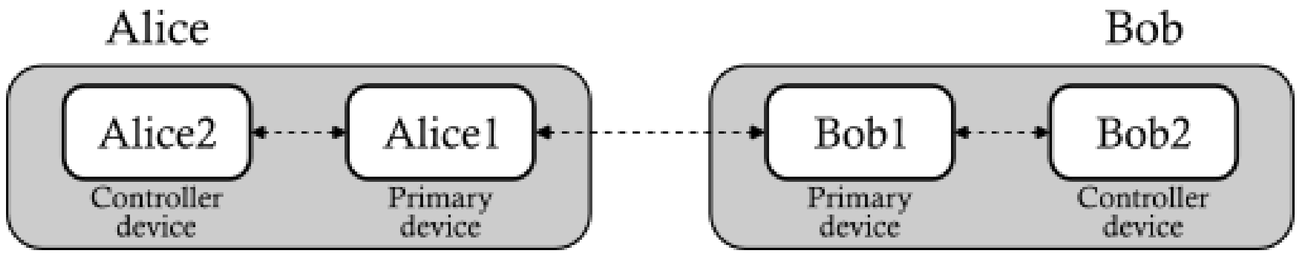
Gizem Akman, Valtteri Niemi
University of Helsinki
Helsinki Institute for Information Technology (HIIT)
Philip Ginzboorg
Aalto University
XAMK
Sampo Sovio
Huawei Technologies Finland

Policy enforcement with split keys



Security Policies

- Sets of rules to decide which people should be granted access to an asset.
- Used for network access control, cloud computing, IoT environments, mobile networks, and enterprise security.

Contributions

(1) propose two variants of mutual policy enforcement (MPE) protocol: signature-based (MPE-SIGN) and MAC-based (MPE-MAC) protocols,
(2) identify several use cases where the MPE protocol can be applied, and
(3) implement the model of MPE-SIGN and MPE-MAC protocols in ProVerif and formally verify their security properties.

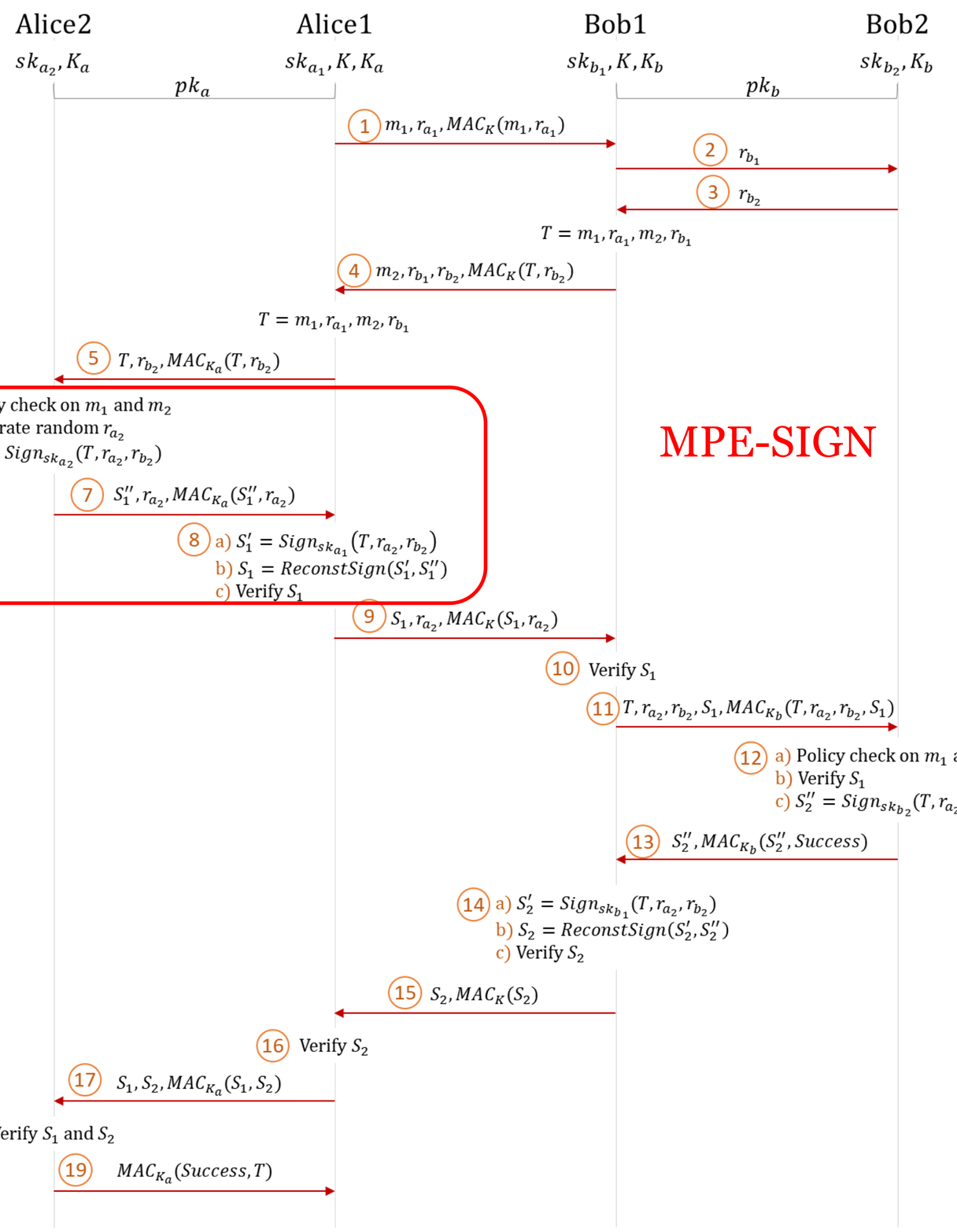
Part of MPE-MAC:

6 a) Policy check on T
b) $M_{13} = MAC_{K_{21}}(T, r_{a_2}, r_{b_2})$
c) $M_{14} = MAC_{K_{22}}(T, r_{a_2}, r_{b_2})$

7 $M_{13}, M_{14}, r_{a_2}, MAC_{K_a}(M_{13}, M_{14}, r_{a_2})$ →

8 a) $M_{23} = MAC_{K_{11}}(T, r_{a_2}, r_{b_2}, M_{13}, M_{14})$
b) $M_{24} = MAC_{K_{12}}(T, r_{a_2}, r_{b_2}, M_{13}, M_{14})$

Main differences between MPE-MAC and MPE-SIGN



MPE-SIGN

Use Cases

- Communicating smart devices
  - Cars in traffic
  - Drones
  - Devices in a smart environment
- Simple IoT devices that collect data
- Ensuring honest-but-curious behavior in multi-party computation
- Machine-to-Machine contract negotiation

MPE-SIGN vs. MPE-MAC

- If the use case is time-sensitive and requires repetition, the MPE-MAC protocol would be more suitable to use because it would incur less computational and communicational costs.
- On the other hand, MPE-SIGN would be more suitable for use cases that require non-repudiation because the signature is involved.

Formal Verification

- We used ProVerif tool for formal verification of our protocols.

```
Query inj-event(Alice1FINISHED(a1,b1,m1,m2)) ==> inj-event(Bob1FINISHED(a1,b1,m1,m2)) && inj-event(Alice2FINISHED(a2,a1,m1,m2)
) && inj-event(Bob2FINISHED(b2,b1,m1,m2)) && inj-event(Alice2PolicyCheck(a2,a1,m1,m2)) && inj-event(Bob2PolicyCheck(b2,b1,m1,m
2)) is true.

Query inj-event(Bob1FINISHED(a1,b1,m1,m2)) ==> inj-event(Alice1sendS1(a1,b1,m1,m2)) && inj-event(Alice2START(a2,a1,m1,m2)) &&
inj-event(Bob2FINISHED(b2,b1,m1,m2)) && inj-event(Alice2PolicyCheck(a2,a1,m1,m2)) && inj-event(Bob2PolicyCheck(b2,b1,m1,m2)) i
s true.

Query inj-event(Alice2FINISHED(a2,a1,m1,m2)) ==> inj-event(Alice1verifyS2(a1,b1,m1,m2)) && inj-event(Bob1FINISHED(a1,b1,m1,m2)
) && inj-event(Bob2FINISHED(b2,b1,m1,m2)) && inj-event(Alice2PolicyCheck(a2,a1,m1,m2)) && inj-event(Bob2PolicyCheck(b2,b1,m1,m
2)) is true.

Query inj-event(Bob2FINISHED(b2,b1,m1,m2)) ==> inj-event(Alice1sendS1(a1,b1,m1,m2)) && inj-event(Alice2START(a2,a1,m1,m2)) &&
inj-event(Bob1sendS1(a1,b1,m1,m2)) && inj-event(Alice2PolicyCheck(a2,a1,m1,m2)) && inj-event(Bob2PolicyCheck(b2,b1,m1,m2)) is
true.
```