The 5G-AKA Authentication Protocol Privacy

Adrien Koutsos
LVS, ENS Paris-Saclay

January 18, 2019

The Authentication and Key Agreement Protocol

The Protocol

AKA is a key exchange protocol between:

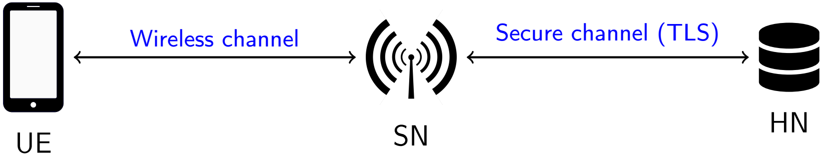
- The user equipment (UE): the mobile phone.
- The serving network (SN): the antenna.
- The home network (HN): the service provider (Free, Orange, SFR ...)

The Authentication and Key Agreement Protocol

The Protocol

AKA is a key exchange protocol between:

- The user equipment (UE): the mobile phone.
- The serving network (SN): the antenna.
- The home network (HN): the service provider (Free, Orange, SFR ...)



Security Goals

Some security goal of AKA

- Mutual authentication between the user (UE) and the network (HN).

Security Goals

Some security goal of AKA

- Mutual authentication between the user (UE) and the network (HN).
- Privacy properties:
  - Confidentiality of the user identity (ID).
  - Unlinkability of the user.

Security Goals

Some security goal of AKA

- Mutual authentication between the user (UE) and the network (HN).
- Privacy properties:
  - Confidentiality of the user identity (ID).
  - Unlinkability of the user.

Actually, there are other security goals

- Authentication of the antenna by the user.
- Authentication of the antenna by the network.
- Authentication of the user by the antenna.
- ...

Security Goals

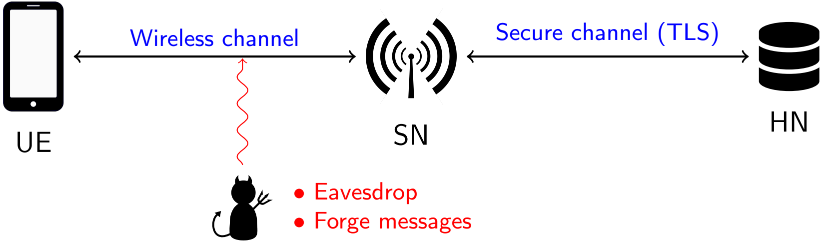
Some security goal of AKA

- Mutual authentication between the user (UE) and the network (HN).
- Privacy properties:
  - Confidentiality of the user identity (ID).
  - Unlinkability of the user.

Actually, there are other security goals

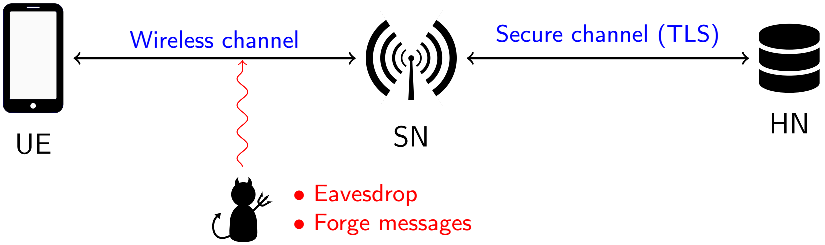
- Authentication of the antenna by the user.
- Authentication of the antenna by the network.
- Authentication of the user by the antenna.
- ...

Protocol Modeling



- Eavesdrop
- Forge messages

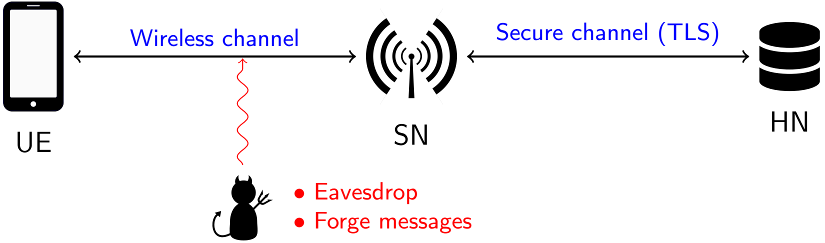
Protocol Modeling



We focus on:

- Mutual authentication between the user (UE) and the network (HN).
- Unlinkability of the user.

Protocol Modeling



We focus on:

- Mutual authentication between the user (UE) and the network (HN).
- Unlinkability of the user.

⟹ We do not model the antenna: we have a two party protocol.

Sequence Numbers

Pseudo Random Number Generation

- On the user side: all crypto primitives are computed in the SIM.
- Hardware PRNG is expensive/slow.

Sequence Numbers

Pseudo Random Number Generation

- On the user side: all crypto primitives are computed in the SIM.
- Hardware PRNG is expensive/slow.

$\Rightarrow$ In 4G-AKA, no PRNG on the mobile phone.

Sequence Numbers

Pseudo Random Number Generation

- On the user side: all crypto primitives are computed in the SIM.
- Hardware PRNG is expensive/slow.

⇒ In 4G-AKA, no PRNG on the mobile phone.

Cryptographic Primitives

- Asymmetric encryption requires randomness.

⇒ 4G-AKA uses only symmetric one-way functions.

Sequence Numbers

Authentication

Authentication protocols need to prevent message replays. In 4G-AKA:

Sequence Numbers

Authentication

Authentication protocols need to prevent message replays. In 4G-AKA:

- The antenna uses a random challenge.
- The mobile phone uses a sequence number SQN:

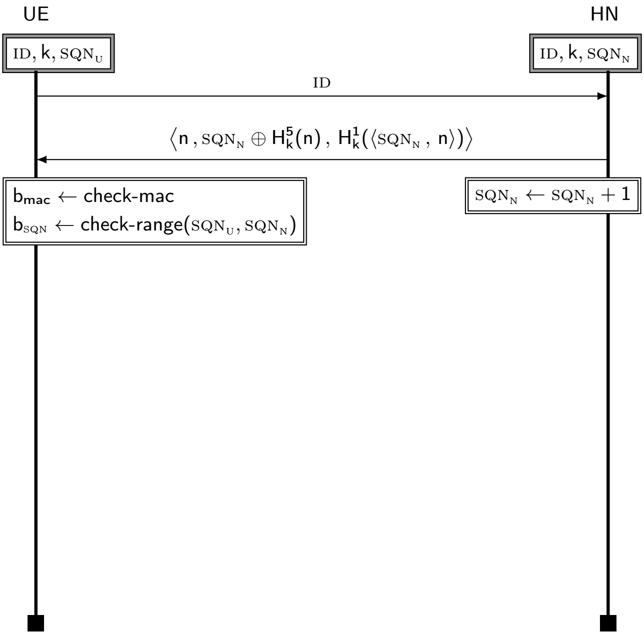
Sequence Numbers

Authentication

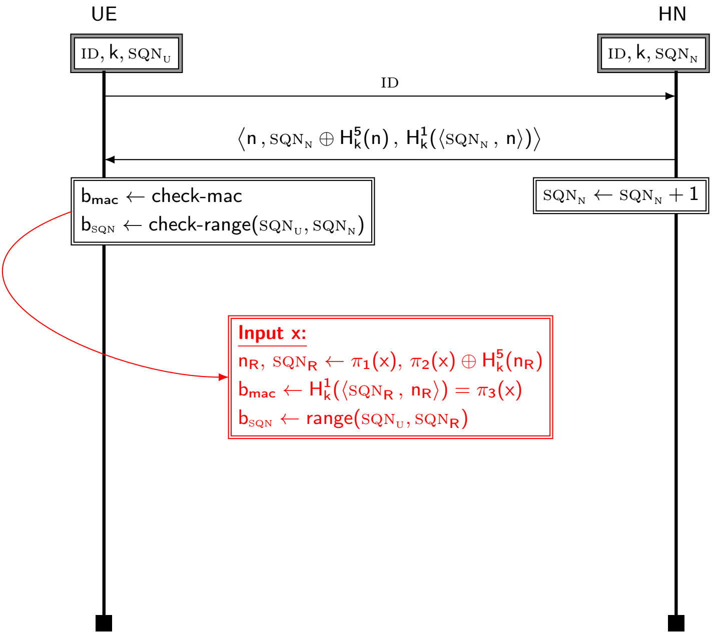
Authentication protocols need to prevent message replays. In 4G-AKA:

- The antenna uses a random challenge.
- The mobile phone uses a sequence number SQN:
  - Incremented after each successful session.
  - Tracked by the user and the antenna (SQN_U and SQN_N).
  - ⇒ De-synchronization possible.

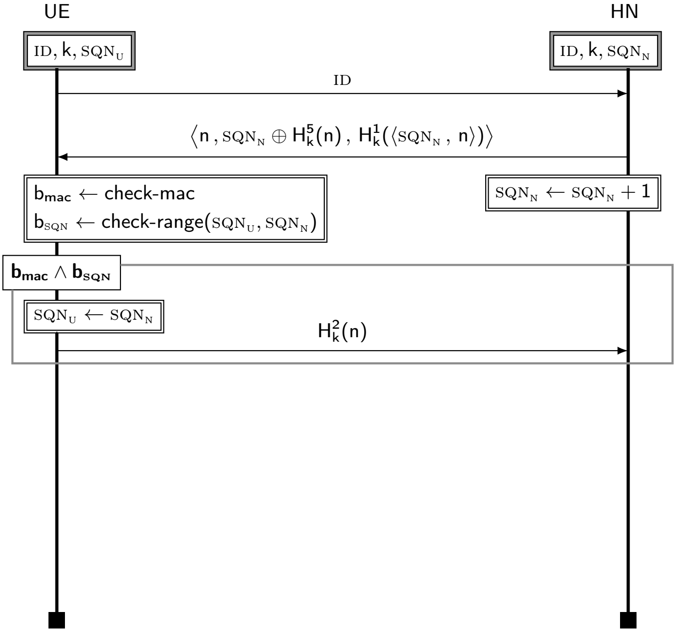
4G-AKA


UE
HN
ID, k, SQN_U
ID, k, SQN_N
ID
⟨n , SQN_N ⊕ H_k^5(n) , H_k^1(⟨SQN_N , n⟩)⟩
b_mac ← check-mac
b_SQN ← check-range(SQN_U, SQN_N)
SQN_N ← SQN_N + 1

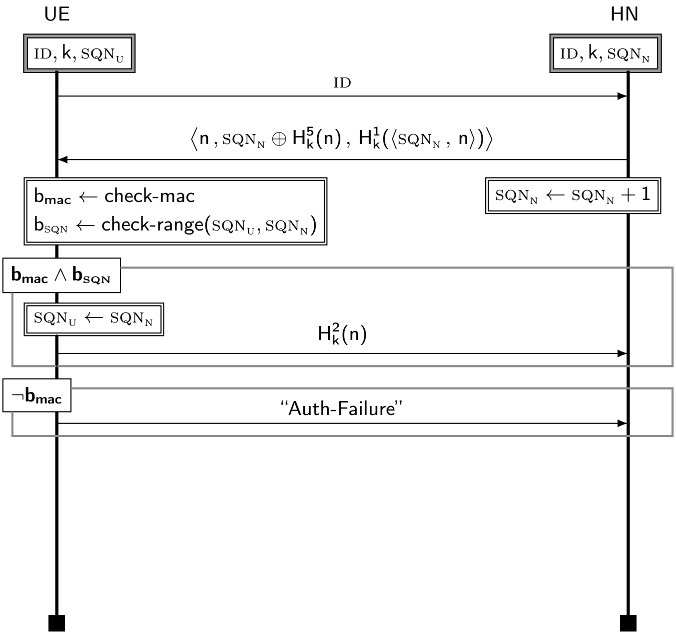
4G-AKA


UE
HN
ID, k, $\mathrm{SQN_U}$
ID, k, $\mathrm{SQN_N}$
ID
$\langle n, \mathrm{SQN_N} \oplus H^5_k(n), H^1_k(\langle \mathrm{SQN_N}, n\rangle)\rangle$
$b_{mac} \leftarrow$ check-mac
$b_{\mathrm{SQN}} \leftarrow$ check-range$(\mathrm{SQN_U}, \mathrm{SQN_N})$
$\mathrm{SQN_N} \leftarrow \mathrm{SQN_N} + 1$
Input x:
$n_R, \mathrm{SQN_R} \leftarrow \pi_1(x), \pi_2(x) \oplus H^5_k(n_R)$
$b_{mac} \leftarrow H^1_k(\langle \mathrm{SQN_R}, n_R\rangle) = \pi_3(x)$
$b_{\mathrm{SQN}} \leftarrow$ range$(\mathrm{SQN_U}, \mathrm{SQN_R})$

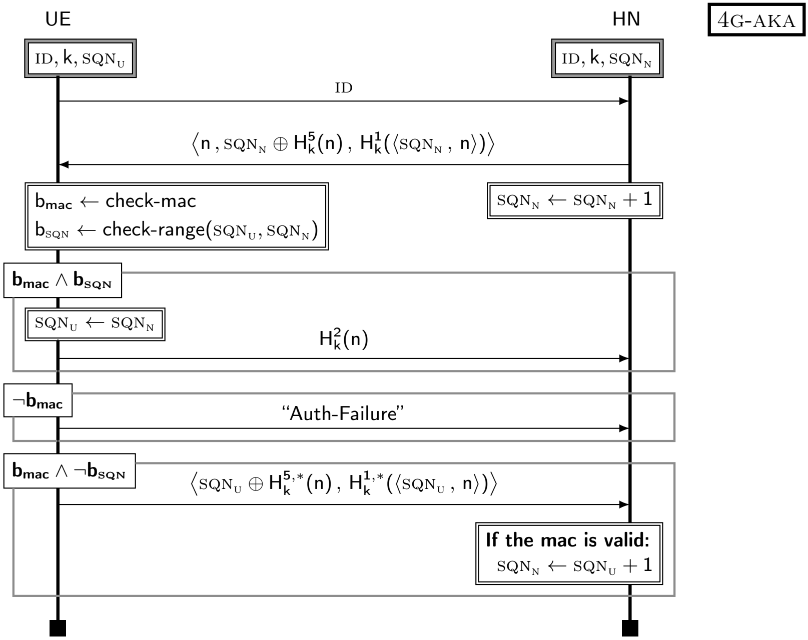
4G-AKA


UE
HN
ID, k, SQN_U
ID, k, SQN_N
ID
⟨n , SQN_N ⊕ H_k^5(n) , H_k^1(⟨SQN_N , n⟩)⟩
b_mac ← check-mac
b_SQN ← check-range(SQN_U, SQN_N)
SQN_N ← SQN_N + 1
b_mac ∧ b_SQN
SQN_U ← SQN_N
H_k^2(n)

4G-AKA



UE
HN
ID, k, SQN_U
ID, k, SQN_N
ID
$\langle n, SQN_N \oplus H^5_k(n), H^1_k(\langle SQN_N, n\rangle)\rangle$
$b_{mac} \leftarrow$ check-mac
$b_{SQN} \leftarrow$ check-range(SQN_U, SQN_N)
$SQN_N \leftarrow SQN_N + 1$
$b_{mac} \wedge b_{SQN}$
$SQN_U \leftarrow SQN_N$
$H^2_k(n)$
$\neg b_{mac}$
"Auth-Failure"


4G-AKA
UE
HN
ID, k, SQN_U
ID, k, SQN_N
ID
⟨n , SQN_N ⊕ H^5_k(n) , H^1_k(⟨SQN_N , n⟩)⟩
b_mac ← check-mac
b_SQN ← check-range(SQN_U, SQN_N)
SQN_N ← SQN_N + 1
b_mac ∧ b_SQN
SQN_U ← SQN_N
H^2_k(n)
¬b_mac
"Auth-Failure"
b_mac ∧ ¬b_SQN
⟨SQN_U ⊕ H^{5,*}_k(n) , H^{1,*}_k(⟨SQN_U , n⟩)⟩
If the mac is valid:
SQN_N ← SQN_U + 1

Privacy in 4G-AKA

Not confidentiality of the user identity

The ID is sent in plain text!

Privacy in 4G-AKA

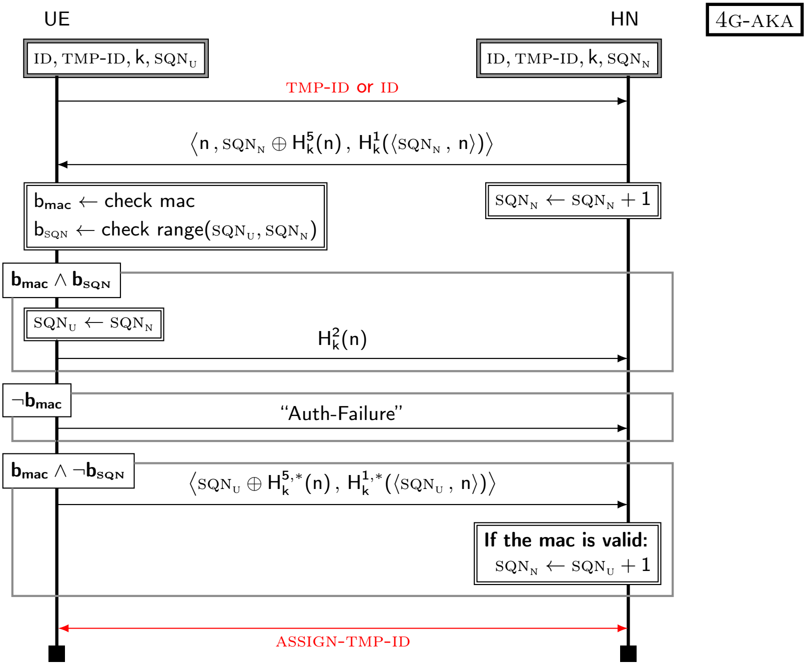
Not confidentiality of the user identity

The ID is sent in plain text!

4G-AKA solution

Use a temporary identity TMP-ID instead of the permanent identity ID:

- The network has a mapping from TMP-IDs to IDs.
- Each TMP-ID should be used at most once.
- The network assigns new TMP-ID after each successful session.


4G-AKA
UE
HN
ID, TMP-ID, k, SQN_U
ID, TMP-ID, k, SQN_N
TMP-ID or ID
⟨n , SQN_N ⊕ H^5_k(n) , H^1_k(⟨SQN_N , n⟩)⟩
b_mac ← check mac
b_SQN ← check range(SQN_U, SQN_N)
SQN_N ← SQN_N + 1
b_mac ∧ b_SQN
SQN_U ← SQN_N
H^2_k(n)
¬b_mac
"Auth-Failure"
b_mac ∧ ¬b_SQN
⟨SQN_U ⊕ H^{5,*}_k(n) , H^{1,*}_k(⟨SQN_U , n⟩)⟩
If the mac is valid:
SQN_N ← SQN_U + 1
ASSIGN-TMP-ID

Privacy in 4G-AKA

Confidentiality of the user identity

Once a temporary identity is set up, the ID is protected if:

- The protocol does not fail.
- The adversary is a passive adversary.

Privacy in 4G-AKA

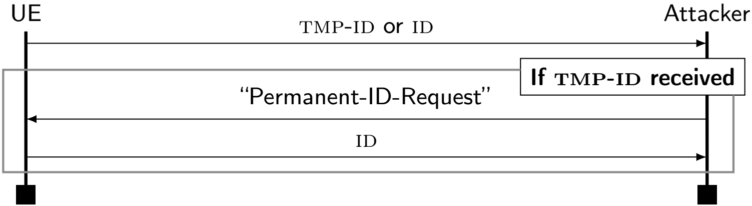
Confidentiality of the user identity

Once a temporary identity is set up, the ID is protected if:

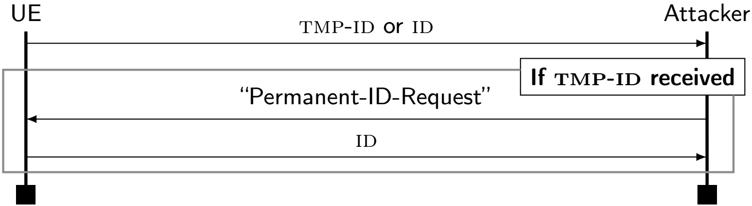
- The protocol does not fail.
- The adversary is a passive adversary.

$\Longrightarrow$ This is not realistic!

The IMSI Catcher Attack [Strobel, 2007]



The IMSI Catcher Attack [Strobel, 2007]



Why this is a major attack

- Reliable: the attack always works.
- Easy to deploy: only need an antenna.
- Large scale: not targeted.

Privacy in 5G-AKA

The 5G-AKA protocol

5G-AKA is the next version of AKA (drafts are available [3GPP, 2018]).

Privacy in 5G-AKA

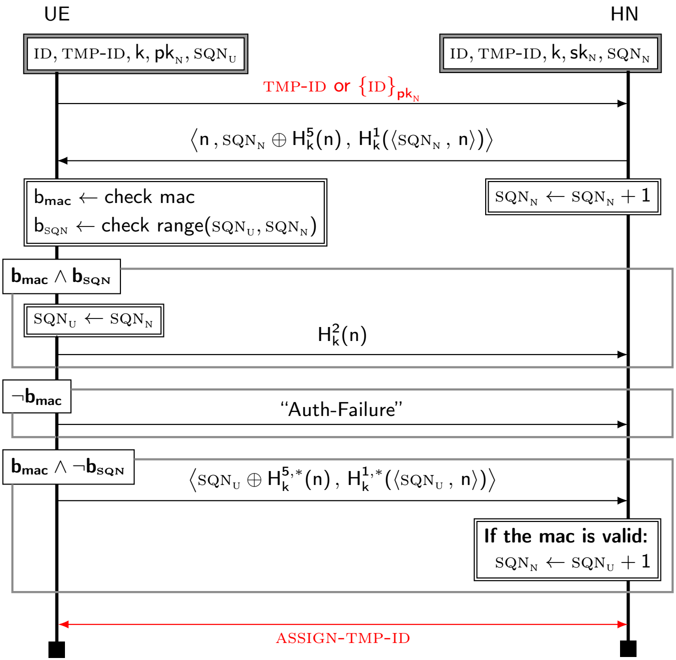
The 5G-AKA protocol

5G-AKA is the next version of AKA (drafts are available [3GPP, 2018]).

3GPP fix for 5G-AKA

Simply encrypt the permanent identity by sending $\{\mathrm{ID}\}_{\mathsf{pk}_{\mathrm{N}}}$

5G-AKA


UE
HN
ID, TMP-ID, k, pk$_N$, SQN$_U$
ID, TMP-ID, k, sk$_N$, SQN$_N$
TMP-ID or $\{ID\}_{pk_N}$
$\langle n, SQN_N \oplus H^5_k(n), H^1_k(\langle SQN_N, n\rangle)\rangle$
$b_{mac} \leftarrow$ check mac
$b_{SQN} \leftarrow$ check range(SQN_U, SQN_N)
$SQN_N \leftarrow SQN_N + 1$
$b_{mac} \wedge b_{SQN}$
$SQN_U \leftarrow SQN_N$
$H^2_k(n)$
$\neg b_{mac}$
"Auth-Failure"
$b_{mac} \wedge \neg b_{SQN}$
$\langle SQN_U \oplus H^{5,*}_k(n), H^{1,*}_k(\langle SQN_U, n\rangle)\rangle$
If the mac is valid:
$SQN_N \leftarrow SQN_U + 1$
ASSIGN-TMP-ID

Privacy in 5G-AKA

Is it enough?

Privacy in 5G-AKA

Is it enough?

For confidentiality of the ID, yes.

Privacy in 5G-AKA

Is it enough?

For confidentiality of the ID, yes.

For unlinkability, no.

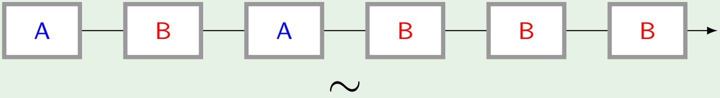
Unlinkability

Linkability Attack

Even if the ID is hidden, an attacker may link sessions of the same user.

Unlinkability

Example

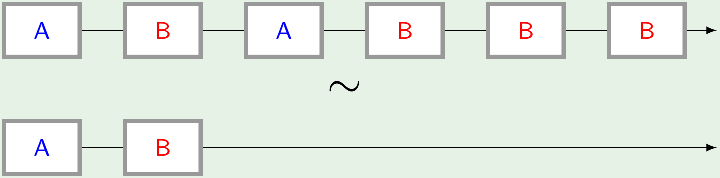


Linkability Attack

Even if the ID is hidden, an attacker may link sessions of the same user.

Unlinkability

Example

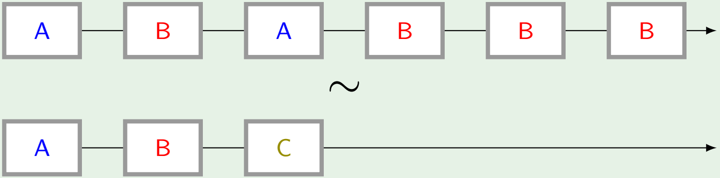


Linkability Attack

Even if the ID is hidden, an attacker may link sessions of the same user.

Unlinkability

Example

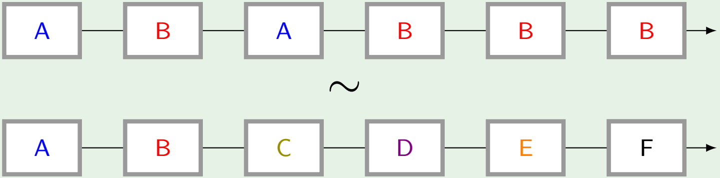


Linkability Attack

Even if the ID is hidden, an attacker may link sessions of the same user.

Unlinkability

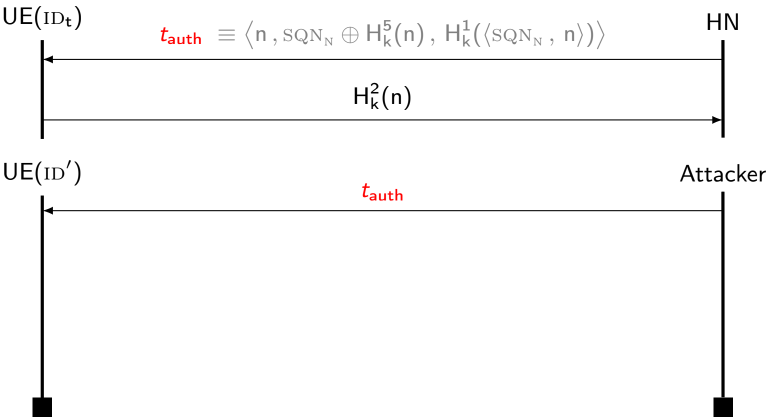
Example



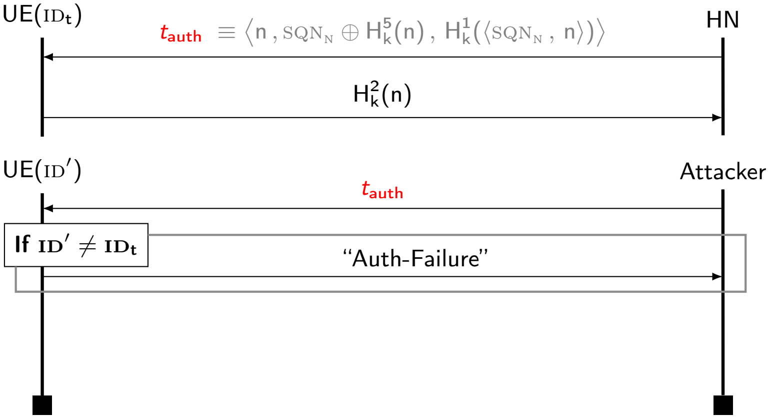
Linkability Attack

Even if the ID is hidden, an attacker may link sessions of the same user.

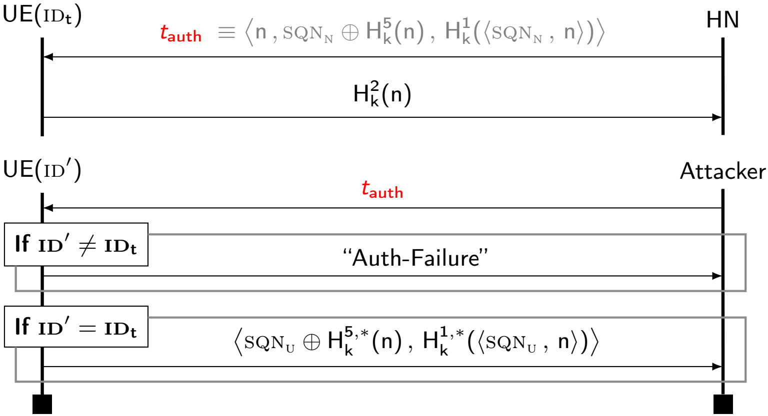
The Failure Message Attack [Arapinis et al., 2012]



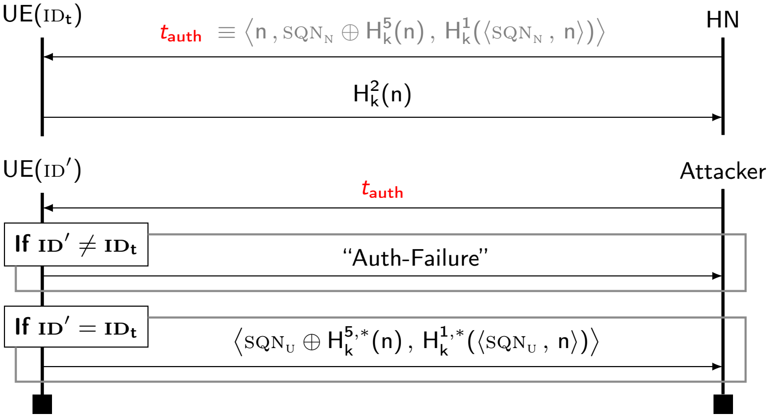
The Failure Message Attack [Arapinis et al., 2012]



The Failure Message Attack [Arapinis et al., 2012]



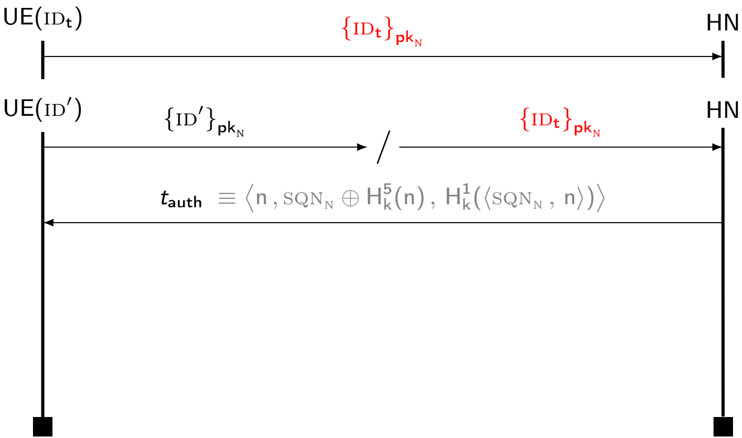
The Failure Message Attack [Arapinis et al., 2012]



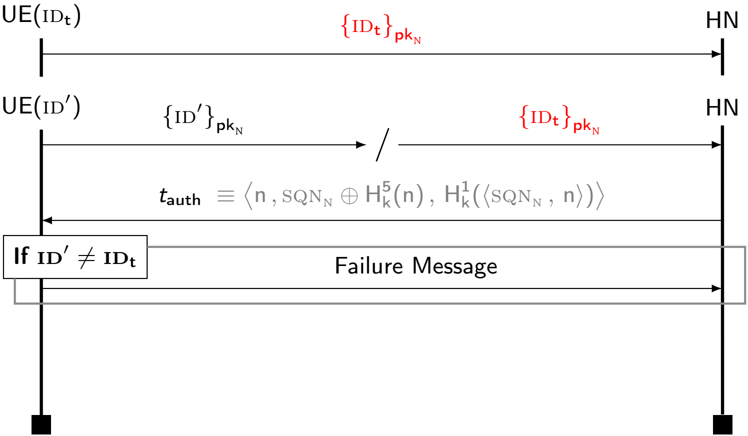
Unlinkability attack

The adversary knows if it interacted with ID$_t$ or ID$'$.

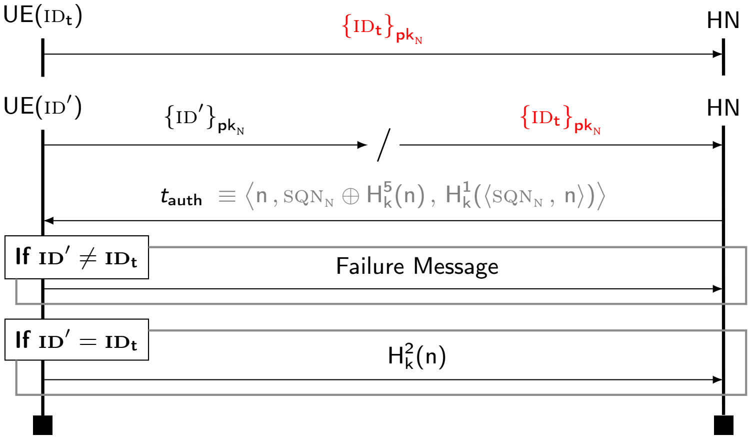
The Encrypted ID Replay Attack [Fouque et al., 2016]



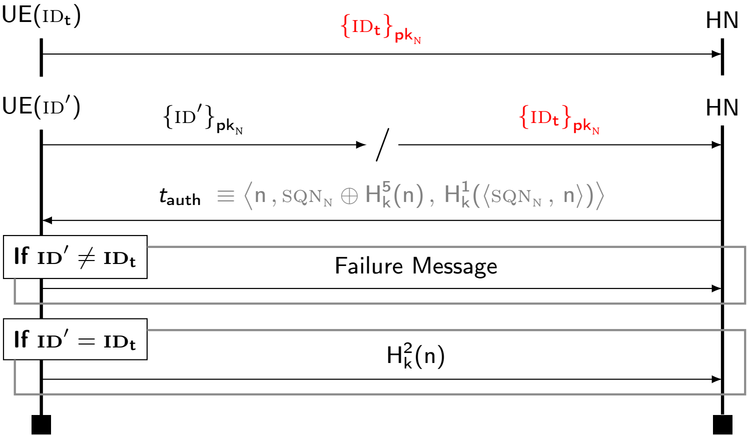
The Encrypted ID Replay Attack [Fouque et al., 2016]



The Encrypted ID Replay Attack [Fouque et al., 2016]



The Encrypted ID Replay Attack [Fouque et al., 2016]



Unlinkability attack

The adversary knows if it interacted with $\mathrm{ID_t}$ or ID'.

New Attack on the PRIV-AKA Protocol

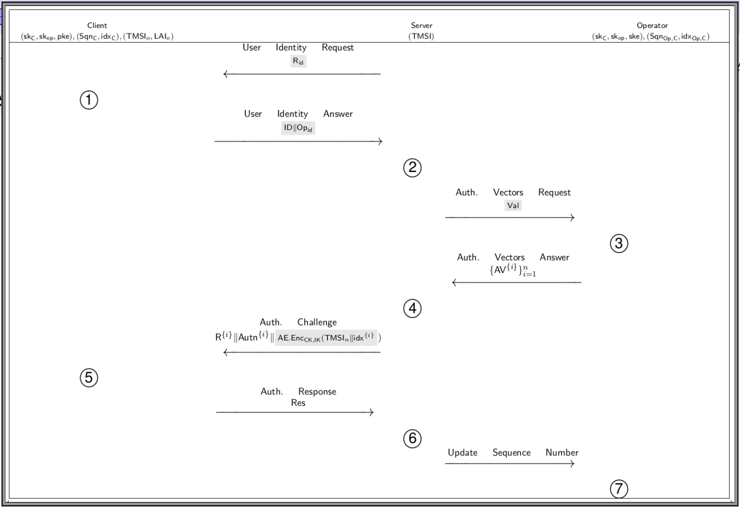
The PRIV-AKA Protocol

The authors of [Fouque et al., 2016] propose a new protocol, PRIV-AKA (claimed unlinkable).

New Attack on the PRIV-AKA Protocol

The PR

The au ... AKA
(claime



New Attack on the PRIV-AKA Protocol

The PRIV-AKA Protocol

The authors of [Fouque et al., 2016] propose a new protocol, PRIV-AKA (claimed unlinkable).

Unlinkability Attack (four sessions)

We found an attack to permanently de-synchronize the user:

- Run a session but keep the last message t_1.
- Re-synchronize the user and the network.

New Attack on the PRIV-AKA Protocol

The PRIV-AKA Protocol

The authors of [Fouque et al., 2016] propose a new protocol, PRIV-AKA (claimed unlinkable).

Unlinkability Attack (four sessions)

We found an attack to permanently de-synchronize the user:

- Run a session but keep the last message t_1.
- Re-synchronize the user and the network.
- Re-iterate the last two steps to get a second message t_2.

New Attack on the PRIV-AKA Protocol

The PRIV-AKA Protocol

The authors of [Fouque et al., 2016] propose a new protocol, PRIV-AKA (claimed unlinkable).

Unlinkability Attack (four sessions)

We found an attack to permanently de-synchronize the user:

- Run a session but keep the last message t_1.
- Re-synchronize the user and the network.
- Re-iterate the last two steps to get a second message t_2.
- Send both t_1 and t_2, which increments $\mathrm{SQN}_{\mathrm{N}}$ by **two**.

New Attack on the PRIV-AKA Protocol

The PRIV-AKA Protocol

The authors of [Fouque et al., 2016] propose a new protocol, PRIV-AKA (claimed unlinkable).

Unlinkability Attack (four sessions)

We found an attack to permanently de-synchronize the user:

- Run a session but keep the last message t_1.
- Re-synchronize the user and the network.
- Re-iterate the last two steps to get a second message t_2.
- Send both t_1 and t_2, which increments $\mathrm{SQN_N}$ by **two**.
- The user is permanently de-synchronized $\Longrightarrow$ unlinkability attack.

Objective

Objective

Design a modified version of AKA, called AKA$^+$, such that:

- Provides some form of unlinkability.

Objective

Objective

Design a modified version of AKA, called AKA^+, such that:

- Provides some form of unlinkability.
- Satisfies the design and efficiency constraints of 5G-AKA.

Objective

Objective

Design a modified version of AKA, called AKA^{+}, such that:

- Provides some form of unlinkability.
- Satisfies the design and efficiency constraints of 5G-AKA.
- Is proved secure.

Random Number Generation in 5G-AKA

Random Number Generation by the User

In 5G-AKA, the user generates a random number only:

- If no TMP-ID is assigned.
- In the session following a de-synchronization.

The AKA$^+$ Protocol

Design Constraints

AKA$^+$ should be as efficient as the 5G-AKA:

- Random number generation (user): at most one nonce per session, and only for re-synchronization or if no TMP-ID is assigned.

The AKA$^+$ Protocol

Design Constraints

AKA$^+$ should be as efficient as the 5G-AKA:

- Random number generation (user): at most one nonce per session, and only for re-synchronization or if no TMP-ID is assigned.
- The user can use only one-way functions and asymmetric *encryption*.

The AKA$^+$ Protocol

Design Constraints

AKA$^+$ should be as efficient as the 5G-AKA:

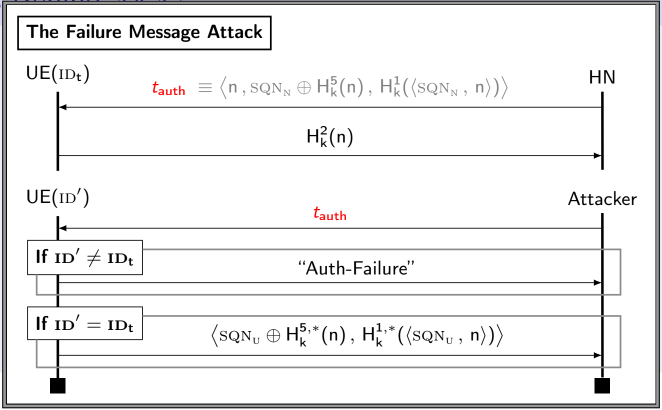
- Random number generation (user): at most one nonce per session, and only for re-synchronization or if no TMP-ID is assigned.
- The user can use only one-way functions and asymmetric *encryption*.
- Network complexity: only three messages per session.

Key Ideas

Key Ideas Behind AKA^{+}

Key Ideas

Key Ideas Behind AKA$^+$



Key Ideas

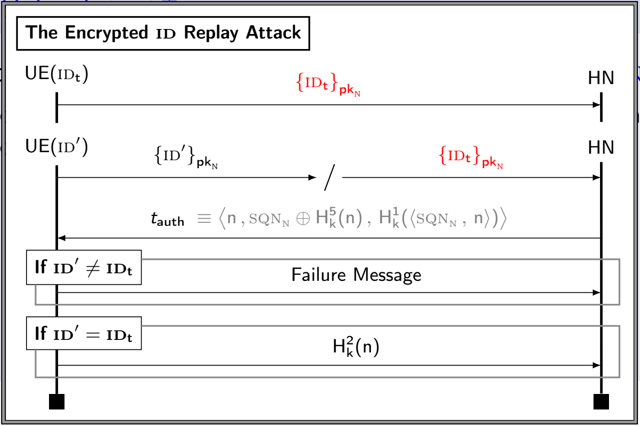
Key Ideas Behind AKA^+

- Postpone re-synchronization to the next session: $\{\langle \text{ID}, \text{SQN}_{\text{U}} \rangle\}_{\mathsf{pk}_{\text{N}}}$.
  - No re-synchronization message $\Longrightarrow$ no failure message attack.
  - No extra randomness for the user.

Key Ideas

Key Ideas

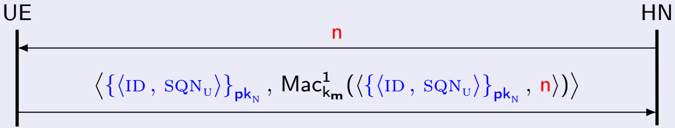
- Postpo … $_{\mathrm{U}}\rangle\}_{\mathsf{pk_N}}$.
  - N … ck.
  - N



Key Ideas

Key Ideas Behind AKA^+

- Postpone re-synchronization to the next session: $\{\langle \text{ID}, \text{SQN}_{\text{U}} \rangle\}_{\mathsf{pk}_{\text{N}}}$.
  - No re-synchronization message $\Longrightarrow$ no failure message attack.
  - No extra randomness for the user.
- Add a challenge n from the HN when using the permanent identity.



Architecture of AKA$^+$

AKA$^+$ Sub-Protocols

- ID sub-protocol:
  - is initiated by the HN with a challenge n.
  - uses the encrypted permanent identity.
  - allows to re-synchronize the UE and the HN.

ID Sub-Protocol

Architecture of AKA$^+$

AKA$^+$ Sub-Protocols

- ID sub-protocol:
  - is initiated by the HN with a challenge n.
  - uses the encrypted permanent identity.
  - allows to re-synchronize the UE and the HN.
- TMP-ID sub-protocol:
  - is initiated by the UE.
  - uses a temporary identity.

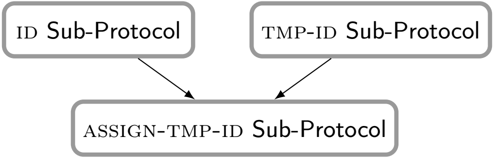
ID Sub-Protocol

TMP-ID Sub-Protocol

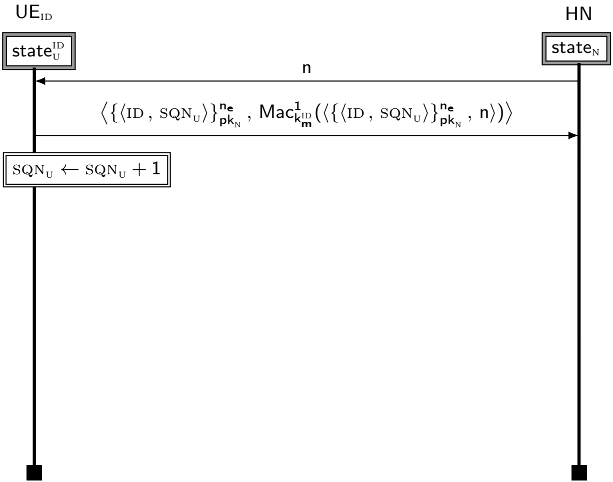
Architecture of AKA$^+$

AKA$^+$ Sub-Protocols

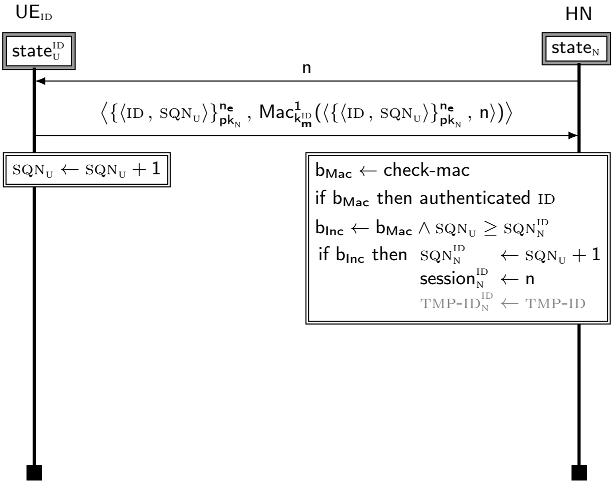
- ID sub-protocol:
  - is initiated by the HN with a challenge n.
  - uses the encrypted permanent identity.
  - allows to re-synchronize the UE and the HN.
- TMP-ID sub-protocol:
  - is initiated by the UE.
  - uses a temporary identity.
- ASSIGN-TMP-ID sub-protocol:
  - assigns a fresh temporary identity to the UE.



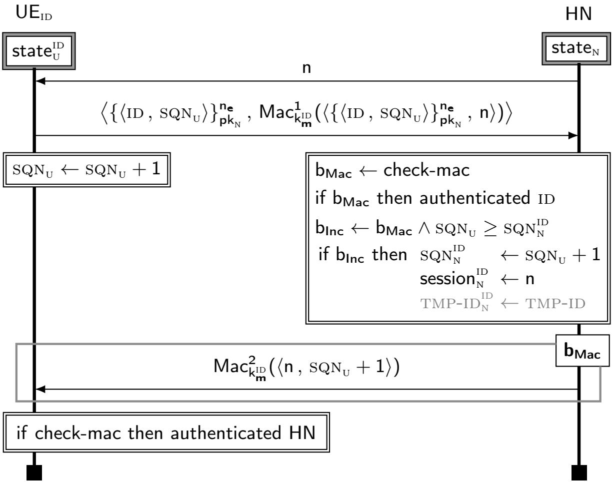
ID Sub-Protocol



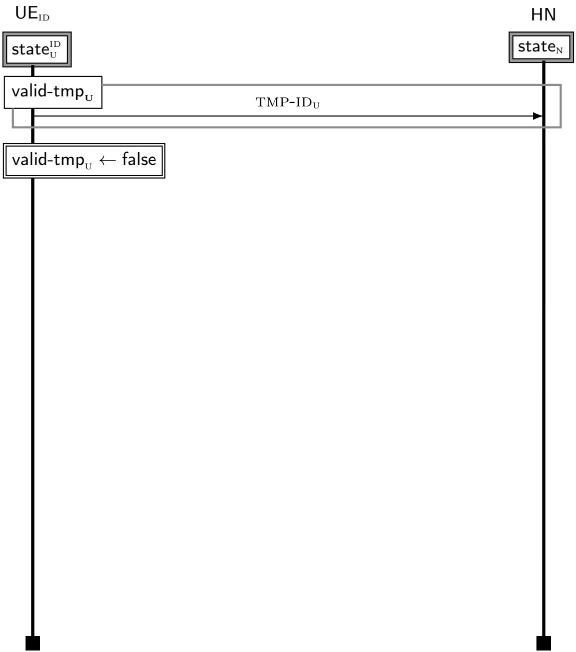
ID
Sub-Protocol



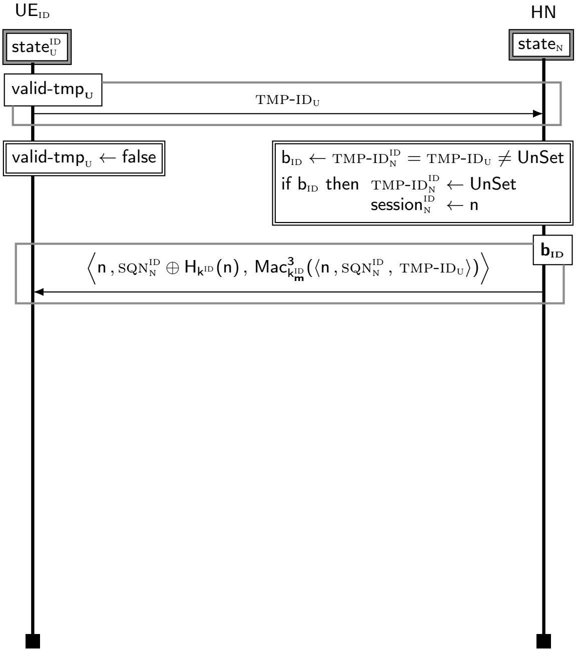
ID Sub-Protocol



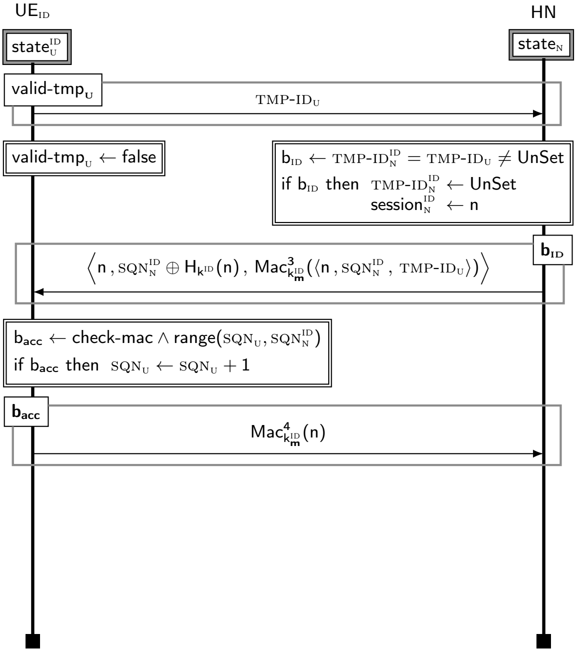
TMP-ID Sub-Protocol



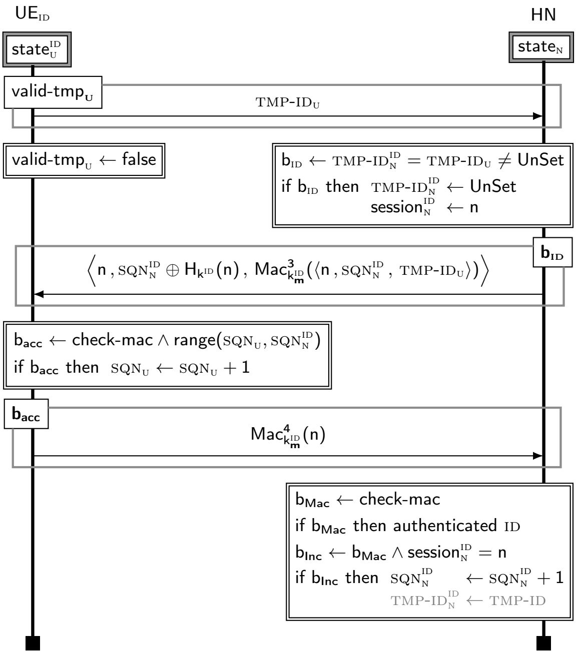
TMP-ID Sub-Protocol



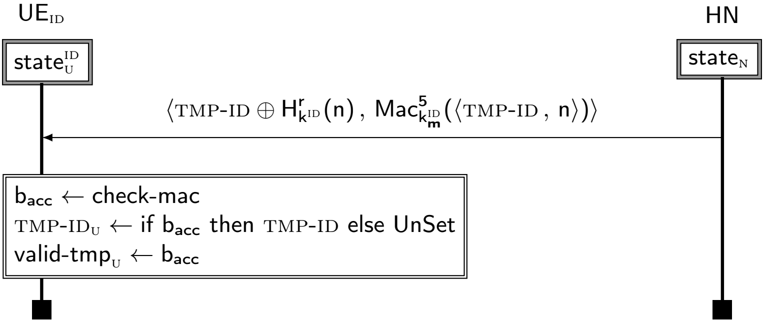
TMP-ID Sub-Protocol



TMP-ID Sub-Protocol



The ASSIGN-TMP-ID Sub-Protocol



Security Proofs

Objective

Formally prove that AKA^+ satisfies:

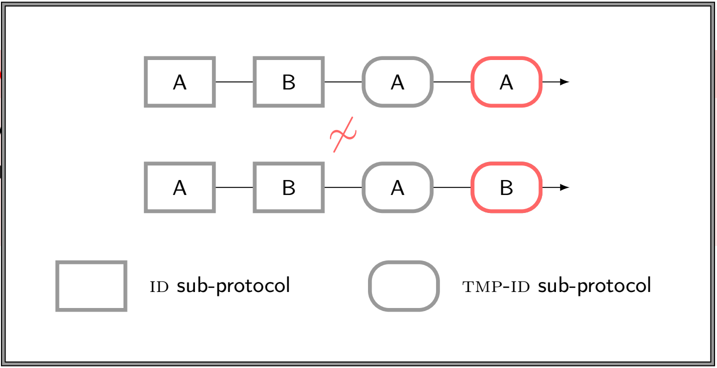
- **mutual authentication**.
- **unlinkability**.

Security Proofs

Objecti

Formally

- mu
- unl



Security Proofs

Objective

Formally prove that AKA^+ satisfies:

- **mutual authentication**.
- ~~**unlinkability**~~ $\Longrightarrow$ σ-**unlinkability**.

The σ-Unlinkability Property

σ-Unlinkability

High level idea: show privacy only for a subset of the standard unlinkability game scenarios.

The σ-Unlinkability Property

σ-Unlinkability

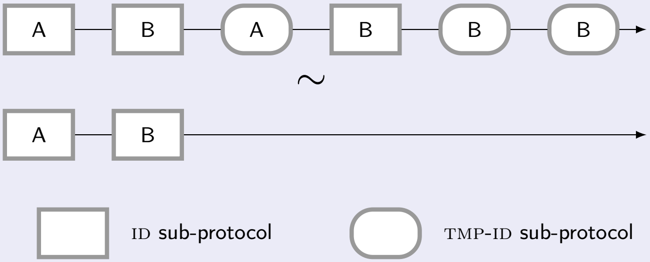
High level idea: show privacy only for a subset of the standard unlinkability game scenarios.

- Game-based definition (like standard unlinkability).
- Parametric property (σ).
- In general, weaker than unlinkability.
- Allow to precisely quantify privacy guarantees.

The σ-Unlinkability Property

Two Indistinguishable Executions

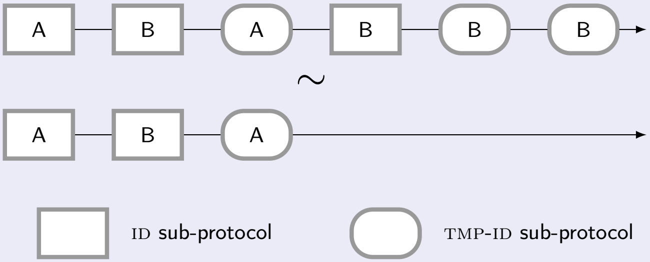
Each time the ID sub-protocol is used, we can change the user's identity.



The σ-Unlinkability Property

Two Indistinguishable Executions

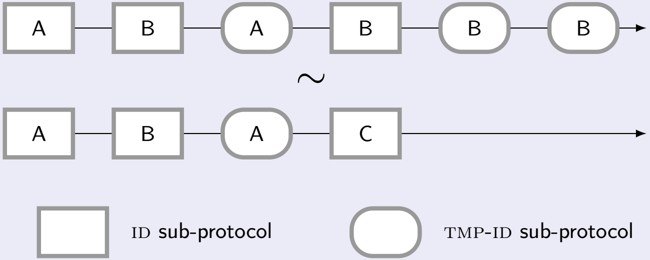
Each time the ID sub-protocol is used, we can change the user's identity.



The σ-Unlinkability Property

Two Indistinguishable Executions

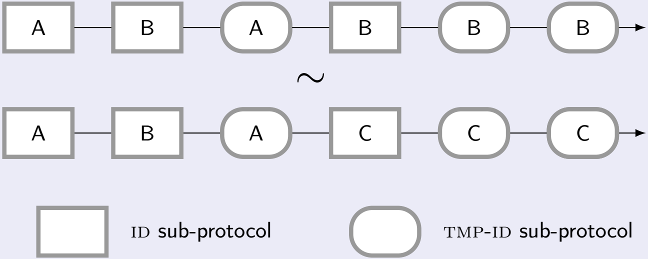
Each time the ID sub-protocol is used, we can change the user's identity.



The σ-Unlinkability Property

Two Indistinguishable Executions

Each time the ID sub-protocol is used, we can change the user's identity.



σ-Unlinkability

Efficiency vs Privacy

There is a trade-off between:

- **Efficiency:** the TMP-ID sub-protocol is faster.
- **Privacy:** the ID sub-protocol provides some privacy.

σ-Unlinkability

Efficiency vs Privacy

There is a trade-off between:

- **Efficiency:** the TMP-ID sub-protocol is faster.
- **Privacy:** the ID sub-protocol provides some privacy.

Remark

- If we use only the ID sub-protocol, we get standard unlinkability.
- All previous attacks are also σ-unlinkability attacks.

Modeling

The Bana-Comon Model [Bana and Comon-Lundh, 2014]

The proof is in the Bana-Comon unlinkability model:

- Messages are modeled by (first-order) **terms**.

Modeling

The Bana-Comon Model [Bana and Comon-Lundh, 2014]

The proof is in the Bana-Comon unlinkability model:

- Messages are modeled by (first-order) **terms**.
- A **security property** $P \sim Q$ is modeled by a **formula** $\vec{u}_P \sim \vec{u}_Q$.

Modeling

The Bana-Comon Model [Bana and Comon-Lundh, 2014]

The proof is in the Bana-Comon unlinkability model:

- Messages are modeled by (first-order) **terms**.
- A **security property** $P \sim Q$ is modeled by a **formula** $\vec{u}_P \sim \vec{u}_Q$.
- **Implementation assumptions** and **cryptographic hypothesis** are modeled by axioms Ax.

Modeling

The Bana-Comon Model [Bana and Comon-Lundh, 2014]

The proof is in the Bana-Comon unlinkability model:

- Messages are modeled by (first-order) **terms**.
- A **security property** $P \sim Q$ is modeled by a **formula** $\vec{u}_P \sim \vec{u}_Q$.
- **Implementation assumptions** and **cryptographic hypothesis** are modeled by axioms Ax.
- We have to show that $\mathsf{Ax} \models \vec{u}_P \sim \vec{u}_Q$.

Modeling: the Protocol

Messages and State

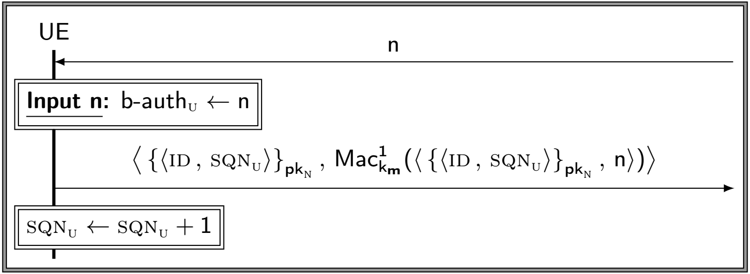
- **Symbolic trace** of actions τ.
  Example: $\tau = \mathsf{UE}_A, \mathsf{HN}, \mathsf{UE}_B, \mathsf{UE}_A$.

Modeling: the Protocol

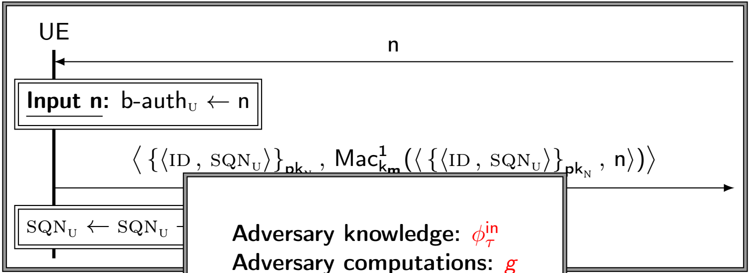
Messages and State

- **Symbolic trace** of actions τ.
  Example: $\tau = \mathsf{UE}_A, \mathsf{HN}, \mathsf{UE}_B, \mathsf{UE}_A$.
- **Symbolic frame** ϕ_τ: sequences of messages observed by the attacker.
- **Symbolic state** σ_τ: current state of the users and the network.

Modeling: the Protocol



Modeling: the Protocol

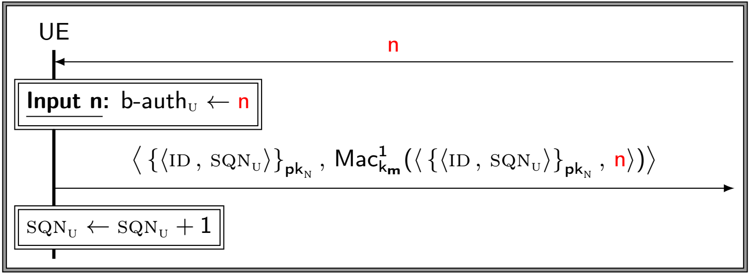


Adversary knowledge: ϕ_τ^{in}

Adversary computations: g

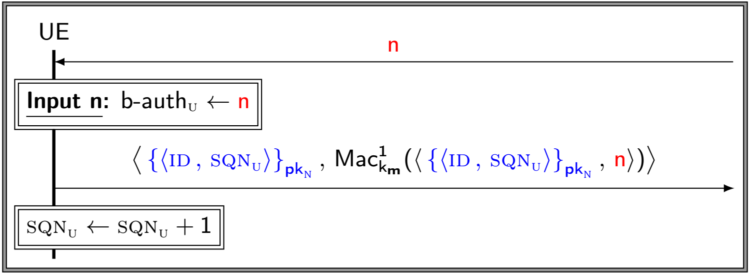
$\Longrightarrow$ **Symbolic input:** $g(\phi_\tau^{in})$

Modeling: the Protocol



$$\sigma_\tau^{\mathsf{up}} \equiv \begin{cases} \\ \mathsf{b\text{-}auth}_{\mathrm{U}} \mapsto g(\phi_\tau^{\mathsf{in}}) \end{cases}$$

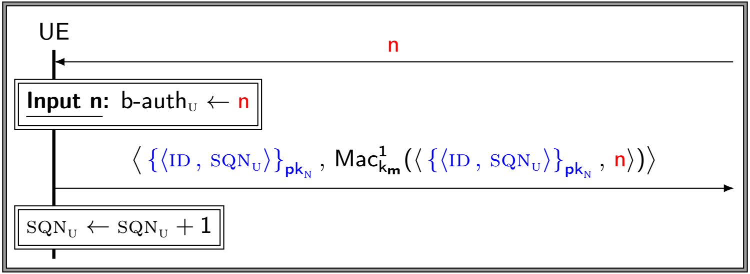
Modeling: the Protocol



$$t_\tau^{\mathsf{enc}} \equiv \{\langle \mathrm{ID}, \sigma_\tau^{\mathsf{in}}(\mathrm{SQN_U}) \rangle\}_{\mathsf{pk_N}}^{\mathsf{n_e}}$$

$$\sigma_\tau^{\mathsf{up}} \equiv \begin{cases} \\ \mathsf{b\text{-}auth_U} \mapsto g(\phi_\tau^{\mathsf{in}}) \end{cases}$$

Modeling: the Protocol

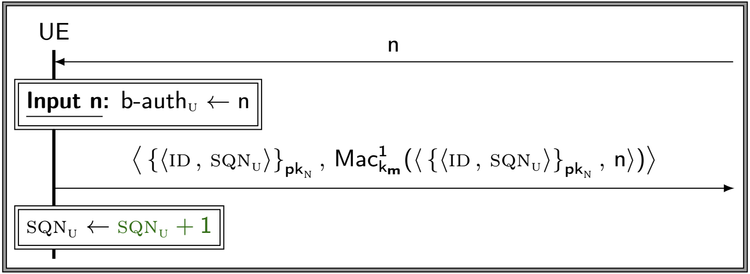


$$t_\tau^{\mathsf{enc}} \equiv \{\langle \mathrm{ID} , \sigma_\tau^{\mathsf{in}}(\mathrm{SQN}_{\mathrm{U}}) \rangle\}_{\mathsf{pk}_{\mathrm{N}}}^{\mathsf{n_e}}$$

$$\phi_\tau \equiv \phi_\tau^{\mathsf{in}}, \left\langle t_\tau^{\mathsf{enc}} , \mathsf{Mac}^1_{\mathsf{k}_\mathsf{m}^{\mathrm{ID}}}(\langle t_\tau^{\mathsf{enc}} , g(\phi_\tau^{\mathsf{in}}) \rangle) \right\rangle$$

$$\sigma_\tau^{\mathsf{up}} \equiv \begin{cases} \\ \mathsf{b\text{-}auth}_{\mathrm{U}} \mapsto g(\phi_\tau^{\mathsf{in}}) \end{cases}$$

Modeling: the Protocol



$$t^{\mathsf{enc}}_{\tau} \equiv \{\langle \mathrm{ID}, \sigma^{\mathsf{in}}_{\tau}(\mathrm{SQN_U})\rangle\}^{\mathsf{n_e}}_{\mathsf{pk_N}}$$

$$\phi_{\tau} \equiv \phi^{\mathsf{in}}_{\tau}, \langle t^{\mathsf{enc}}_{\tau}, \mathsf{Mac}^1_{\mathsf{k^{ID}_m}}(\langle t^{\mathsf{enc}}_{\tau}, g(\phi^{\mathsf{in}}_{\tau})\rangle)\rangle$$

$$\sigma^{\mathsf{up}}_{\tau} \equiv \begin{cases} \mathrm{SQN_U} \mapsto \mathsf{suc}(\sigma^{\mathsf{in}}_{\tau}(\mathrm{SQN^{ID}_U})) \\ \text{b-auth}_{\mathrm{U}} \mapsto g(\phi^{\mathsf{in}}_{\tau}) \end{cases}$$

$$\sigma_{\tau} \equiv \sigma^{\mathsf{in}}_{\tau} \cdot \sigma^{\mathsf{up}}_{\tau}$$

Base Axioms

Proposition: Mac Unforgeability

If Mac is an EUF-MAC function, then the following axiom is valid:

$$\overline{\mathsf{verify}_{\mathsf{k_m}}(s, m) \rightarrow \bigvee_{u \in \mathcal{S}} s = \mathsf{Mac}_{\mathsf{k_m}}(u)} \qquad (\text{EUF-MAC})$$

Base Axioms

Proposition: Mac Unforgeability

If Mac is an EUF-MAC function, then the following axiom is valid:

$$\overline{\mathsf{verify}_{\mathsf{k_m}}(s, m) \rightarrow \bigvee_{u \in \mathcal{S}} s = \mathsf{Mac}_{\mathsf{k_m}}(u)} \qquad (\text{EUF-MAC})$$

Where:

- $\mathcal{S}$ is the set of subterms of s, m of the form $\mathsf{Mac}_{\mathsf{k_m}}(\_)$.
- $\mathsf{k_m}$ appears only in Mac key position in s, m.

Base Axioms

Proposition: Mac Unforgeability

If Mac is an EUF-MAC function, then the following axiom is valid:

$$\overline{\mathsf{verify}_{\mathsf{k_m}}(s, m) \rightarrow \bigvee_{u \in \mathcal{S}} s = \mathsf{Mac}_{\mathsf{k_m}}(u)} \qquad (\text{EUF-MAC})$$

Where:

- $\mathcal{S}$ is the set of subterms of s, m of the form $\mathsf{Mac}_{\mathsf{k_m}}(\_)$.
- $\mathsf{k_m}$ appears only in Mac key position in s, m.

Example

$$\phi \equiv \mathsf{Mac}_{\mathsf{k_m}}(t_1), \mathsf{Mac}_{\mathsf{k_m}}(t_2), \mathsf{Mac}_{\mathsf{k'_m}}(t_3)$$

$$\mathsf{verify}_{\mathsf{k_m}}(g(\phi), \mathsf{n}) \ \rightarrow$$

Base Axioms

Proposition: Mac Unforgeability

If Mac is an EUF-MAC function, then the following axiom is valid:

$$\frac{}{\mathsf{verify}_{\mathsf{k_m}}(s, m) \to \bigvee_{u \in \mathcal{S}} s = \mathsf{Mac}_{\mathsf{k_m}}(u)} \quad (\text{EUF-MAC})$$

Where:

- $\mathcal{S}$ is the set of subterms of s, m of the form $\mathsf{Mac}_{\mathsf{k_m}}(\_)$.
- $\mathsf{k_m}$ appears only in Mac key position in s, m.

Example

$$\phi \equiv \mathsf{Mac}_{\mathsf{k_m}}(t_1), \mathsf{Mac}_{\mathsf{k_m}}(t_2), \mathsf{Mac}_{\mathsf{k'_m}}(t_3)$$

$$\mathsf{verify}_{\mathsf{k_m}}(g(\phi), \mathsf{n}) \ \to \ (g(\phi) = \mathsf{Mac}_{\mathsf{k_m}}(t_1) \vee g(\phi) = \mathsf{Mac}_{\mathsf{k_m}}(t_2))$$

Inference Rules

Function Application

If you cannot distinguish the arguments, you cannot distinguish the images.

$$\frac{x_1, \ldots, x_n \sim y_1, \ldots, y_n}{f(x_1, \ldots, x_n) \sim f(y_1, \ldots, y_n)} \text{ FA}$$

Theorem

Definition

For every τ, we let $\underline{\tau}$ be τ where we use a fresh identity each time we run the $\underline{\text{ID sub-protocol}}$.

Theorem

Definition

For every τ, we let $\underline{\tau}$ be τ where we use a fresh identity each time we run the $\underline{\text{ID sub-protocol}}$.

Lemma

For every τ, there is a derivation using Ax of the formula $\phi_\tau \sim \phi_{\underline{\tau}}$.

Theorem

Definition

For every τ, we let $\underline{\tau}$ be τ where we use a fresh identity each time we run the ID sub-protocol.

Lemma

For every τ, there is a derivation using Ax of the formula $\phi_\tau \sim \phi_{\underline{\tau}}$.

Theorem

*The AKA^+ protocol is σ-unlinkable for **an arbitrary number of agents and sessions** when:*

- *The asymmetric encryption $\{\_\}_{\_}^{\_}$ is IND-CCA1.*
- *H and H^r (resp. Mac^1–Mac^5) satisfy jointly the PRF assumption.*

Remarks and Proof

Remarks

- This is against an **active attacker**.
- We show this for an **arbitrary number of agents and sessions**.

Remarks and Proof

Remarks

- This is against an **active attacker**.
- We show this for an **arbitrary number of agents and sessions**.

Proof

The proof is by induction over the symbolic trace τ. Finding the invariant requires some work, as it needs to:

- anticipate what will be needed latter (e.g. encryptions).
- match the left and right views of the adversary on the state.

Remarks and Proof

Remarks

- This is against an **active attacker**.
- We show this for an **arbitrary number of agents and sessions**.

Proof

The proof is by induction over the symbolic trace τ. Finding the invariant requires some work, as it needs to:

- anticipate what will be needed latter (e.g. encryptions).
- match the left and right views of the adversary on the state. E.g.:

$$\begin{array}{l}\text{if } \sigma_\tau(\mathsf{sync}_{\mathrm{U}}^{\mathrm{ID}}) \\ \quad \text{then } \sigma_\tau(\mathrm{SQN}_{\mathrm{U}}^{\mathrm{ID}}) - \sigma_\tau(\mathrm{SQN}_{\mathrm{N}}^{\mathrm{ID}}) \\ \quad \text{else } \bot\end{array} \sim \begin{array}{l}\text{if } \sigma_{\underline{\tau}}(\mathsf{sync}_{\mathrm{U}}^{\mathrm{ID}_{\underline{\tau}}}) \\ \quad \text{then } \sigma_{\underline{\tau}}(\mathrm{SQN}_{\mathrm{U}}^{\mathrm{ID}_{\underline{\tau}}}) - \sigma_{\underline{\tau}}(\mathrm{SQN}_{\mathrm{N}}^{\mathrm{ID}_{\underline{\tau}}}) \\ \quad \text{else } \bot\end{array}$$

Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, all others known unlinkability attacks still applies.

Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, all others known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.

Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, all others known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.
- We proposed the AKA^+ protocol, which satisfies the design constraints of 5G-AKA.

Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, all others known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.
- We proposed the AKA^+ protocol, which satisfies the design constraints of 5G-AKA.
- We defined the notion of σ-unlinkability.

Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, all others known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.
- We proposed the AKA^+ protocol, which satisfies the design constraints of 5G-AKA.
- We defined the notion of σ-unlinkability.
- We proved in the BC logic that AKA^+ is σ-unlinkability.
- We also proved that AKA^+ provides mutual authentication.

Thanks for your attention

References I

[3GPP, 2018] 3GPP (2018).
Ts 33.501: Security architecture and procedures for 5g system.

[Arapinis et al., 2012] Arapinis, M., Mancini, L. I., Ritter, E., Ryan, M., Golde, N., Redon, K., and Borgaonkar, R. (2012).
New privacy issues in mobile telephony: fix and verification.
In *the ACM Conference on Computer and Communications Security, CCS'12*, pages 205–216. ACM.

[Bana and Comon-Lundh, 2014] Bana, G. and Comon-Lundh, H. (2014).
A computationally complete symbolic attacker for equivalence properties.
In *2014 ACM Conference on Computer and Communications Security, CCS '14*, pages 609–620. ACM.

References II

[Fouque et al., 2016] Fouque, P., Onete, C., and Richard, B. (2016).
Achieving better privacy for the 3gpp AKA protocol.
PoPETs, 2016(4):255–275.

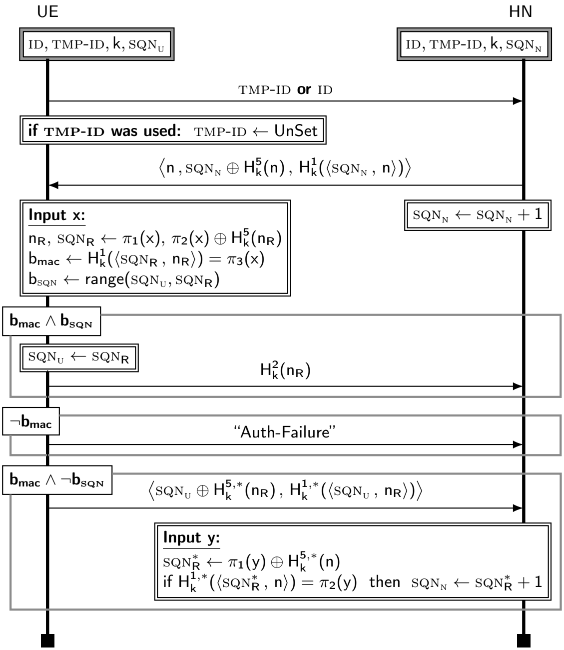
[Strobel, 2007] Strobel, D. (2007).
Imsi catcher.
Ruhr-Universität Bochum, Seminar Work.

No Pre-Fetching of Authentication Vectors

From the 3GPP specification for 5G-AKA ([3GPP, 2018], p. 37)

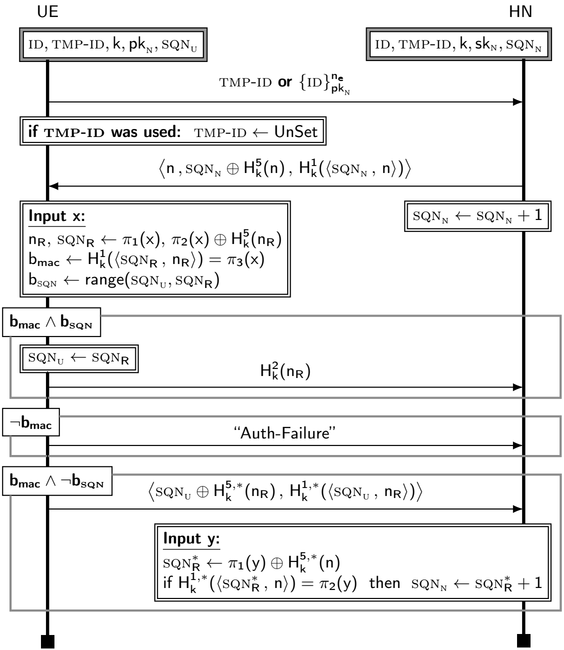
5G AKA does not support requesting multiple 5G AVs, neither the SEAF pre-fetching 5G AVs from the home network for future use.

4G-AKA

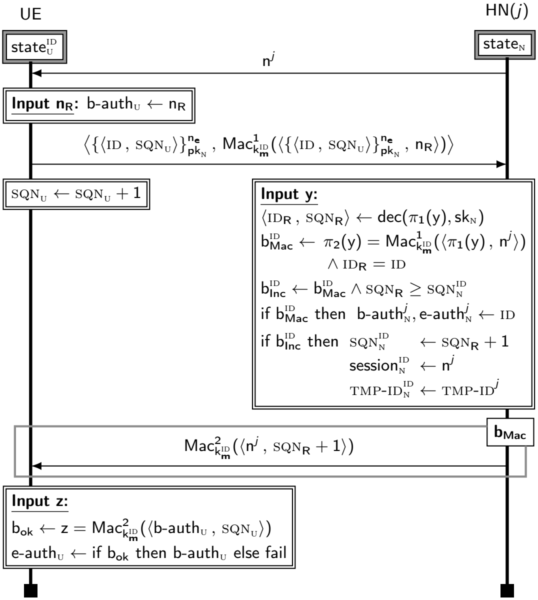


UE
HN
ID, TMP-ID, k, SQN_U
ID, TMP-ID, k, SQN_N
TMP-ID or ID
if TMP-ID was used: TMP-ID ← UnSet
⟨n , SQN_N ⊕ H_k^5(n) , H_k^1(⟨SQN_N , n⟩)⟩
Input x:
n_R, SQN_R ← π_1(x), π_2(x) ⊕ H_k^5(n_R)
b_mac ← H_k^1(⟨SQN_R , n_R⟩) = π_3(x)
b_SQN ← range(SQN_U, SQN_R)
SQN_N ← SQN_N + 1
b_mac ∧ b_SQN
SQN_U ← SQN_R
H_k^2(n_R)
¬b_mac
"Auth-Failure"
b_mac ∧ ¬b_SQN
⟨SQN_U ⊕ H_k^{5,*}(n_R) , H_k^{1,*}(⟨SQN_U , n_R⟩)⟩
Input y:
SQN_R^* ← π_1(y) ⊕ H_k^{5,*}(n)
if H_k^{1,*}(⟨SQN_R^* , n⟩) = π_2(y) then SQN_N ← SQN_R^* + 1

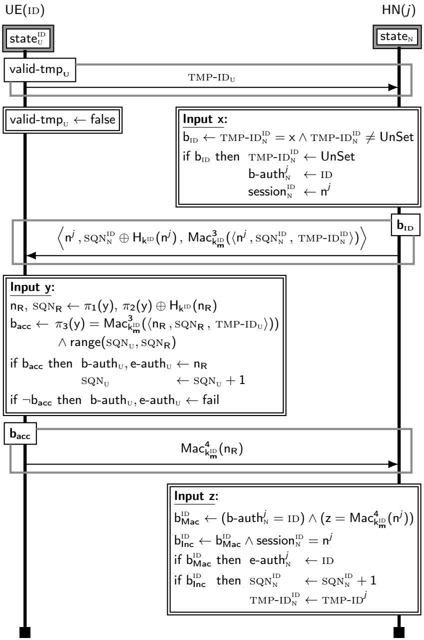
5G-AKA


UE
HN
ID, TMP-ID, k, pk_N, SQN_U
ID, TMP-ID, k, sk_N, SQN_N
TMP-ID or {ID}^{n_e}_{pk_N}
if TMP-ID was used: TMP-ID ← UnSet
⟨n , SQN_N ⊕ H^5_k(n) , H^1_k(⟨SQN_N , n⟩)⟩
Input x:
n_R, SQN_R ← π_1(x), π_2(x) ⊕ H^5_k(n_R)
b_mac ← H^1_k(⟨SQN_R , n_R⟩) = π_3(x)
b_SQN ← range(SQN_U, SQN_R)
SQN_N ← SQN_N + 1
b_mac ∧ b_SQN
SQN_U ← SQN_R
H^2_k(n_R)
¬b_mac
"Auth-Failure"
b_mac ∧ ¬b_SQN
⟨SQN_U ⊕ H^{5,*}_k(n_R) , H^{1,*}_k(⟨SQN_U , n_R⟩)⟩
Input y:
SQN^*_R ← π_1(y) ⊕ H^{5,*}_k(n)
if H^{1,*}_k(⟨SQN^*_R , n⟩) = π_2(y) then SQN_N ← SQN^*_R + 1

ID Sub-Protocol



TMP-ID Sub-Protocol



The ASSIGN-TMP-ID Sub-Protocol

UE $\quad$ $\mathsf{state}_{\mathrm{U}}^{\mathrm{ID}}$

HN(j) $\quad$ $\mathsf{state}_{\mathrm{N}}$

$\mathsf{e\text{-}auth}_{\mathrm{N}}^{\mathrm{ID}} = \mathbf{ID}$

HN → UE: $\langle \textsc{tmp-id}^j \oplus \mathsf{H}^{\mathsf{r}}_{\mathsf{k}^{\mathrm{ID}}}(\mathsf{n}^j)\, ,\ \mathsf{Mac}^{5}_{\mathsf{k}^{\mathrm{ID}}_{\mathbf{m}}}(\langle \textsc{tmp-id}^j\, ,\ \mathsf{n}^j \rangle)\rangle$

Input x:

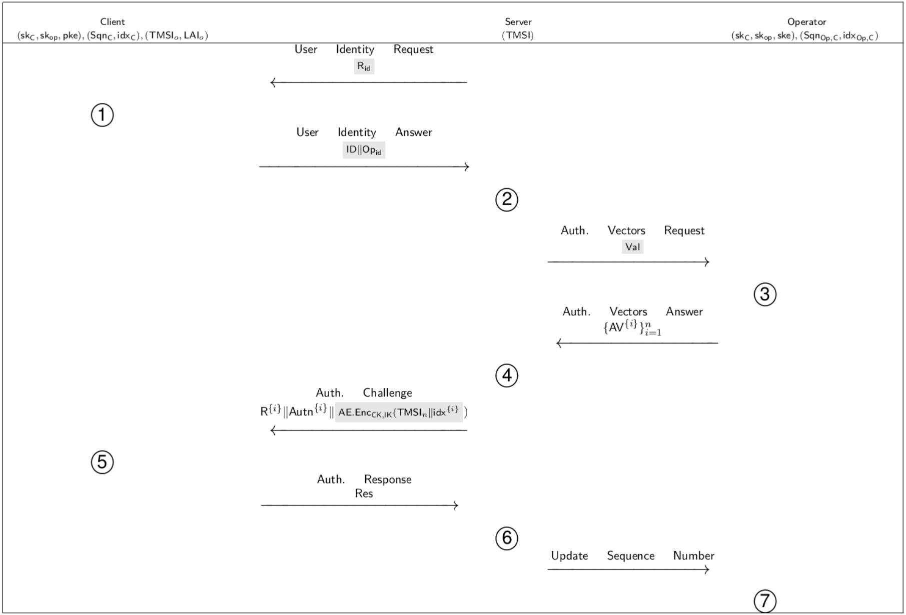
$\textsc{tmp-id}_{\mathsf{R}} \leftarrow \pi_1(\mathsf{x}) \oplus \mathsf{H}^{\mathsf{r}}_{\mathsf{k}^{\mathrm{ID}}_{\mathbf{m}}}(\mathsf{e\text{-}auth}_{\mathrm{U}})$

$\mathsf{b}_{\mathsf{acc}} \leftarrow \left(\pi_2(\mathsf{x}) = \mathsf{Mac}^{5}_{\mathsf{k}^{\mathrm{ID}}_{\mathbf{m}}}(\langle \textsc{tmp-id}_{\mathsf{R}}\, ,\ \mathsf{e\text{-}auth}_{\mathrm{U}}\rangle)\right)$
$\qquad \wedge (\mathsf{e\text{-}auth}_{\mathrm{U}} \neq \mathsf{fail})$

$\textsc{tmp-id}_{\mathrm{U}} \leftarrow$ if $\mathsf{b}_{\mathsf{acc}}$ then $\textsc{tmp-id}_{\mathsf{R}}$ else UnSet

$\mathsf{valid\text{-}tmp}_{\mathrm{U}} \leftarrow \mathsf{b}_{\mathsf{acc}}$

PRIV-AKA [Fouque et al., 2016]



PRIV-AKA [Fouque et al., 2016]

Client	Server	Operator
①: Compute the identifier: If $\mathsf{flag}_{\mathsf{TMSI}} := 0$ then ID = TMSI. Else, $\mathsf{ID} = \mathsf{PKE.Enc}_{\mathsf{pke}}(f_5(\mathsf{keys}, \mathsf{R}_{\mathsf{id}}, \mathsf{IMSI}, \mathsf{idx}_C) \| \mathsf{R}_{\mathsf{id}} \| \mathsf{IMSI} \| \mathsf{idx}_C)$. $\mathsf{flag}_{\mathsf{TMSI}} := 1$. ——— ⑤: Compute AK using $\mathsf{R}^{\{i\}}$. Recover $\mathsf{Sqn}^{\{i\}}$ (from AK). Check Mac_S value. Compute: IK, CK; Retrieve the received index and the new TMSI. If abort caused or the AE does not verify, set $\mathsf{flag}_{\mathsf{TMSI}} := 1$ and increment: $\mathsf{idx}_C := \mathsf{idx}_C + 1$. Else, check validity of $\mathsf{Sqn}^{\{i\}}$, i.e if one of the following conditions is correct: – $\mathsf{Sqn}_C = \mathsf{Sqn}^{\{i\}}$. – $\mathsf{Sqn}_C = \mathsf{inc}(\mathsf{Sqn}^{\{i\}})$ and $\mathsf{idx}^{\{i\}} = \mathsf{idx}_C + 1$. If the first condition is accepted: reset the index idx_C, update the sequence number $\mathsf{Sqn}_C = \mathsf{inc}(\mathsf{Sqn}_C)$. If the second condition is accepted: idx_C=idx_C+1. Compute $\mathsf{Res} := \mathcal{F}_1^*(\mathsf{keys}, \mathsf{R}^{\{i\}}, \mathsf{Sqn}^{\{i\}}, \mathsf{Res}_S, \mathsf{AMF})$. Update the internal index. Allocate the new TMSI. $\mathsf{flag}_{\mathsf{TMSI}} := 0$.	②: Process the identifier ID: If the identifier is a TMSI then Val = IMSI. Otherwise, $\mathsf{Val} = (\mathsf{ID}, \mathsf{R}_{\mathsf{id}})$. ——— ④: Store $\{\mathsf{AV}^{\{i\}}\}_{i=1}^{n}$. Choose $\mathsf{AV}^{\{i\}}$ one by one in order. Then, it sends the authentication challenge and the new couple $(\mathsf{TMSI}_n, \mathsf{idx}^{\{i\}})$ encrypted and authenticated by the session keys. ——— ⑥: If the authentication of the client is verified ($\mathsf{Res} \stackrel{?}{=} \mathsf{Mac}_C$), then they ask to the server the update of its sequence number. Otherwise, the protocol is aborted.	③: Verify the identity of the client with Val. If this holds, retrieve idx_C, set $\mathsf{idx}_{\mathsf{Op},C} := \mathsf{idx}_C$ Generate $(\mathsf{R}^{\{1\}}, \ldots, \mathsf{R}^{\{n\}})$. Denote: $\mathsf{keys} := (\mathsf{sk}_C, \mathsf{sk}_{\mathsf{op}})$. For each $i = 1, \ldots, n$, compute: $\mathsf{Mac}_S \leftarrow \mathcal{F}_1(\mathsf{keys}, \mathsf{R}^{\{i\}}, \mathsf{Sqn}^{\{i\}}, \mathsf{Res}_S, \mathsf{AMF})$, $\mathsf{Mac}_C \leftarrow \mathcal{F}_1^*(\mathsf{keys}, \mathsf{R}^{\{i\}}, \mathsf{Sqn}^{\{i\}}, \mathsf{Res}_S, \mathsf{AMF})$, $\mathsf{CK} \leftarrow \mathcal{F}_3(\mathsf{keys}, \mathsf{R}^{\{i\}}, \mathsf{Sqn}^{\{i\}}, \mathsf{Res}_S, \mathsf{AMF})$, $\mathsf{IK} \leftarrow \mathcal{F}_4(\mathsf{keys}, \mathsf{R}^{\{i\}}, \mathsf{Sqn}^{\{i\}}, \mathsf{Res}_S, \mathsf{AMF})$, $\mathsf{AK} \leftarrow \mathcal{F}_5(\mathsf{keys}, \mathsf{R}^{\{i\}}, \mathsf{Res}_S)$, $\mathsf{Autn}^{\{i\}} \leftarrow (\mathsf{Sqn}^{\{i\}} \oplus \mathsf{AK}) \| \mathsf{AMF} \| \mathsf{Mac}_S$, $\mathsf{Sqn}^{\{i\}} \leftarrow \mathsf{inc}(\mathsf{Sqn}^{\{i-1\}})$, $\mathsf{AV}^{\{i\}} := (\mathsf{R}^{\{i\}}, \mathsf{CK}, \mathsf{IK}, \mathsf{Autn}^{\{i\}}, \mathsf{Mac}_C, \mathsf{idx}^{\{i\}})$, with $\mathsf{Sqn}^{\{1\}} := \mathsf{Sqn}_{\mathsf{Op},C}$, $\mathsf{idx}^{\{1\}} := \mathsf{idx}_{\mathsf{Op},C}$, $\forall i \neq 1, \mathsf{idx}^{\{i\}} = 0$. End for. ——— ⑦: Update the sequence number: $\mathsf{Sqn}_{\mathsf{Op},C} \leftarrow \mathsf{inc}(\mathsf{Sqn}_{\mathsf{Op},C})$. Reset the index $\mathsf{idx}_{\mathsf{Op},C}$.

Licenses

- Smart-phone icon: Gregor Hagedorn, CC-BY-SA-3.0
- Database icon: Font Awesome, CC-BY-4.0