, the matching reference metrics come from the CN, so that there is no need to convey the reference metrics to the PVE.

The 3-stage process of performing HV is described below:

Stage 1. The H(e)NB executes a secure boot process, step by step, according to the locally pre-configured policy.

E.g. the critical core code needs to be checked against the expected value (e.g. BIOS, OS loader).

Stage 2. The H(e)NB executes the local part of trusted boot process. Additional components are loaded one by one according to the locally pre-configured policy.
Based on a locally stored policy (i.e. rule), the integrity measurements are taken and are grouped into two types: 1) Type 1, for local validation. 2) Type 2, for network validation.
For the type 1 measurements, the TrE uses pre-stored reference metrics to validate their integrity in the H(e)NB and determines the local validation result. Then it stores the validation result for network checking. For the type

2 measurements, TrE collects and prepares the measurement data (i.e. TrE signs the data) for sending to the network verifier for further validation.

Stage 3. The network executes remote validation to complete the trusted boot process.

- At the end of the boot procedure, the TrE sends a signed status message to the network, including the Type 1 local validation results and Type 2 measurements;

- The network verifier analyzes and assesses the two parts. As to the actual measurement values, it compares them with the expected check-values stored in the network and obtained a remote validation result.

The network makes a H(e)NB access control decision according to the validation results (locally and remotely). Based on the two assessment results, the core network determines whether H(e)NB is allowed to continue the access procedure, or whether H(e)NB is compromised and needs to be isolated or needs to be repaired by OAM.

7.5.3 Analysis of Device Integrity Validation

Editor's Note: The current content of this section is obsolete and needs to be updated after an agreed threat analysis and feasibility study of all non-autonomous validation methods is completed.

Various methods for performing device validation are analyzed.

The following properties are relevant for a selection:

- Root of trust: Both variants require an immutable root of trust (SW and possibly data) to exist in the device.

- Execution of validation check:

  - The remote validation variant requires the existence of an attestation server within the operator network, which must be provided with device type and SW version specific validation check data. This results in considerable management effort for this server including push of new version validation check data from the manufacturer to the operator.
    In addition a remote attestation protocol has to be specified, which is either 3GPP specific, or gives a close binding to a specific validation and attestation method, if taken from some other standardisation body.

  - The autonomous validation variant requires the provisioning of the device itself with validation check data, e.g. together with the SW downloaded. This requires the device to be able to check the integrity of the validation check data, which can be accomplished by signing this data by the manufacturer, and including the root certificate of the manufacturer into the root of trust of the device.

- Handling of multiple backhaul links: If more than one backhaul link is established, then for remote validation the successful validation has to be ensured for every link establishment (cf. sub-clause 7.7.1).

  - In case of remote validation this can be achieved either by some information infrastructure in the network keeping track of the validation state of each device, or by performing the remote validation separately for each link establishment.

  - In case of autonomous validation, the successful establishment of the link, which includes successful authentication of the device, is by itself proof of the passed validation check.

Editor's Note: It needs to be clarified why the claim in the above that a successful establishment of a secure backhaul link itself should be treated by itself proof of the passed validation check.

From the above it is seen that the security level of both variants is not very different, as both rely on an immutable root of trust in the device.

Editor's Note: It needs to be clarified why the security level of the autonomous validation and a remote validation should be considered as not very different from each other;.

But the required management is different, requiring for the remote validation case an additional server, specification of an additional attestation protocol, and more complex management procedures for manufacturer and operator.

Editor's Note: It needs to be verified if any real or perceived disadvantage of remote validation, such as the added complexity, would outweighs its merits on balance.

Editor's Note: A semi-autonomous validation, with some signaling about the outcome of local device integrity check sent from the H(e)NB to the SeGW, may also need to be considered.

Autonomous validation is the simplest method and requires the least additional efforts. But the core network has little control and knowledge of the H(e)NB's trust state. Even a small error may lead to a startup failure which may also require maintenance personnel to perform manual recovery of the device. It introduces maintainence issues at the expense of protocol simplicity. Both AV and RV methods has their own merits and pitfalls. The HV method can be used to take advantage of the merits of the two methods.

HV has the equivalent network traffic load and complexity of SAV. Unlike in the SAV where the actual measurement value is not sent to the PVE, the HV sends the actual measurement values to the PVE. This requires the PVE to have to store the reference values for the measurements and update them as necessary. In either cases, the policy (i.e.rule) should be pre-configured in H(e)NB.

7.5.4 Study of Device Integrity Validation Methods

7.5.4.1 Terms of Reference

The following investigations and clarifications are seen as necessary beyond the existing descriptions in the present document:

1 Threat models /description of attacks and clean derivation of security features of validation from the threat model.

2 Threat analysis with explicit relation to the different validation methods:

1. Which threats/attacks may be countered by autonomous validation?
2. Which additional threats/attacks identified in the present document may be countered by "explicit" (non-autonomous) validation, which are not caught by autonomous validation?
3. Are there (other) existing countermeasures available for the threats identified in 2.2., which do not rely on validation?

3 Specify the "open interfaces" for full vendor interoperability. This is common in 3GPP and shall allow implementation of H(e)NBs and NEs independently, based on specification only.

1. What are the measurement values to be stored and transferred in a manner which is independent from H(e)NB architecture and implementation?
2. What requirements apply to the transfer of information received from the H(e)NB as a result of validation (transport over existing channels, binding of validation and authentication, etc.)?

4 Specify the procedures and architectures in the network which are necessary for full vendor interoperability.

1. What are the possible reactions in SeGW or H(e)MS on this detailed information received from the H(e)NB as a result of validation, in case of differences to the expected information?
2. How is the expected set of information, e.g. the measurement values, determined by Validation Entity, e.g. dependent on vendor, HW type, and SW version?
3. Where do the reference values used by the Platform Validation Entity come from (push by vendor, pull by MNO, ...)? What is needed from the infrastructure to support this (Network elements, interfaces)?
4. What are the relations to existing and proposed H(e)NB S/W distribution methods and channels included in TR069 (e.g. for H(e)MS based update of H(e)NB SW)?

5 Describe remediation methods and their security implications.

1. What remediation methods (repairing, re-loading of SW in secure way, etc.) are possible on a suspected compromised device?
2. How do validation reporting methods assist the remediation from (suspected) compromised state of H(e)NB?

6 What is the trade-off between added security and cost / complexity (cost / benefit trade-off) between countermeasures and effort?

7.5.4.2 Scope of Study

- All validation methods for H(e)NBs need a threat and feasibility study in a TR giving answers to the items in the section on "necessary investigations" above.

- For all validation methods, the study also needs to examine the trade-off between added security and cost/complexity. Such trade-off is essential if the H(e)NB is to be a low-cost device, with low CAPEX /OPEX in MNO networks.

- No pCRs on extensions to validation concepts are agreed for the TS 33.320 before the findings of the feasibility study and recommendation in TR are available. Changes concerning validation methods shall be incorporated into the TS only if they are deemed by SA3 to represent "stable solutions."

7.5.4.3 Threat Analysis of Validation Methods

7.5.4.3.1 General

The following sections detail the threat analysis study of various validation methods under consideration. The analysis includes security requirements, threats and countermeasures, and a conclusions section.

7.5.4.3.2 Security Requirements for AUv

Requirements which could be addressed by AuV are as follows (extracted and summarised from section 5):

6. "The booting process of the H(e)NB shall be additionally secured by cryptographic means."

> The stages of validation involve verification of a data authentication pattern, e.g. a signed hash, on the blocks of code to be verified.

7. "Software updates and configuration changes for the H(e)NB shall be cryptographically signed (by operator or H(e)NB supplier) and verified configuration".

> Validation involves verification of a data authentication pattern, e.g. a signed hash, on the blocks of code to be verified.

13. "H(e)NB should be run with minimized network services (disabled or firewalled), and test[ed] regular[ly] for a securely verifiable system state."

> Validation can verify the executable firewall code and firewall settings if the latter are embedded in the code block to be verified.

17. "It shall not be possible to override the operator's policy at a H(e)NB."

> Validation verifies the executable code which ensures that this is the case.

29. "OAM server and/or operator network should be able to assess the trustworthiness of the H(e)NB's state and its capabilities for secure communication with OAM."

> Validation provides a means of verifying the state of executable code blocks in the H(e)NB.

7.5.4.3.3 Threats and Counter-Measures Applicable to AuV

The following table shows the mapping of the relevant security requirements applicable to AuV, listed above in 7.5.4.3.2, onto countermeasures and how those counter-measures are mapped onto threats. This analysis, see the right-hand column of the table, thus produces the list of threats which can potentially be mitigated by validation. Even though the cross-referencing of CMs to threats also throws up threats 1, 5, 15 and 27, they have been omitted from the table, because validation is not relevant to them.

SECURITY REQUIREMENT APPLICABLE TO AuV	ASSOCIATED COUNTER MEASURES	THREAT(S) ASSOCIATED WITH THE CMs
6, 17	CM2	2, 3, 4, 6, 7, 8, 9, 10, 12, 18, 19, 21, 23,
7, 29	CM6	7
13	CM1	4, 12, 14, 16, 17

The associated countermeasures (CMs) are listed below for convenience. The threats are not listed, for reasons of brevity.

CM1 Mutual authentication and Security tunnel establishment mechanisms

CM2 TrE of H(e)NB

CM6 Security mechanisms for OAM

CMs 1 and 6 are preventive measures that are not related to validation, so autonomous validation provides a complementary CM which detects an attack if the existing CM fails.

CM2 is strongly related to validation.

7.5.4.3.4 Security Requirements Applicable to SAV

Requirements 6,7,13,17 and 29 are all addressed by SAV, as they were addressed by AuV.

Additional requirements which could be addressed by SAV specifically are as follows (extracted and summarised from section 5):

23. Access to H(e)NB local management interface by the H(e)NB owner, if allowed by the operator, shall require authentication and authorization and shall not allow modification to operator controlled information, e.g. H(e)NB licensed radio interface parameters. If the operator allows local management access by the H(e)NB owner, The H(e)NB owner shall be able to select the authorization password.

SAV provides a means of notifying the network of any unauthorised change to operator controlled information, e.g. H(e)NB licensed radio interface parameters , if said information is stored as part of the component which is integrity-checked and reported to the PVE. The network can then make the decision as to what action is necessary.

Editor's note: it is FFS whether or not SAV can be a useful counter-measure for other security requirements which involve location, e.g. 12 and 18.

7.5.4.3.5 Threats and Counter-Measures Applicable to SAV

The threats that are addressed by SAV in the same way as they are addressed by AuV are listed below, as listed in section 7.5.4.3.2.

SECURITY REQUIREMENT Applicable to SAV in the same way as applicable to AuV	ASSOCIATED COUNTER MEASURES	THREAT(S) ASSOCIATED WITH THE CMs
6, 17	CM2	2, 3, 4, 6, 7, 8, 9, 10, 12, 18, 19, 21, 23,
7, 29	CM6	7
13	CM1	4, 12, 14, 16, 17

The associated countermeasures (CMs), as identified in section 7.5.4.3.2, are:

CM1 Mutual authentication and Security tunnel establishment mechanisms

CM2 TrE of H(e)NB

CM6 Security mechanisms for OAM

CMs 1 and 6 are preventive measures that are not related to validation, so SAV (as does AuV) provides a complementary CM which detects an attack if the existing CM fails.

CM2 is strongly related to all validation methods, including SAV.

Editor's note: it is FFS whether or not SAV can be a useful counter-measure for recognized threats which involve location, e.g. threats 11 and 24.

7.5.4.3.6 Analysis and Conclusions

7.5.4.3.6.1 Autonomous Validation

1. AuV can be employed as a counter-measure for threats 2, 3, 4, 6, 7, 8, 9, 10, 12, 14, 16, 17, 18, 19, 21, 23
2. For threats 4, 7, 12, 14, 16, 17, counter-measures are described in the section 5 which do not involve validation. However, those counter-measures are preventive measures. If the counter-measures fail, then AuV, in accordance with good security practice, provides methods for detecting the attack in the H(e)NB.
3. For threats 2, 3, 4, 6, 7, 8, 9, 10, 12, 18, 19, 21, 23, a counter-measure is described which would be part of AuV.

7.5.4.3.6.2 Semi-Autonomous Validation

SAV can be employed as a counter-measure for threats 2, 3, 4, 6, 7, 8, 9, 10, 12, 14, 16, 17, 18, 19, 21, 23, i.e. the same threats as AuV.

For threats 4, 7, 12, 14, 16, 17, counter-measures which do not involve validation are already described in section 5. However, those counter-measures are preventive measures. If the counter-measures fail, then SAV is more feature-rich than AuV, in that while SAV and AuV both provide methods for detecting the attack in the H(e)NB, SAV also notifies the network of the integrity status of the H(e)NB so that appropriate actions can be taken

Editor's note: it is FFS whether or not SAV can be a useful counter-measure for other security requirements and threats which involve locatio and if, in such cases, SAV could add significant value over and above AuV

7.5.4.3.6.3 Hybrid Validation

Editor's Note: Threat analysis of Hybrid Validation is needed in this section. The analysis should describe what threats (from those that are listed in section 5 of this document) that are addressed by the Hybrid Validation method and how they are addressed.

7.5.4.4 Answers to Questions Concerning Autonomous Validation

This section presents the answers to the questions raised in section 7.5.4.1 as they apply to autonomous validation only.

1. **Threat models /description of attacks and clean derivation of security features of validation from the threat model.**

   A detailed description of the threats and the derived validation security features is provided in section 7.5.4.4 Threat Analysis of Validation Methods. Security requirements for validation are listed which are derived from the threat models in section 5 of the present document.

2. **Threat analysis with explicit relation to the different validation methods:**

   1. **Which threats/attacks may be countered by autonomous validation?**

      See section 7.5.4.3.4.1. for the list of threats that may be countered by autonomous validation.

   2. **Which additional threats/attacks identified in identified in the present document may be countered by "explicit" (non-autonomous) validation, which are not caught by autonomous validation?**

      See section 7.5.4.3.4.1

   3. **Are there (other) existing countermeasures available for the threats identified in 2.2., which do not rely on validation?**

The present document does not provide text that provides a direct mapping from threats (described in section 5) to countermeasures (described in section 6.3). Such a mapping can only be identified by combining the mappings from the threats to the requirements as described in section 6.1 and the mapping from the requirements to the countermeasures that can fulfill them as described in Table 3 in section 6.3.

The following is a result of combining those mappings. For brevity, the mappings themselves are not replicated. Only findings arising from the identified mappings are shown below:

1) Section 6.3 of the present document describes existing countermeasures which are independent of integrity checking and validation and could be considered to mitigate the threats in section 2.2.

2) Table 3, section 6.3 presents several ways that CM 2 (TrE in H(e)NB) may be used to fulfil the requirements that map to threats 16, 19, 20, 21, and 22. These are intended to be preventive measures, whereas Autonomous Validation would also provide a detection mechanism for those threats.

3) For threat 24, there is no countermeasure proposed in section 6.3 that maps to address this threat, whereas validation would provide mitigation.

3. Specify the "open interfaces" for full vendor interoperability. This is common in 3GPP and shall allow implementation of H(e)NBs and NEs independently, based on specification only.

In AuV, the device integrity check is performed locally. In case of a successful integrity check, the device connects to the SeGW and attempts to authenticate. The authentication procedure is then perfomed in the standard manner using IKEv2 as described in section 7.6 and therefore no additional interfaces are required.

1. What are the measurement values to be stored and transferred in a manner which is independent from a H(e)NB architecture and implementation?

In AuV, no information is transferred from the H(e)NB to the network.

However in AuV, the local integrity measurements are compared with trusted reference values. These trusted reference values are the digests of the SW and data components defined and generated by the manufacturer and stored in the H(e)NB. These are specific to each manufacturer and do not need to be specified in terms of standardization.

One aspect that could be standardized is the minimum level of acceptable security for the integrity check algorithm. For example, the trusted reference values must be computed using SHA-1 or equal or better algorithm.

2. What requirements apply to the transfer of information received from the H(e)NB as a result of validation (transport over existing channels, binding of validation and authentication, etc.)?

In AuV, no information is transferred from the H(e)NB to the network.

4. Specify the procedures and architectures in the network which are necessary for full vendor interoperability.

AuV is a local function to the H(e)NB and does not require any additional network support for the procedure itself. If components of the H(e)NB are updated on the device, the trusted reference values for the components should also be updated on the device. Existing mechanisms from the manufacturer and/or the operator that support remote update of software of the H(e)NB can be reused to support remote update of the corresponding trusted reference values for the updated software.

1. What are the possible reactions in SeGW or H(e)MS on this detailed information received from the H(e)NB as a result of validation in case of differences to the expected values?

In AuV, if the integrity measurement values differ from the trusted reference values then the device will not attempt to authenticate with the SeGW.

2. How is the expected set of information, e.g. the measurement values determined by Validation Entity, e.g. dependent on vendor, HW type, and SW version?

There is no validation entity for AuV and hence a PVE is not required. Since AuV is a local function to the H(e)NB, the set of measurement values and trusted reference values are specific to the manufacturer of the H(e)NB.

3. **Where do the reference values used by the Platform Validation Entity come from (push by vendor, pull by MNO, ...)? What is the needed from the infrastructure to support this? (Network elements, interfaces)?**

   In AuV, no network entity such as a Platform Validation Entity exists.

4. **What are the relations to existing and proposed H(e)NB S/W distribution methods and channels included in TR069 (e.g. for H(e)MS based update of H(e)NB SW)?**

   TR069 provides for mechanisms to update software for CPE equipment. This protocol may be used to update the SW in a similar manner to CPE.

5. **Describe remediation methods and their security implications.**

   Currently AuV does not support remediation upon a failure of device integrity checking.

6. **What is the trade-off between added security and cost / complexity (cost / benefit trade-off) between countermeasures and effort?**

   The security and other benefits conferred by the Autonomous Validation (AuV) against a system that does not employ AuV but employs other non-validation countermeasures are described in the Threat Analysis section 7.5.4.3.4.

   If AuV is not employed, the TrE, which is listed as Countermeasure 2, then cannot be made trustworthy using device integrity checking and validation processes. Rather, such a TrE must be implemented as a closed environment which could be trusted only because of its closed nature. Such an H(e)NB system with a closed TrE is used as the baseline for the trade-off analysis given below.

   Against such a system, a system that has AuV (and a TrE that depends on and utilizes the functionality of AuV) would imply the following costs and benefits.

Table 7.5.4-1. Cost Benefit Analysis of Autonomous Validation

Entity	Cost	Benefits
H(e)NB system with closed TrE w/o integrity checking and validation	• N/A (this is the baseline)	• N/A (this is the baseline)
H(e)NB system with TrE that is: 1) integrity-checked by a RoT, and 2) checks the integrity of other components of the H(e)NB	• Large decrease in maintenance and personnel costs, due to the reduced need to have onsite physical maintenance for some types of failure • Potentially large decrease in platform costs, since a H(e)NB system that uses a TrE backed up by AuV: 1) Does not need to be implemented in a large, closed platform 2) Does not need to execute all firmware within a large, closed TrE, to become trustworthy • Small increase of complexity/cost due to implementing integrity checking (done by the RoT to the TrE, and by the TrE to the rest of the H(e)NB), • Small increase of complexity/cost due to the need to provision the Trusted Reference Values (TRV) on the device.	• Ability to make the TrE a very small entity • Ability to easily change /upgrade software and still assure trusted operation • Ability to detect any modified component • Ability to protect the network from access by H(e)NBs with compromised components, by binding authentication to validation

Additionally, the trade-off analysis for the proposed device distress indication is outlined in the table below.

Table 7.5.4-2. Distress Indication Cost Benefit Analysis

Entity	Cost	Benefit
Core Network	• Minimal H(e)MS fucntionality to handle distress signal information	• Ability to put a compromised device in a black list • Optional remote remediation reduces frequency of costly onsite maintenance procedures
H(e)NB	• Small immutable FBC to support distress signal transmission to designated H(e)MS • Optionally additional functionality to support full normal (excluding TrE) code update	• Notifies CN that device is in distress • Optional ability to support replacement of compromised normal code • Reduces need to have onsite physical maintenance for some types of failure • Ability to address an integrity check failure that may occur due to a mis-match between a code version and its TRVs even if the code itself is not compromised

7.5.4.5 Answers to Questions Concerning Semi Autonomous Validation

This section presents the answers to the questions raised in section 7.5.3.1 as they apply to Semi-Autonomous validation only.

1. **Threat models /description of attacks and clean derivation of security features of validation from the threat model.**

   The following security requirements in the present TR can be fulfilled by validation: 6, 7, 13, 17, 29. The threats onto which these security requirements map are listed below. Threats which map onto those security requirements, but to which validation is not relevant, are omitted.

2. **Threat analysis with explicit relation to the different validation methods:**

   1. **Which threats/attacks may be countered by Semi-autonomous validation?**

      SAV can be a counter-measure for threats 2, 3, 4, 6, 7, 8, 9, 10, 12, 14, 16, 17, 18, 19, 21, 23 28 In addition to detecting these threats, SAV includes mechanisms for the network to be informed that an attack has been detected, to make the decisions as to whether to block or allow network access and for the H(e)NB to recover from an attack by being remediated by the network.

      Editor's note: it FFS whether or not location-based threats such as 11 and 24 can be mitigated by SAV

   2. **Which additional threats/attacks identified in the TR may be countered by "explicit" (non-autonomous) validation, which are not caught by autonomous validation?**

      Editor's note: it FFS whether or not location-based threats 11 and 24 can be mitigated by SAV and, if so, if SAV offers a significant advantage over AuV

   3. **Are there (other) existing countermeasures available for the threats identified in 2.2., which do not rely on validation?**

      For threats 4, 7, 12, 14, 16, 17, 28, counter-measures are described in the present TR which do not rely on validation. However, those counter-measures are purely preventive measures. If the preventive measures fail, then validation, in accordance with good security practice, provides methods for detecting the attack in the H(e)NB. SAV further includes mechanisms for the network to discover that an attack on particular functionality of the H(e)NB has been detected, to make the decisions as to whether to block or allow network access, and to remediate specific functionality of the H(e)NB through existing software update procedures.

      For threats 2, 3, 4, 6, 7, 8, 9, 10, 12, 18, 19, 21, 23, 28, the present TR describes a counter-measure (#2) which is expressed in terms of functions which would actually be part of any validation method.

3. **Specify the "open interfaces" for full vendor interoperability. This is common in 3GPP and shall allow implementation of H(e)NBs and NEs independently, based on specification only.**

   The following interfaces are suggested for Semi-Autonomous Validation

(N.B. For further information on open interfaces, see the answers for ToR Question 4 below)

1. **What are the measurement values to be stored and transferred in a manner which is independent from H(e)NB architecture and implementation?**

   In SAV the H(e)NB performs the local device integrity verification in three stages as defined in section 7.5.2.4. In doing so, it compares the integrity check measurements for components with the corresponding expected trusted reference values. These components can be the firmware, operating system, RF firmware etc. The components are defined by the manufacturer of the system.

   In SAV it is not the list of stage 3 components that have failed integrity checks that is reported to the PVE. Rather, what is reported is a list of functionalities (of the H(e)NB) each of which can be impacted by failure of any of the components. The relationship between the components and functionalities is described below and is proprietary to the manufacturer of the H(e)NB. The list of functionalities may be standardized.

   Since the components are the quanta on which the integrity checks are performed, how to define the components can be decided by the manufacturers. A list of functionalities is associated with each component. When a component fails an integrity check, a list of functionality that is associated with the component(s) that failed the integrity check(s) can be constructed. The components are organized in the order of their integrity checking order.

Editor's Note: The feasibility of creating such a list of functionalities is FFS.

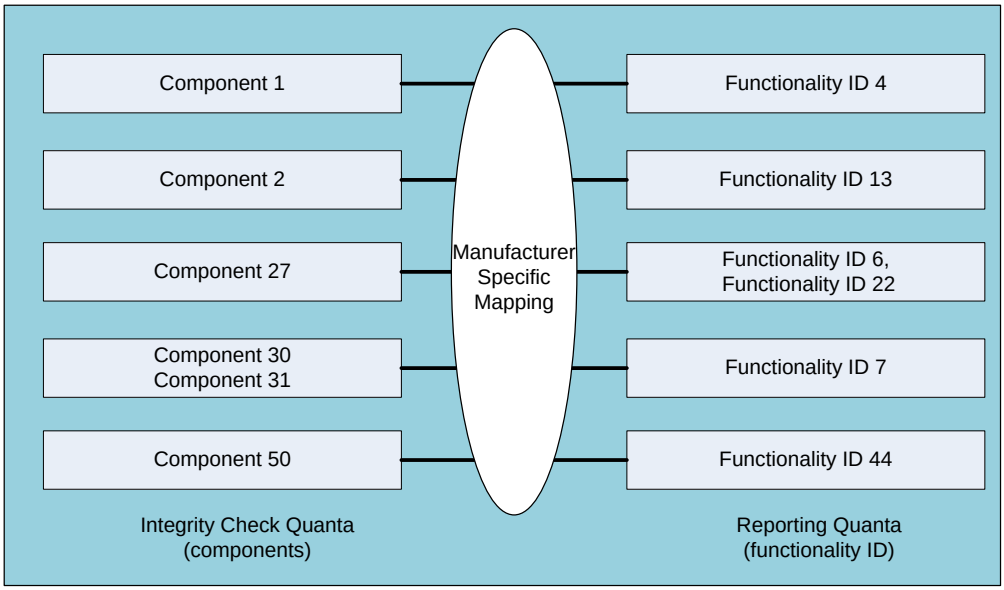


Figure 7.5.4.5-1 Components and Functionality

In SAV, following a stage 3 component check failure, a list of functionalities corresponding to the components that failed the integrity checks is created and reported to the PVE. The PVE, based on the list of reported functionalities, determines the validation status of the device and communicates any recommended action(s) to the SeGW and if applicable to the H(e)MS.

A stage 1 or 2 component check failure is not handled by the mechanisms of SAV.

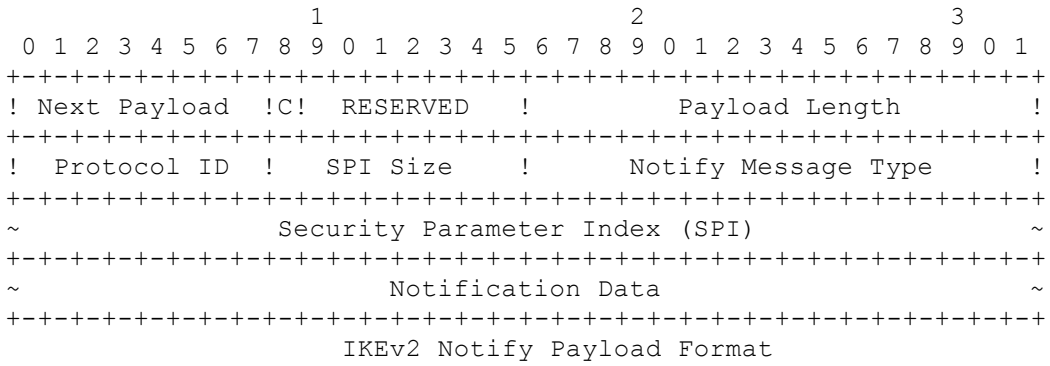
2. **What requirements apply to the transfer of information received from the H(e)NB as a result of validation (transport over existing channels, binding of validation and authentication, etc.)?**

   In SAV, the release of sensitive keys and functionality needed for device authentication is allowed only if the stage 1 components (for TrE) and the stage-2 components (for basic communication functionality) have successfully passed integrity checks and have been loaded and started.

   In addition to such conditional release of sensitive keys and sensitive functions needed for the authentication functionality, a further mechanism of binding between the validation and authentication is provided by the two processes using the same IKEv2 message exchanges:

- First, the device certificate used for device authentication and the list of the functionalities (obtained as result of the local integrity checking of components and mapping of these components to the functionalities) are sent to the SeGW in the IKEv2 IKE_AUTH_REQ message. Therefore, the code that composes the IK_AUTH_REQ message should itself have been successfully integrity checked, and moreover, should have the functionality to collect and put in the same message both the device certificate and the SAV functionality list. The SeGW extracts and forwards the list of failed functionalities to the PVE. The PVE decides the future actions and indicates the results (e.g. recommended actions) to the SeGW and in some cases to the H(e)MS.

- Secondly, the information sent from the H(e)NB in SAV is sent over the IKEv2 NOTIFY message field. A list of the failure functionalities is included in the NOTIFY payload in TLV (Type, Length, Value) format to accommodate variable length messages. For example, if during the integrity verification process, the H(e)NB determines four components have failed, the H(e)NB would determine the list of failed functionalities and the NOTIFY field would contain the codes indicating the number of functionality IDs that are impacted by failure of integrity checks for components and list of the functionality ID for those functionalities associated with the integrity-check-failed components. If there are no failed components then the field specifying the number of functionality IDs reported is set to NULL.



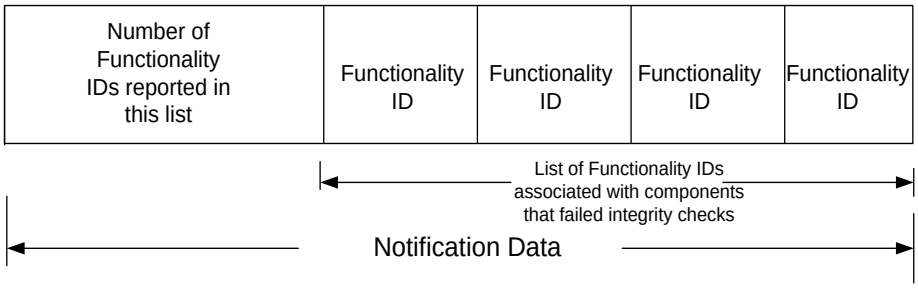


Figure 7.5.4.5-2. Example Format of SAV Notification Data for reporting of List of Functionality IDs in Semi-Autonomous Validation.

4. Specify the procedures and architectures in the network which are necessary for full vendor interoperability.

This study provides details of the SAV procedure and the interaction of various network entities. The network entities involved and their additional functions specific to SAV are:

1. PVE: a new network entity for H(e)NB device validation, and also for specifying the post-validation actions to be taken by the SeGW and (possibly) the H(e)MS.
2. SeGW: enforces post-validation access control as indicated by the PVE.
3. H(e)MS (possibly): Provides remediation to the H(e)NB as indicated by the PVE.

In the case where the H(e)NB connects to the H(e)MS via the SeGW over an IPSec tunnel, the SAV reporting mechanism (whereby the H(e)NB reports the list of failed functionality IDs to the SeGW) can use the interface that is already defined in the present document and specified in section 5 of TS 32.583. The IKEv2 protocol, which is used for authentication of the device, can be used to include all additional information required to report the list of failed functionality IDs.

There are some interfaces that may need to be standardized in order to support SAV in ways that facilitiate interoperability. The newly needed interfaces are those that connect the PVE to the SeGW and the H(e)MS.

The following diagram, adapted from the network architecture diagram in Figure 4.1.1-1 of 32.583 v8.1.0 shows a possible example of the network architecture that supports SAV, when the H(e)NB connects to the H(e)MS via a SeGW. The new interfaces needed in this case are:

1. I-pve-SeGW: Interface between the PVE and the SeGW:
2. I-pve-HTM: Interface between the PVE and the TR-069 Manager of the H(e)MS

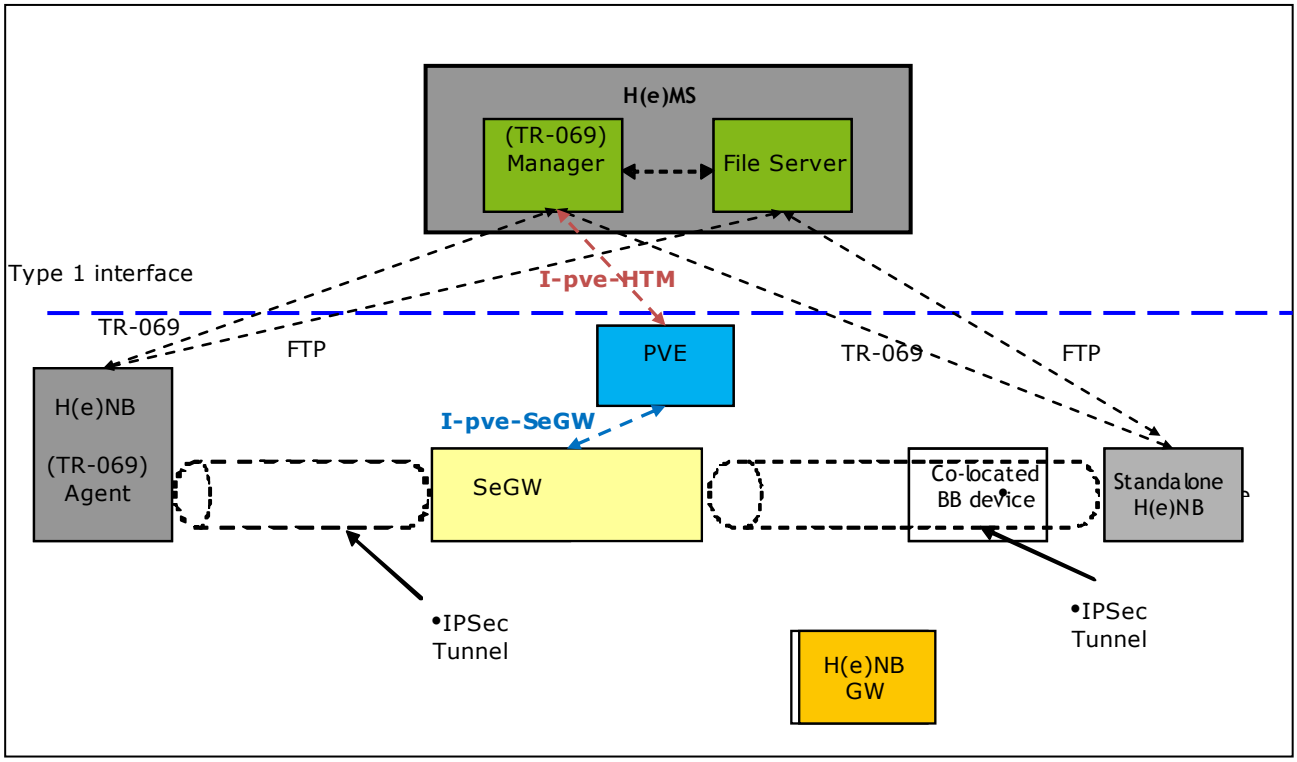


Figure 7.5.4.5-3. Possible Network Architecture for SAV and Interfaces for the PVE, when H(e)NB connects to the H(e)MS via SeGW.

1. I-pve-SeGW is needed:
    1. For the SeGW to contact the PVE and send the list of functionality IDs coresonding to the components that failed integrity check
    2. For the PVE to indicate access control actions to be taken by the SeGW based on the result of the validation.
2. I-pve-HTM is needed:
    1. For the PVE to request the H(e)MS TR069 manager to schedule remote software update based on the validation result

In the case where the H(e)NB connects to the H(e)MS over the public Internet, the procedure of SAV could be performed over transport-level protocols. The H(e)NB could send the list of functionality ID's corresponding to the failed components to the PVE via the H(e)MS, where the transport of the list of functionality IDs could be carried in a TR069 message (e.g. *Inform* message). Message formats such as proposed in Figure 2 could be used. The list of functionaliy IDs corresponding to the failed components can then be forwarded from the H(e)MS to the PVE, where validation assessment is made, and a recommended action could be sent from the PVE to the H(e)MS which could then enforce the actions. Post-validation access control action would be performed by the H(e)MS in this case.

The following diagram shows a possible example of the architecture for the network that supports SAVwhen the H(e)NB connects to the H(e)MS over the public Internet. The new interface needed in this case is:

1. I-pve-HTM: Interface between the PVE and the TR-069 Manager of the H(e)MS

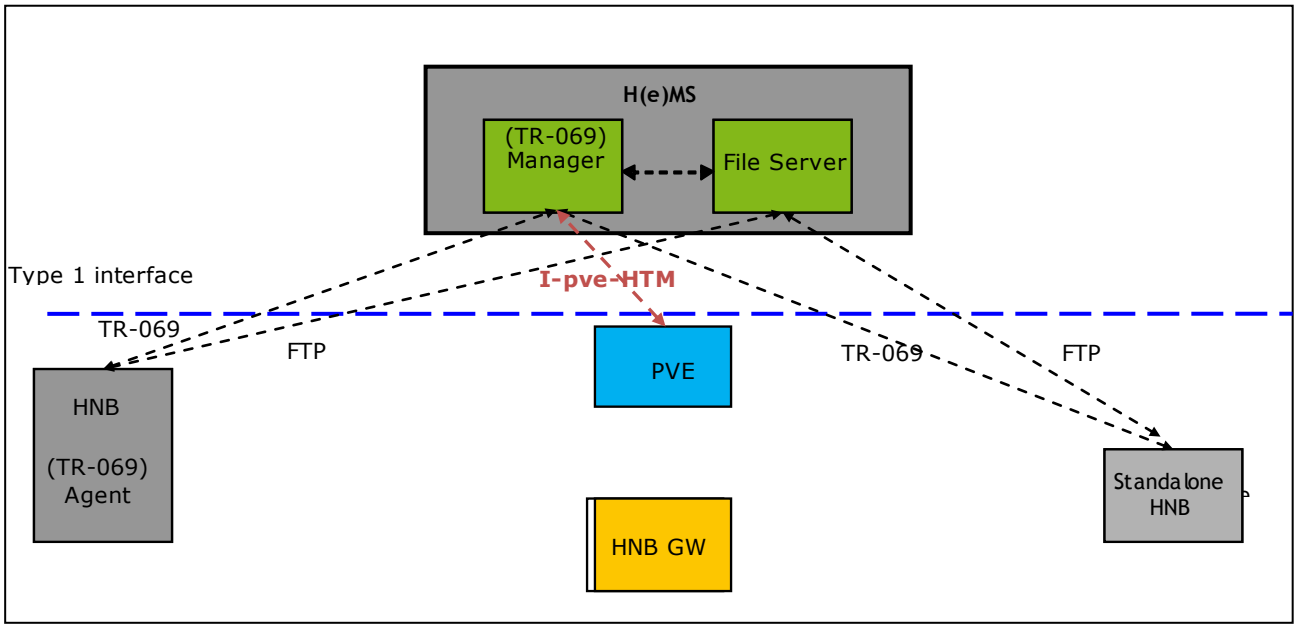


Figure 7.5.4.5-4. Possible Network Architecture for SAV and Interfaces for the PVE, when H(e)NB connects to the H(e)MS over public Internet

1. I-pve-HTM is needed:

    1. For the H(e)MS to send the PVE a list of Functionality IDs, corresponding to the failed components, that it receives from the H(e)NB (e.g. in a TR069 Inform message); and

    2. For the PVE to indicate actions to be undertaken by the H(e)MS, based on the result of PVE's validation of the H(e)NB, and

    3. For the PVE to request the H(e)MS TR069 manager to schedule a remote software update based on the validation result

1. **What are the possible reactions in SeGW or H(e)MS on this detailed information received from the H(e)NB as a result of validation in case of differences to the expected values?**

An implementation independent table comprising H(e)NB functionality IDs together with actions to be taken can be maintained by the PVE. The SeGW extracts the list of functionality IDs from the Notify payload field (in the case of SAV via SeGW) and passes them to the PVE. Alternatively, a suitable protocol message (e.g. TR069 Inform) can be used to convey the list of functionality IDs from the H(e)NB to the H(e)MS, which can then be forwarded from the H(e)MS to the PVE (in the case of SAV over public Internet). The PVE can then decide the subsequent actions to be taken by the SeGW or H(e)MS and can relay these decisions (and possibly also the validation results) to the respective entities.

If the authentication and validation are successful then the device is allowed full network access. If the the device integrity check fails for stage 1 or stage 2 code, which are pre-designated by the manufacturer and contains the code necessary for authentication and communication with SeGW, along with the code for the TrE, then the TrE is either not started (in the case of stage 1 failure) or it cannot complete device authentication (in the case of stage 2 failure).

However, if the device fails the integrity check for some functionalities not included in stage 1 or stage 2 code, then based on the PVE policy, the H(e)NB may be given partial access to the network, access only to the H(e)MS or it may be quarantined. The SeGW (or H(e)MS in the case where H(e)NB connects to it over public Internet) is instructed to perform access control. If a SW update is required then the H(e)MS is instructed to perform an update which may be scheduled for a later time or may be immediate.

In order to facilitiate interoperability regarding the reporting structure for SAV, the list of the functionalities that can be reported in a SAV message should be standardized. The mapping between components and their

corresponding functionalities can be left to manufacturer-specific implementation. The following table shows a possible list of functionalities, derived from 3GPP specifications TR 22.220, 23.830, 25.467, 25.468, 25.469, 25.820, 25.967, and 32.821, that could be included as part of the standardized list. The table also shows possible actions that can be taken by the H(e)NB, SeGW, and H(e)MS, upon indication of failure of the functionalities. The actions for the H(e)NB could be pre-configured in the H(e)NB and enforced on the device. The PVE indicates actions to the SeGW and the H(e)MS. This list may be expanded to include more as envisioned.

Editor's Note: The feasibility of creating such a list of functionalities is FFS

Table 7.5.4.5-1 Failed Functionalities and possible actions

Functionality ID	Failed Functionality Description	H(e)NB Action	SeGW Action	H(e)MS Action
1	H(e)MS subsystem	Partial Access Allowed	Allow complete access	Schedule SW Update
2	Uu interface	Partial Access Allowed	Allow complete access	Schedule SW Update
3	Iuh interface	Partial Access Allowed	Allow complete access	Schedule SW Update
4	Transport Address Mapping	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
5	QoS Management	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
6	UE Baseband System	Partial Access Allowed	Allow complete access	Schedule SW Update
7	UE Radio Frequency System	Partial Access Allowed	Allow complete access	Schedule SW Update
8	Local IP Access	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
9	UE Registration for HNB	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
10	UE Access control management	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
20	Managed Remote Access	Full Access Allowed	Allow complete access	Schedule SW Update
21	Charging	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
22	Emergency Services	Full Access Allowed	Allow complete access	Immediate SW Update
26	HNB support for legacy CN	Partial Access Allowed	Allow complete access	Schedule SW Update
27	Inbound Handover Support	Partial Access Allowed	Allow complete access	Schedule SW Update
28	Roaming	Partial Access Allowed	Allow complete access	Schedule SW Update
40	Time and Clock Management	Partial Access Allowed	Allow complete access	Schedule SW Update
41	CSG management	Access to H(e)MS only	Access to H(e)MS only	Immediate SW Update
42	Mobility Management	Partial Access Allowed	Allow complete access	Schedule SW Update
43	NAS Node selection function	Partial Access Allowed	Allow complete access	Schedule SW Update
44	Configuration Settings	Partial Access Allowed	Allow complete access	Schedule SW Update

2. **How is the expected set of measurement values determined by Validation Entity, e.g. dependent on vendor, HW type, and SW version?**

The validation entity is the PVE in the SAV. The PVE does not maintain measurements; instead it maintains the mapping between the functionalities and the various actions to be taken if those functionalities are reported in a SAV message, indicating their corresponding components have failed an integrity check. As noted previously, the full list of functionality IDs should be standardized.

Furthermore, in SAV, the H(e)NB maintains the list of trusted reference values corresponding to the measured components. Since the components are manufacturer specific, the manufacturer decides these component lists. However, to provide interoperability a common specification of the language of the functionality list which is portable and readable by both the device and the network entity is necessary. The mapping of components measured to the functionality ID is left to the manufacturer.

One aspect that could be addressed is to standardize the minimum level of acceptable security for the integrity measurement algorithm. For example, the trusted reference values could be required to be computed using SHA-1 or other algorithms offering a similar or higher level of security.

3. **Where do the reference values used by the Platform Validation Entity come from (push by vendor, pull by MNO, ...)? What is the needed from the infrastructure to support this? (Network elements, interfaces)?**

In SAV, the PVE does not maintain the trusted reference values, as mentioned in (2) above. Instead, the PVE needs to maintain a mapping of the (reported) functionality IDs to actions to be taken by SeGW and/or H(e)MS. If the PVE is operator controlled, how such mapping data is configured in the PVE is operator specific. If the PVE is a third-party entity, how such mapping should be configured will probably need involvement of that third party and the operator. In both cases, standardization of the configuration of the mappings is not needed.

4. **What are the relations to proposed H(e)NB S/W distribution methods and channels included in TR069 (e.g. for H(e)MS based update of H(e)NB SW)?**

In order to support regular software updates, TR069 can be used. The manufacturer's trusted reference values must also be included as part of the normal SW update procedure.

5. Describe remediation methods and their security implications.

Fine-grained remediation steps can be taken if any stage 3 component fails integrity check. The operator could attempt, for example, a targeted patching of software component(s) corresponding to the reported functionalities.

1. **What remediation methods (repairing, re-loading of SW in secure way, etc.) are possible on a suspected compromised device?**

Remediation is possible because when the network receives a SAV report of suspected compromised stage-3 functionality, it can trust that the TrE and basic communication functionality are intact and have loaded, and therefore can reload software in a secure way.

The benefit of SAV is that when a non-critical, stage-3 component has failed an integrity check, the functionality ID(s) corresponding to that component can be reported to the PVE. The network has the option of performing a targeted software update of the compromised component and its corresponding trusted reference value, using existing remote software update procedures.

2. **How validation reporting methods assist the remediation from (suspected) compromised state of H(e)NB?**

SAV reporting consists of a list of functionality IDs corresponding to the components that have failed integrity checking. This information assists the remediation process on a (suspected) compromised device in two ways:

(1) network access control for (suspected) compromised device

(2) identification of specific components for software update

SAV enables the network to allow the H(e)NB to have basic network access even when some of the functionality of the H(e)NB may not be operational. Therefore, SAV assists remediation by allowing the network operator to schedule a remote software update immediately or at a later time, based on e.g. the criticality of the components that need to be updated and/or the network's bandwidth, etc.

The operator can also identify the component(s) that need to be updated on the H(e)NB by receiving the reported functionality ID(s) from the H(e)NB and having the mapping from the functionality IDs to the components.

6. What is the trade-off between added security and cost / complexity (cost / benefit trade-off) between countermeasures and effort?

The security and other benefits conferred by SAV against a system that does not employ SAV but employs other non-validation countermeasures are described in the Threat Analysis section 7.5.3.3.4.

If no integrity checking of any kind was employed, then allowing security-sensitive functions to be stored, managed and executed outside of the hardware protected part of the H(e)NB would be a significant risk. Any compromise or failure of such functions would be very difficult to detect, diagnose and remediate. However, for the purpose of trade-off analysis given in Table 2 below, such a non-integrity-check H(e)NB system is used as the common baseline against which all validation methods could be compared.

SAV provides an additional security benefit of increased service availability, over other validation methods that include integrity checking and automatic disabling of authentication upon failure of any component, by enabling the network operator to make informed choices about allowing or blocking access based on the report it gets from the H(e)NB regarding the health of the various functionalities on the device.

The main added cost of SAV to the network operator is the cost of the PVE, but the complexity of the PVE can be minimal in both complexity and maintenance. In the simplest case, a PVE can be implemented as a stateless entity with a look up table function. The PVE can be merged with H(e)MS as a functionality, too. The added cost to the H(e)NB would be minimal, since any additional cost due to the slightly more complex functionality required of the TrE to perform SAV would likely be very minimal in both development cost and provisioning cost.

The overall operational cost of SAV is decreased considerably, if the reduced OPEX cost, due to increased visibility and control resulting in a reduced need to provide customer support, send personnel or perform manual maintenance, is greater than any incremental OPEX cost, due to the need to maintain the PVE and its functions.

Table 7.5.4.5-2. Cost Benefit Analysis of Semi Autonomous Validation

Entity	Cost	Benefits
H(e)NB system with TrE that is: 1) Integrity-checked by a RoT, and 2) Checks the integrity of other components of the H(e)NB	• Large decrease in maintenance and personnel costs, due to the reduced need to have onsite physical maintenance for some types of failure • Potentially large decrease in platform costs, since a H(e)NB system that uses a TrE backed up by SAV: 1. Does not need to be implemented in a large, closed platform 2. Does not need to execute all firmware within a large, closed environment to become trustworthy ↑ Small increase of complexity due to: 1. Implementation of integrity checking (done by the RoT to the TrE, and by the TrE to the rest of the H(e)NB), 2. The need to provision the Trusted Reference Values (TRV) on the device.	+ Fewer service calls and less frustration for hosting party + Users of UEs connected to H(e)NB experience less service interruptions + Provide increased network availability + Provide recovery mechanism for (suspected) compromised components + Ability to make the TrE a very small entity + Ability to easily change/upgrade software and still assure trusted operation + Ability to detect any unauthorized component modifications + Ability to report specific functionality IDs for those components which fail integrity check + Allow the H(e)NB to execute with basic functionality when non-critical components fail an integrity check

Network/PVE	↑ Small increase to support validation functionality (however can be combined as part of H(e)MS)	+ Reduces H(e)NB recalls and returns + Reduces volume of H(e)NB hosting party customer support calls + Reduces churn and improves sales + Protects network from access by compromised H(e)NBs by binding authentication to validation. N.B. this is also true of AuV + Assists network to make fine-grained decisions on access control for the H(e)NB + Alerts H(e)MS to perform targeted remote software or configuration update + Ability to grant H(e)NB network access with support for some basic functionality when non critical functional failures are reported + Ability to block H(e)NB network access until a SW update is complete + Ability to block H(e)NB network access until a configuration change is complete

7.5.4.6 Answers to Questions Concerning Hybrid Validation

The following investigations and clarifications are seen as necessary beyond the existing descriptions in TR 33.820:

2. Threat models /description of attacks and clean derivation of security features of validation from the threat model.

3. Threat analysis with explicit relation to the different validation methods:

   1. Which threats/attacks may be countered by autonomous validation?

      6 (booting H(e)NB with fraudulent software (“re-flashing”)

      8 (Physical tampering with H(e)NB

      16 (denial of service attacks against core network

      19 (mis-configuration of H(e)NB)

      20 (mis-configuration of ACL or compromised of ACL)

      21 (radio resource management tampering)

      22 (masquerade as a valid H(e)NB)

   2. Which additional threats/attacks identified in the TR may be countered by "explicit" (non-autonomous) validation, which are not caught by autonomous validation?

      TBD

   3. Are there (other) existing countermeasures available for the threats identified in 2.2., which do not rely on validation?

      None

4. Specify the "open interfaces" for full vendor interoperability. This is common in 3GPP and shall allow implementation of H(e)NBs and NEs independently, based on specification only.

    1. What are the measurement values to be stored and transferred in a manner which is independent from H(e)NB architecture and implementation?

        There are three types of reference measurement values that are used for Hybrid Validation:

        1. stored in the H(e)NB protected by hardware-based secure storage provided by TrE
        2. stored in the H(e)NB protected by software-based secure storage provided by TrE
        3. stored in Platform Validation Entity

        The reference metrics held in the PVE are provided by the H(e)NB vendor and the means of transferring is to be determined by PVE vendor and H(e)NB vendor. These reference metrics are per component and each value is 20 bytes.

    2. What requirements apply to the transfer of information received from the H(e)NB as a result of validation (transport over existing channels, binding of validation and authentication, etc.)?

        Th Notify Payload within IKEv2 shall be used to provide transfer of information from H(e)NB validation. It is explicitly bound to the authentication of H(e)NB via IKEv2. If the validation fails, the IKEv2 shall terminate in error. This applies after AuV succeeds and errors are reported.

5. Specify the procedures and architectures in the network which are necessary for full vendor interoperability.

    1. What are the possible reactions in SeGW or H(e)MS on these detailed measurement values in case of differences to the expected values?

        The PVE, which can be co-located with SeGW or H(e)MS, will inform SeGW or H(e)MS of the error conditions in case the measurement values are different than the expected values that are stored in the PVE.,

    2. How is the expected set of measurement values determined by Validation Entity, e.g. dependent on vendor, HW type, and SW version?

        The expected set of measurement values is given by the vendor (HW vendor, SW vendor, third party vendor) when the components are installed. The initial values are provisioned in the PVE before the H(e)NB powers on. Subsequent expected set of measurement values are given when the component is updated or upgraded.

    3. Where do the reference values used by the Platform Validation Entity come from (push by vendor, pull by MNO, ...)? What is the needed from the infrastructure to support this? (Network elements, interfaces)?

        The reference values used by PVE are given by vendors (HW, SW, FW, and/or third party) when the components are installed and/or upgraded. There are no additional network elements, interfaces when the PVE is co-located with H(e)MS or SeGW.

    4. What are the relations to H(e)NB S/W distribution methods and channels included in TR069 (e.g. for interface needed to SeGW H(e)MS based update of H(e)NB SW)?

        If PVE is co-located in H(e)MS, the existing infrastructure based on TR069 can be fully re-utilized. Interface to SeGW may need to be extended to support TR069.

6. Describe remediation methods and their security implications.

    1. What remediation methods (repairing, re-loading of SW in secure way, etc.) are possible on a suspected compromised device?

        Remediation is done based on the nature of the compromise. In case of severe SW or HW compromise, repairing H(e)NB may require sending the device to repair facility authorized by operator and/or HW vendor.

    2. How validation reporting methods assist the remediation from (suspected) compromised state of H(e)NB?

        Validation reporting and/or validation of components by network provides explicit evidence of H(e)NB validation state. It provideds exact details and nature of the compromise of the individual component(s) within H(e)NB.

7. What is the trade-off between added security and cost / complexity (cost / benefit trade-off) between countermeasures and effort?

   Additional threats and attacks not addressed by the Autonomous Validation can be easily addressed. In addition to knowing implicitly the core components that are validated as a process of secure startup, the core network also has the ability to validate addition components and/or configurations that are not validated during the secure startup and has the ability to know the exact state of all of the components explicitly.

   Notify Payload is already proposed to be used for carrying the validation data between H(e)NB and SeGW, which can be extended to PVE. Since IKEv2 is used and it supports the Notify Payload, there is no additional complexity on the IKEv2 exchange. The SeGW needs to be extended to process the additional Notify Payload. In case PVE is co-located in SeGW, there is no additional complexity in terms of interface. When PVE is co-located in H(e)MS, the existing interface and security between SeGW and H(e)MS is fully re-utilized. In case PVE is standalone (not recommended), there is additional interface between PVE and SeGW and between PVE and H(e)MS. Interface to SeGW may need to be extended to support TR069.

7.6 Authentication Implementation Options

The following described the various implementation options that can be used for authentication options. Some generic mechanisms maybe considered as option for either device authentication or host party authentication, such as the option described below.

7.6.1 Generic Authentication

7.6.1.1 General

This section describes mechanisms to be used for the authentication principles as described in section 7.1.

Editor's Note: The term AKA credential used below may undergo revision by SA3 if seen as necessary.

7.6.1.2 EAP-AKA-based Client Authentication

7.6.1.2.1 General

This solution may be used for device authentication (step a1 according to section 7.1) or for hosting party authentication (step b1 according to section 7.1).

The H(e)NB is provided with an appropriate AKA credential enabling to use EAP-AKA, e.g. within IKEv2 for authentication and set-up of IPsec security associations between the SeGW and the H(e)NB. The SeGW is authenticated by the H(e)NB with the SeGW certificate during the IKEv2 protocol run. Afterwards the SeGW is acting as EAP authenticator and forwards the EAP protocol messages to the AAA server, which retrieves an authentication vector from AuC via HSS/HLR. By completing the EAP-AKA authentication successfully, H(e)NB and core network (via AuC) are authenticated mutually.

NOTE 1: For this authentication concept it is possible that the appropriate AKA credentials could be stored in a removable or irremovable Trusted Environment (TrE). However, a removable TrE if used for storage of device authentication credentials does not by itself lead to the authentication of the H(e)NB device. Consequently, any, possibly illegitimate or compromised, device would be able to access the operator's IP network with a valid AKA credential, unless additional measures are taken (see 7.3).

NOTE 2: Depending on operator needs, existing HLR/HSS element and interfaces may be used for this purpose. The allocation of IMSI ranges and possible restrictions for these IMSIs in HLR attributes are out of the scope of this technical report.

7.6.1.2.2 Assumptions at H(e)NB

Appropriate AKA credentials must be provided to the H(e)NB.

If EAP-AKA is used for device authentication, then the related credential has to be provided to the Trusted Environment (TrE) of the H(e)NB. If used for this purpose, to allow mutual authentication as required for device authentication, either the inherent properties of AKA to also authenticate the home network may be used, or for the authentication of the SeGW, the root certificate of the operator should be installed at the H(e)NB.

If EAP-AKA is used for authentication of the hosting party, the related credential is stored on a Host Party Module (HPM).

7.6.1.2.3 Assumptions for Storage of AKA Credential

For device authentication, the appropriate AKA credential is stored and the related application executed in a trusted environment, called Trusted Environment (TrE). A definition of Trusted Environment is given in Section 7.2.2. which should be irremovable.

For the hosting party authentication, the appropriate AKA credential is stored and the related application executed in a secure environment, called hosting party module (HPM). A definition of Hosting Party Module is given in Section 7.2.1.

7.6.1.2.4 Assumptions in Core Network

The SeGW acts as EAP authenticator and relays authentication information to the AAA server. The AAA server retrieves an authentication vector from AuC via HSS/HLR.

The HSS/HLR contains an entry for the device and/or hosting party. The HSS/HLR is able to distinguish between authorizations of AKA credentials associated with UEs, associated with H(e)NB devices and/or associated with H(e)NB hosting parties, e.g. by subscription profile data.

7.6.1.2.5 Authentication Flow

EAP-AKA is run within IKEv2 between H(e)NB and SeGW for mutual authentication of H(e)NB and core network or for authentication of the hosting party.

7.6.1.2.6 Impacts on Core Network

A AAA server is required as modified network element.
The authentications of the H(e)NBs generate additional processing load, and network load for HLR/ HSS.

Additional storage capacity is required in HLR/HSS for the H(e)NB or hosting party entries.

7.6.1.2.7 Authentication Identifier

An identifier is needed for the authentication protocol to indicate the identity of the H(e)NB device or the hosting party, e.g. in the identification information elements of the protocol used. Also the access control in SeGW is based on this identifier. The identifiers used for AKA-based authentication should be globally unique.

The identifier for AKA-based authentication is the IMSI of the AKA credential. For this purpose, these IMSIs have to be marked in HLR/HSS as used for H(e)NBs, e.g. by allocating dedicated ranges or by adding specific attributes.

NOTE: The implementation of the related HLR/HSS entry is out of scope of this document.

7.6.1.3 Certificate-based Client Authentication

7.6.1.3.1 General

Authentication is based on device certificate for H(e)NB and network certificate for the core.

The H(e)NB authenticates with the built-in device certificate to the SeGW. For this purpose, the SeGW verifies the H(e)NB device certificate. In order to enforce the access control, the verified device identity is looked up in a whitelist maintained by the H(e)NB device identity server. The whitelist is a positive list which collects the device identities of those H(e)NB devices that are allowed by the operator to be connected to the core network due to valid contracts.

The SeGW is authenticated by the H(e)NB based on server certificate. This is no different from SeGW authentication used together with EAP-AKA based device authentication.

NOTE 1: This section focuses on authentication and does not consider access control.

NOTE 2: This section only describes usage of certificate-based authentication to device authentication, as the currently known use cases propose EAP-AKA for the authentication of a hosting party. In principle, also the application of certificate-based authentication to hosting party authentication is possible.

NOTE 3: It is out of scope of this technical report if also variants with the initial enrollment based on vendor certificates and the further authentications based on operator certificates might be possible.

7.6.1.3.2 Assumptions at H(e)NB

The H(e)NB is provisioned with a device certificate and the associated private key generated by the vendor. This device certificate allows the authentication of the H(e)NB by the SeGW (and thus the operator network).

The credential (private key) must be stored in a TrE.

NOTE: It is out of scope of this technical report to define how a list of trusted root certificates or cross-certification by the vendor CA is used to authenticate the SeGW.

7.6.1.3.3 Assumptions in Core Network

A H(e)NB device identity server is available in the core network as network element providing additional functionality. This server manages a whitelist holding the information about valid device identities of H(e)NBs.
The SeGW must be provided with an appropriate certificate for H(e)NB device certificate validation.

NOTE: The H(e)NB identity server is not necessarily implemented as a physical server, but may be co-located with other functions.

7.6.1.3.4 Authentication Flow

IKEv2 with certificates used for authentication may be run between H(e)NB and SeGW to mutually authenticate the H(e)NB and the SeGW. This allows also the binding of Hosting party authentication according to step b1 of section 7.1.

In use cases which only deploy device authentication (bundled authentication of hosting party according to step b2 of section 7.1, or no hosting party authentication at all), also other certificate-based authentication protocols, e.g. TLS with mutual authentication, may be used.

7.6.1.3.5 Impacts on Core Network

A H(e)NB device identity server is required in the operator's network. The H(e)NB authentications do not affect the HLR/HSS signalling.

7.6.1.3.6 Certificate Management

The certificate management has to cover cases of authorized changes of H(e)NB owner or operator. This includes:

- Private sale of H(e)NB without involvement of vendor or retailer
- Change of operator

NOTE 1: It is out of scope of this technical report how to handle certificate management for authorized changes.

The certificate management also has to handle compromise of certificates.

NOTE 2: No revocation could be chosen, if the trade-off between loss caused by compromise of certificates and CAs and additional cost for revocation methods suggests this. Note, that for this case still the whitelists mentioned above allow the disabling of single H(e)NBs.

NOTE 3: It is out of scope of this technical report to decide if revocation is needed and, in case revocation is needed, how it is handled.

The certificate management has to cover certificate lifetime since the expected lifetime of a H(e)NB may be longer than the validity periods usually chosen for certificates.

NOTE 4: It is out of scope of this technical report how long the expiry times of certificates may be and how to handle expired certificates, if expiry may be expected.

7.6.1.3.7 Authentication Identifier

An identifier is needed for the authentication protocol to indicate the identity of the H(e)NB device, e.g. in the identification information elements of the protocol used. Also the access control in SeGW is based on this identifier. This identifier must also appear in an attribute of the related certificate.

If certificate based authentication is used for H(e)NB device authentication, then global uniqueness of the device identifier used for authentication is required. In general, the identifier may be any name which can be inserted in an appropriate attribute of the related certificate. In alignment with TS33.310 [12], clauses 6.1.3b and 6.1.3 and RFC4945 [17] section 5.1.3.6, this name should be a Fully Qualified Domain Name (FQDN), as it is to be used in the H(e)NB device certificate. If the device identifier is provisioned by the manufacturer of the device, then the device identifier should be composed of a globally unique manufacturer identity and an identity local to the manufacturer, e.g. a serial number. This provides at the same time global uniqueness and the freedom of the manufacturer to assign identifiers locally,

The definition of the exact authentication identifier format is left to other documents.

NOTE 1: A globally unique, FQDN formatted identifier would be appropriate for H(e)NB identity, allowing the vendor to use different solutions. The definition of the exact structure of the FQDN is considered to be outside the scope of this document. .

NOTE 2: It is desirable to be able to use the same device identity for both IPSec and TLS, but details are out of scope for this study.

7.6.2 Device Authentication

EAP-AKA and certificate both can be used for device authentication.

7.6.2.1 General

A pre-requisite assumption here is that H(e)NB validation and device authentication should be performed sequentially. At power up, the H(e)NB validation should precede device authentication.

It is also assumed that the H(e)NB's TrE should compute within itself the RES and AUTH parameters needed for the IKE_AUTH request messages that the H(e)NB needs to send to the SeGW during the device authentication procedures.

If certificate of the SeGW needs to be verified by the H(e)NB, the computations required for such verification should use the SeGW's and the certification authorities' public key(s) for root certificate(s), as well as any other data required for such verification. To prevent manipulation of public key material or any other required data on the H(e)NB, by which a malicious party could be enabled to impersonate an SeGW, the public keys and their certificates, as well as any other required data, such as certificate revocation lists, need to be protected both when they are provisioned to the H(e)NB and during the H(e)NB's operational lifetime. Therefore, such keys and other data should be stored in the H(e)NB's TrE. Further, the cryptographic operations using them should preferably be performed entirely within the H(e)NB's TrE.

7.6.2.2 EAP-AKA based

This authentication, which represents step a as described in section 7.1, is based on EAP-AKA for H(e)NB. EAP-AKA is run within IKEv2 between H(e)NB and SeGW for mutual authentication of H(e)NB and core network. The IKEv2 EAP-AKA authentication will follow the 3GPP TS 33.234 specification [10].

In order to bind the AKA credential to the device identity, which is essential for device authentication, the AKA credential has to be provisioned in the Trusted Environment (TrE) of the H(e)NB.

The call-flow on Figure 5. shows the EAP-AKA based mutual authentication between the H(e)NB, the SeGW, and the core network.

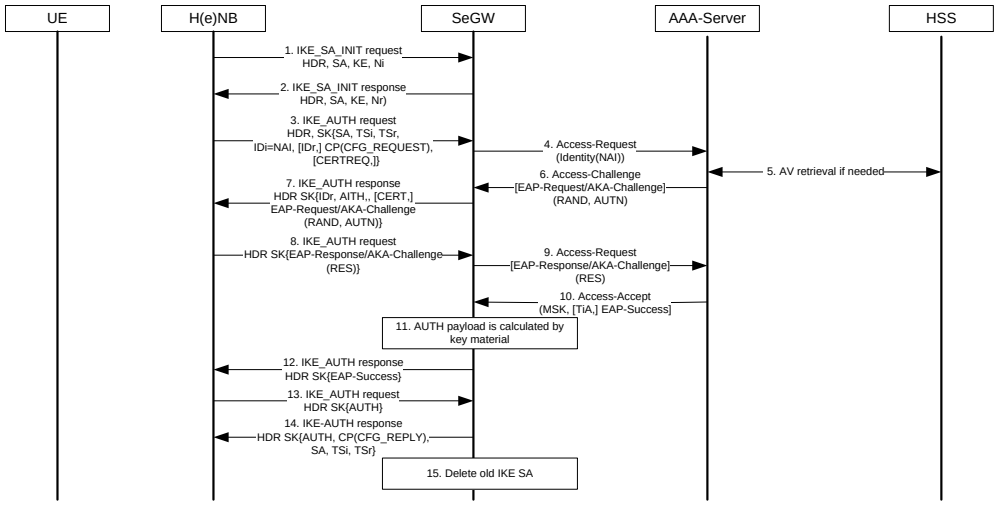


Figure 5: EAP-AKA-based authentication

1. The H(e)NB sends an IKE_SA_INIT request to the SeGW.

2. The SeGW sends IKE_SA_INIT response.

3. The H(e)NB sends its identity in the IDi payload in this first message of the IKE_AUTH phase, and begins negotiation of child security associations. AUTH is omitted to inform the SeGW that the H(e)NB want to perform EAP authentication. Configuration payload is carried in this message if the H(e)NB's remote IP address needs to be configured dynamically. The H(e)NB also requests a certificate from the SeGW.

   NOTE: The user profile selected by NAI presented in the IDi payload enforces the choice of authentication (certificate, EAP-AKA, or certificate-EPA-AKA multiple authentication).

4. The SeGW sends the Authentication Request message with an empty EAP AVP to the 3GPP AAA Server, containing the identity received in IKE_AUTH message sent in step 2.

5. If necessary, the AAA Server should fetch the device profile and authentication vectors from HSS/HLR.

   NOTE: It is out of scope of this technical report to decide whether the appropriate platform for device authentication is an HSS/HLR or a different authenticating entity is needed to be defined is FFS.

6. The AAA Server initiates the authentication challenge.

7. The SeGW send IKE_AUTH response to H(e)NB. The EAP message received from the AAA Server (EAP-Request/AKA-Challenge) is included in order to start the EAP procedure over IKEv2. The SeGW's identity, a certificate, and the AUTH parameter which is used to protect the previous message it sent to the H(e)NB (in the IKE_SA_INIT exchange) are included in this message in case that the H(e)NB need to authenticate the SeGW based on the certificate of the SeGW.

8. The H(e)NB responds to the authentication challenge. The only payload (apart from the header) in the IKEv2 message is the EAP message. The H(e)NB checks the authentication parameters in case that the H(e)NB need to authenticate the SeGW based on the certificate of the SeGW. Computation of the EAP-AKA RES parameter should be performed within the H(e)NB's TrE. The SeGW's root signing public key and any other data required for such verification should be stored in the TrE. To enhance security, the verification could be performed entirely in the TrE

9. The SeGW forwards the EAP-Response/AKA-Challenge message to the AAA Server.

10. When all checks are successful, the AAA Server sends the Authentication Answer including an EAP success and the key material to the SeGW. This key material should consist of the MSK generated during the authentication process.

11. The MSK should be used by the SeGW to generate the AUTH parameters in order to authenticate the IKE_SA_INIT phase messages.

12. The EAP Success message is forwarded to the H(e)NB over IKEv2.

13. The H(e)NB should take its own copy of the MSK as input to generate the AUTH parameter to authenticate the first IKE_SA_INIT message. The H(e)NB then forwards the AUTH parameter is sent to the SeGW. Verification of the EAP-AKA parameter AUTH should be performed within the H(e)NB's TrE.

14. The SeGW checks the correctness of the AUTH received from the H(e)NB and calculates the AUTH parameter which authenticates the second IKE_SA_INIT message. The SeGW should send the assigned Remote IP address in the configuration payload (CFG_REPLY), if the H(e)NB requested for a Remote IP address through the CFG_REQUEST. Then the AUTH parameter is sent to the H(e)NB together with the configuration payload, security associations and the rest of the IKEv2 parameters and the IKEv2 negotiation terminates.

15. If the SeGW detects that an old IKE SA for that H(e)NB already exists, it will delete the IKE SA and send the H(e)NB an INFORMATIONAL exchange with a Delete payload in order to delete the old IKE SA in H(e)NB.

7.6.2.3 Certificate-based

This authentication, which represents step a as described in section 7.1, is based on device certificate for H(e)NB and network certificate for the core.

IKEv2 with certificates used for authentication is run between H(e)NB and SeGW to mutually authenticate the H(e)NB and the SeGW.

The IKEv2 certificate-based mutual authentication is executed according to IETF RFC-4306 [11]. The certificate handling and profiles will adhere to 3GPP TS 33.310 [12], except that as noted above in clause 7.6.1.3.7, the device identifier should be a FQDN. For this reason, the subjectAltName field should contain an FQDN and not an IP address, even if no DNS is available.

NOTE: The precise mechanisms which are to be used for verifying the validity of SeGW and H(e)NB certificates (e.g. CRL, OCSP, whitelist) are out of scope of this technical report.

Certificate-based authentication does not require additional measures to ensure the binding between the device identity and the authentication credential. The credential (private key) must be stored in a TrE.

The call-flow on Figure 6. shows the simple certificate based mutual authentication between the H(e)NB and the SeGW.

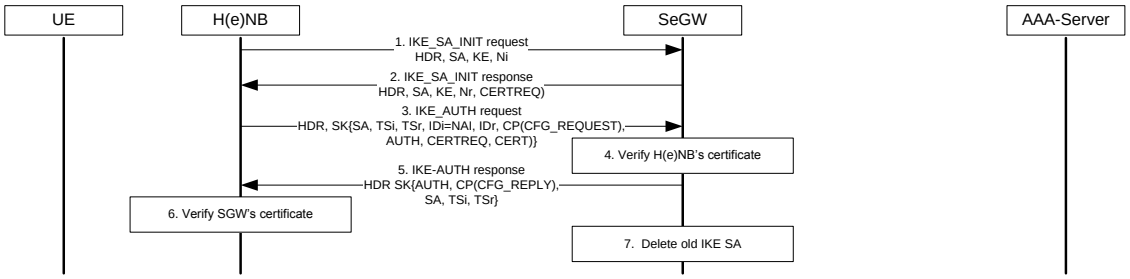


Figure 6: Certificate-based authentication

1. The H(e)NB sends an IKE_SA_INIT request to the SeGW.

2. The SeGW sends IKE_SA_INIT response, requesting a certificate from the H(e)NB.

3. The H(e)NB sends its identity in the IDi payload in this first message of the IKE_AUTH phase, and begins negotiation of child security associations. The H(e)NB sends the AUTH payload, its own certificate, and also requests a certificate from the SeGW. Configuration payload is carried in this message if the H(e)NB's remote IP address needs to be configured dynamically. Computation of the AUTH parameter is performed within the H(e)NB's TrE.

NOTE: The user profile selected by NAI presented in the IDi payload enforces the choice of authentication (certificate, EAP-AKA, or combined authentication).

4. The SeGW checks the correctness of the AUTH received from the H(e)NB and calculates the AUTH parameter which authenticates the second IKE_SA_INIT message. The SeGW verifies the certificate received from the H(e)NB.

NOTE: The mechanism for the SeGW to verify access authorization (e.g. checking against a whitelist) is FFS.

5. The SeGW sends the AUTH parameter and its certificate to the H(e)NB together with the configuration payload, security associations, and the rest of the IKEv2 parameters and the IKEv2 negotiation terminates. The Remote IP address is assigned in the configuration payload (CFG_REPLY), if the H(e)NB requested for a Remote IP address through the CFG_REQUEST.

6. The H(e)NB verifies the SeGW's certificate with its stored root certificate. The SeGW's root signing public key and any other data required for such verification should be stored in the TrE. To enhance security, the verification could be performed entirely in the TrE

NOTE: It is optional, but not required for the H(e)NB to further validate the SeGW's certificates, as it is under control of the operator.

7. If the SeGW detects that an old IKE SA for that H(e)NB already exists, it will delete the IKE SA and send the H(e)NB an INFORMATIONAL exchange with a Delete payload in order to delete the old IKE SA in H(e)NB.

7.6.3 Hosting Party Authentication

7.6.3.1 Bundled with the Device Authentication

The authentication of the hosting party is bundled with the device authentication, i.e. there is no additional authentication step after the device authentication. This authentication represents step b2 as described in section 7.1.

EAP-AKA and certificate both can be used when the authentication of the hosting party is bundled with the device authentication. The authentication flow of the bundled authentication is similar to the device authentication.

7.6.3.2 Stand-alone Hosting Party Authentication

7.6.3.2.1 Device Authentication based on Certificate and Hosting Party Authentication based on EAP-AKA

Device Authentication may optionally be followed with an EPA-AKA-based Hosting Party Authentication exchange. This authentication represents step b1 as described in section 7.1. The IKEv2 certificate-based mutual authentication is executed according to IETF RFC-4306 [11], followed by IKEv2's multiple authentication procedure defined in IETF RFC4739 [13]. The certificate handling and profiles will adhere to 3GPP TS 33.310 [12], although certificate enrolment and certificate revocation are not required. The IKEv2 EAP-AKA authentication will follow the 3GPP TS 33.234 [10] specification.

As in the case of device authentication, a pre-requisite assumption is that H(e)NB validation and device authentication should be performed sequentially. At power up, the H(e)NB validation should precede device authentication.

It is also assumed that the H(e)NB's TrE should perform within itself all computation of the AUTH required for certificate based device authentication, and the HPM is responsible for computing the RES and AUTH parameters for the EAP-AKA based hosting party authentication.

The call-flow on Figure 7. shows the certificate based mutual authentication between the H(e)NB and the SeGW, followed by an EAP-AKA auth exchange between the H(e)NB/HPM and the AAA server.

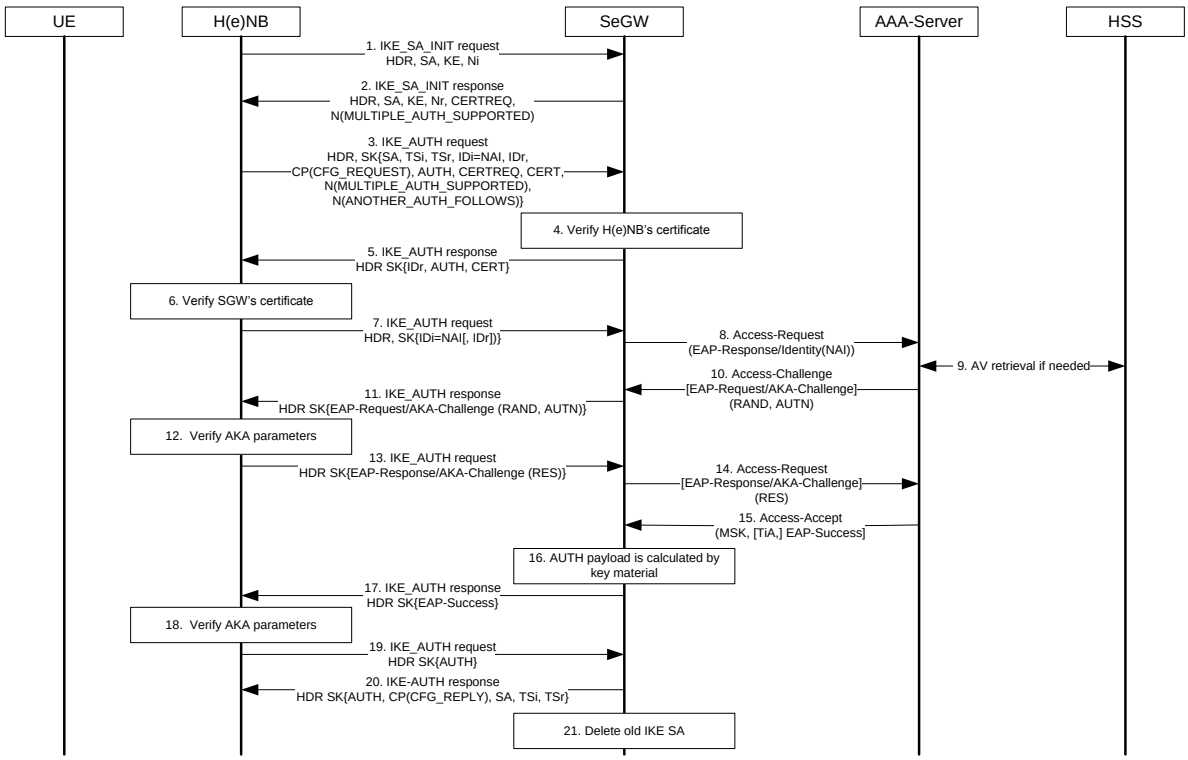


Figure 7: Combined certificate and EAP-AKA-based authentication

1. The H(e)NB sends an IKE_SA_INIT request to the SeGW.

2. The SeGW sends IKE_SA_INIT response, requesting a certificate from the H(e)NB. The SeGW indicates that it support Multiple Authentication by including the MULTIPLE_AUTH_SUPPORTED payload.

3. The H(e)NB inserts its identity in the IDi payload in this first message of the IKE_AUTH phase, computes the AUTH parameter within its TrE, and begins negotiation of child security associations. The H(e)NB then sends the AUTH payload, its own certificate, and also requests a certificate from the SeGW. Configuration payload is carried in this message if the H(e)NB's remote IP address needs to be configured dynamically. The H(e)NB indicates that it support Multiple Authentication and that it wants to do a second authentication by including the MULTIPLE_AUTH_SUPPORTED and ANOTHER_AUTH_FOLLOWS attributes.

NOTE: The user profile selected by NAI presented in the IDi payload enforces the choice of authentication (certificate, EAP-AKA, or combined authentication).

4. The SeGW checks the correctness of the AUTH received from the H(e)NB and calculates the AUTH parameter which authenticates the second IKE_SA_INIT message. The SeGW verifies the certificate received from the H(e)NB.

NOTE: The mechanism for the SeGW to verify access authorization (e.g. checking against a whitelist) is FFS.

5. The SeGW sends the AUTH parameter and its certificate to the H(e)NB.

6. The H(e)NB verifies the SeGW's certificate with its stored root certificate. The SeGW's and the root CA's public key and certificates, as well as any other data required for such verification should be stored in the TrE. To enhance security, the verification could be performed entirely in the TrE

NOTE: It is optional, but not required for the H(e)NB to further validate the SeGW's certificates, as it is under control of the operator.

7. The H(e)NB sends another IKE_AUTH message with the AUTH payload omitted to inform the SeGW that the H(e)NB want to perform EAP authentication.

8. The SeGW sends the Authentication Request message with an empty EAP AVP to the 3GPP AAA Server, containing the identity received in IKE_AUTH message sent in step 2.

9. The AAA Server should fetch the user profile and authentication vectors from HSS/HLR.
10. The AAA Server initiates the authentication challenge.
11. The SeGW send IKE_AUTH response to H(e)NB. The EAP message received from the AAA Server (EAP-Request/AKA-Challenge) is included in order to start the EAP procedure over IKEv2. The SeGW's identity, its certificate, and the AUTH parameter which is used to protect the previous message it sent to the H(e)NB (in the IKE_SA_INIT exchange) are included in this message.
12. The H(e)NB checks the AUTH authentication parameters in case that the H(e)NB need to authenticate the SeGW based on the certificate of the SeGW. The H(e)NB processes the EAP challenge message and uses the HPM for verification of the AUTN and generating the RES parameters. Optionally, processing of the whole EAP challenge message, including verification of the received MAC with the newly derived keying material is performed within the H(e)NB's HPM as specified in ETSI TS 102.310 [14].
13. The H(e)NB sends the IKE_AUTH response with the AKA-Challenge to the SeGW.
14. The SeGW forwards the EAP-Response/AKA-Challenge message to the AAA Server.
15. When all checks are successful, the AAA Server sends the Authentication Answer including an EAP success and the key material to the SeGW. This key material should consist of the MSK generated during the authentication process.
16. The MSK should be used by the SeGW to generate the AUTH parameters in order to authenticate the IKE_SA_INIT phase messages.
17. The EAP Success message is forwarded to the H(e)NB over IKEv2.
18. The H(e)NB should take its own copy of the MSK as input to generate the AUTH parameter to authenticate the first IKE_SA_INIT message. Computation of the AUTH parameters performed within the H(e)NB's HPM.
19. The AUTH parameter is sent to the SeGW.
20. The SeGW checks the correctness of the AUTH received from the H(e)NB and calculates the AUTH parameter which authenticates the second IKE_SA_INIT message. The SeGW should send the assigned Remote IP address in the configuration payload (CFG_REPLY), if the H(e)NB requested for a Remote IP address through the CFG_REQUEST. Then the AUTH parameter is sent to the H(e)NB together with the configuration payload, security associations and the rest of the IKEv2 parameters and the IKEv2 negotiation terminates.
21. If the SeGW detects that an old IKE SA for that H(e)NB already exists, it will delete the IKE SA and send the H(e)NB an INFORMATIONAL exchange with a Delete payload in order to delete the old IKE SA in H(e)NB.

The basic IKEv2 multiple and EAP-AKA authentication will conform to 3GPP TS 33.234 [10] and IETF RFC-4739 [13].

Editor's Note: It is ffs how certificate used will affect the architecture.

7.6.3.2.2 Binding of HPM ID and Device ID

The authentication system comprises the following entities:

H(e)NB, the equipment of home node B with a HPM inserted in. Every equipment has a unique EI (Equipment Identity) representing itself. The H(e)NB_EI is assigned by manufacturer. The H(e)NB_EI is stored securely in the TrE of the H(e)NB.

SeGW, Security Gateway, representing operator's core network to perform mutual authentication with H(e)NB.

HLR/AAA server, Home Location Register for H(e)NB, including Authentication Center . Also, HLR stores the records of H(e)NB_EIs corresponding every HPM_ID, presenting the binding relationship of the H(e)NB_EI and the HPM_ID .AAA server performs binding authentication based on the records.

SeGW forwards the EI of H(e)NB received from this H(e)NB itself to the HLR/AAA server. The HLR/AAA server compares it with the record. If they're the same, then it can be ascertained that the H(e)NB is the legitimate equipment binding to the HPM.

For the trustworthiness of the authentication assertions conveyed by the protocols described below, it is of paramount importance that all sensitive data remain protected by the TrE on the H(e)NB and the HPM. This means in particular that authentication secrets of the H(e)NB, representing the binding authentication of the H(e)NB, and the H(e)NB_EI should be securely stored in the TrE. Furthermore, the Hosting Party Identity and corresponding authentication secrets should also be securely stored in and processed only by the HPM. Secure channels should be used to transport all these data to the SeGW.

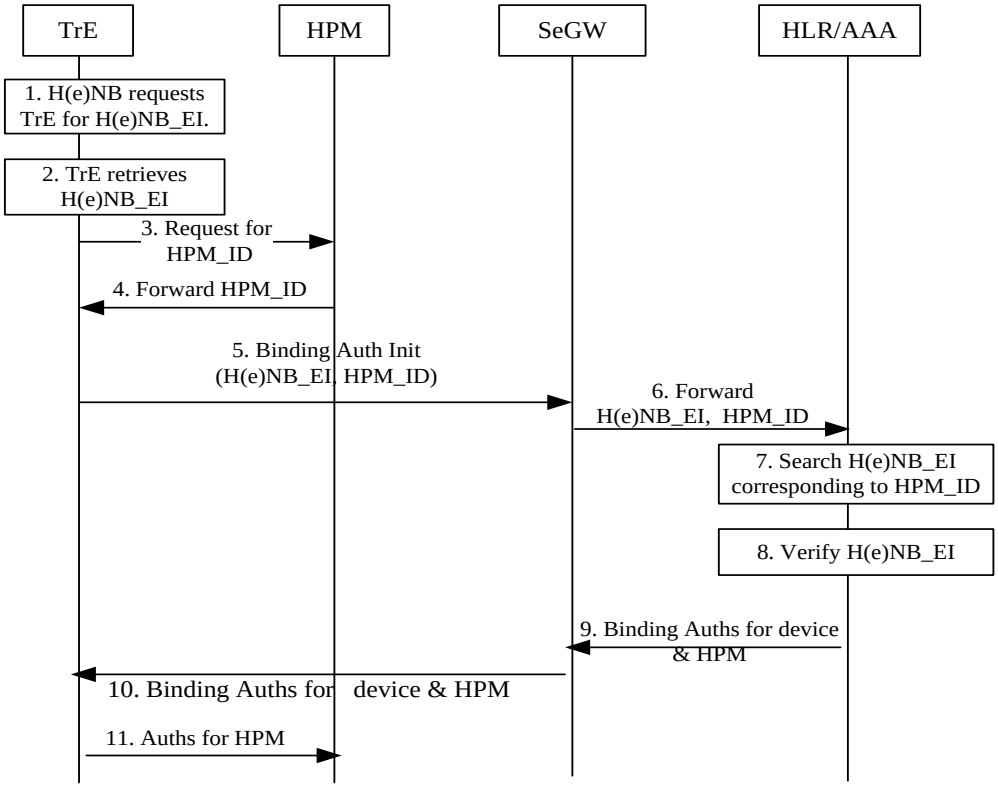


Figure 8: Binding Authentication

(1) H(e)NB requests the TrE for H(e)NB_EI for binding authentication.

(2) The TrE retrieves the H(e)NB_EI that it securely holds.

(3) The TrE requests the HPM to forward HPM_ID to the TrE. If the TrE is capable of setting up a secure channel with the HPM, then the TrE should first set up a secure channel between it and the HPM, and send the request for HPM_ID protected by the secure channel.

(4) The HPM forwards the HPM_ID to the TrE. If a secure channel is set-up in step (3) above between the TrE and the HPM, the forwarding of HPM_ID from the HPM to the TrE should be protected by the secure channel.

(5) The TrE sends, through H(e)NB, a request, including HPM_ID and H(e)NB_EI, to the SeGW to originate the binding authentication process .

(6) The SeGW forwards the HPM_ID and H(e)NB_EI to the HLR/AAA to request the binding record .

(7) After receiving the HPM_ID, HLR/AAA searches for the H(e)NB_EI corresponding the HPM_ID.

(8) HLR/AAA verifies the H(e)NB_EI from SeGW. If it is the same with the registered record, binding authentication process succeeds. It can be judged that the HPM_ID is inserted in the legitimate equipment.

(9) The HLR/AAA responds to SeGW the binding Auth result.

(10) SeGW forwards to H(e)NB the binding Auth result, which is forwarded to the TrE in the H(e)NB.

(11) The TrE forwards the binding Auth result for the hosting party to the HPM. If a secure channel has been set-up between the TrE and the HPM, the forwarding of the hosting-party binding Auth result should be done under the secure channel.

Editor's Note: It is FFS whether HPM should not receive the HeNB_EI from the TrE and perform the binding authentication procedure itself.

This method depends on H(e)NB_EI sent by the entity is true, not forged.

There are two ways to achieve the prerequisite.

1, H(e)NB_EI is treated as onboard token secret. It is stored in H(e)NB a secure domain i.e. from which outsider cannot retrieve it.

Meanwhile, it is transport in cryptograph encrypted by a key (e.g, CK derived from AKA algorithms or Ki stored in HPM.)

Editor's Note: It is ffs what additional requirements (e.g. additional provisioning of keys and/or additional protocol runs) are introduced by the requirement of the encrypted transmission of the H(e)NB_EI.

2, SeGW performs device authentication to verify the H(e)NB_EI before binding authentication.

If a combinations of the HPM with an onboard certificate is used， the binding process would be as following:

(1) Each H(e)NB is provisioned with a shared secret during production. The H(e)NB_EI——>shared secret list are configured in SeGW or other core network equipment.

The SeGW perform pre-shared mode IKE agreement with H(e)NB to verify the said H(e)NB_EI is true.

(2) Each H(e)NB is configured a digital certificate.

The SeGW perform certificate mode IKE agreement with H(e)NB to verify the said H(e)NB_EI is true.

Equipment certificate or the pre-shared key can be pre-configured by H(e)NB equipment manufacturers.

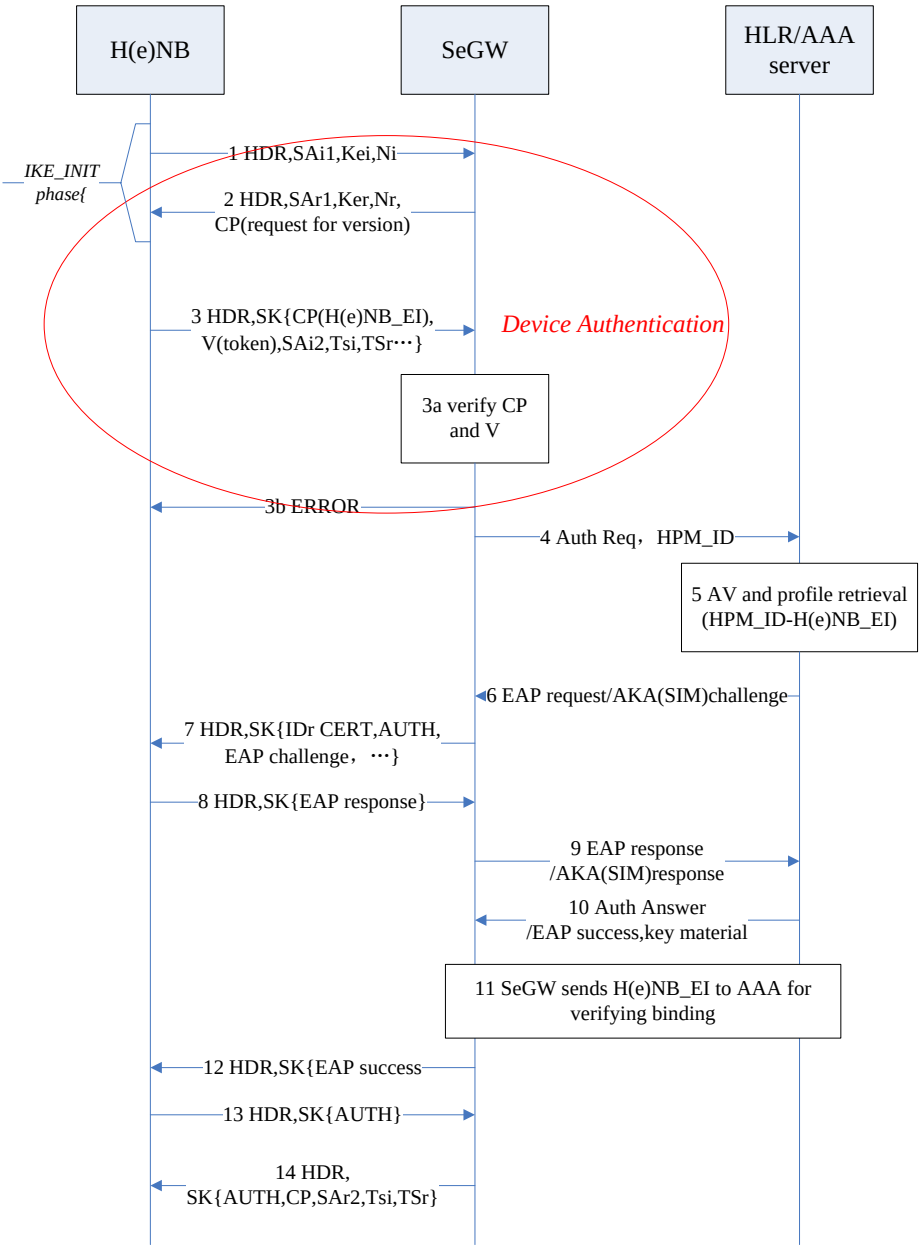


Figure 9: Device and Binding Authentication

In step 2, CP is sent from SeGW to H(e)NB for requesting version information (version payload has been defined in IKEv2 protocol).

In step 3, H(e)NB_EI is carried in version payload. A token is carried in vendor payload or in version payload. Token is calculated by a secret (private key or shared key) and NONCEi and NONCEr.

In step 3a, SeGW verify token.

In step 5, AAA get binding of HPM_ID and H(e)NB_EI;

In step 11, SeGW sends H(e)NB_EI and HPM_ID to AAA for verifying binding.

Editor's Note: It is ffs how to cryptographically bind the two authentications.

7.6.4 Relations to Trusted Environment

Editor's Note: It is ffs how an available trusted environment can cause synergy effects.

7.7 Backhaul Security Mechanisms

7.7.1 Backhaul Connection Security

Setting up a secure backhaul connection between H(e)NB and the operator network requires mutual authentication between H(e)NB and network.. A validation of the platform integrity, i.e. the validation of the properties of the TrE, must be included in or tightly bound to this device authentication (cf. clause 7.1).

There are several ways to ensure the backhaul connection security:

- if there is one SeGW only as endpoint for all communication between H(e)NB and network the platform integrity is validated during the device authentication of the first backhaul connection. Then either all traffic is tunneled through this backhaul connection, or additional backhaul connections between H(e)NB and this SeGW may be established, e.g. for QoS or performance reasons.. The binding of the device authentications of the additional backhaul connections to the validation of the platform integrity is ensured by the derivation of child SAs for the additional backhaul connections from the SA of the primary backhaul connection for which the platform integrity was verified.

- if backhaul connections to different network elements not behind the same SeGW are foreseen for the H(e)NB, e.g. one connection to a SeGW for signaling and user plane, and a separate connection to OAM server (also comprising an SeGW according to this document), then the following possibilities arise:

  - For any backhaul connection to be set up, the device authentication procedure initiates a separate validation of the platform integrity. This requires all backhaul connection endpoints to be able to perform such validation and to have access to the currently valid validation check data for each H(e)NB possibly connected to this endpoint.

  - The platform integrity is validated only during the device authentication of the first backhaul connection. The device authentication of any other (secondary) backhaul connection that is to be set up relies on the validation of the platform integrity that was checked in conjunction with the device authentication of the first backhaul connection.
    This approach requires additional complexity e.g. an additional mechanism that keeps track of the state of the validation of the platform integrity for each device. This state information must be integrity protected and available to any possible endpoint of a backhaul connection that is to be set up. This may require an additional network element to store this information and which must be contacted by each possible endpoint.

NOTE: If the platform integrity is validated only during the device authentication of the first backhaul connection, then the policy for expiry of the platform validation has to be considered separately. Different policies are mentioned here as examples:
(a) If the first backhaul connection is closed, the corresponding validation of the platform integrity must not be used any more as a base for other device authentications, which has to be reflected in the state information. This requires the endpoint of the first backhaul connection to report the closure of this connection to the entity storing the state information.
(b) If the first backhaul connection is closed, then also all other connections which relied on the platform validation of this connection are closed. This requires keeping track of all backhaul connections in different endpoint which rely on the platform validation of this device, to inform the endpoints of the need to close the secondary backhaul connections.

7.7.2 Backhaul Traffic Protection for H(e)NB

7.7.2.1 General

All signalling, bearer, and management plane traffic over the interface between H(e)NB and SeGW should be sent through an IPsec ESP tunnel (with NAT-T UDP encapsulation as necessary) that is established as a result of the authentication procedure. Encryption should use the negotiated cryptographic algorithm, based on core network policy, enforced by the SeGW. This policy should conform to one of the ciphering profiles described in 3GPP TS 33.234 and 33.210; any other profile should only be used after a careful security analysis.

The H(e)NB and SeGW set up Secure Association (SA) pair through which all traffic is sent. A single negotiated ciphering algorithm is applied to the connection.

7.7.2.2 Establishment of a Secure Tunnel

The H(e)NB should set up at least one IPsec tunnel, i.e. a pair of unidirectional SAs between H(e)NB and SeGW. The H(e)NB should initiate the creation of the SA i.e. it should act as initiator in the Traffic Selector negotiation. Upon successful authentication, the SeGW allocates IP address to the H(e)NB.

The H(e)NB and SeGW should use the IKEv2 mechanisms for detection of NAT, UDP encapsulation for NAT Traversal, H(e)NB initiated NAT keep-alive, IKE and IPsec SA rekeying, and Dead Peer Detection (DPD).

The ciphering mode is negotiated during connection establishment. During setup of the tunnel, the H(e)NB includes a list of supported encryption algorithms as part of the IKE signalling, which include the mandatory and supported optional algorithms defined in the IPsec profile. The SeGW selects one of the crypto suites specified in 3GPP TS 33.234, and signals this to the H(e)NB.

7.7.2.3 Supporting QoS

The support of QoS between the H(e)NB and the SeGW via DSCP marking of a single SA pair or multiple Child SAs is FFS.

NOTE: Details of QoS support are out of scope of this technical report.

7.8 Location Locking mechanisms

7.8.1 Overview of Location Locking

Several threats identified in this document are related to operating the H(e)NB in inappropriate locations. In order to counter these threats, three steps can be distinguished:

1. Identification of the H(e)NB location
2. Authentication (verification) of the location information
3. Authorisation of H(e)NB operation

7.8.2 Comparison Security of H(e)NB Location Identification Methods

H(e)NB location can be provided by different means, by different parties. SA2 discussed in [3] methods to get the location of a H(e)NB. RAN3 gave a discussion in [4] to H(e)NBs on methods of certifying H(e)NB location

NOTE: The term ‘location certification’ used in [4] should be further understood by SA3 before the same term can be considered for this TR.

There are two types of location identification methods for H(e)NB, such as:

1. using the location information on neighbouring cells or UEs to obtain location identification of H(e)NB and

2. using the location information available locally to the H(e)NB itself.

Other methods can be classified as one of the above two types.

Discussion and decision the security principle of H(e)NB location identification methods is needed.

Types of location identification for H(e)NB	Basic requests of security	List of methods	Security analyses
Localization of H(e)NB based on neighbouring cells or UEreports	The neighbour reporting should be trusted. The reports should be fresh.	*From the surrounding macro-cells which can detect H(e)NB*	A rogue H(e)NB can move in the *macro-cells* location.
		From end-users who are using H(e)NB	A rogue H(e)NB can replay a previously stored data. The end-user should be of trust for this method to be secure enough.
Localization of H(e)NB by itself	The location information from H(e)NB must be trustworthy and collected in real-time and protected from a replay attack.	*From fixed access line end point (DSLAM)*	This method requires collaboration between the mobile operator and the fixed-access line operator, which are not necessary the same entity. The good side is that the collected information may be trusted. However, H(e)NBs are typically intermittent base stations, they can be switched on and off at anytime by their owners. Each time a H(e)NB pops up, the mobile network operator will have to check its location and this can put undesirable burden on the fixed-access operator.
		From WAN IP address and allocated ranges ("Whois")	In particular due to NAT (Network Address Translation), a rogue H(e)NB can easily impersonate its IP address, unless the procedure involves a trusted STUN server which can certify the public address.
		GPS in the H(e)NB	A rogue H(e)NB can replay a previous data unless specific countermeasures are in place.
		The H(e)NB can detect surrounding macro-cells	A rogue H(e)NB can replay the location data registered in a previous location although it has moved unless specific countermeasures are in place.
		The H(e)NB may embed a receiver of some radio standards (radio, TV,…) and find location from a radio signature computed from received signals	A replay is possible unless specific countermeasures are in place.

Table 5: Comparison security of H(e)NB Location Identification Methods

Location information should be stored securely. Further, the various location functions of the H(e)NB should be carried out within the TrE and if the H(e)NB receives location information of messages from a macro cell, UEs, SeGW or Access point Home Register (AHR), or transmits then such information should be handled securely within the TrE and protected while in transit or storage.

The functions of the H(e)NB for location identification fall into the category of security-sensitive functions and have to be protected from malicious manipulation. Since the TrE provides a secure execution and storage environment within the H(e)NB, such protection should be provided by the TrE for these functions and information. The TrE should

- Check the integrity of the location identification functions during a boot process,

- Check the integrity of location identification information obtained in the Identification step,
- Protect location information obtained from the identification step if such information needs to be stored within the H(e)NB temporarily for reasons of processing or auditing.

However, the location identification functions themselves do not necessarily have to be carried out within the TrE itself.

7.8.3 Location Authentication

Analysis of location identification methods above showed several possible attacks that may cause a false location to be reported. Location Authentication intends to prohibit such attacks or to make them adequately difficult to succeed. In addition to protecting a single location identification method, a combination of two or more methods may also be used to verify the reported location information.

The location authentication functions of the H(e)NB fall into the category of security-sensitive functions and have to be protected from malicious manipulation. Such protection should be provided by the TrE. The TrE should

- Check the integrity of the location authentication functions during a boot process,
- Protect certain information required for location authentication, such as authentication credentials used for location authentication, if such information needs to be stored within the H(e)NB.

However, the location authentication functions themselves do not necessarily have to be carried out within the TrE itself.

NOTE: As given in the Table 6 above, the sources of location identification have a varying degree of reliability. When considering execution of location-related functions in TrE, it has to be noted that the overall security of the results is no more reliable than the input. Thus the trade-off has to be evaluated if execution in TrE gives more advantages as compared to the increased complexity of TrE which may reduce overall security.

Editor's Note: Integrity assurance of location information data from the Identification step may not be needed in some situations.

7.8.4 Location Authorisation

Authorisation of H(e)NB operation in the identified and authenticated location is an operator decision.

7.8.5 Solutions

The core network obtains the information of the H(e)NB location and compares it with the corresponding H(e)NB location information stored. If they match, then the core network grants H(e)NB service access based on the H(e)NB location information.

The location information of the H(e)NB can be obtained from

- the IP address of the broadband access device, or
- the information of macro-cells surrounding the H(e)NB, or
- the location information from the GPS embedded in the H(e)NB itself or in the UE which is camping on the H(e)NB.

In this section, several specific solutions are described as follows.

7.8.5.1 Solution based on IP Address

A H(e)NB is normally connected to the IP network via some access device (e.g. DSL modem, cable modem, home router, etc.) and has an IP address assigned by broadband access provider. By binding the physical ports of the broadband access network with the geographical information, the operator can locate the H(e)NB.

The assigned IP address, user identification and location information related to IP address are stored in the network database. According to the IP address sent by H(e)NB, the mobile core network can query the network database to obtain the port number(s) bound with the IP address, and/or the address information (even the longitude and latitude values).

The IP network should provide interface for the H(e)NB operator. Through the interface, the mobile core network is able to query the geographic location information based on the IP address,.

The NASS(Network Attachment Subsystem) standard in TISPAN [5]has defined the interface. The above network-based database can be the CLF (connectivity Session Location and Repository Function) element .CLF registers the following information provided by NACF(network access configuration function), and make them relevant: the IP address located to the fixed access point, the network location information, and geography location information. CLF provides e2 interface for service layer entity. The reference document [6] gives e2 interface specification based on Diameter protocol.

The entity used to query CLF can be the home register of H(e)NB. We name it AHR in order to distinguish with the UE home register HLR..

NOTE 1: The solution described above is valid for access networks according to TISPAN NASS. Describing solutions for other access networks is not in scope of the current document.

NOTE 2: An IP-address based only method is not reliable regarding the location since the H(e)NB could be connected to the Local Area Network of the hosting party through a Virtual Private Network. Barring development of a hitherto unknown method of detecting such tunnelling within the LAN the IP-addressed based method should not be used stand alone but should be combined with at least one of the other methods to increase the reliability of the solution. In addition the access network must have some means to prevent IP spoofing.

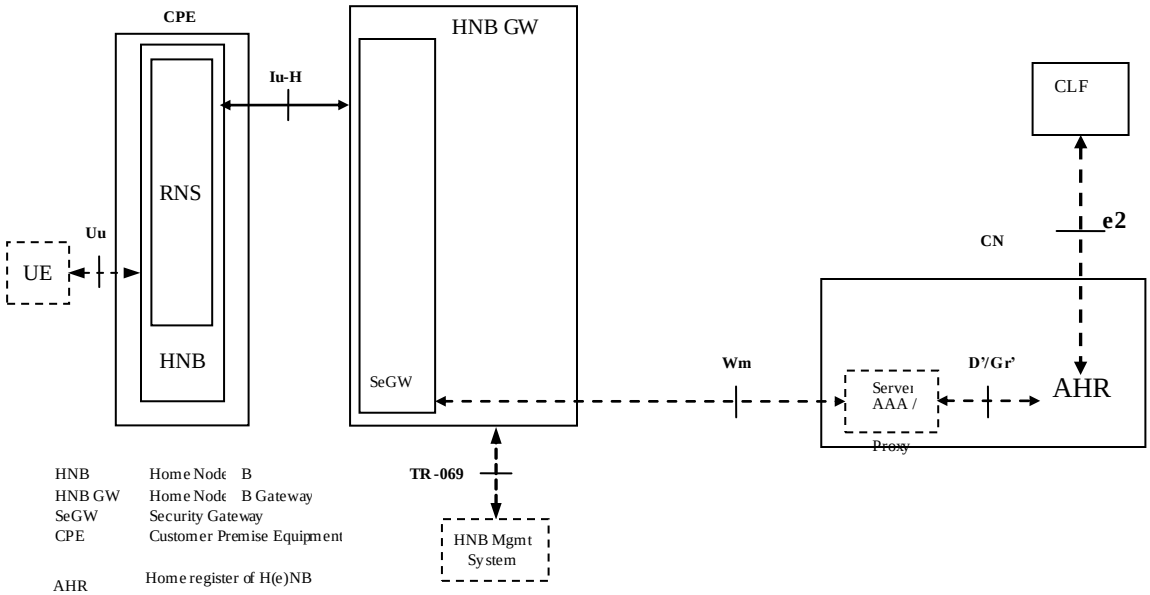


Figure 10: H(e)NB System Architecture Extension for Location Identification

The location locking mechanism consists of two steps:

1, The registration of H(e)NB location information ;

2, The authentication of H(e)NB location.

Step1 is a Location Registration step, and only occurs when H(e)NB powers on for the first time and connects to core network through IP backhaul.

Step 1 consists of the following sub-steps: :

1a, H(e)NB sends a request message to AHR, carrying its IP address in this message.

1b, AHR sends a location information query message to the CLF, carrying the received IP address.

1c, Based on the IP address, the CLF queries its database to obtain the access line location identifier of the H(e)NB, which is used to identify the access line location of the H(e)NB ,such as the port numbers bound with the IP addresses,

1d, AHR determines the location of the H(e)NB based on the obtained identifier. AHR registers the location of this H(e)NB.

1e, AHR replies response message to H(e)NB.

After step1, AHR can store access line location identifier as an attribute of H(e)NB profile, treating it as a criterion to judge the location with.

NOTE 1: If the contract location exists in AHR already, the registration step is not needed. The contract location can be defined by operator when H(e)NB service is subscribed .The same applies as for the other methods below.

NOTE 2: The description above does not include that the contract of the operator with the hosting party may require a certain location of the H(e)NB, e.g. because of regulatory requirements. This has to be checked and verifies by the network. Thus the AHR may be pre-configured with a "required location" of the H(e)NB. Exact procedures for this are not in scope of this document. The same applies as for the other methods below.

Step2 is Location Authentication step. This step occurs every time when H(e)NB requests to access network. Therefore, there is no need for registration, unlike in Step 1 above. Step 2 consists of the following sub-steps:

2a, H(e)NB sends access request message to AHR, carrying its IP address in this message.

2b, According to the new IP address, AHR queries the CLF again to obtain the access line location identifier.

2c, AHR authenticates whether the access line location identifier stored in AHR corresponds to the location identifier it newly retrieves from CLF based on IP address obtained from the H(e)NB. If it is the same, it can be ascertained that the H(e)NB location is not changed.

2d, AHR replies to H(e)NB the location authentication result in a response message,

NOTE 3: The location information is stored in AHR as a subscription profile is more reasonable. Thus, the messages related to location authentication between AHR and H(e)NB is forward by SeGW. The role of the SeGW in enforcing admission of the H(e)NB to the network is omitted here for simplicity. The same applies to the other methods below.

H(e)NB's location can be authenticated using the above Step 2. However， proxy attacks may be possible: a proxy server may take on the same IP address as a legitimately registered H(e)NB when the H (e)NB is relocated to another area. Such a proxy server then may be able to disguise as the legitimate H(e)NB, as far as location is concerned.

7.8.5.2 Solution based on H(e)NB Reports of Neighbouring Macro-cells

To be located on the basis of macro cell information, a H(e)NB must be installed in the coverage of a macro cell, have a 3G or 2G receiver, and is able to switch to the receiver working state to scan the neighbouring macro 3G or 2G cells of the H(e)NB.

The location locking mechanism based on macro-cells is similar to the method using IP addresses but the location information is presented in the form of information about macro-cells, such as PLMN ID， LAI or Cell ID .

Step 1 The registration of H(e)NB location information

After an H(e)NB is powered on, it scans the neighbouring macro cells in a receiver mode. Then the H(e)NB sends a H(e)NB Location Registration Request message to the AHR. The message carries the information such as location area and cell ID of the neighbouring macro cells. The AHR registers the cell ID of the neighbouring macro cells as an attribute of H(e)NB profile, and sends a H(e)NB Response message to the H(e)NB.

Step 2 The authentication of H(e)NB location.

The H(e)NB sends an Access Request message to the AHR. The message carries the information such as location area and cell ID of the neighbouring macro cells. AHR compares the information of neighbouring macro cells with the saved H(e)NB profile to determine whether to allow the H(e)NB to connect to the network through the bound cell or location area. If the information of neighbouring macro cells does not match the H(e)NB profile, the AHR returns a H(e)NB Access Response message to refuse the H(e)NB access and indicates "invalid location" as the cause value. If the information of neighbouring macro cells matches the H(e)NB profile, the AHR returns a H(e)NB Access Response to allow the H(e)NB access.

The security of this solution can be enhanced if H(e)NBs are required to report not only static information such as the location area and cell ID as described above, but also other information that is both dynamic and difficult for an attacker to generate. One way to generate such information would be for the AHR and macro cells to share a secret key. Macro cells can then use this key to generate and transmit a keyed hash of {cell ID || timestamp}. An H(e)NB claiming to be near some macro cell would then have to provide the AHR with a recent keyed hash from that macro cell, which could then be verified by the AHR.

Editor's Note: It needs to be confirmed whether macro cells already broadcast suitable timestamp.

7.8.5.3 Solution based on IP Address and H(e)NB Reports of Neighbouring Macro-cells

A macro cell has a large coverage area and therefore, simply using the cell information may not meet accuracy requirements for certain use cases. Using a combination of the IP address and the macro cell information could improve the accuracy.

The process is described as following:

Step 1 The registration of H(e)NB location information.

1a, H(e)NB sends the request message to AHR, carrying its IP address and neighbouring cell ID in this message.

1b, AHR sends a Location Information Query message to CLF, carrying the received IP address. Based on the IP address, AHR queries the CLF in order to obtain the access line location identifier bound with the H(e)NB IP address.

1d, According to the access line location identifier and neighbouring macro cell ID, AHR determines the home area of the H(e)NB. AHR stores the access line location identifier of this H(e)NB together with the received cell ID as attributes of the H(e)NB.

Step 2 The authentication of H(e)NB location.

2a, AHR receives the Access request message from H(e)NB, which carries its IP address and the cell ID of the surrounding macro cell.

2b, According to the new IP address, AHR queries CLF again to obtain the access line location identifier, which is used to identify the access line location of the H(e)NB.

2c, AHR judges whether the new obtained access line location identifier is the same with the stored one, and additionally whether the received cell ID is the same with the stored ones. If they are both the same, it can be ascertained that the H(e)NB location is not changed.

2e, AHR replies to H(e)NB the location authentication result in the access response message.

Note that under this proposed scheme, even if H(e)NB moves to another unregistered address, H(e)NB may still be located within the same macro cell. This arrangement may improve the security of the location authentication scheme. Since an attacker has to surmount two barriers to commit fraud here, i.e. the IP address and false macro-cell information.

7.8.5.4 Solution based on UE Information

H(e)NB can detect the UE nearby. The UE position may represent the H(e)NB location. Thus the H(e)NB location can be verified if only one of surrounding UE is equipped with GPS. AHR stores the range of valid location area. UE signs its location information and sends the signed information to H(e)NB. The H(e)NB signs the received information with its identity. Then H(e)NB sends the signed information to CN to validate if the location is the legal scope..

NOTE: The description above gives the basic principle. If such method is selected for deployment, necessary adaptations of the UE and suitable protocols for transmission of location data from UE to H(e)NB have to be considered.

7.8.5.5 Solution based on UE information and H(e)NB Reports of Neighbouring Macro-cells

Combining these two sources of location information offers several advantages over either on its own. An attacker may be able to modify or forge GPS-based UE location information (and to a lesser extent, A-GPS location information). Checking this information against reported neighbouring macro cells requires the attacker to compromise two independent sources of information. As discussed earlier, macro cells have a large coverage area, so location information based solely on them may not be accurate enough for some uses (such as E-911). UE-based location information will improve this accuracy.

7.8.5.6 Solution based on (A-)GPS in H(e)NB

When the H(e)NB has built in GPS or Assisted-GPS (A-GPS) capability, its location information may be obtained via the (A-)GPS within the H(e)NB and subsequently can be sent from the H(e)NB to CN during access request. GPS may not work very well, however, in some indoor environments. A-GPS will improve indoor performance considerably.

7.8.6 Re-locking of H(e)NB Location

7.8.6.1 Same Location for H(e)NB

When there is no change to the location of H(e)NB and H(e)NB recovers from unavailability (e.g. power off, break down etc), H(e)NB shall perform the re-locking of its location according to the three steps in section 7.8.1.

7.8.6.2 Different Locations for H(e)NB

As long as the location of H(e)NB is changed (e.g. a H(e)NB is moved from a house to another house by its owner), its move of location should be monitored by operators.

When H(e)NB is placed in a new location, H(e)NB follows the three steps in section 7.8.1 for the re-locking of its new location on condition that the availability of the new location is verified by the operators.

7.8.7 H(e)NB Location Policy Options and Configuration

Which of the above approaches to use depends on a number of factors, such as security level and accuracy level according to operator policy, H(e)NB capability（whether (A-)GPS installed, or macro coverage. A policy may be applied to assist in determining the method to be used. It is suggested that the policy is pre-configured in H(e)NB, e.g. by operator configuration data ,and H(e)NB automatically adapt to it.

Using IP address alone may not be secure enough. GPS, and to a lesser extent (A-)GPS, may not work well in some indoor environments, and both may add cost to the H(e)NB.

One example of a policy based on the factors above is shown here. Other policies are of course possible. Prioritization and possible exclusion of certain methods may be an extension of the operator policy, not reflected in the example table below.

Scene	Policy
No macro cell exist, and no GPS installed	H(e)NB cannot register reliable location
Macro cell exists and system has high security requirement	IP address + Macro cell
Macro cell exits and one of surrounding UEs is equipped with (A-)GPS	Macro cell + UE information
(A-)GPS installed in H(e)NB	GPS information+ IP address
(A-)GPS installed in H(e)NB and Macro cell exists	GPS information + IP address + Macro cell
One of surrounding UE is equipped with GPS	UE information

Table 6 Example Location Identification Policy Table

7.9 Access Control Mechanisms for H(e)NB

7.9.1 Non-CSG Method

The access control mechanism at connection establishment for the non-CSG capable UE accessing to HNB or the UE accessing to non-CSG capable HNB is handled in [16].

Editor's Note: More work is needed to determine what input, if any, SA3 needs to provide on the handling of ACL-based access control at handover. This work should include an investigation into which other 3GPP standards documents define this procedure.

7.9.2 CSG Method

The access control mechanism at connection establishment for the CSG capable UE accessing to CSG capable H(e)NB based on CSG concepts is handled in [19] (HNB) and [20] (HeNB).

Editor's Note: More work is needed to determine what input, if any, SA3 needs to provide on the handling of CSG-based access control at handover. This work should include an investigation into which other 3GPP standards documents define this procedure. Note that the documents listed above do not provide a complete description.

7.9.3 Access List Management

7.9.3.1 Overall Model and Requirements

Following steps or procedures are necessary for overall access list management; each of these faces security issues and thus have security requirements:

- Management of access lists should be secure, this includes:
  - Creation
  - Modification, like:
    - Deletion of UEs and/or H(e)NBs

- Addition of UEs and/or H(e)NBs
- Expiration

- Access:
  - Initial connection
  - Idle mobility
  - Handover
- Temporary access
- Deployment and migration from Non-CSG solution to complete CSG capable network

A high level access list management and access control related message sequence for the CSG case is given in Figure 11. From this we can capture the following requirements:

- Authenticated messages for communication regarding access list management and access control
- Authorized access list management
- Access control for each connectivity
  - Binding the H(e)NB with UE ID
  - Checking whether the H(e)NB is authenticated by the network
  - With authentication of H(e)NB by the UE, this is to prevent issues like UE connecting to a wrong H(e)NB, UE disconnected by a rogue H(e)NB leading to change in Allowed CSG List etc.
- Deployment security also considering migration

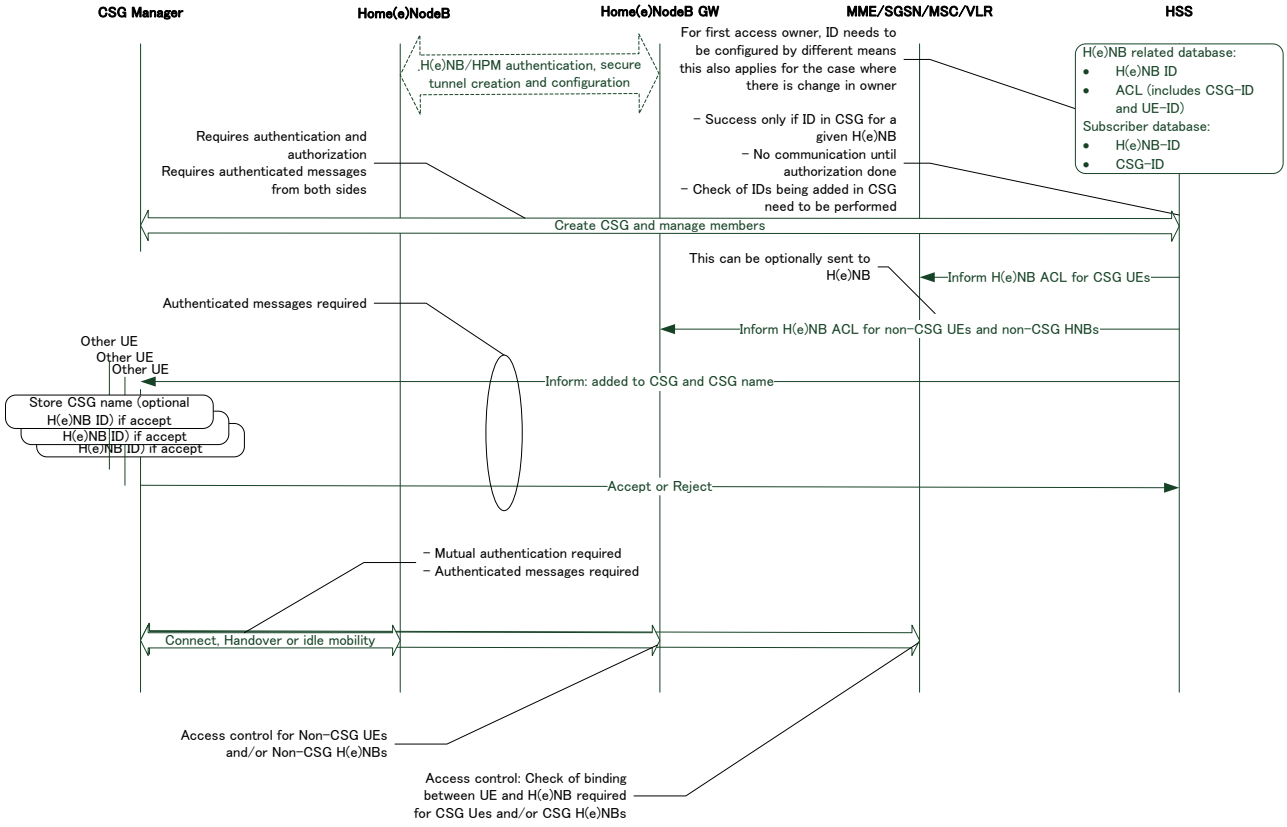


Figure 11: High level message sequence for access list management and access control for the CSG case.

TS 22.220 contains the following requirement regarding the storage of CSG Lists and CSG Type in the USIM of the UE:

- *"The UE shall contain a user controlled list of allowed CSG identities (Allowed CSG List). It shall be possible to store the Allowed CSG List in the USIM. When available, the list on the USIM shall be used."*
- *"In addition to the Allowed CSG list, the UE shall maintain an operator controlled list of allowed CSG identities (Operator CSG list). It shall be possible to store the Operator CSG list in the USIM. When available, the list on the USIM shall be used"*
- "It shall be possible to store the CSG Type in the USIM. As an option, the CSG Type may be stored in the ME. If the CSG Type is present in the USIM, a CSG Type stored in the ME shall be ignored."

7.10 Security Mechanisms for OAM

In case that OAM server is placed inside the operator's network, i.e., behind a SeGW:

- OAM traffic could be protected hop-by-hop. Between H(e)NB and SeGW, OAM traffic is protected by IPsec tunnel. Network security mechanisms could be used to protect OAM traffic between SeGW and OAM server when the path from SeGW to OAM server is considered as insecure.
- OAM traffic could also be protected end-to-end. A secure tunnel, e.g. TLS, is established between H(e)NB and OAM server. OAM traffic is protected by such a security tunnel.

In case that OAM Server is placed outside the operator's network, i.e., the H(e)NB is directly connected to the OAM server, OAM server is exposed to attackers located in insecure network. Though secure tunnel, e.g. TLS, could be used to partially protect OAM traffic and prevent OAM server from being attacked, the risk of placing OAM server outside the operator's network is higher than that of placing the OAM server inside the operator's network. Robust security mechanisms between OAM server and H(e)NB should be carefully designed and implemented in this case.

Editor's Note: OAM's own security mechanisms may still need to be considered.

7.11 Clock Synchronization Security Mechanisms for H(e)NB

7.11.1 General

The following sections describe the various implementation options that can be used for clock synchronization security mechanism.

7.11.2 Based on Secure Backhaul Link between H(e)NB and SeGW

The clock server should be located behind the SeGW, the communication between the clock server and H(e)NB is protected by the secure backhaul link between H(e)NB and the SeGW.

NOTE 1: It may be possible to leave some of the clock related signalling messages unprotected. There may be some security risks leaving some of clock synchronization messages unprotected, e.g., DoS attack to core network or H(e)NB. Care should be taken in considering what messages are to be protected or not protected. The detailed messages that are considered will be FFS;

NOTE 2: There may be bandwidth, delay or jitter problems if all of the time synchronisation (e.g. IEEE 1588) traffic is protected. This should be taken into consideration in clock synchronisation protection.

NOTE 3: If delays for time synchronization become excessively long because bandwidths required to transport secured (or unsecured) time synchronization packets, the need to rely more on internal clock for time-related operations will be greater, since time synchronization may not take place at sufficiently high frequencies. Therefore, SA3 may need to investigate measures to ensure and protect the availability and accuracy of local time-clocks within the H(e)NB.

Editor's Note: Provisioning of the clock server name needs to be considered as additional step for the purpose of comparing against different solutions.

7.11.3 Based on Security Protocols of the Clock Synchronization Protocols

Existing clock synchronization protocols with built-in security protocols can be used. For example, the Network Time Protocol (NTP) defines Autokey Specification to secure the clock synchronization, and IEEE 1588 also defines a security extension in its specification.

When a Clock Synchronization Protocol is used for clock synchronization for H(e)NB, the security protocols can be used as Clock Synchronization Security mechanisms for H(e)NB.

NOTE 1: When the time server is reached via the unsecured Internet, this opens up the risk of DNS attacks and IP address spoofing outside the operator network. In addition the H(e)NB has additional open ports (e.g. for NTP) accessible from the Internet, which may make hardening of the device against Internet based attacks harder.

NOTE 2: In case NTP is used, the scaling of the Secure NTP to the scale of the number of H(e)NBs (i.e. the NTP server handling different credentials for each H(e)NB) is to be considered before decision.

7.12 H(e)NB Distress Indication

7.12.1 General Requirement

If the device integrity check for one or more components fails, then this implies that either those components are compromised or that the corresponding trusted reference values are out of step with the code base on the device. Since a H(e)NB that fails device integrity verification may not be able to perform the authentication procedure, it may not be able to communicate with the network, and the network would not know that the device is unable to attach to the network. To mitigate this problem, the H(e)NB could initiate communications with the network to indicate that it is in distress, thereby alerting the network so that it will know that the device is unable to authenticate to, or communicate with, the SeGW. Optionally, the details of the distress indication message could be expanded to facilitate a network-initiated update of the normal code image. In this context, "normal code" means any executable code which has to be verified by the TrE but it excludes any functions of the TrE itself.

7.12.2 Distress Communication Function

The H(e)NB should be equipped with a distress communication function, the principal purpose of which is to facilitate transmission of a distress indication message to the network, in case the H(e)NB fails device integrity verification and is therefore unable to authenticate to or to communicate with the network in the normal manner.

Editors Note: As this function is management related, it needs alignment with SA5

The distress communication function should be executed if the device integrity verification fails, including any failure which prevents normal communications with the network. The distress communication function should contain at least all necessary functions, methods and credentials needed for communication with the entity in the network that is responsible for receiving the distress indication from the device.

The distress communication function should be stored separately from the "normal" code, so that it can operate as a fallback function, to be invoked if higher-level functions are found to be corrupted .

Optionally, the distress communication function may include operations necessary to receive a full, remote software update of the entire normal code image of the device from the network.. This distress communication function would not be used to recover from a case where the TrE has failed its integrity check.

7.12.3 H(e)NB Distress Indication Procedure using Distress Communication Function

H (e)NB devices should implement a secure start-up which allows the device to perform device authentication procedures if and only if designated elements of local device integrity verification are successful. If any of the designated components fail their integrity check, the device should be considered as having failed its integrity check and should initiate its distress indication procedure. In the distress indication procedure, the device should execute the distress communication function which would contact a network entity (e.g. a pre-designated H(e)MS) to indicate that it is in distress.

The distress communication functionality should include operations for the H(e)NB to send a distress indication message to the pre-designated network entity in case of integrity verification failure of the H(e)NB.

The distress indication message should be sent to the pre-designated network entity using one of the following methods (the list does not imply any order of priority):

- Secure connection and authentication via TLS
- Secure connection and authentication via IKE/IPsec
- Secure connection and authentication via pre-shared keys and symmetric cryptography
- Connection without authentication or communications security

7.12.4 Optional Procedure for Replacement of Normal Code Image Using Distress Communication Function

Optionally, the H(e)NB distress communication function may facilitate remote replacement of all of the normal code image of the device, if the network initiates such a procedure as a result of the received distress indication message. This does not apply to the case where any related functions of the TrE have failed their integrity check.

The H(e)NB distress communication function may utilize TR-069 status/capability checking and SW download functions for such remediation.

The optional replacement process for the normal code image of the H(e)NB should also include replacement of the corresponding trusted reference value(s).

Upon completion of the optional normal code image and/or validation value(s) replacement process, the H(e)NB should reboot.

7.12.5 Requirements for Distress Communication Function and Distress Indication Message

Requirements on the distress communication function and distress indication message should include the following:

1. The code for the distress communication function should be securely stored separately from the "normal code" within the device
2. The distress communication function should be loaded and started in case of a failed secure start-up.
3. The address of a pre-designated network entity (e.g. a pre-designated H(e)MS) should be stored in the distress communication function
4. The distress communication function should send a distress indication message to the pre-designated network entity. The distress indication message information element should include the device ID (as derived from RoT secure storage).
5. The distress communication function should be capable of connecting to the pre-designated network entity.
6. The pre-designated network entity, upon receipt of the H(e)NB distress indication message should know that the device has failed its integrity verification and requires maintenance
7. The distress communication function may optionally include functionality to facilitate a full replacement and rebuild of the normal code image and replacement of the corresponding trusted reference value(s) through a process initiated by the pre-designated network entity. Upon completion of this optional process the H(e)NB should reboot

8 Conclusions

8.1 Authentication

In this study device authentication was identified as the essential precondition for H(e)NB security. Besides, it is obvious that the integrity of the device must be validated and the authentication must be tied to the validated device for any device authentication.

For device authentication, EAP-AKA based and certificate-based methods are described in this document.

For optional hosting party authentication, an EAP-AKA based method is described in this document which can be combined with the certificate-based device authentication.

It is recommended that the credentials and critical security functions for device authentication shall be protected inside a hardware-based Trusted Environment that is securely integrated into the H(e)NB and that device authentication shall be securely bound to device integrity validation.

Editor's note: The term "hardware based" may need to be further clarified.

It is also recommended that a single, certificate-based device authentication solution, coupled with the Trusted Environment, is standardized as a mandatory part of Release 9 and may be combined with optional EAP-AKA-based hosting party authentication which should also be standardized as part of Release 9. As only part of the deployment scenarios require the hosting party authentication, it is recommended that hosting party authentication is optional to implement in H(e)NB.

NOTE: If within the timeframe of the Release 9 specification work a hardware-based Trusted Environment turns out to be not feasible for certificate based device authentication and device integrity validation, then an EAP-AKA based solution based on an embedded UICC could be re-considered for device authentication providing that a mechanism can be provided to bind the EAP-AKA based authentication to the device integrity validation.

It is acknowledged that pre-Release 9 H(e)NB device authentication solutions can exist, e.g. based on EAP-AKA or certificate based authentication, which can offer an acceptable security level prior to Release 9 compliant solutions becoming available.

It is also recommended to use IKEv2 as authentication protocol since it includes the establishment of a secure backhaul connection between the H(e)NB and the SeGW based on IPsec, and also supports binding of device authentication and the optional hosting party authentication.

Editor's Note: More conclusions need to be added.

8.2 Location Security

It is recommended that the standard require the use of at least one of the available location data sources as provided by the H(e)NB during the discovery process as currently specified by RAN3 in TS 25.467 [16]. It is recommended that the standard allow for the optional use of additional location data as available. The standard should note that the reliability and accuracy of location information may depend on deployment scenarios.

8.3 Device Validation

It is recommended that autonomous validation of H(e)NBs shall be required as mandatory. This recommendation provides an acceptable balance between the achieved level of security and the necessary effort for its implementation and management and provides a quick realization of the minimum validation requirement for R9 timeframe. In addition, it does not affect any interfaces to and functions of network elements, and it is extensible with features of other validation methods, if seen as necessary in the future.

Further study of the potential benefits of Semi-Autonomous Validation, Hybrid Validation, or other potential validation methods is recommended. The topics to be investigated include the content and transport of a possible validation protocol, suitability of such protocol for the open interfaces specified for H(e)NB, management and infrastructure

requirements, and an evaluation of security advantages vs. incurred effort for the different possible realizations of other validation methods.

If other validation methods become stable within the time frame of Release 9, then optional support of another validation method is also recommended.

This recommendation preempts neither any concrete architectural realization thereof on the part of the network, nor any implementation option on the part of the device.

Annex A:
Security mechanisms for OAM

A.1 Mechanism to verify the software updates

Software update process will influent the integrity state of H(e)NB, One principle to verify the changes for the H(e)NB resulted by the software updates is that the platform integrity of the H(e)NB is verified before and after the software updates by the core network.

The overview process of the mechanism can be depicted in Figure A.1,

1. The software updates is initiated between the OAM server and the H(e)NB
2. The OAM server requests the PVE to verify the integrity of the H(e)NB.
3. The PVE checks the security status of the H(e)NB.
4. The PVE sends the result to the OAM server.
5. OAM server checks the received result. If it is correct then OAM server performs the software updates, and computes the expected RIM (Reference Integrity Metric)of the H(e)NB after software updates and the RIM of the software the H(e)NB updated.
6. The software updates process is performed, during which the OAM server sent the RIM of the updated software to the H(e)NB.
7. H(e)NB installs the received software locally .After the software updates, the integrity of the H(e)NB will be re-measured to get the current TIM(Target Integrity Metric) of the H(e)NB.

Based on the received RIM of the software and the old TIM , H(e)NB computes expected TIM of the H(e)NB after software updates , and compares it with the current TIM to certify the software has been installed correctly .

This step may be executed depending on the specific policy of the H(e)NB.

8. The H(e)NB sent a message to the OAM server about the completion, including the current TIM of the H(e)NB.
9. The OAM server compares the TIM of the H(e)NB with the expected RIM in local, if they are matched, then updates the RIM of the H(e)NB in local, otherwise, appropriate measures should be taken.

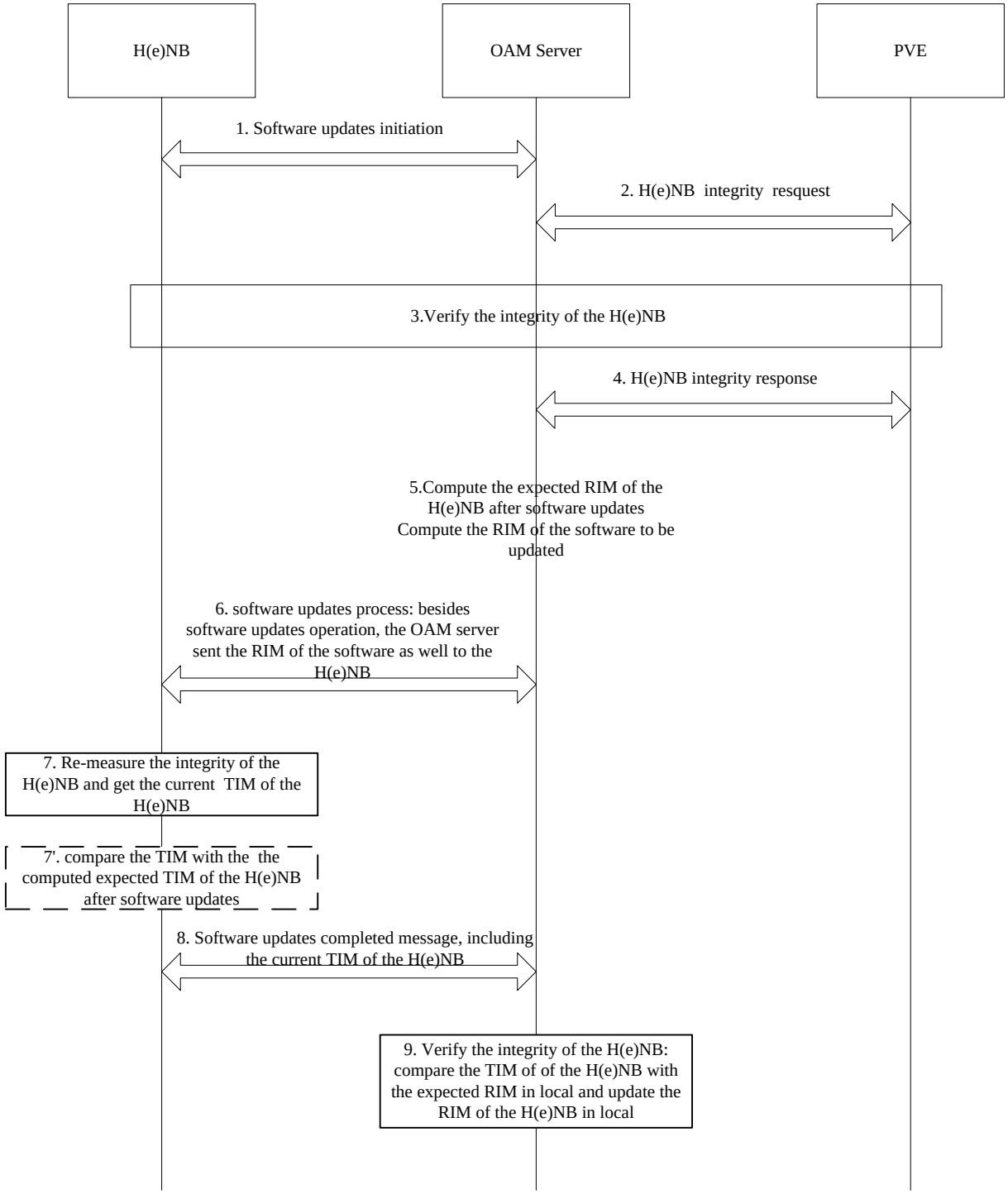


Figure A.1 Overview process of the Mechanism to verify the software updates

NOTE 1: In FigureA.1 the OAM server is an element lies in the core network. Other entities needed in implementation, such as SeGW of the network are omitted for the sake of simplicity of the process description.

NOTE 2: This is one possible method of performing software update verification which references TCG.

A.2 Another method to verify the software updates

The following gives another method to verify the software updates, this method supposes that there is a root K sharing between the TrE in H(e)NB and the HLR/HSS, it can be the root K for EAP-AKA for H(e)NB authentication.

The following are the steps:

1. The software updates is initiated between the OAM server and the H(e)NB
2. The OAM server computes the RIM (Reference Integrity Metric)of the H(e)NB;
3. The OAM server sends the RIM and the H(e)NB ID (IDh)to the ICE(Integrity check entity).
4. ICE(Integrity check entity) sends the IDh to HLR/HSS
5. HLR/HSS generate a random R and calculate a session key CK by the R and the root K.
6. HLR/HSS sends the (IDh, R, CK) to the ICE.
7. ICE cryptographically protect RIM(e.g. digital signature /or confidentiality) by CK. Suppose the result of digital signature for RIM is Sr, the result of confidentiality RIM is Cr. ie. Sr=Sig(CK, RIM), Cr=En(CK, RIM), Sig is a digital signature algorithm, En is a encryption algorithm.
8. ICE send (IDh, R, RIM, Sr) /or (IDh, R, Cr) to the OAM server.
9. The software updates process is performed, during which the OAM server sent the (IDh, R, RIM, Sr) /or (IDh, R, Cr) of the updated software to the H(e)NB.
10. The TrE in the H(e)NB calculate session key CK' by the recieved R and the root K in the TrE, and compute the RIM' by the recieved software, and then compute Sr'/ or Cr' by RIM' and CK'.
11. The TrE compares the Sr and Sr'/ or the Cr and Cr'. if they are equal then TrE save the Sr(/or Cr).
12. The TrE installs the received software locally.

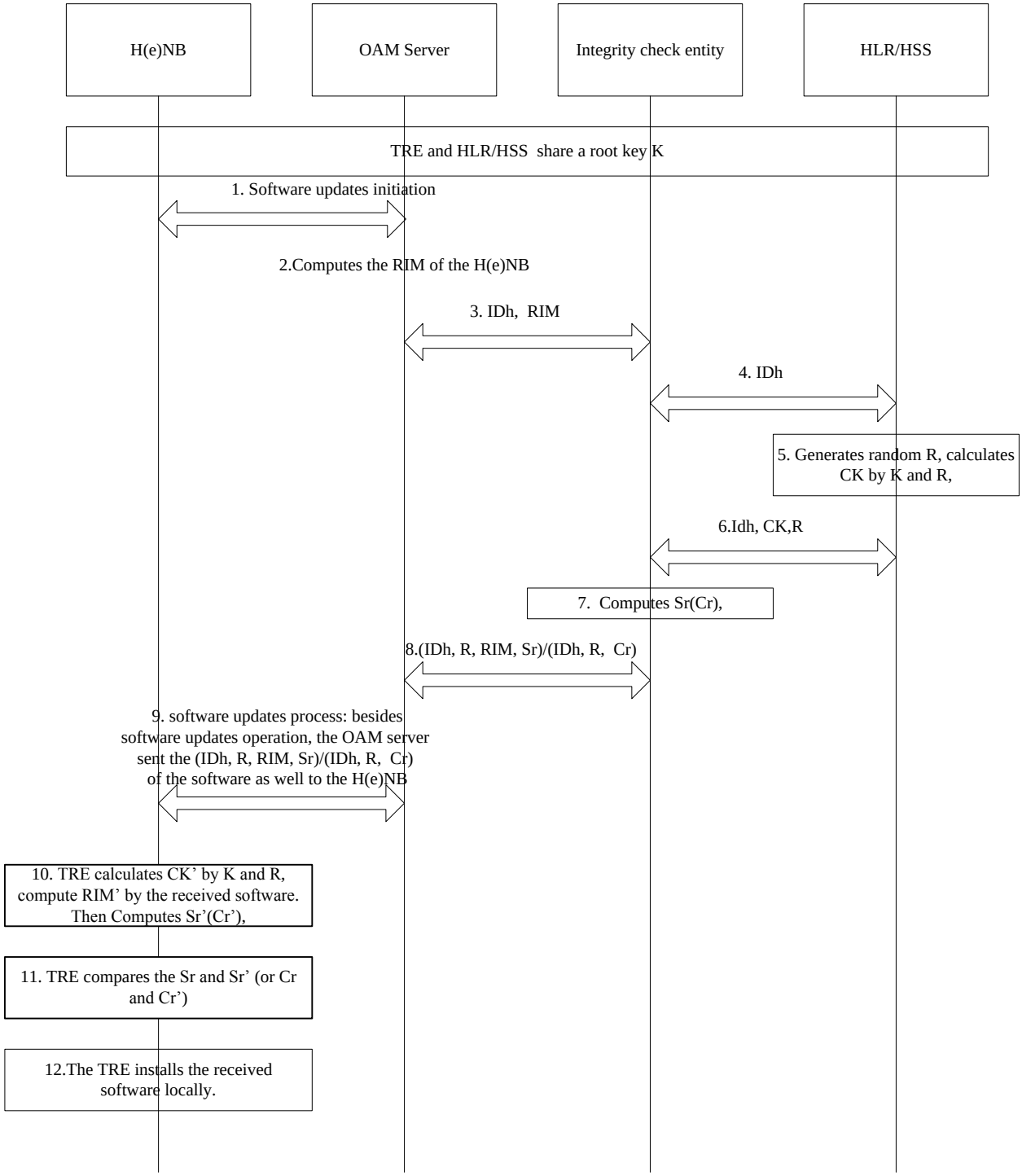


Figure A.2 Overview process of the Mechanism to verify the software updates

NOTE 1: The HLR/HSS here can't be the same as used for AKA, it can be an another independent equipment.

NOTE 2: There is no intereface between OAM SERVER and the HLR/HSS now, if this solution is used, it must implement this intereface.

NOTE 3: The trust model here is such that the operator's OAM server must determine validity of the software. The H(e)NB can't directly validate that the software comes from the correct manufacturer.

Annex B:
TrE Types and Corresponding Interfaces

Many different implementation possibilities can be considered for types of TrE's and interfaces for them. The following give examples with illustrations of some of the TrE implementation possibilities.

Thin TrE

A "Thin" TrE may be a TrE with the minimum necessary resources and functionality. The minimum set of resources and functionality a thin TrE should provide include:

- Capability to compute and send to the SeGW the parameters needed for device authentication of the H(e)NB
- Functions for H(e)NB validation, including code-integrity check of the rest of the H(e)NB at boot time;
- Some crypto capabilities and some protected memory (for persistent keys, etc) that can be used to enable protected interfaces to other building blocks of the H(e)NB and also to securely store H(e)NB_ID and/or TrE_ID and other authentication credentials;
- A true random number generator (TRNG);

Such a "Thin" TrE may utilize, and may critically depend on, some external protected resources such as more cryptographic functions and more protected memory, which may be accessible only to the TrE itself. Such resources should be accessed via protected interfaces to establish trustworthiness of the TrE and in particular to enable the secure start-up process of the TrE and H(e)NB. Other resources and functional building blocks of the H(e)NB may be security sensitive and may also need to be accessed via protected interfaces. Unprotected interfaces may connect the TrE to general purpose memory, part of which may be used to extend the TrE's secure storage capacity, and other non-sensitive resources of the H(e)NB.

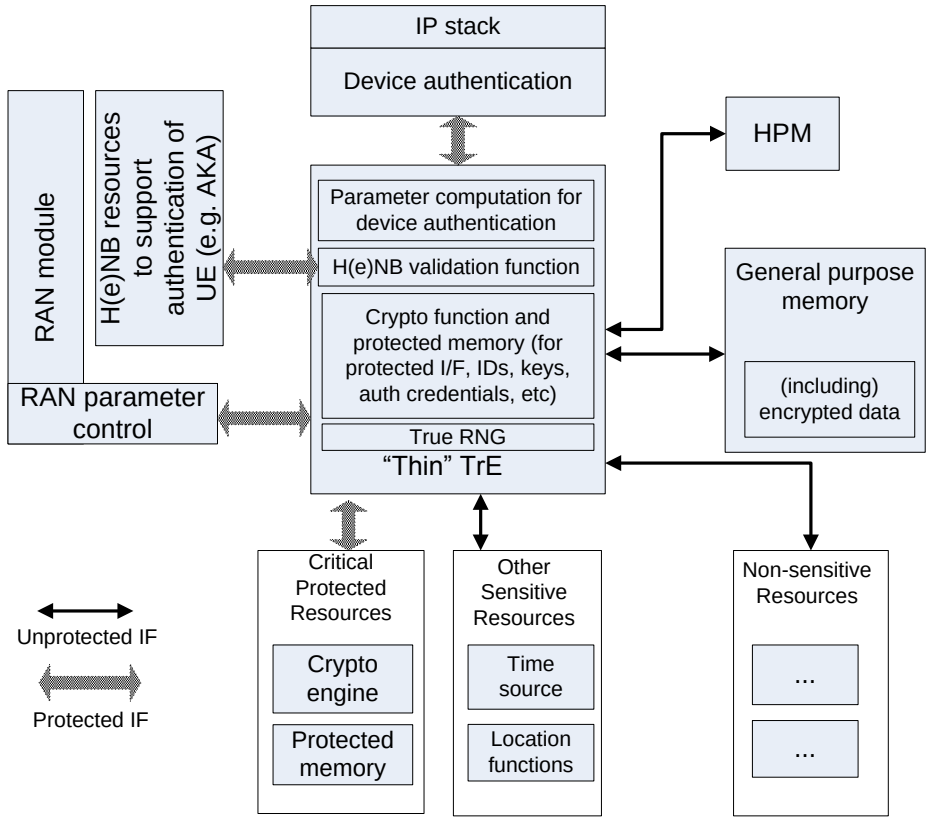


Figure B.1. A "Thin" TrE and its Interfaces

Thicker TrEs

A "Thick" TrE as depicted in the figure below may include within itself more resources and functionalities than a Thin TrE. Such resources and functionalities may include the same functions outlined for a Thin TrE and one or more of the following:

- More capable crypto resources and protected memory than those that are needed to just provide cryptographically protected interfaces for the TrE;
- Resources that enable full device authentication (including authentication method selection) over IKEv2 from within the TrE;
- H(e)NB procedures that support UE's AKA or GBA procedures;
- Capability to verify digital certificate for example from the SeGW;

A "Thick" TrE would interface with other security-sensitive parts of the HeNB via protected interfaces, and with non-sensitive parts of the HeNB via unprotected interfaces, as illustrated in the figure.

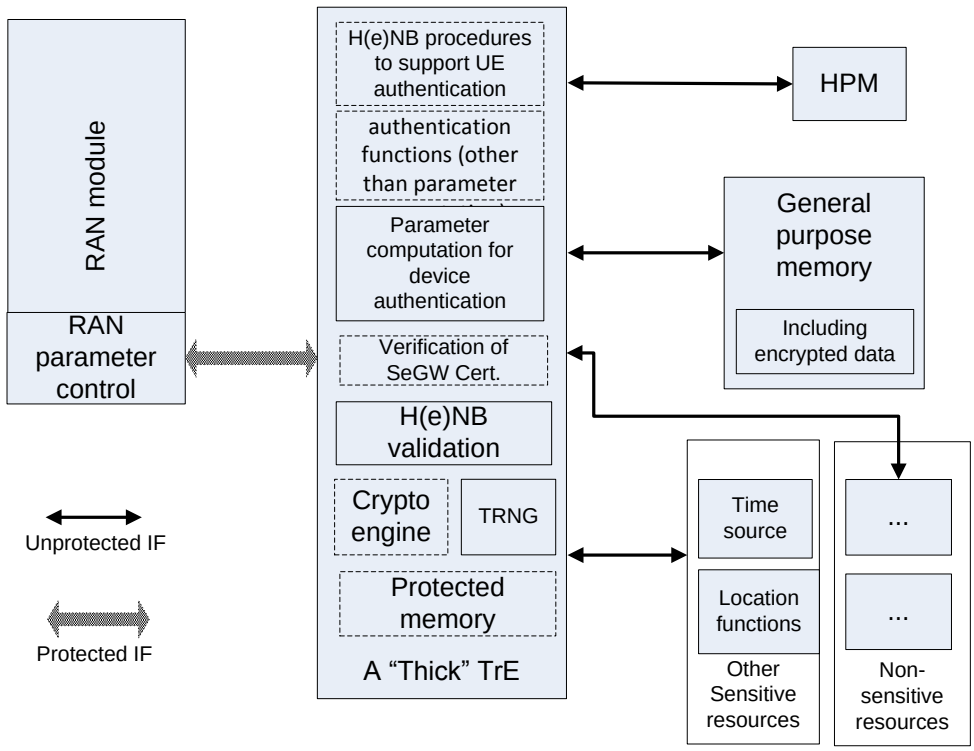


Figure B.2. A "Thick" TrE and its Interfaces

Annex C:
Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2007-12	SA3#49bis				Initial version	0.0.0	0.0.1
2008-02	SA3#50				Inclusion of content based on S3-080041, S3-080130, S3-080208	0.0.1	0.1.0
2008-04	SA3#51				Inclusion of content based on S3-080387,S3-080437,S3-080444,S3-080459	0.1.0	0.2.0
2008-07	SA3#52				Inclusion of content based on S3-080730, S3-080711, S3-080907, S3-080732, S3-080778, S3-080779, S3-080798, S3-080917, S3-080918, S3-080919	0.2.0	0.3.4
2008-09					MCC clean up for presentation to SA	0.3.4	1.0.0
2008-11	SA3#53				Inclusion of content based on accepted contributions	1.1.0	1.2.0
2009-01	SA3#54				Inclusion of content based on accepted contributions	1.2.0	1.3.0
2009-03	SA-43	SP-090131			Presentation to SA for approval	1.3.0	2.0.0
2009-03					SA approval	2.0.0	8.0.0
2009-06	SA-44	SP-090270	9	-	Clarification of Section 7.4 Authentication Method Selection	8.0.0	8.1.0
2009-06	SA-44	SP-090270	26	-	Remove Duplicate Section Number in 7.6.1.3.3	8.0.0	8.1.0
2009-06	SA-44	SP-090270	7	-	Update to H(e)NB system architecture	8.0.0	8.1.0
2009-06	SA-44	SP-090270	8	-	Update of Access Control Mechanisms for H(e)NB	8.0.0	8.1.0
2009-06	SA-44	SP-090270	14	-	CR to section 7.2.2.3 H(e)NB Authentication	8.0.0	8.1.0
2009-06	SA-44	SP-090270	12	-	Clarification of Certificate Profile	8.0.0	8.1.0
2009-06	SA-44	SP-090270	11	-	Conclusions on Location Security	8.0.0	8.1.0
2009-06	SA-44	SP-090270	3	-	Further description of Semi-Autonomous validation	8.0.0	8.1.0
2009-06	SA-44	SP-090270	1	1	Correction of text on autonomous validation	8.0.0	8.1.0
2009-06	SA-44	SP-090270	13	-	CR to H(e)NB validation	8.0.0	8.1.0
2009-06	SA-44	SP-090270	25	1	Clarification of the text on Hosting Party authentication in the conclusion section	8.0.0	8.1.0
2009-06	SA-44	SP-090270	24	-	Complement of section 7.2.2.3 H(e)NB Authentication	8.0.0	8.1.0
2009-06	SA-44	SP-090270	21	2	H(e)NB handover threat	8.0.0	8.1.0
2009-06	SA-44	SP-090270	2	2	Introducing semi-autonomous validation	8.0.0	8.1.0
2009-09	SA-45	SP-090520	40	2	CSG Requirements	8.1.0	8.2.0
2009-09	SA-45	SP-090520	36	-	Conclusions on Validation	8.1.0	8.2.0
2009-09	SA-45	SP-090520	30	1	Validation	8.1.0	8.2.0
2009-09	SA-45	SP-090520	37	-	H(e)NB validation	8.1.0	8.2.0
2009-09	SA-45	SP-090520	36	-	Device Re-validation	8.1.0	8.2.0
2009-09	SA-45	SP-090520	33	-	analysis of device integrity methods	8.1.0	8.2.0
2009-12	SA-46	SP-090815	42	2	Enhancements to SAV Description	8.2.0	8.3.0
2009-12	SA-46	SP-090815	43	1	H(e)NB Distress Indication	8.2.0	8.3.0
2009-12	SA-46	SP-090815	45	1	Storage of CSG Lists and CSG Type in the USIM	8.2.0	8.3.0
2009-12	SA-46	SP-090815	46	2	Analysis of Device Integrity Validation Methods	8.2.0	8.3.0
2009-12	SA-46	SP-090815	48	1	Hybrid Validation Questionnaire	8.2.0	8.3.0
2009-12	SA-46	SP-090815	50	2	Feasibility Study for Semi-Autonomous Validation	8.2.0	8.3.0