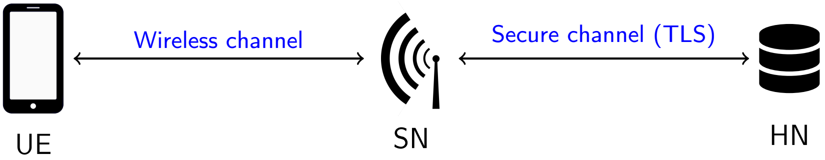
# The 5G-AKA Authentication Protocol Privacy

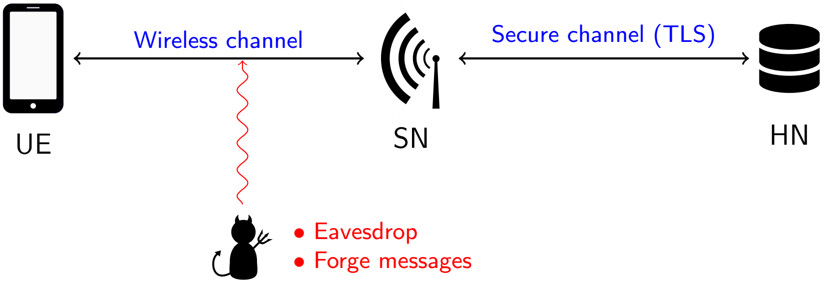
Adrien Koutsos
LSV, CNRS, ENS Paris-Saclay

June 19, 2019

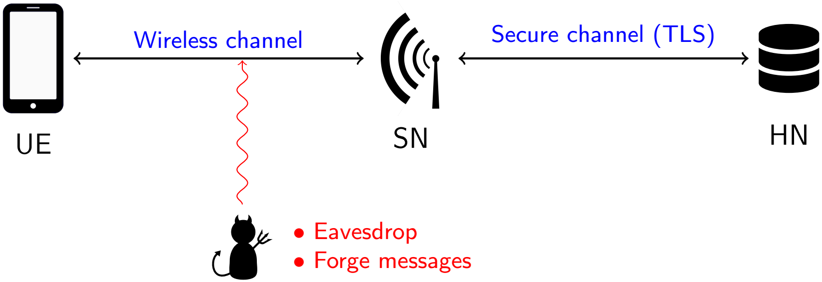
# Authentication and Key Agreement Protocol



# Authentication and Key Agreement Protocol



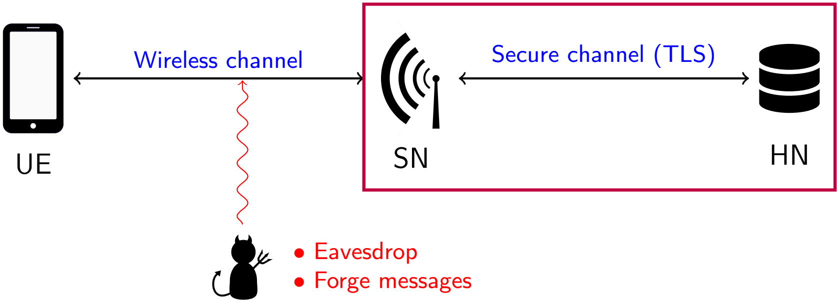
# Authentication and Key Agreement Protocol




We focus on:

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

# Authentication and Key Agreement Protocol




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

## 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 service provider uses a random challenge.
- The mobile phone uses a sequence number SQN:

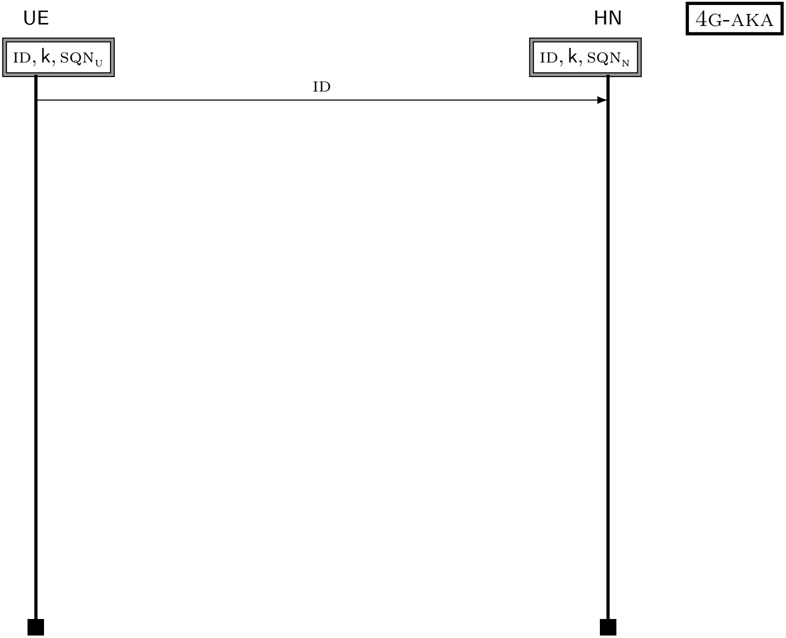
# Sequence Numbers

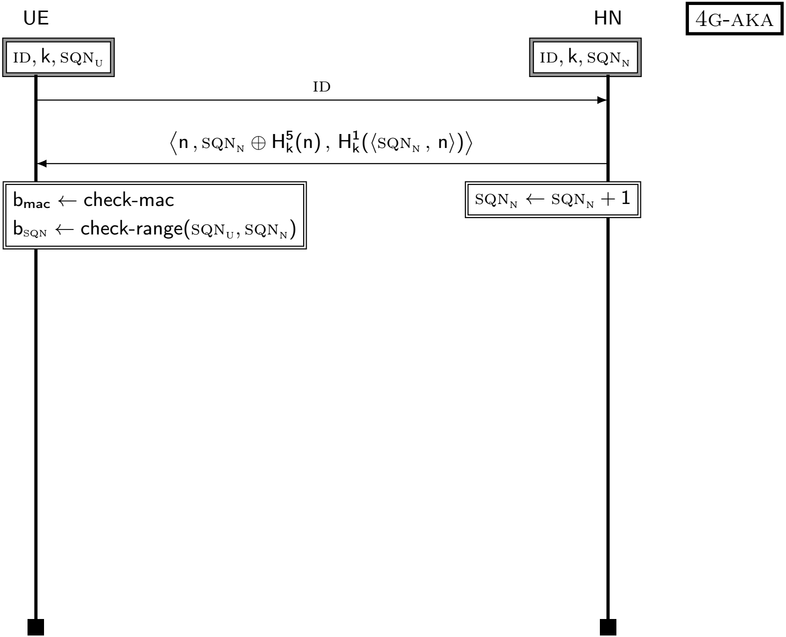
## Authentication

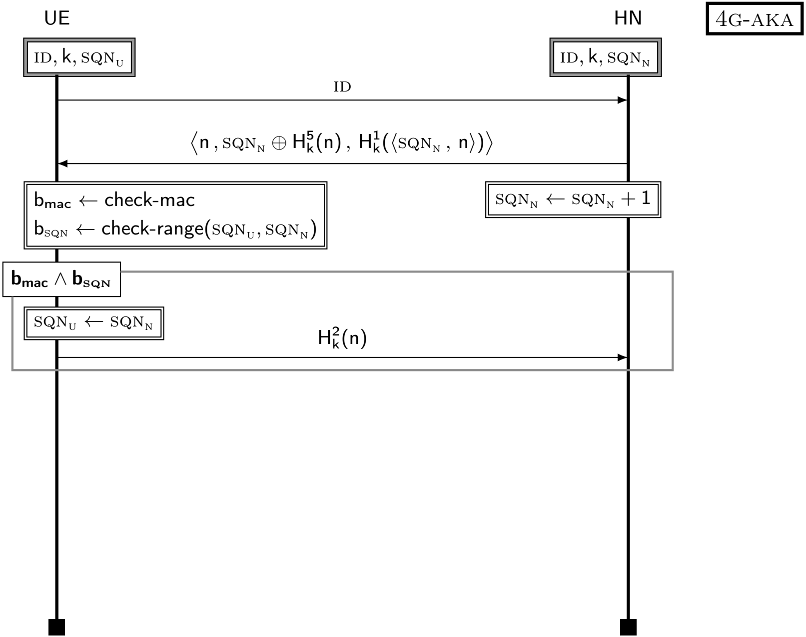
Authentication protocols need to prevent message replays. In 4G-AKA:

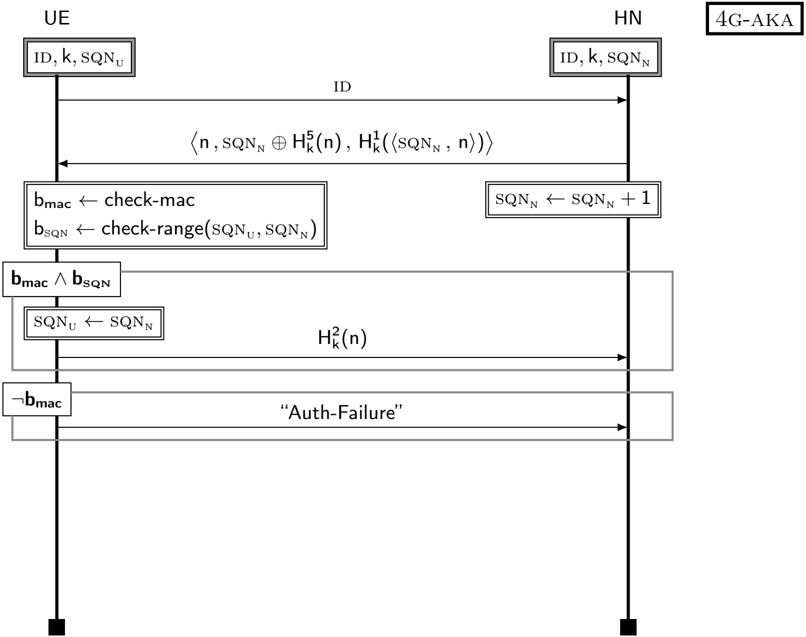
- The service provider uses a random challenge.
- The mobile phone uses a sequence number $\text{SQN}$:
  - Incremented after each successful session.
  - Tracked by the user and the service provider ($\text{SQN}_\text{U}$ and $\text{SQN}_\text{N}$).

$\Rightarrow$ De-synchronization possible.

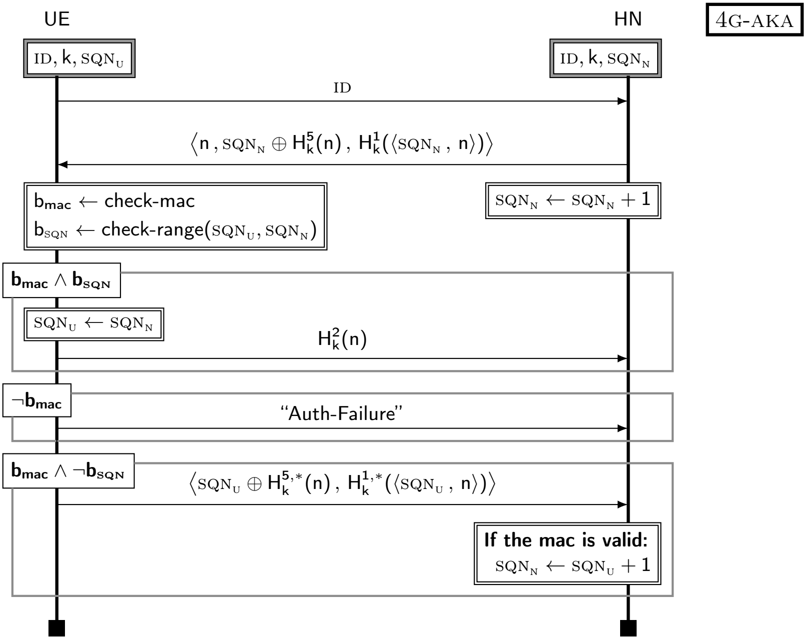

UE
HN
4G-AKA
ID, k, SQN_U
ID, k, SQN_N
ID


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, 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
⟨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)
¬b_mac
"Auth-Failure"



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
$b_{SQN} \leftarrow$ check-range$(\mathrm{SQN_U}, \mathrm{SQN_N})$
$\mathrm{SQN_N} \leftarrow \mathrm{SQN_N} + 1$
$\mathbf{b_{mac}} \wedge \mathbf{b_{SQN}}$
$\mathrm{SQN_U} \leftarrow \mathrm{SQN_N}$
$H^2_k(n)$
$\neg \mathbf{b_{mac}}$
"Auth-Failure"
$\mathbf{b_{mac}} \wedge \neg \mathbf{b_{SQN}}$
$\langle \mathrm{SQN_U} \oplus H^{5,*}_k(n), H^{1,*}_k(\langle \mathrm{SQN_U}, n\rangle)\rangle$
If the mac is valid:
$\mathrm{SQN_N} \leftarrow \mathrm{SQN_U} + 1$

# Privacy in 4G-AKA

**Not confidentiality of the user identity**

The ID is sent in plain text!

# Privacy in 4G-AKA

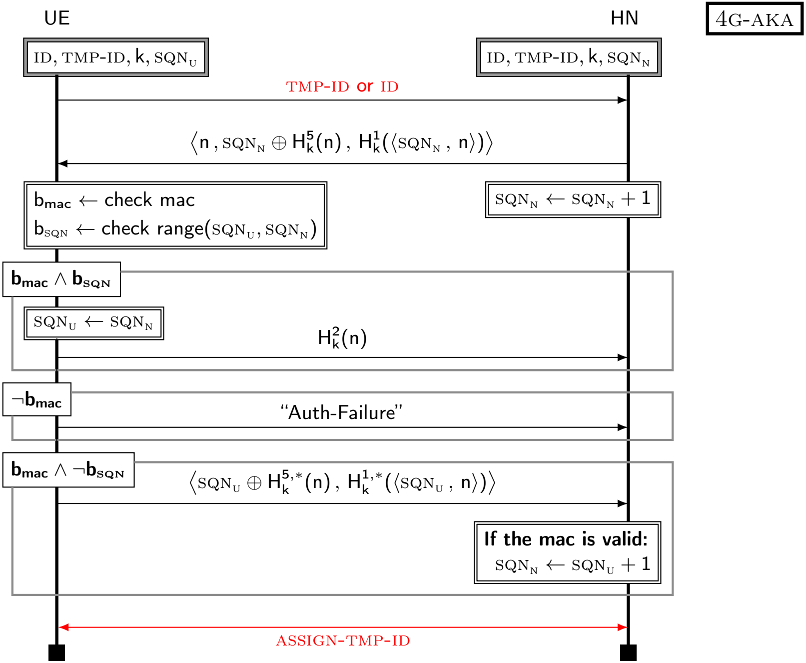
## Not confidentiality of the user identity

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, $\mathrm{SQN_U}$
ID, TMP-ID, k, $\mathrm{SQN_N}$
TMP-ID or 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$(\mathrm{SQN_U}, \mathrm{SQN_N})$
$\mathrm{SQN_N} \leftarrow \mathrm{SQN_N} + 1$
$b_{mac} \wedge b_{\mathrm{SQN}}$
$\mathrm{SQN_U} \leftarrow \mathrm{SQN_N}$
$H^2_k(n)$
$\neg b_{mac}$
“Auth-Failure”
$b_{mac} \wedge \neg b_{\mathrm{SQN}}$
$\langle \mathrm{SQN_U} \oplus H^{5,*}_k(n), H^{1,*}_k(\langle \mathrm{SQN_U}, n\rangle)\rangle$
If the mac is valid:
$\mathrm{SQN_N} \leftarrow \mathrm{SQN_U} + 1$
ASSIGN-TMP-ID

# Privacy in 4G-AKA

### Confidentiality of the user identity

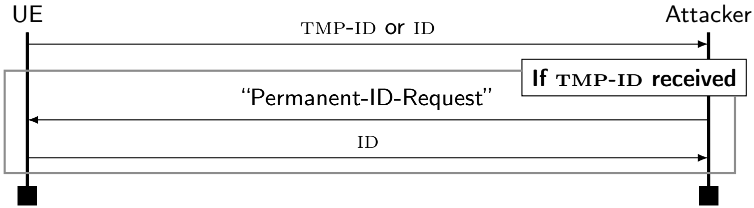
The ID is protected as long as the protocol does not fail.

# Privacy in 4G-AKA

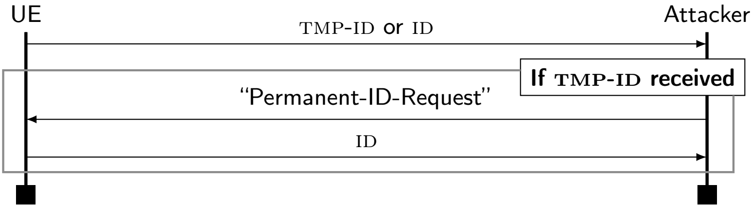
### Confidentiality of the user identity

The ID is protected as long as the protocol does not fail.
$\Longrightarrow$ This only works against a passive adversary.

# 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]).

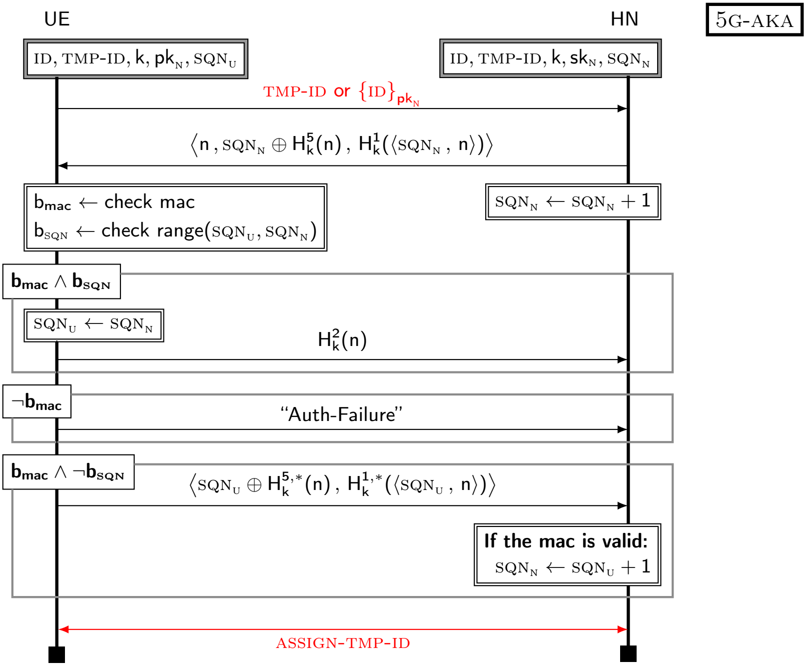
# Privacy in 5G-AKA

## 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

## Unlinkability Attack

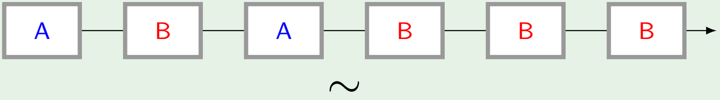
Even if the ID is hidden, an attacker may link sessions of the same user.

# Unlinkability

## Unlinkability Attack

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

## Example of an Unlinkability Scenario

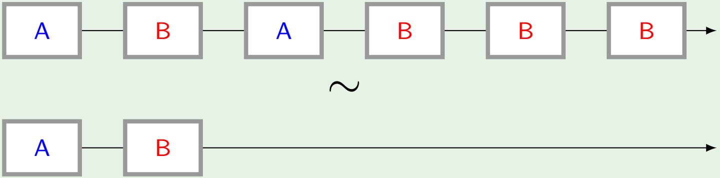


# Unlinkability

## Unlinkability Attack

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

## Example of an Unlinkability Scenario

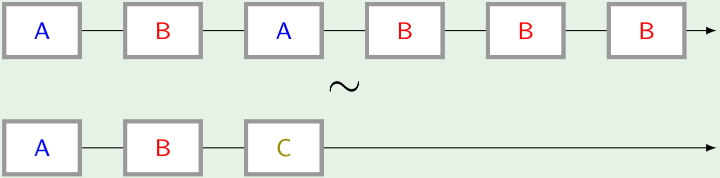


# Unlinkability

## Unlinkability Attack

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

## Example of an Unlinkability Scenario

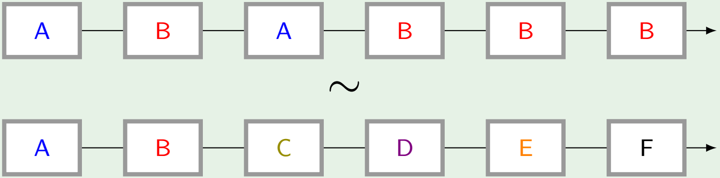


# Unlinkability

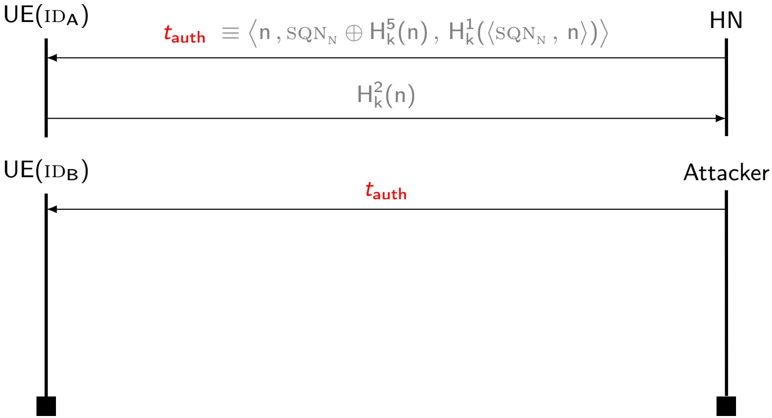
## Unlinkability Attack

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

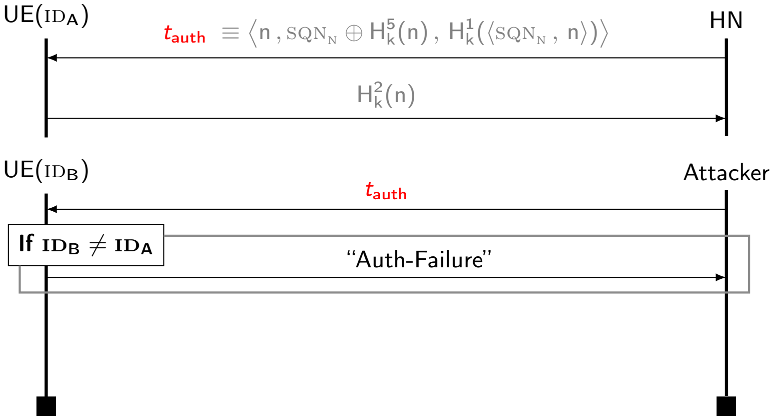
## Example of an Unlinkability Scenario



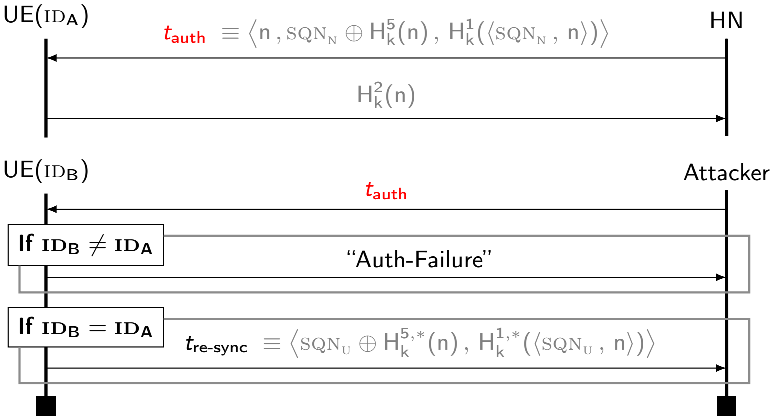
# The Failure Message Attack [Arapinis et al., 2012]



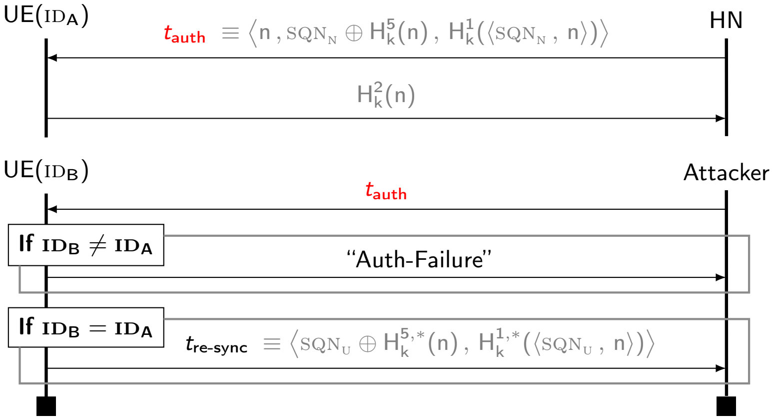
# The Failure Message Attack [Arapinis et al., 2012]



# 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 $\mathrm{ID_A}$ or $\mathrm{ID_B}$.

# Objective

## Objective

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

- Provides some form of unlinkability.

# Objective

## Objective

Design a modified version of AKA, called $\text{AKA}^+$, 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$^+$, that:

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

# 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 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 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:

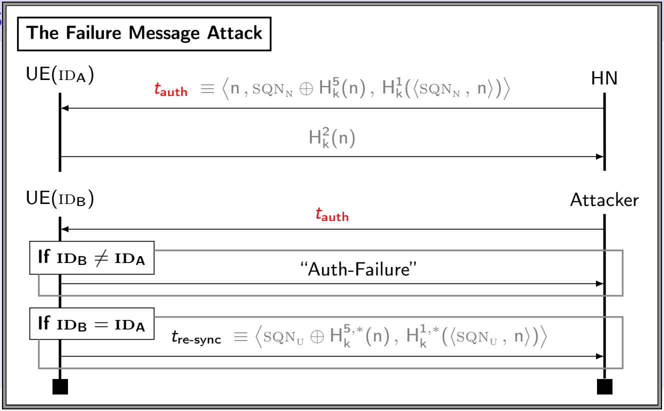
- Random number generation (user): at most one nonce per session, and only if no TMP-ID is assigned.
- The user can use only one-way functions and asymmetric *encryption*.
- Network complexity: try to have only three messages per session.

# Key Ideas

## Key Ideas Behind $\textsc{aka}^+$

# Key Ideas

## Key Ideas



# Key Ideas

## Key Ideas Behind $\textsc{aka}^+$

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

# Key Ideas

## Key Ideas Behind $\text{AKA}^{+}$

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

# Key Ideas

## Key Ideas Behind AKA$^+$

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

UE $\xleftarrow{\mathsf{n}}$ HN

UE $\xrightarrow{\left\langle \{\langle \mathrm{ID}\ ,\ \mathrm{SQN_U} \rangle\}_{\mathsf{pk_N}}\ ,\ \mathsf{Mac}^1_{\mathbf{k_m}}(\langle \{\langle \mathrm{ID}\ ,\ \mathrm{SQN_U} \rangle\}_{\mathsf{pk_N}}\ ,\ \mathsf{n}\rangle)\right\rangle}$ HN

# Architecture of AKA$^+$

## AKA$^+$ Sub-Protocols

- ID sub-protocol 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 uses the encrypted permanent identity.
  - allows to re-synchronize the UE and the HN.
- TMP-ID sub-protocol uses a temporary identity.

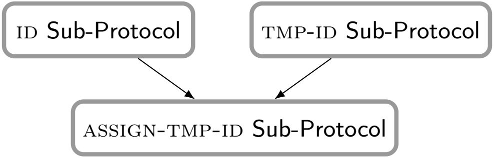
ID Sub-Protocol

TMP-ID Sub-Protocol

# Architecture of AKA$^+$

## AKA$^+$ Sub-Protocols

- ID sub-protocol uses the encrypted permanent identity.
  - allows to re-synchronize the UE and the HN.
- TMP-ID sub-protocol uses a temporary identity.
- ASSIGN-TMP-ID assigns a fresh temporary identity to the UE.



# Security Proofs

## Objective

Formally prove that $\textsc{aka}^+$ satisfies:

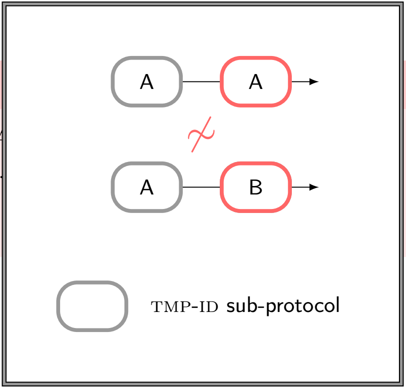
- **mutual authentication**.
- **unlinkability**.

# Security Proofs

## Objective

Formally prove that $A$

- **mutual authen**
- **unlinkability**.



# Security Proofs

## Objective

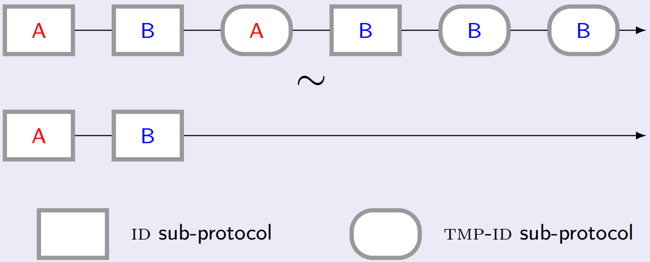
Formally prove that $\textsc{aka}^+$ satisfies:

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

# The $\sigma$-Unlinkability Property

## Two Indistinguishable Executions

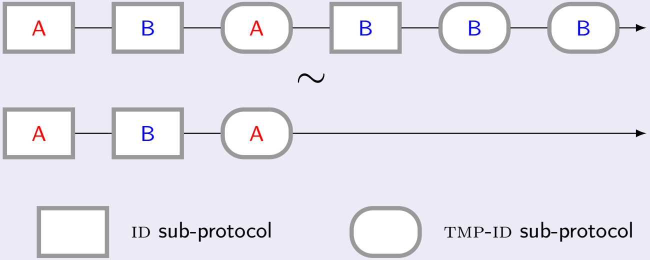
Each time the ID sub-protocol is used, we can change the user's identity.



# The $\sigma$-Unlinkability Property

## Two Indistinguishable Executions

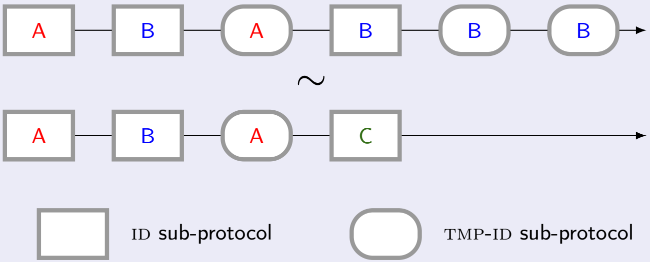
Each time the ID sub-protocol is used, we can change the user's identity.



# The $\sigma$-Unlinkability Property

## Two Indistinguishable Executions

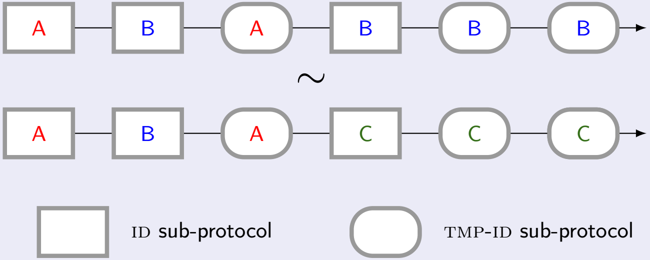
Each time the ID sub-protocol is used, we can change the user's identity.



# The $\sigma$-Unlinkability Property

## Two Indistinguishable Executions

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



# 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 $\mathsf{Ax}$.

# 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 Ax $\models \vec{u}_P \sim \vec{u}_Q$.

# Theorem

## Theorem

*The* $\textsc{aka}^{+}$ *protocol is* $\sigma$*-unlinkable for* ***an arbitrary number of agents and sessions*** *when:*

- *The asymmetric encryption* $\{\_\}_{\_}^{\_}$ *is* $\textsc{ind-cca}_1$.
- *H and* $H^r$ *(resp.* $Mac^1$*–*$Mac^5$*) satisfy jointly the* $\textsc{prf}$ *assumption.*

# Conclusion

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

# Conclusion

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

# Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, several known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.
- We proposed the $\text{AKA}^{+}$ protocol, which tries to satisfy the design constraints of 5G-AKA.

# Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, several known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.
- We proposed the $\text{AKA}^+$ protocol, which tries to satisfy the design constraints of 5G-AKA.
- We defined the notion of $\sigma$-unlinkability.

# Conclusion

- While 5G-AKA prevents the IMSI-catcher attack, several known unlinkability attacks still applies.
- We gave a new unlinkability attack against PRIV-AKA.
- We proposed the $\text{AKA}^+$ protocol, which tries to satisfy the design constraints of 5G-AKA.
- We defined the notion of $\sigma$-unlinkability.
- We proved in the BC logic that $\text{AKA}^+$ is $\sigma$-unlinkability.
- We also proved that $\text{AKA}^+$ 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.

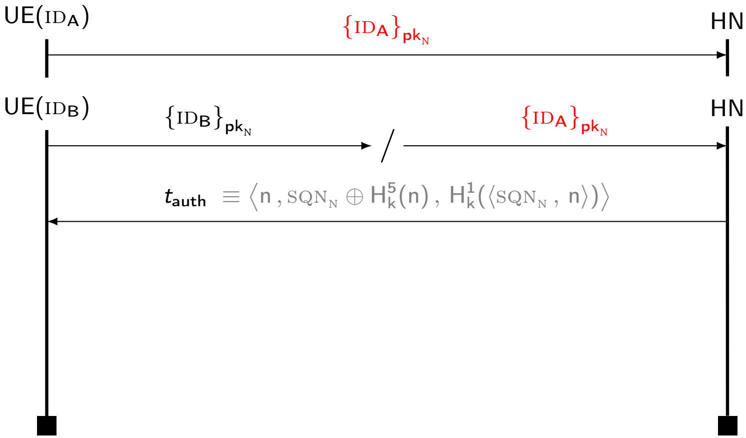
[Bana and Comon-Lundh, 2014] Bana, G. and Comon-Lundh, H. (2014).
A computationally complete symbolic attacker for equivalence properties.
In *2014 ACM Conference on Computer and Communications Security, CCS '14*, pages 609–620. ACM.

# References II

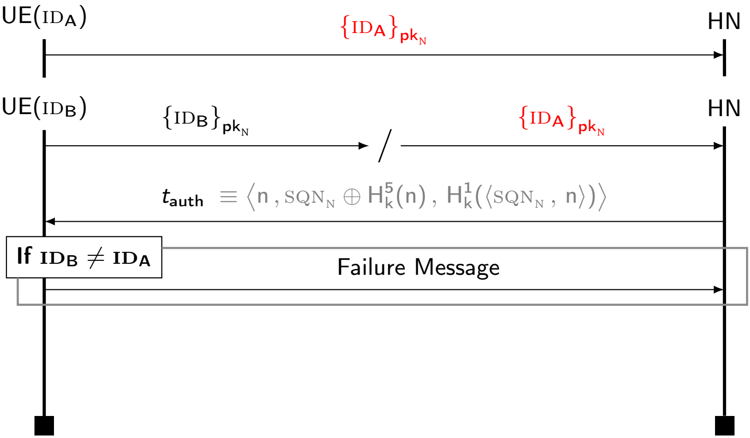
[Fouque et al., 2016] Fouque, P., Onete, C., and Richard, B. (2016).
Achieving better privacy for the 3GPP AKA protocol.
*PoPETs*, 2016(4):255–275.

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

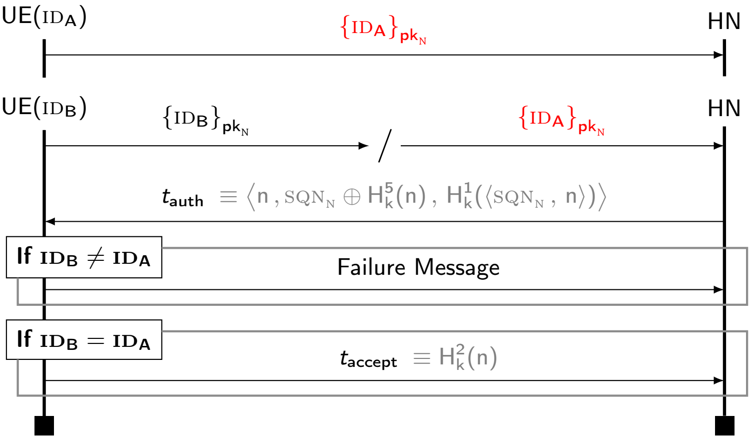
# The Encrypted ID Replay Attack [Fouque et al., 2016]



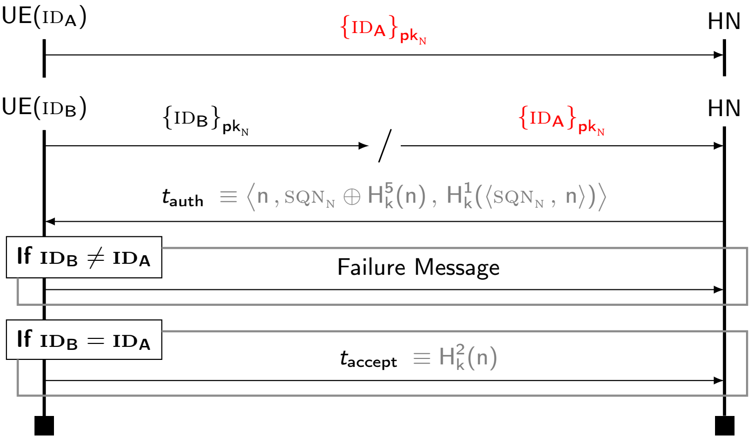
# The Encrypted ID Replay Attack [Fouque et al., 2016]



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



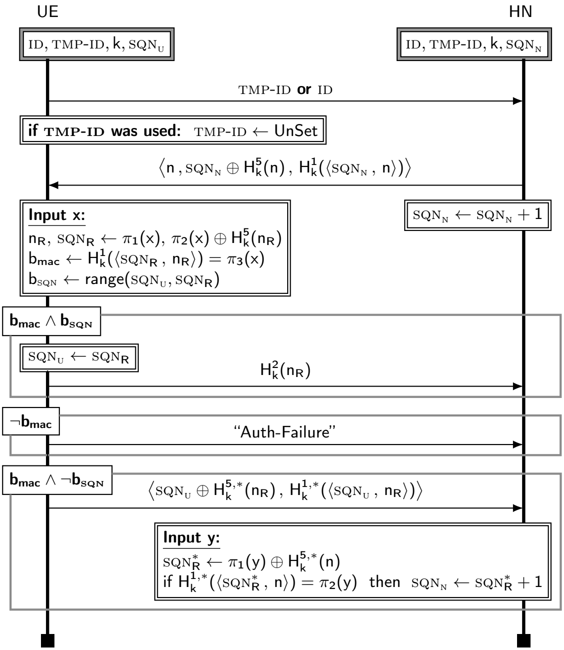
# The Encrypted ID Replay Attack [Fouque et al., 2016]

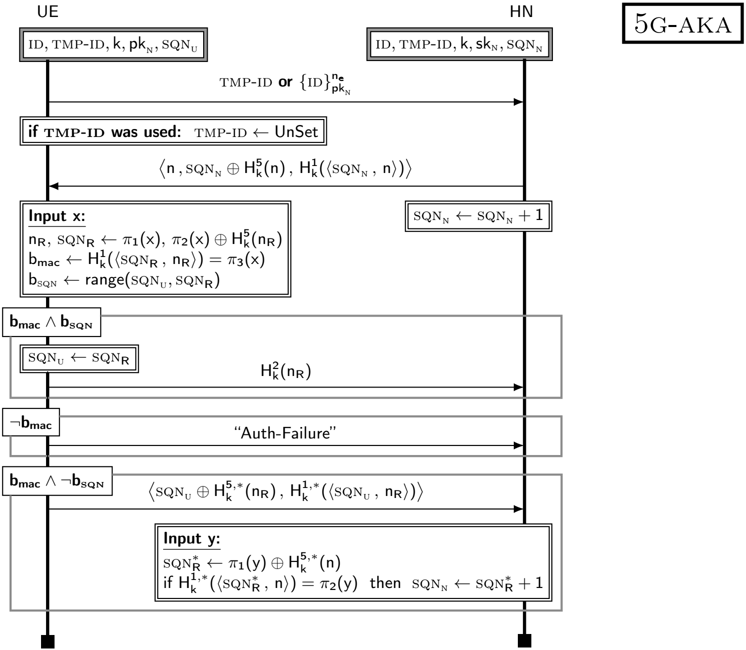



**Unlinkability attack**

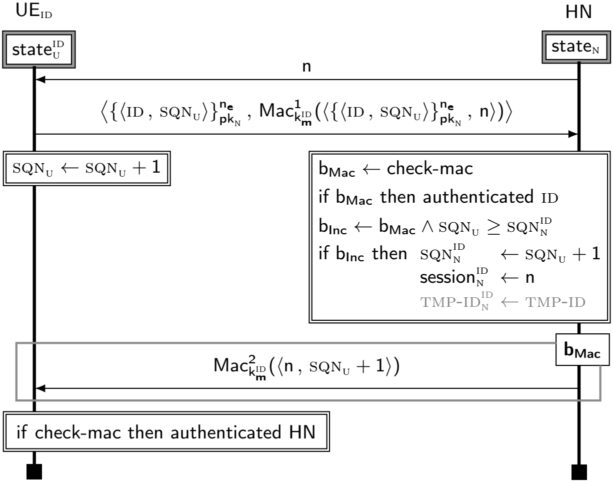
The adversary knows if it interacted with ID$_A$ or ID$_B$.

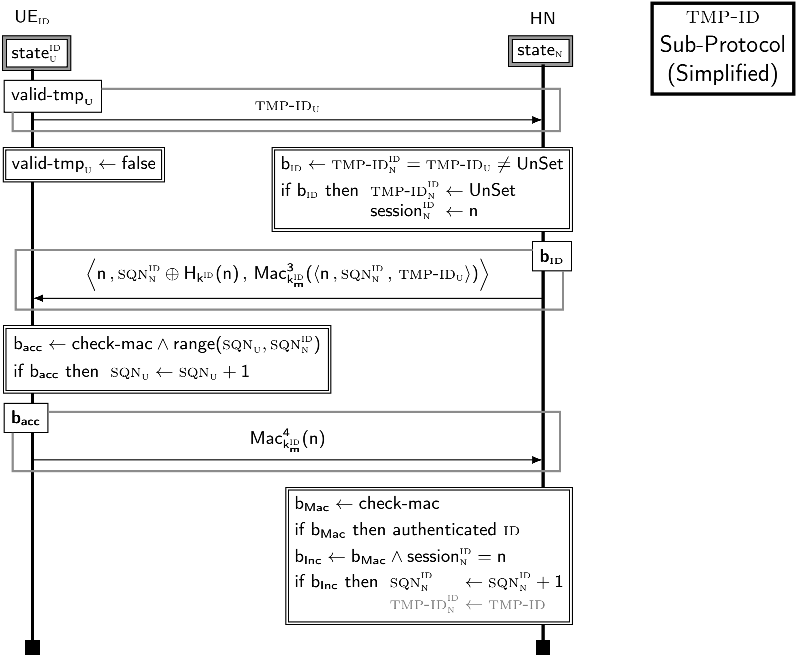
4G-AKA


UE
HN
ID, TMP-ID, k, $\mathrm{SQN_U}$
ID, TMP-ID, k, $\mathrm{SQN_N}$
TMP-ID **or** ID
**if TMP-ID was used:** TMP-ID $\leftarrow$ UnSet
$\langle \mathsf{n}, \mathrm{SQN_N} \oplus \mathsf{H}^5_\mathsf{k}(\mathsf{n}), \mathsf{H}^1_\mathsf{k}(\langle \mathrm{SQN_N}, \mathsf{n}\rangle)\rangle$
**Input x:**
$\mathsf{n_R}$, $\mathrm{SQN}_\mathsf{R} \leftarrow \pi_1(\mathsf{x}), \pi_2(\mathsf{x}) \oplus \mathsf{H}^5_\mathsf{k}(\mathsf{n_R})$
$\mathsf{b_{mac}} \leftarrow \mathsf{H}^1_\mathsf{k}(\langle \mathrm{SQN}_\mathsf{R}, \mathsf{n_R}\rangle) = \pi_3(\mathsf{x})$
$\mathsf{b}_{\mathrm{SQN}} \leftarrow \mathsf{range}(\mathrm{SQN_U}, \mathrm{SQN}_\mathsf{R})$
$\mathrm{SQN_N} \leftarrow \mathrm{SQN_N} + 1$
$\mathbf{b_{mac}} \wedge \mathbf{b_{SQN}}$
$\mathrm{SQN_U} \leftarrow \mathrm{SQN}_\mathsf{R}$
$\mathsf{H}^2_\mathsf{k}(\mathsf{n_R})$
$\neg\mathbf{b_{mac}}$
"Auth-Failure"
$\mathbf{b_{mac}} \wedge \neg\mathbf{b_{SQN}}$
$\langle \mathrm{SQN_U} \oplus \mathsf{H}^{5,*}_\mathsf{k}(\mathsf{n_R}), \mathsf{H}^{1,*}_\mathsf{k}(\langle \mathrm{SQN_U}, \mathsf{n_R}\rangle)\rangle$
**Input y:**
$\mathrm{SQN}^*_\mathsf{R} \leftarrow \pi_1(\mathsf{y}) \oplus \mathsf{H}^{5,*}_\mathsf{k}(\mathsf{n})$
if $\mathsf{H}^{1,*}_\mathsf{k}(\langle \mathrm{SQN}^*_\mathsf{R}, \mathsf{n}\rangle) = \pi_2(\mathsf{y})$ then $\mathrm{SQN_N} \leftarrow \mathrm{SQN}^*_\mathsf{R} + 1$

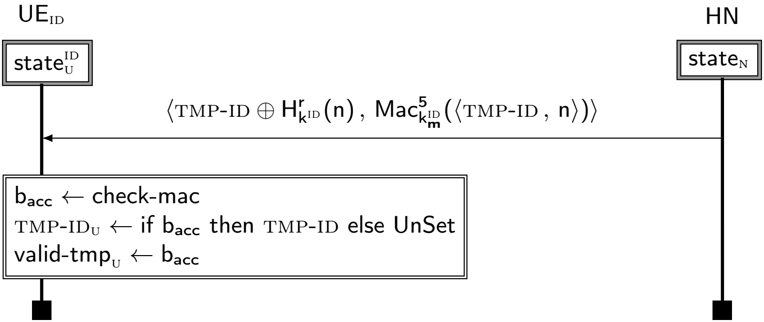

5G-AKA
UE
HN
ID, TMP-ID, k, pk_N, SQN_U
ID, TMP-ID, k, sk_N, SQN_N
TMP-ID or {ID}^{n_e}_{pk_N}
if TMP-ID was used: TMP-ID ← UnSet
⟨n, SQN_N ⊕ H^5_k(n), H^1_k(⟨SQN_N, n⟩)⟩
Input x:
n_R, SQN_R ← π_1(x), π_2(x) ⊕ H^5_k(n_R)
b_mac ← H^1_k(⟨SQN_R, n_R⟩) = π_3(x)
b_SQN ← range(SQN_U, SQN_R)
SQN_N ← SQN_N + 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
(Simplified)

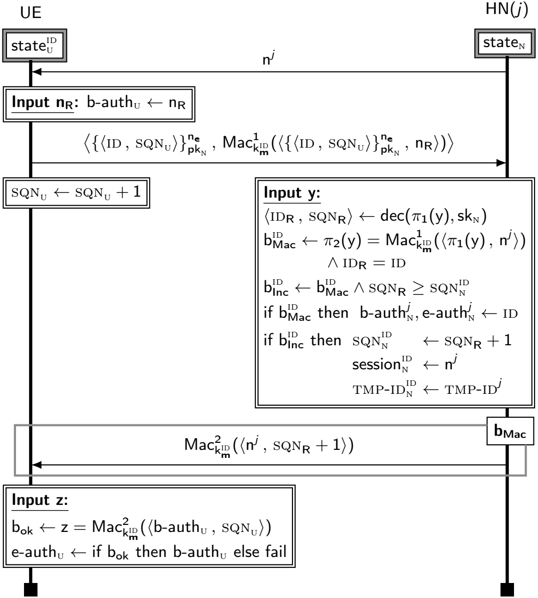



TMP-ID Sub-Protocol (Simplified)
UE_ID
HN
state_U^ID
state_N
valid-tmp_U
TMP-ID_U
valid-tmp_U ← false
b_ID ← TMP-ID_N^ID = TMP-ID_U ≠ UnSet
if b_ID then TMP-ID_N^ID ← UnSet
session_N^ID ← n
b_ID
⟨n, SQN_N^ID ⊕ H_k^ID(n), Mac^3_k_m^ID(⟨n, SQN_N^ID, TMP-ID_U⟩)⟩
b_acc ← check-mac ∧ range(SQN_U, SQN_N^ID)
if b_acc then SQN_U ← SQN_U + 1
b_acc
Mac^4_k_m^ID(n)
b_Mac ← check-mac
if b_Mac then authenticated ID
b_Inc ← b_Mac ∧ session_N^ID = n
if b_Inc then SQN_N^ID ← SQN_N^ID + 1
TMP-ID_N^ID ← TMP-ID

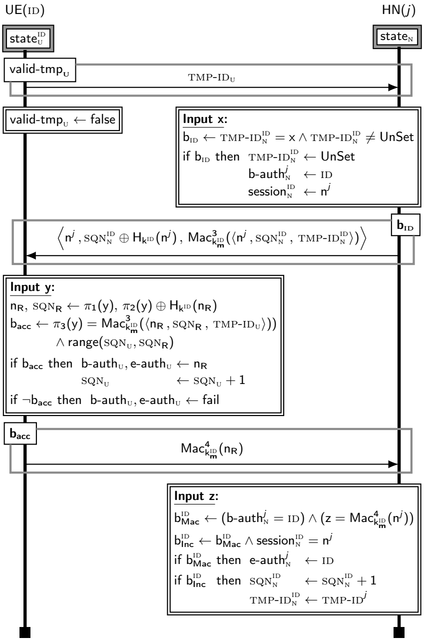
# The ASSIGN-TMP-ID Sub-Protocol (Simplified)



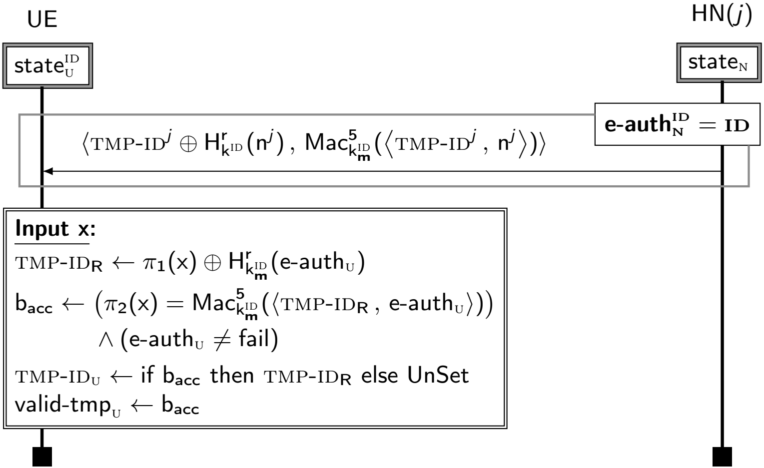
ID
Sub-Protocol



TMP-ID Sub-Protocol



# The ASSIGN-TMP-ID Sub-Protocol



# New Attack on the PRIV-AKA Protocol

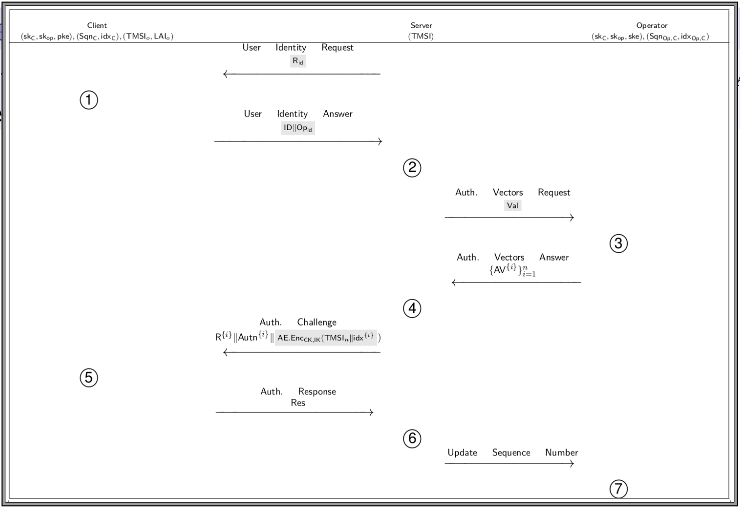
## The PRIV-AKA Protocol

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

# New Attack on the PRIV-AKA Protocol

## The PI

The au AKA
(claime



# New Attack on the PRIV-AKA Protocol

## The PRIV-AKA Protocol

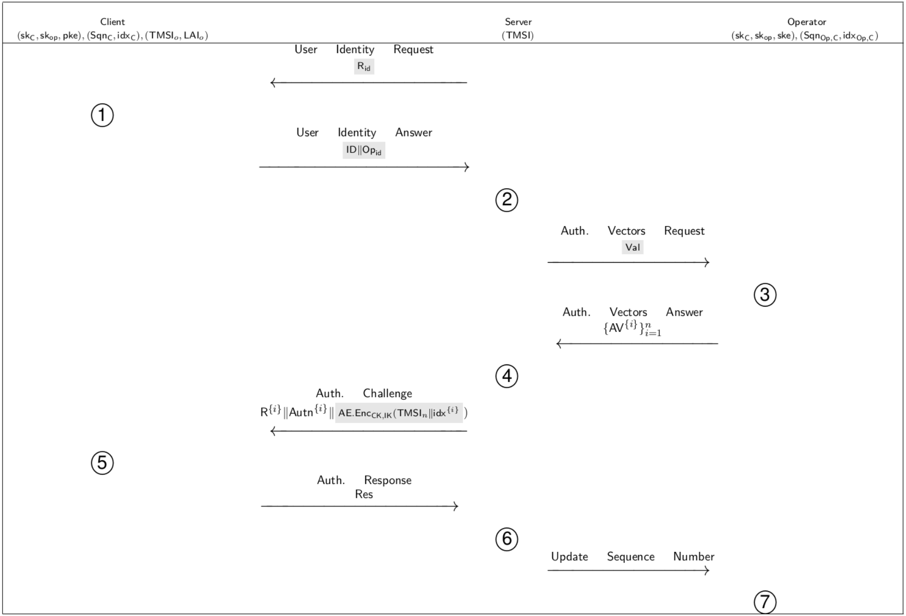
The authors of [Fouque et al., 2016] propose a new protocol, PRIV-AKA (claimed unlinkable).

## 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.
- 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.

# PRIV-AKA [Fouque et al., 2016]



# PRIV-AKA [Fouque et al., 2016]

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