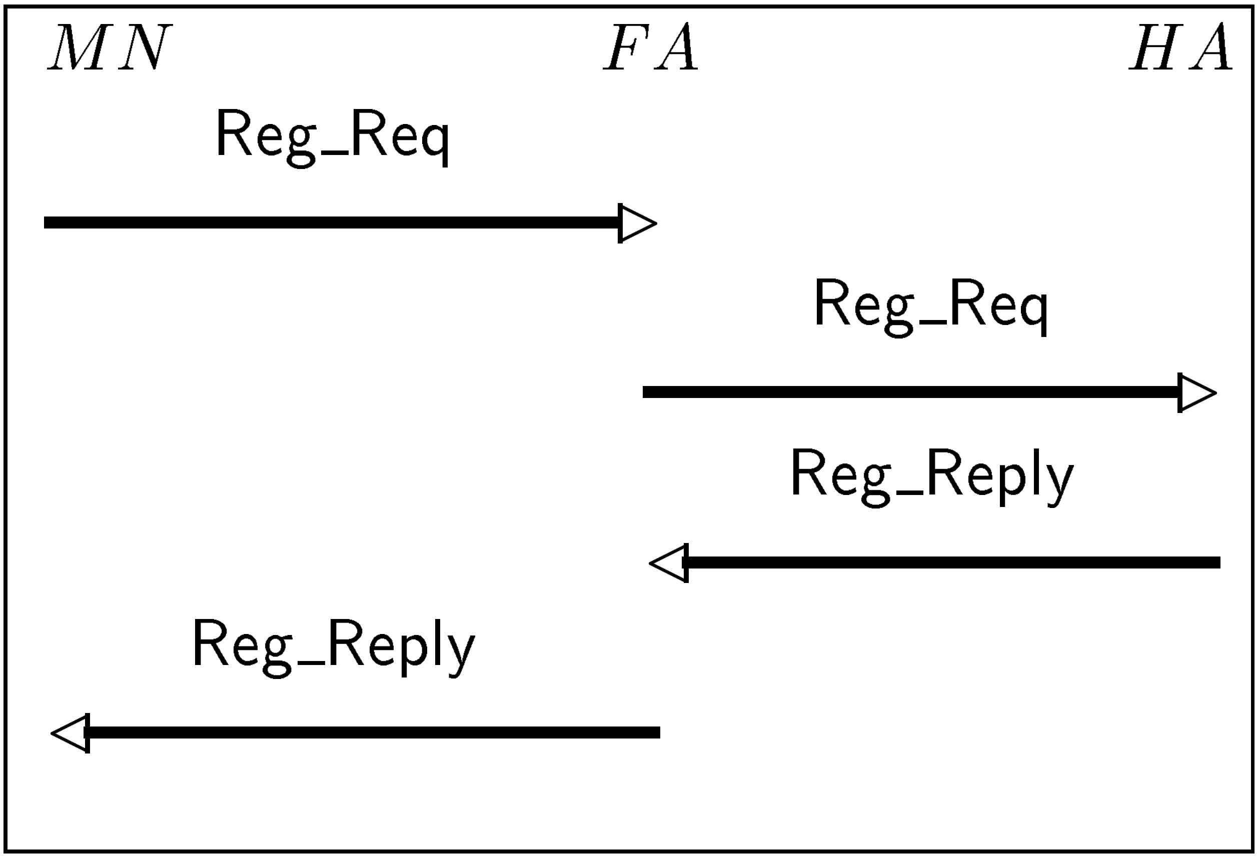
Security for Mobile IP

Amir Herzberg   Hugo Krawczyk   Charlie Perkins

IBM T. J. Watson Research Center

Hawthorne, New York

Quick Review of Mobile IP



Mobile-IP messages are authenticated using all available keys (K_{MH}, K_{FH}, K_{MF}).

Security Requirements for Mobility

- Authenticated bind: ‘MN is at FA’
- Authorization of MN to FA by HA
- Authorization of FA to MN by HA
- Securing the ‘air-link’ (MN to FA)
- Anonymity and Location Privacy
- Higher layers: access control, billing, . . .

Alternative Possible Assumptions

Too Strong:

- Trust FA, HA-FA link (GSM, CDPD)
- Only passive attacks on MN-FA (air) link (CDPD)

Reasonable:

- Trust FA only for anonymity of HA and for integrity/confidentiality after registration.
- Assume a prior FA-HA key.

Message Authentication in Mobile-IP

- Message authentication function $MAC_k()$: knowing $MAC_k(y)$ for choosen y does not reveal $MAC_k(x)$.

- Default MAC in Mobile-IP:
  $MAC_k(x) = MD5(k||x||k)$

- $MD5$ is hash function: hard to find α and β such that $MD5(\alpha) = MD5(\beta)$.

- Assumption: $MD5$ with key prefix and suffix is a good MAC.

- Good hash and bad MAC:
  $hash(k||x) = k||MD5(x)$.

Notations for Message Authentication (MAC):

- $\{m\}_K$ is message m together with its MAC using key K.
- $\{m\}_{MH}$ is message m together with its MAC using key K_{MH} shared between MN and HA.
- $\{m\}_{FH}$ is message m together with its MAC using key K_{FH} shared between FA and HA.
- $\{m\}_{MF}$ is message m together with its MAC using key K_{MF} shared between MN and FA.

Bind Authentication

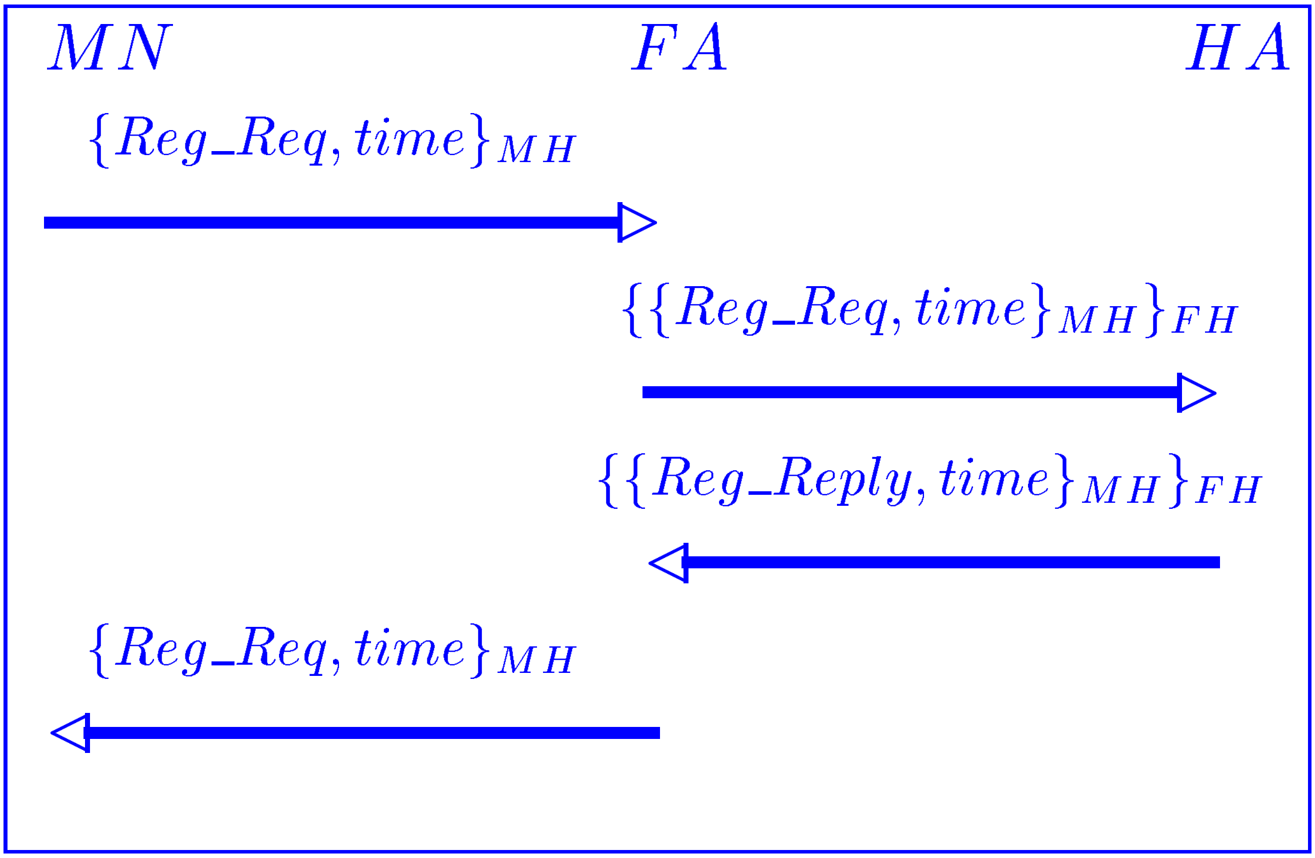
Goal: Bind ('HA at FA') agreed by both HA and FA.

Requires: HA verify reg_req authenticated and fresh.

Preventing replay:

- <u>secure</u> <u>global</u> <u>synchronized</u> time (NTP, GPS)
- resetable clocks (usually synchronized)
- random challenges

Preventing replays using secure NTP



Pros: Simple, no state, no extra flows.

Cons: Requires <u>secure</u> NTP in MN <u>and</u> HA.

Preventing replays using resetable clocks

- Same, but MN may lose sync.

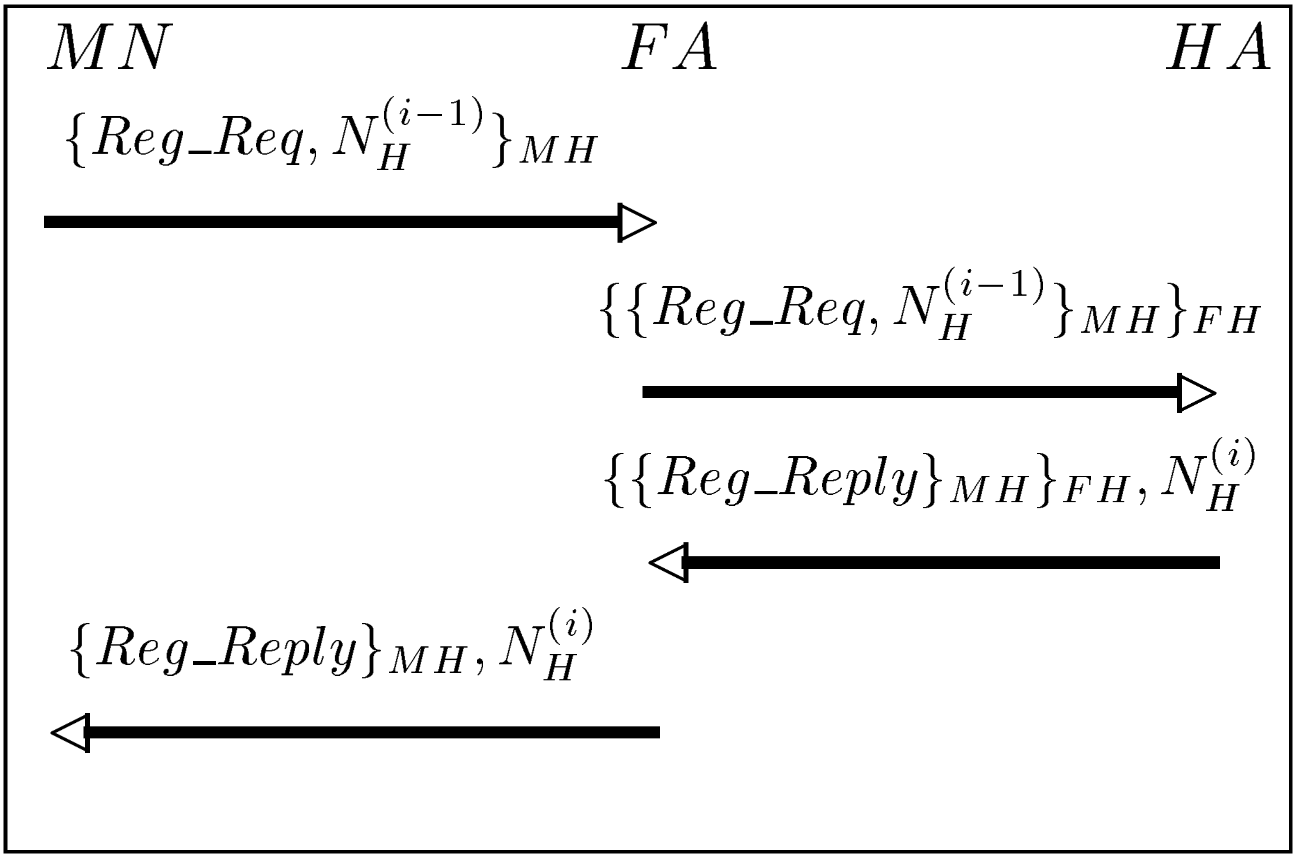
- If MN lost sync, it uses time from rejection reg_reply.

Pros: No NTP, no state, 2 flows in sync.

Cons:
- Adds 2 flows for MN when time lost.
- Separate MN clock.
- Secure time in HA (secure NTP?)
- Lunch-break attack (change clock $\Rightarrow$ future reg_req)
- Offline, chosen plaintext attack

Preventing replays with nonces

Use saved nonce N_H from last registration.



Pros: No NTP, normally 2 flows, more secure.

Cons: 2 additional flows if state lost. State for nonces in HA. Extra bandwidth for next nonce.

Dealing with Retransmissions and Loss

Bad Scenario on Noisy Line:

1. HA receives reg_req with nonce $N_H^{(i)}$.
2. reg_reply with $N_H^{(i+1)}$ is lost.
3. MN resends reg_req with $N_H^{(i)}$.
4. HA rejects, sends new nonce $N_H^{(i+1)}$.
5. $i \leftarrow i + 1$; MN sends another reg_req
6. repeat from 1...

Dealing with Retransmissions and Loss

Correct: HA keeps previous nonce.

1. HA receives reg_req with nonce $N_H^{(i)}$.

2. reg_reply with $N_H^{(i+1)}$ is lost.

3. MN resends reg_req with $N_H^{(i)}$.

4. HA identifies a retransmission, sends OK reg_reply with $N_H^{(i+1)}$ (lifetime adjusted)

5. MN receives one (or both) replies and all is well!

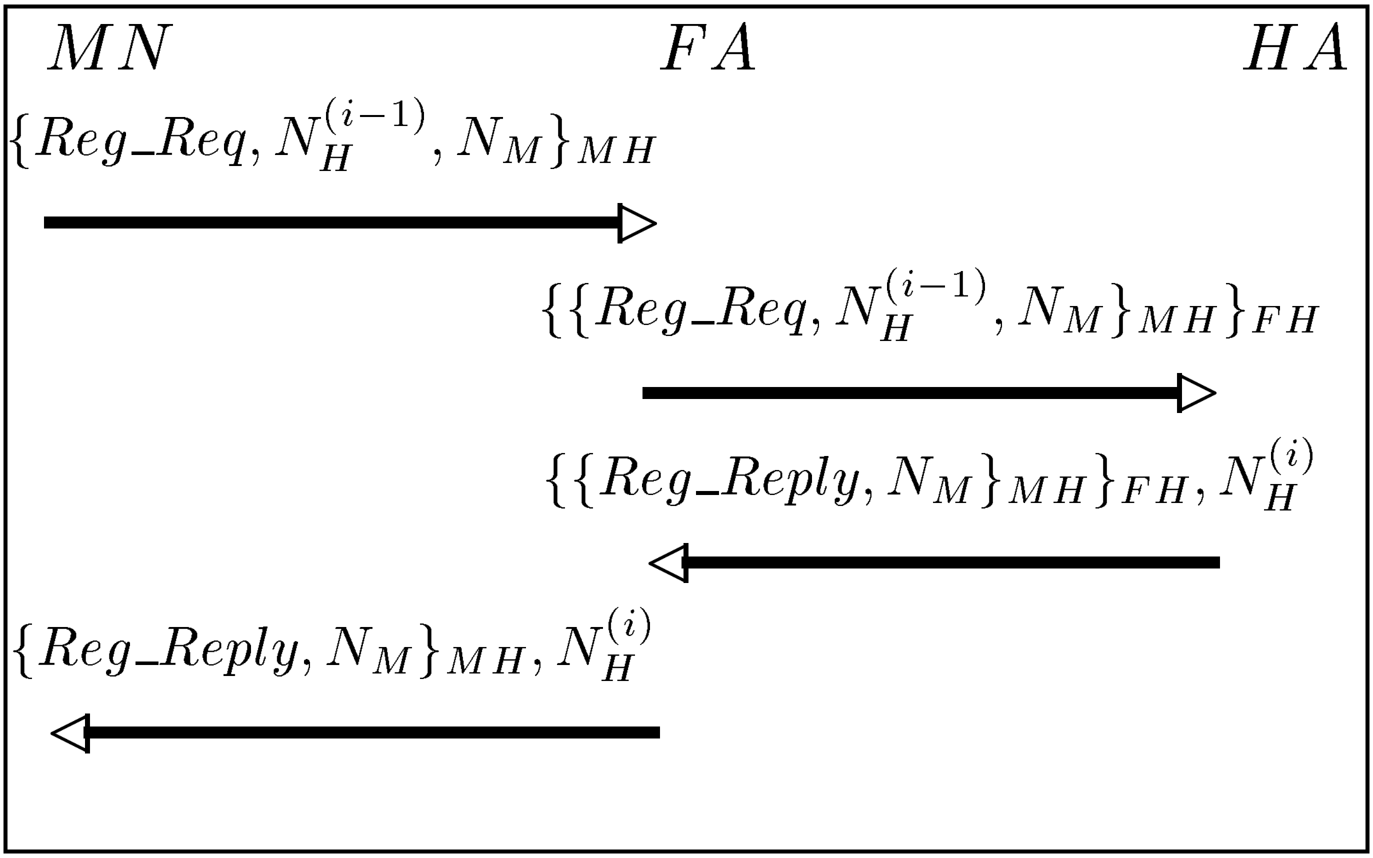
Denial of Service Attack

Attack: flood the MN with old rejection reg_reply messages, causing it to use wrong nonces in the reg_req messages.

Note: attacker should also remove every reg_req which uses the correct nonce. Is this a risk?

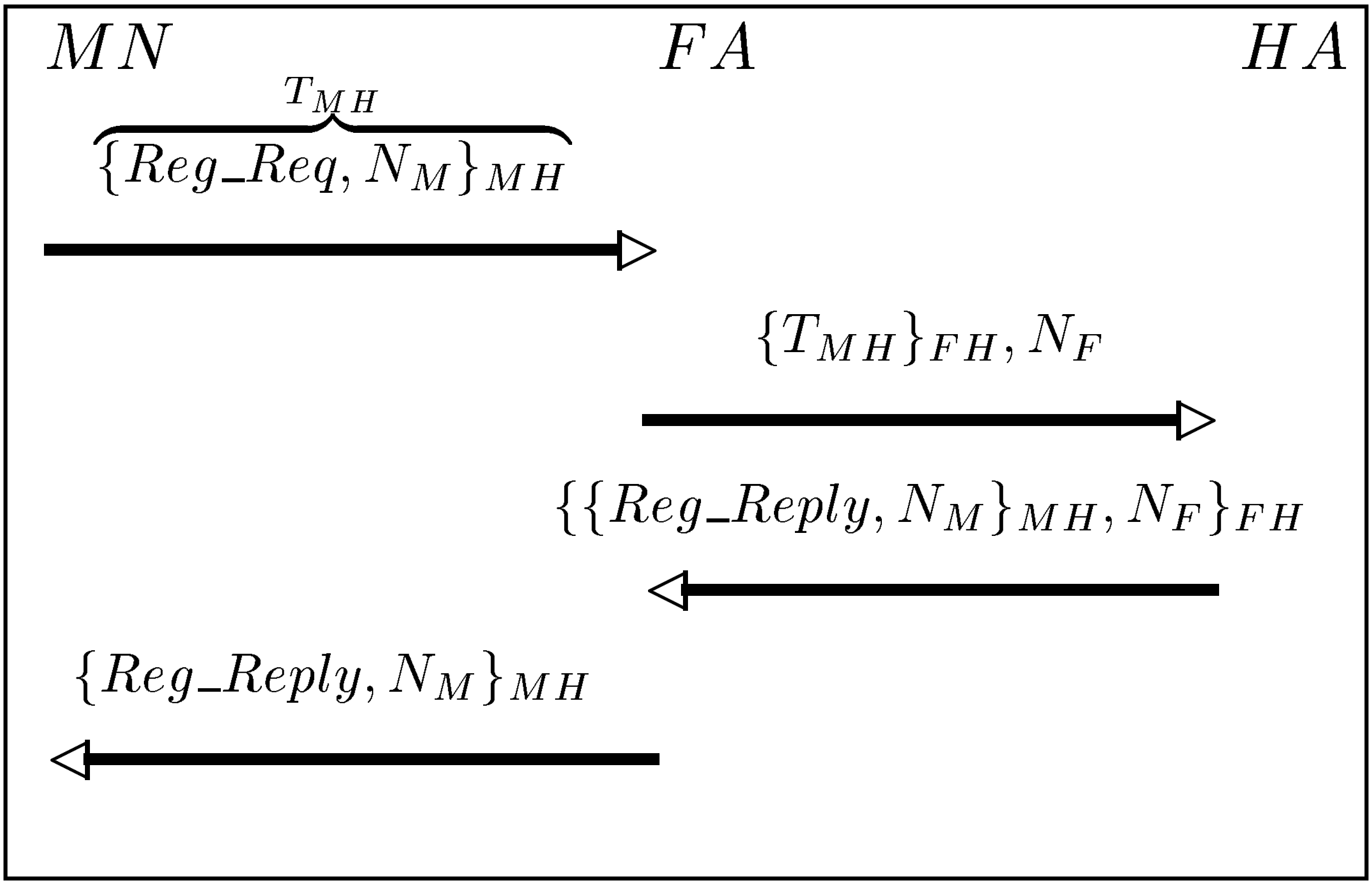
Fix: MN sends its own nonce N_M with reg_req, ignores reg_reply without N_M.

Nonce Protocol with Anti-Flooding:



Authorization to FA and MN

Solution: use nonces N_F, N_M.

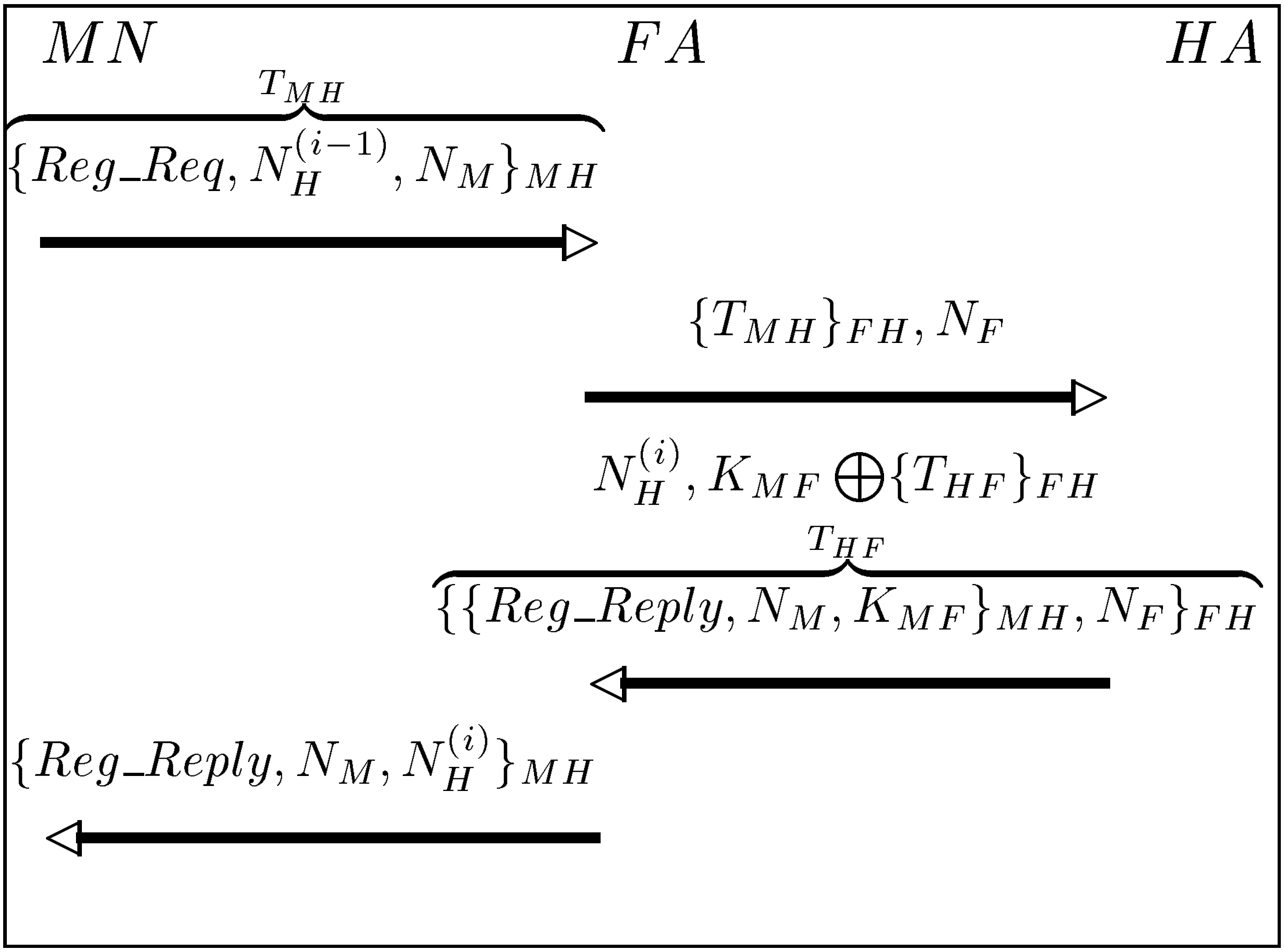


Note: $sizeof(nonces) \geq 32b$!!

Supplying Session Key to MN and FA

- Who needs session key K_{MF}?
  - Secure communication over ‘air-link’ (if no end-to-end security): encryption, message authentication.
  - FA may need to authenticate messages to prevent fraud (‘hitchhiking’ on a MN).
- Alternatives to provide session key:
  - Assume FA and MN have it.
  - To be done outside mobile-IP.
  - An extension of mobile-IP.

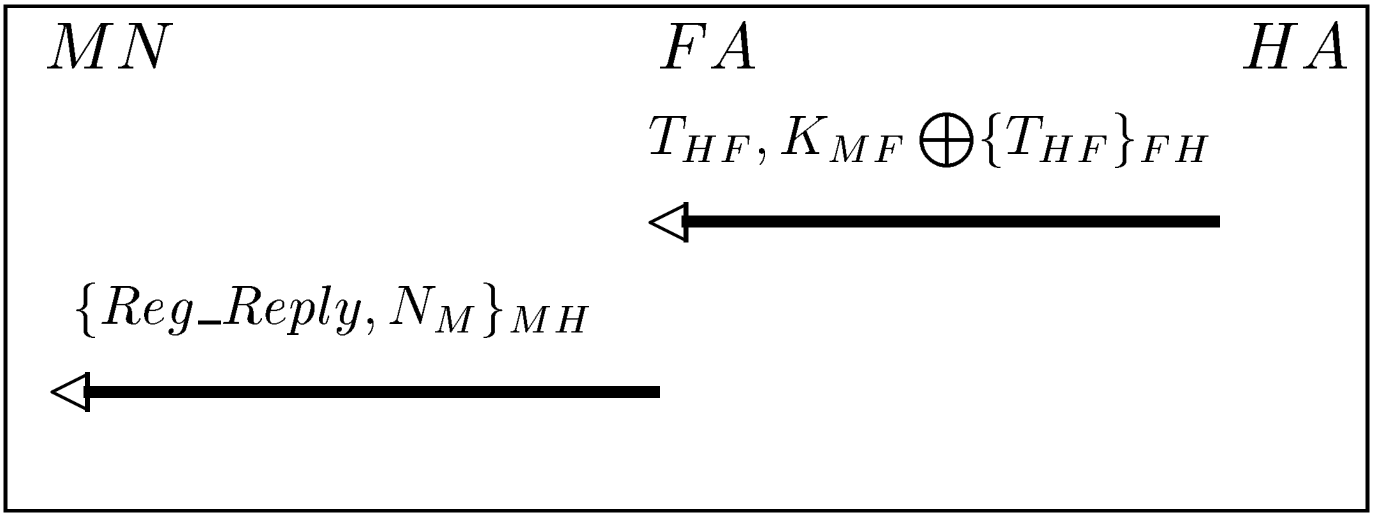
Mobile-IP with Session Key Extension



Note: K_{MF} is only authenticated, not explicitly sent.

How is K_{MF} passed?

- $K_{MF} = \{N_H^{(i)}\}_{MH}$ so MN, HA can compute it.
- FA computes K_{MF} from $\{T_{HF}\}_{FH}$.
- Both MN and FA authenticate K_{MF}.
- Nonces (N_M, N_F) ensure no replay of keys.



Overhead of Session Key exchange

- In MN: one more MAC.
- In FA: one more MAC.
- In HA: two more MAC.
- Communication on air: None.
- Communication between HA and FA: one more block (64bits?).
- Exportable (req. commerce dept approval)

Security Checklist

- $\surd$ Message Authentication: only MN or HA can send.
- $\surd$ Identification of HA vs. MN: different formats for reg_req and reg_reply.
- $\surd$ No replay: ping-pong of nonces.
- $\surd$ Key revealed $\Rightarrow$ MAC is not pseudorandom function. Hence: if MAC is 'good' pseudo-random function, key not revealed.

Generating Randomness for Nonces, Keys

- Initial value: use LSbits of timer, measured time between user key strokes, noise diode...
- Generation: use pseudo-random generator, $r_i = f(r_{i-1})$, e.g. $r_i = MD5(C_1, r_{i-1}, C_2)$ where C_1, C_2 are constants.
- In computation, include randomness from other sources: $r_i = f(r_{i-1}, noise)$.

Anonymity and Privacy Issues

- What is to be protected:
  - The identity of the mobile.
  - The identity of the home.
  - The location of the mobile
- Who is it protected against:
  - An eavesdropper on the air-link.
  - An eavesdropper on FA-HA link.
  - The FA.
  - Other FAs.
- Which kinds of attacks.

Anonymity solutions

- Hide mobile identity:
  - Different pseudonym $K_H(ID_M, salt)$ each registration
  - Key K_H known only to Home
  - New pseudonym passed encrypted
  - No state/table in Home
- Hide home identity:
  - First registration of MN at FA: Reveal. Alternatives: Diffie-Hellman (only against passive attack) or trusted 'clearing house'.
  - Subsequent registrations: use $K_F(ID_H, salt)$ similar to above