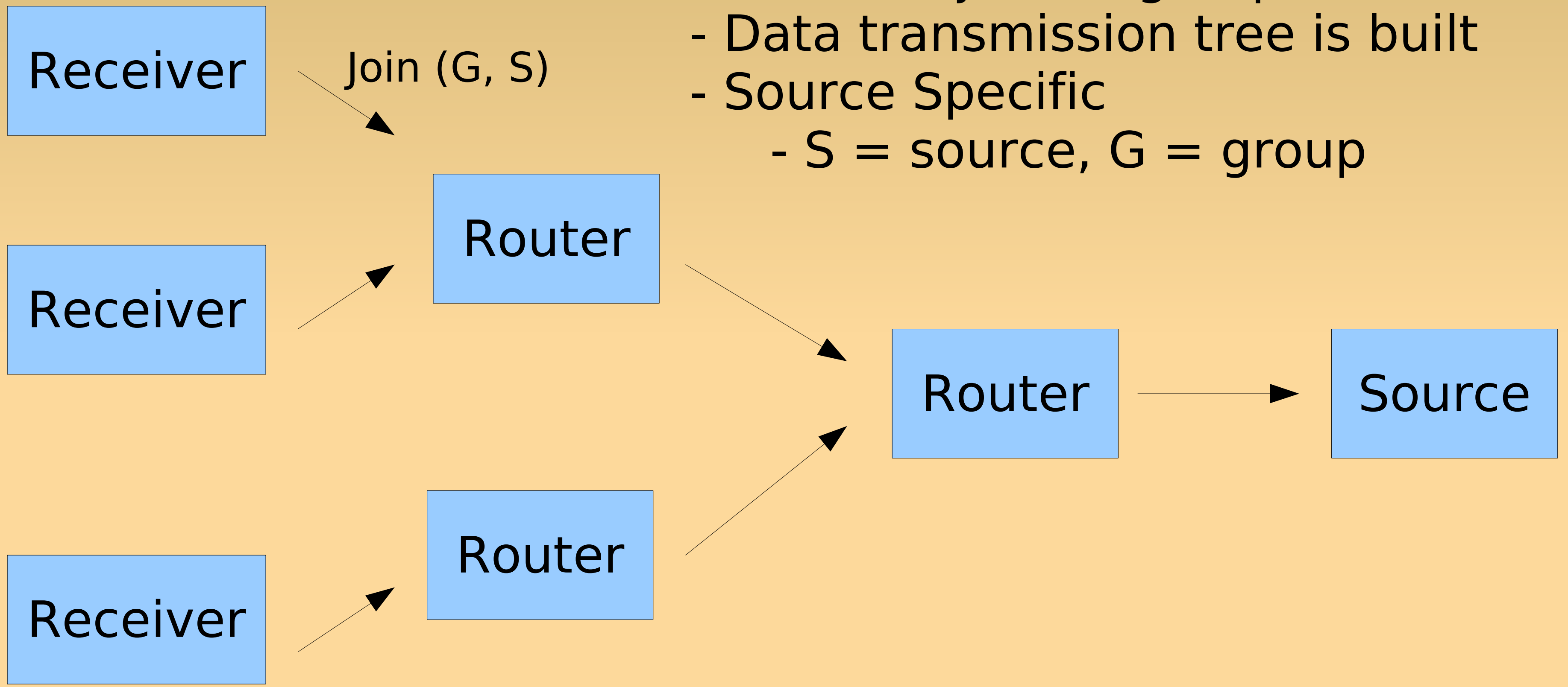
Key Management in IP Multicast

17.11.2006

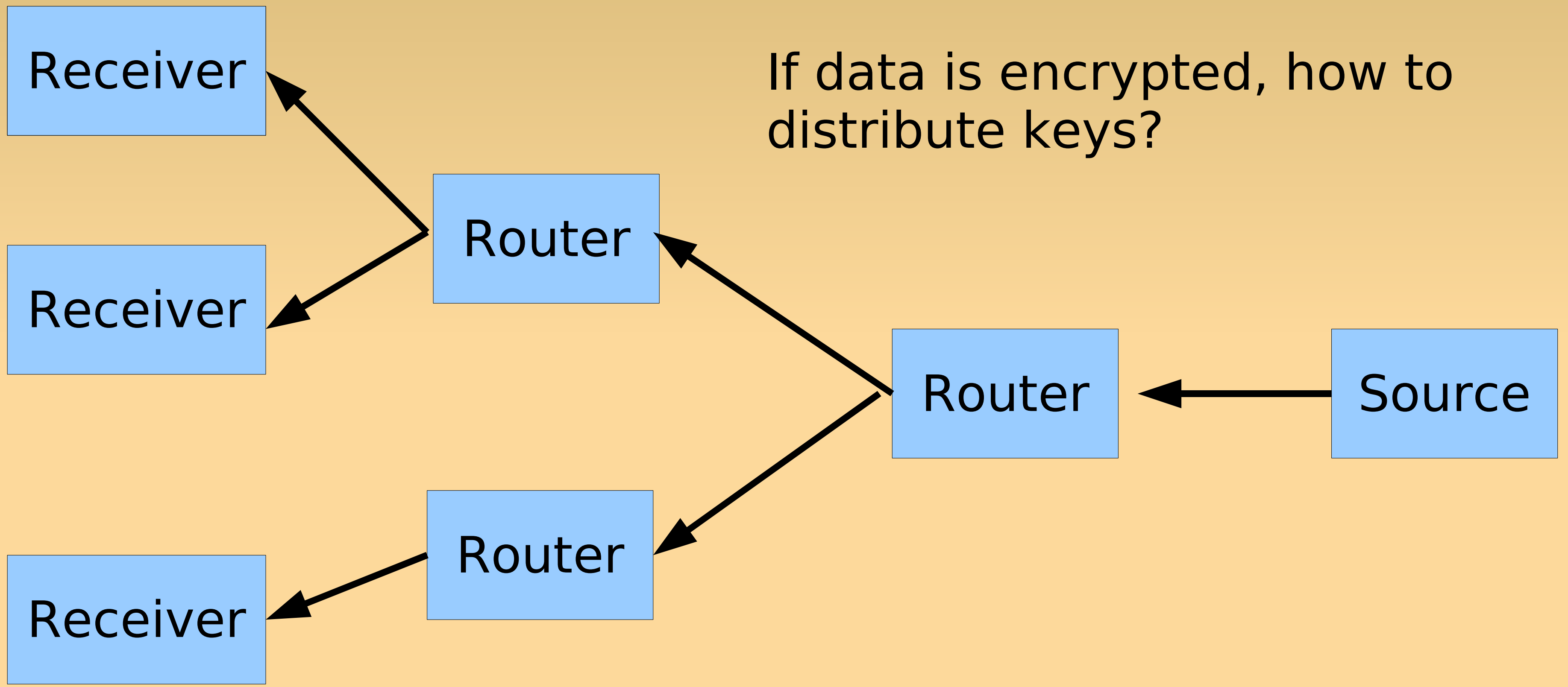
Petri Jokela
petri.jokela@nomadiclab.com

IP multicast

- Receiver joins a group
- Data transmission tree is built
- Source Specific
    - S = source, G = group



IP multicast



Data is multiplied at routers

If data is encrypted, how to distribute keys?

Multicast Security

- IETF Multicast Security (MSEC) WG
- Initial target:
  - Security for one source, large number of receivers
- Currently:
  - Finish current work before next summer
  - Rechartering?

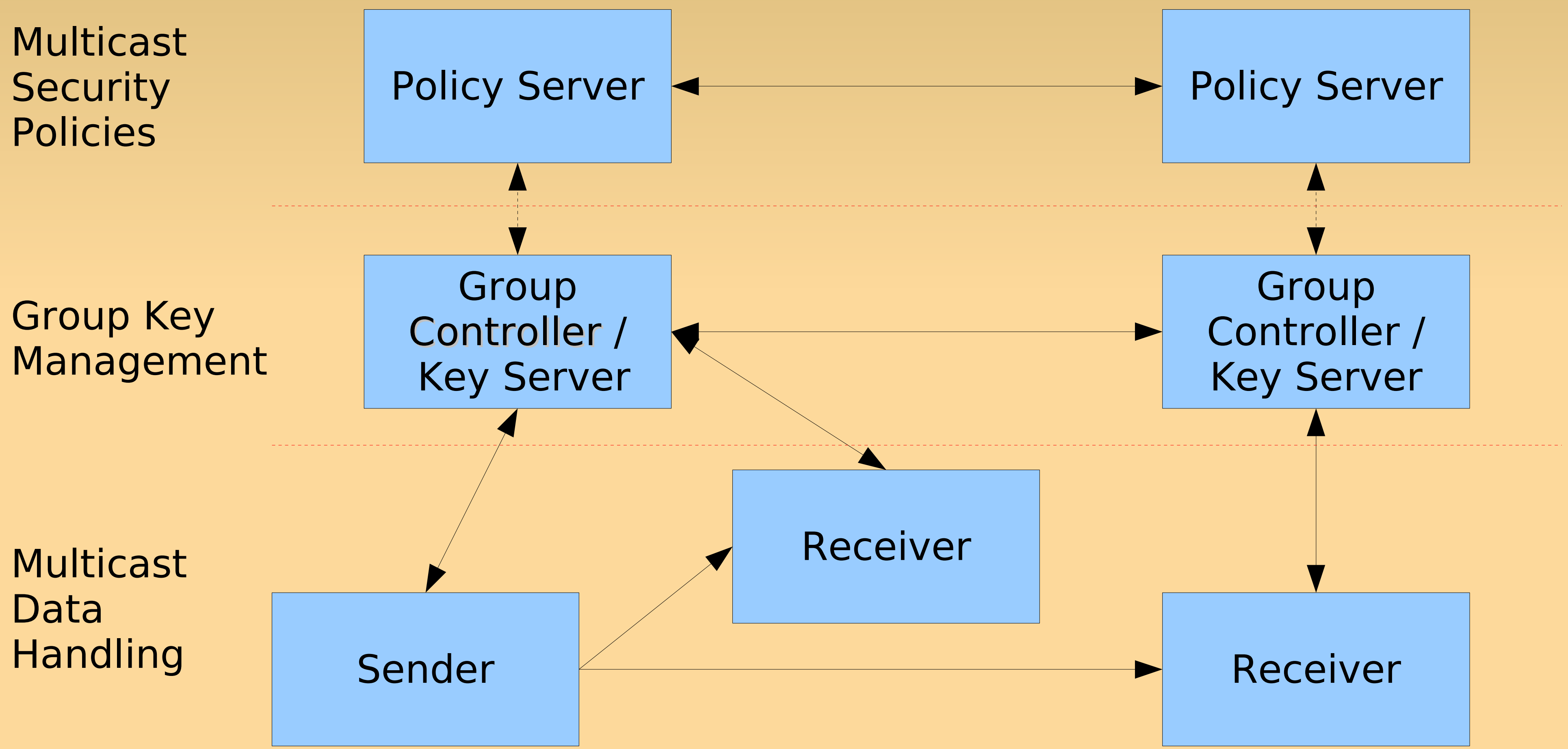
Multicast Security issues

- Integrity of data
  - Receiver: data is not modified
- Secrecy
  - Data cannot be seen by non-group members
- Source authentication
  - The data is coming from the correct source
  - With shared traffic encryption keys, requires other functions in IP multicast
  - Not considered in Key Management

Keying

- Keys
  - Shared: Traffic Enc. Key (TEK), Key Enc. Keys (KEK)
  - Point-to-Point: Registration association
- Problem: How to distribute shared keys?
  - Currently we have centralized server
- Use point-to-point link to deliver the KEK
  - Use a KEK to encrypt TEK; deliver e.g. using multicast data path
- Re-key: member joins or leaves a group

Security Architecture

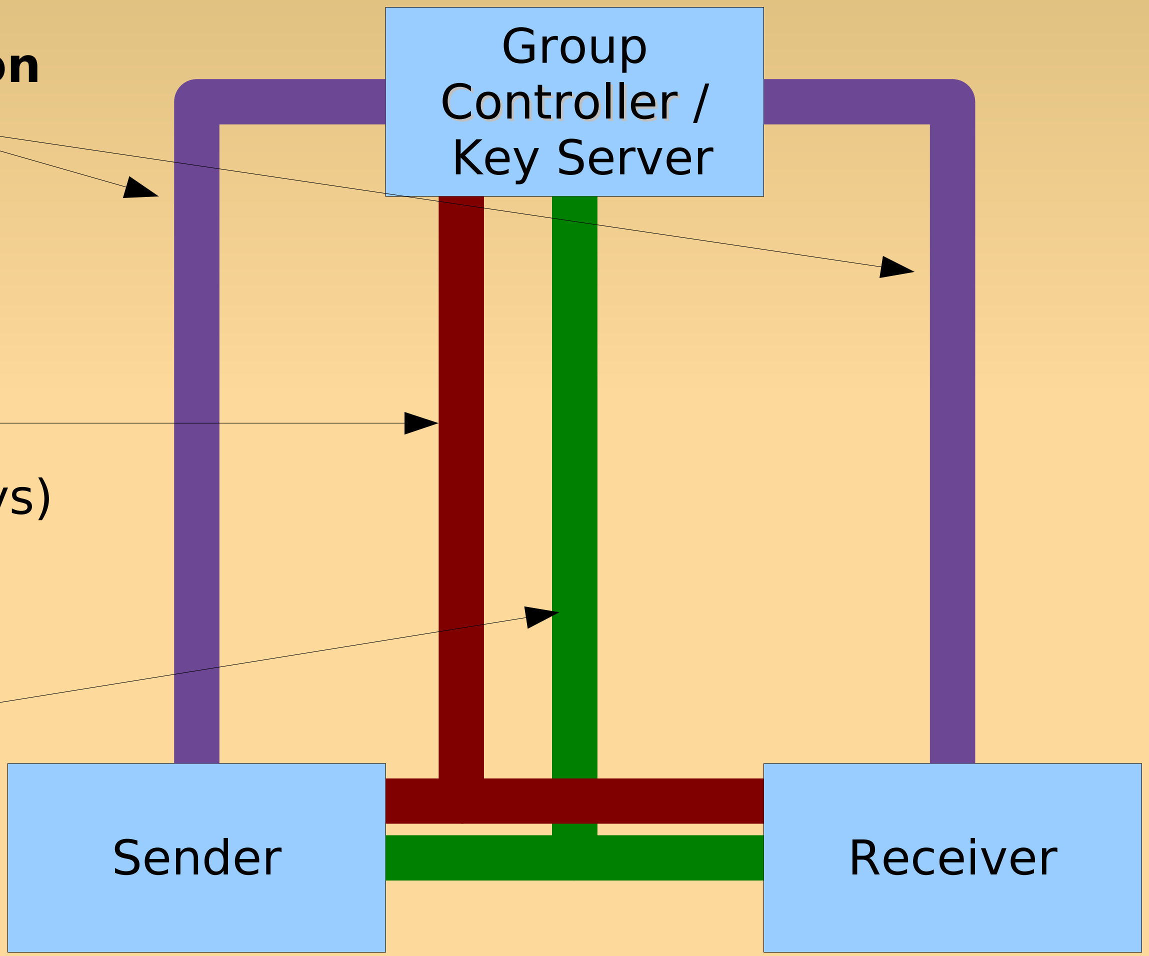


Group Security Association

Registration association
- point-to-point

Re-key association
- Shared keys
    - o (Key Encryption Keys)
- Re-key message e.g. using multicast

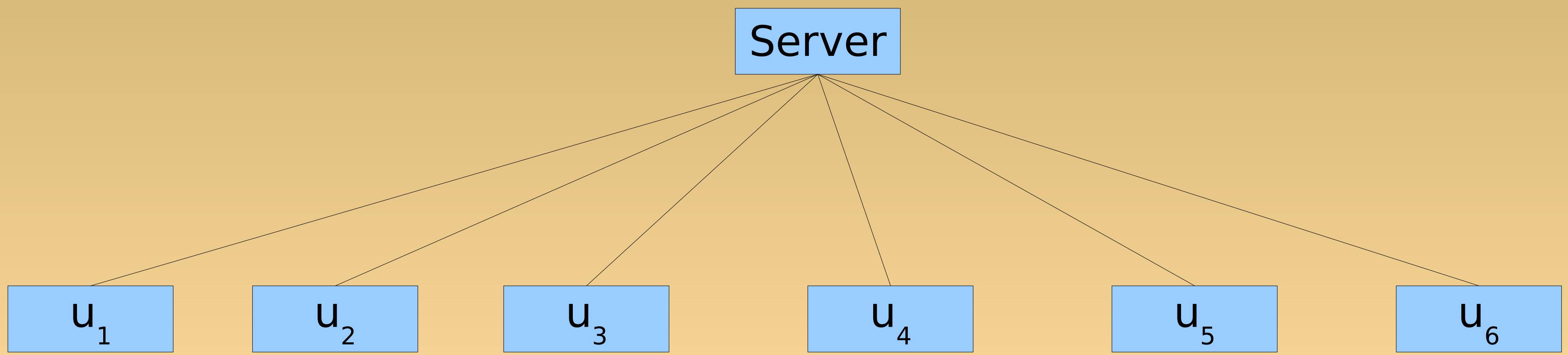
Data association
- Shared key
- Data transmission



Logical Key Hierarchy

Logical Key Hierarchy

Keys and hierarchy



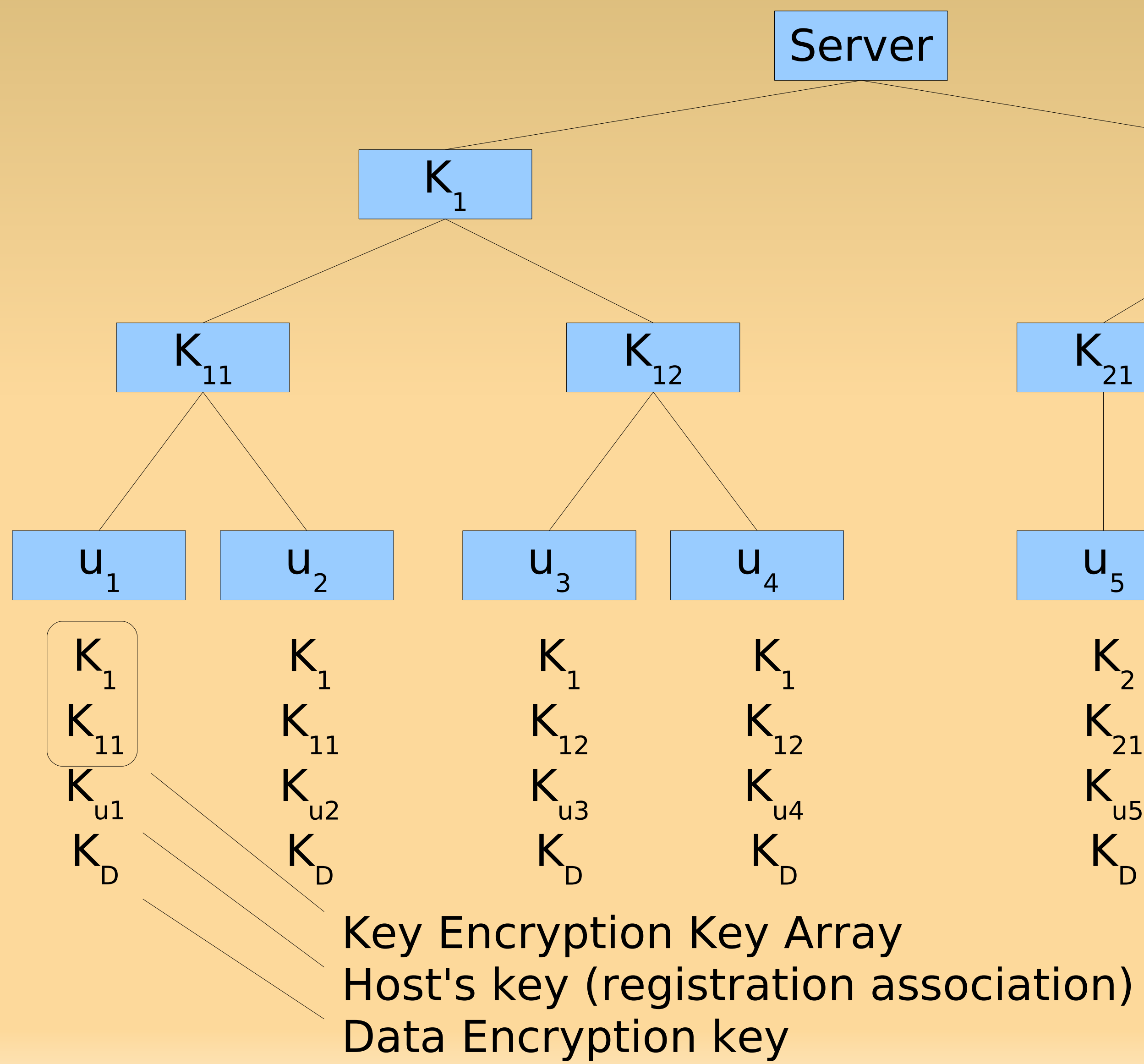
- Change in group (originally n users)
  - Only TEK + Ku_n
    - join: n+1 encryptions (TEK with Ku)
    - leave: n-1 encryptions (TEK with Ku)
  - TEK + 1 KEK + Ku_n
    - join: 1 encr. (TEK with KEK), n encr. (TEK with Ku)
    - leave: n-1 encr. (KEK with Ku), 1 encr (TEK with KEK)

Logical Key Hierarchy

- RFC2627
- Key Encryption Keys hierarchically
  - Less encryption operations
  - Less transmitted messages
- GSAKMP and GDOI define this as optional
- Defined but is it used?
  - E.g. not in 3GPP

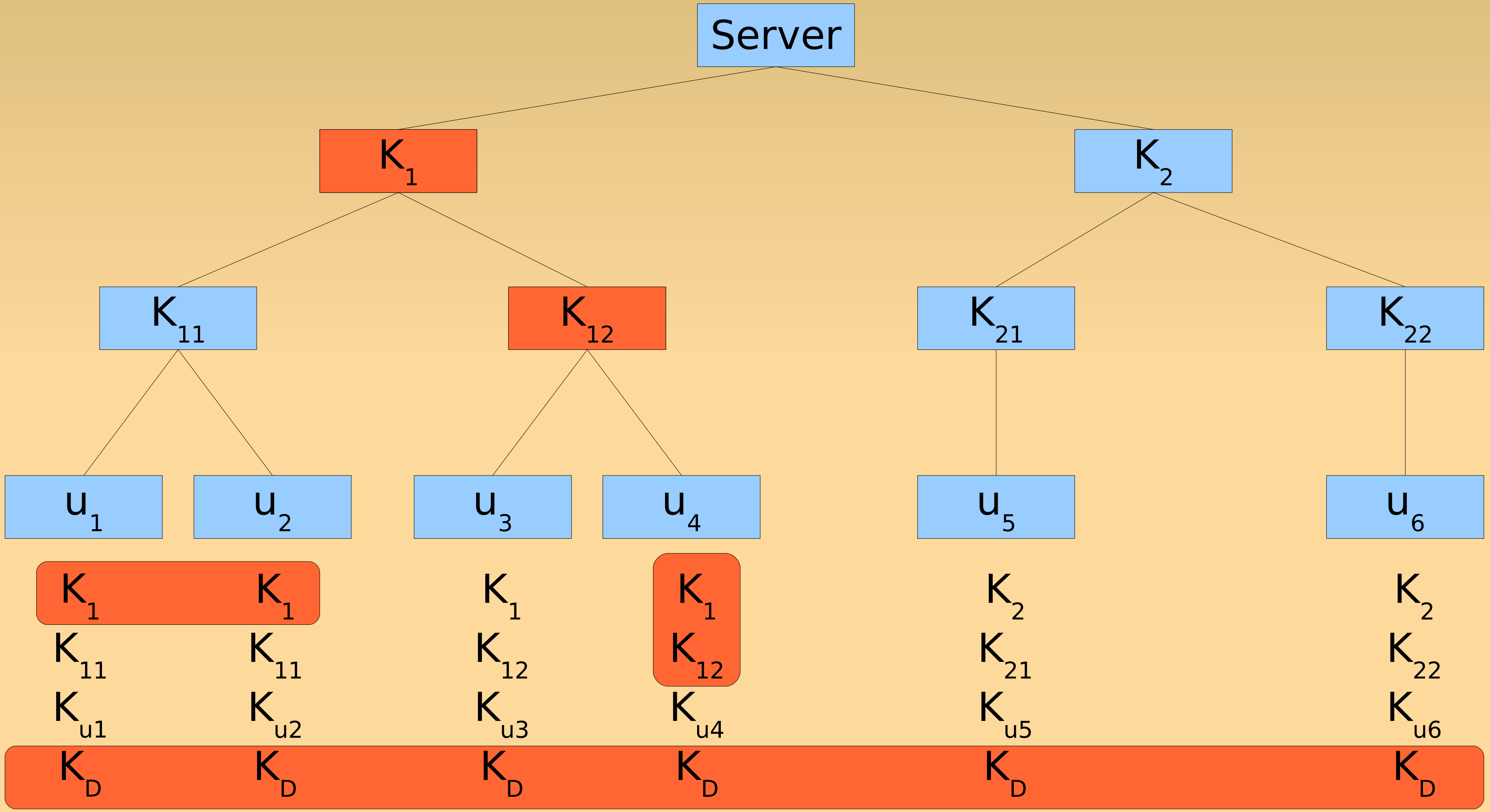
Logical Key Hierarchy

Keys and hierarchy



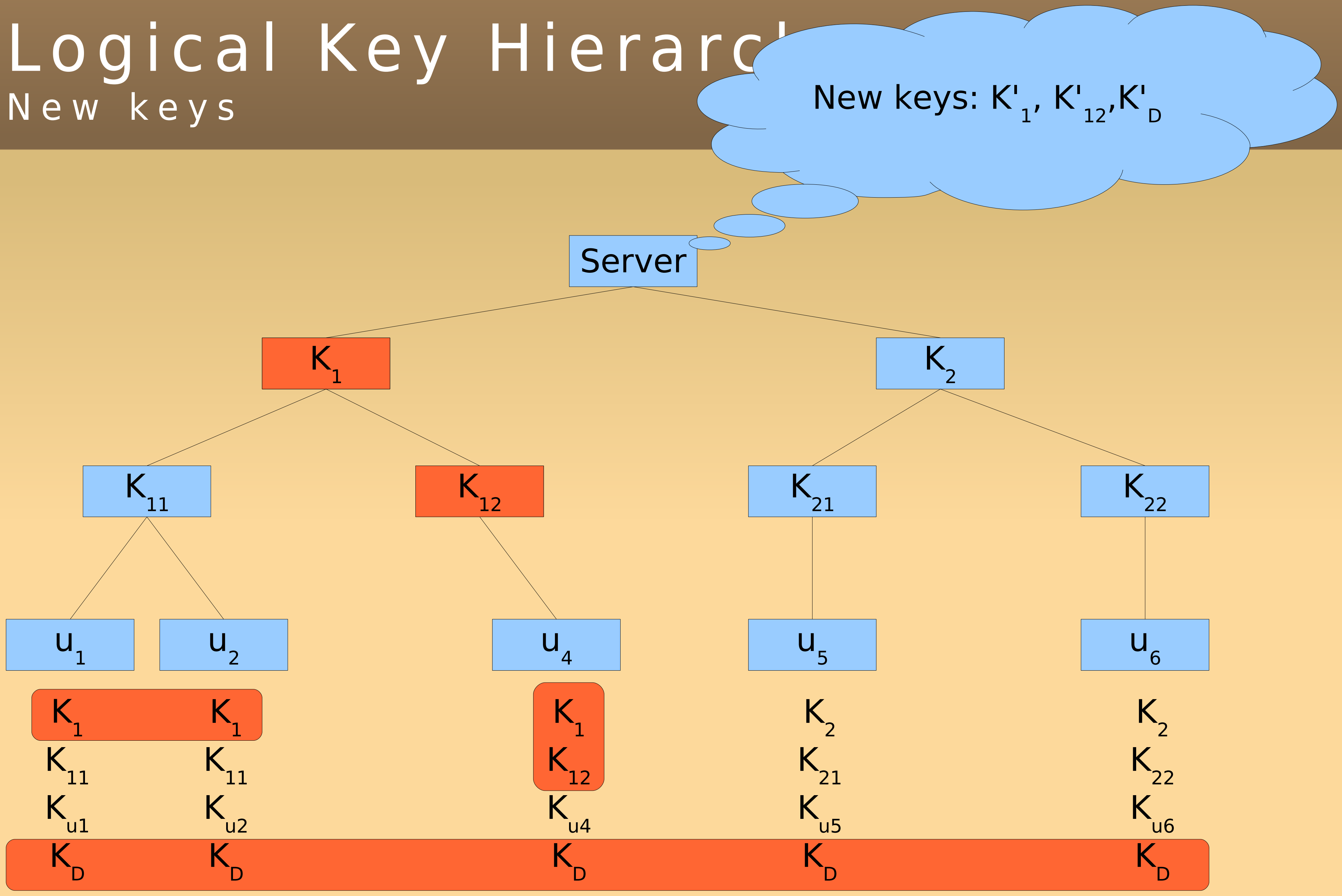
Logical Key Hierarchy

Node leaving, keys that have to be renewed



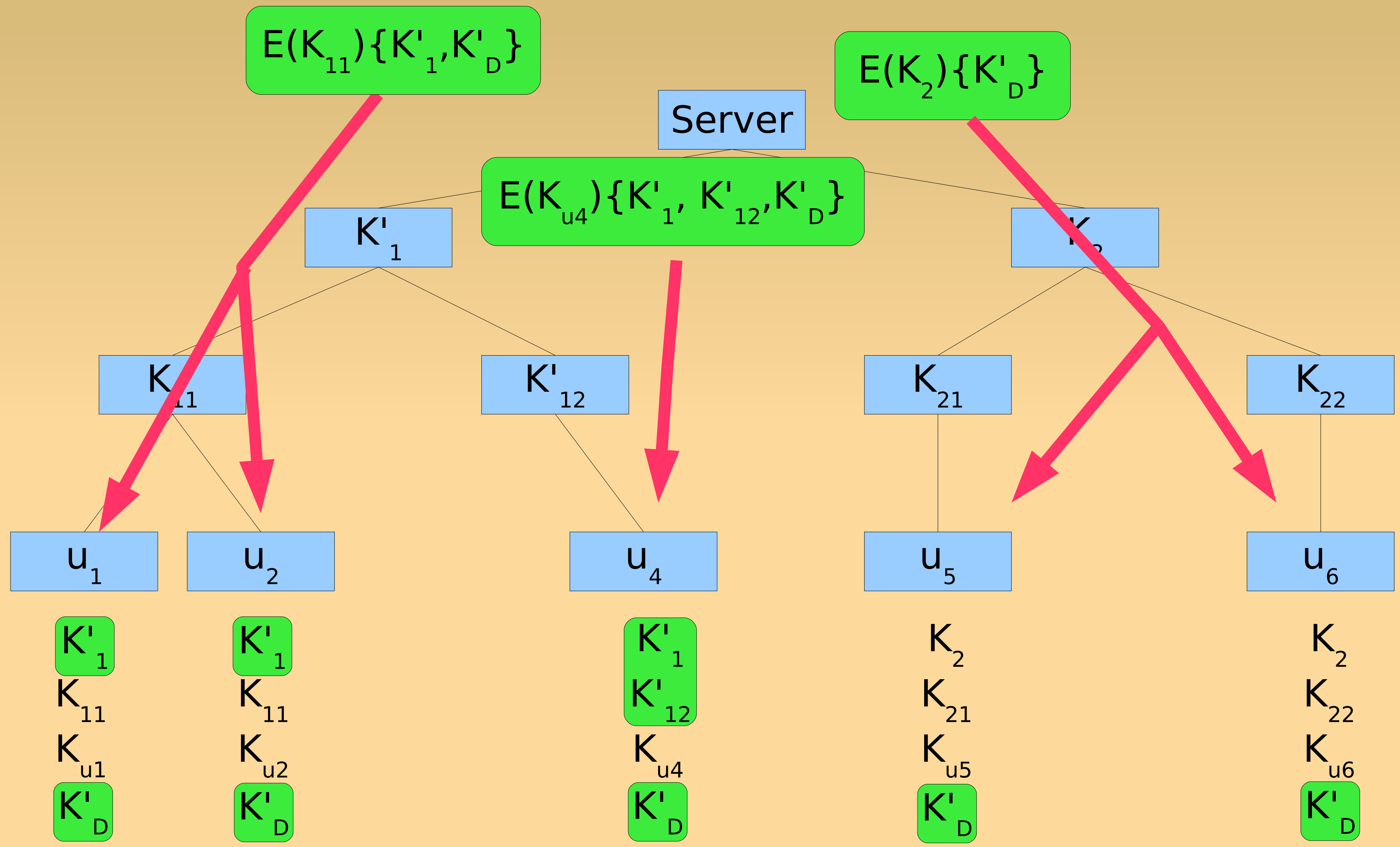
Logical Key Hierarch

New keys



Logical Key Hierarchy

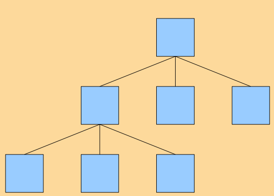
Keys and hierarchy

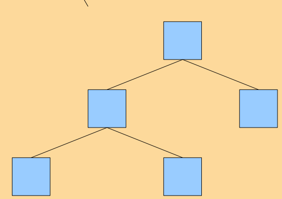


Logical Key Hierarchy

Table of required storage and re-key transmissions

Users	*Degree*	*Storage per User*	*Re-key transmissions (single key)*	*Re-key transmissions (multi key)*
8	2	4	5	3
9	3	3	5	4
16	2	5	7	4
2048	2	12	21	11
2187	3	8	20	14
131072	2	18	33	17
177147	3	12	32	22



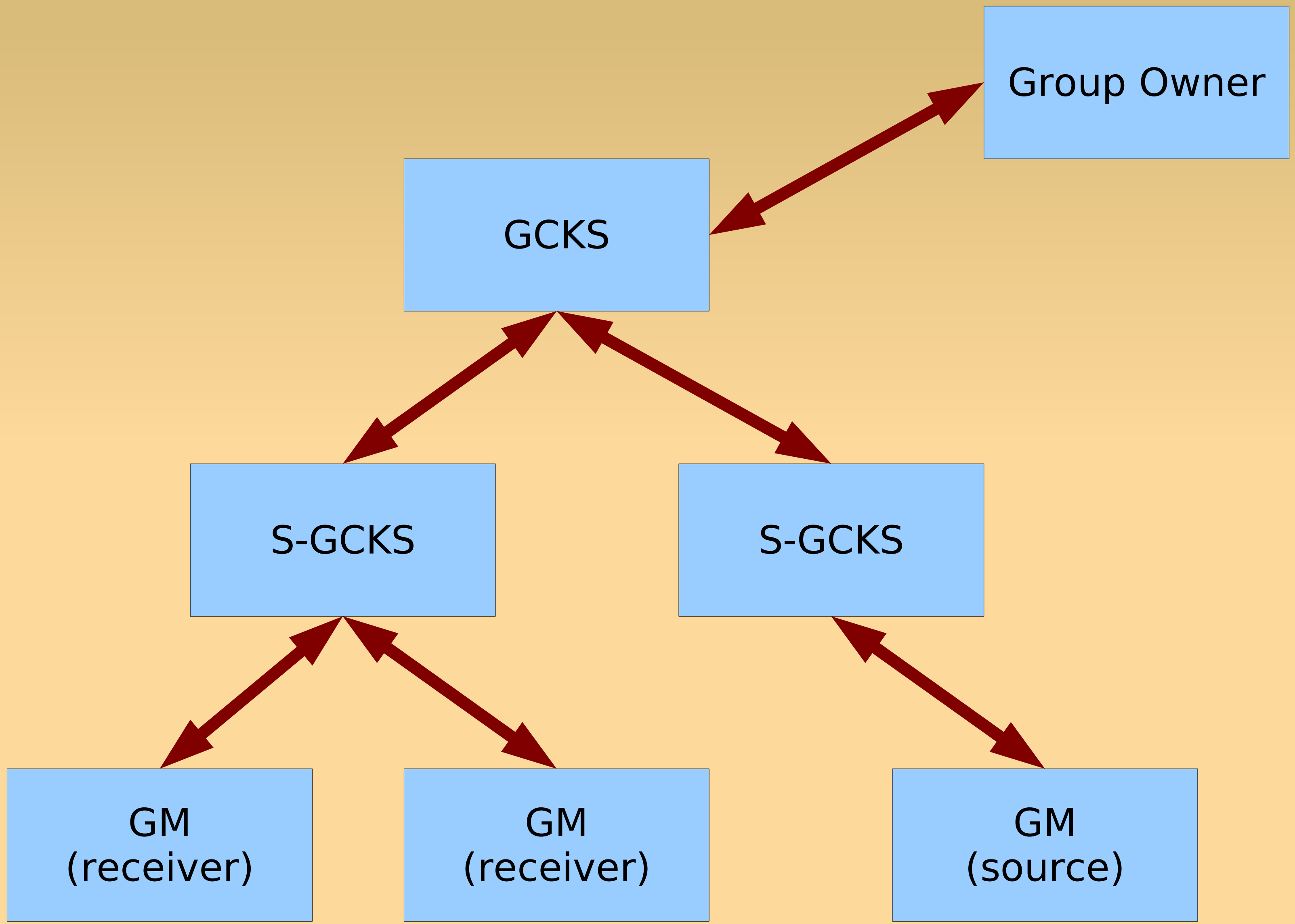


Key Exchange Protocols

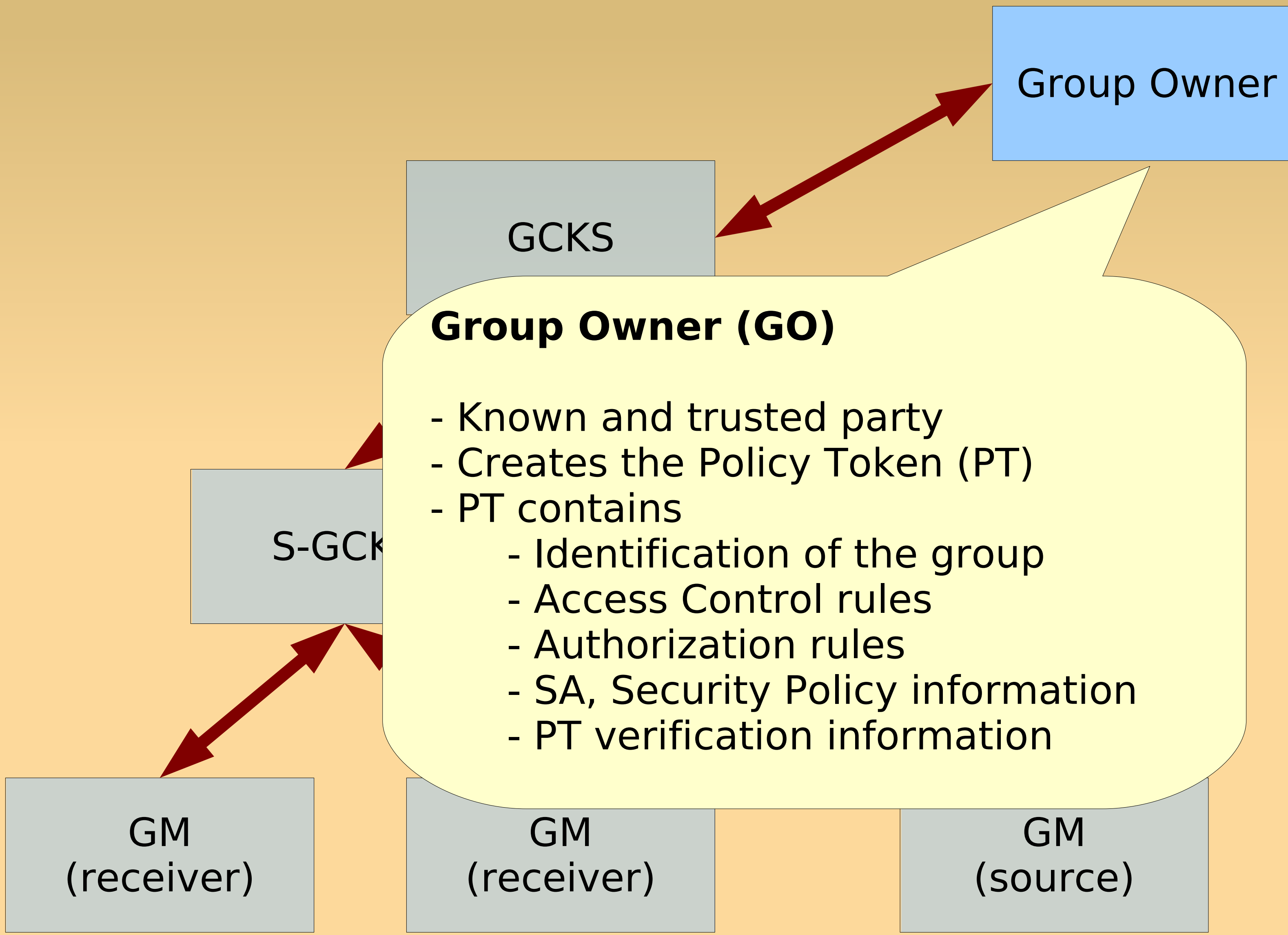
- Protocols defined in the IETF
  - Group Security Association Key Management Protocol (GSAKMP)
  - Multimedia Internet Keying (MIKEY)
  - Group Domain of Interpretation (GDOI)
- All define only multicast keying
  - Re-key SA, Data SA
- Registration not defined
  - E.g. IKE used for creating registration SA

Group Security Association Key Management Protocol (GSAKMP)

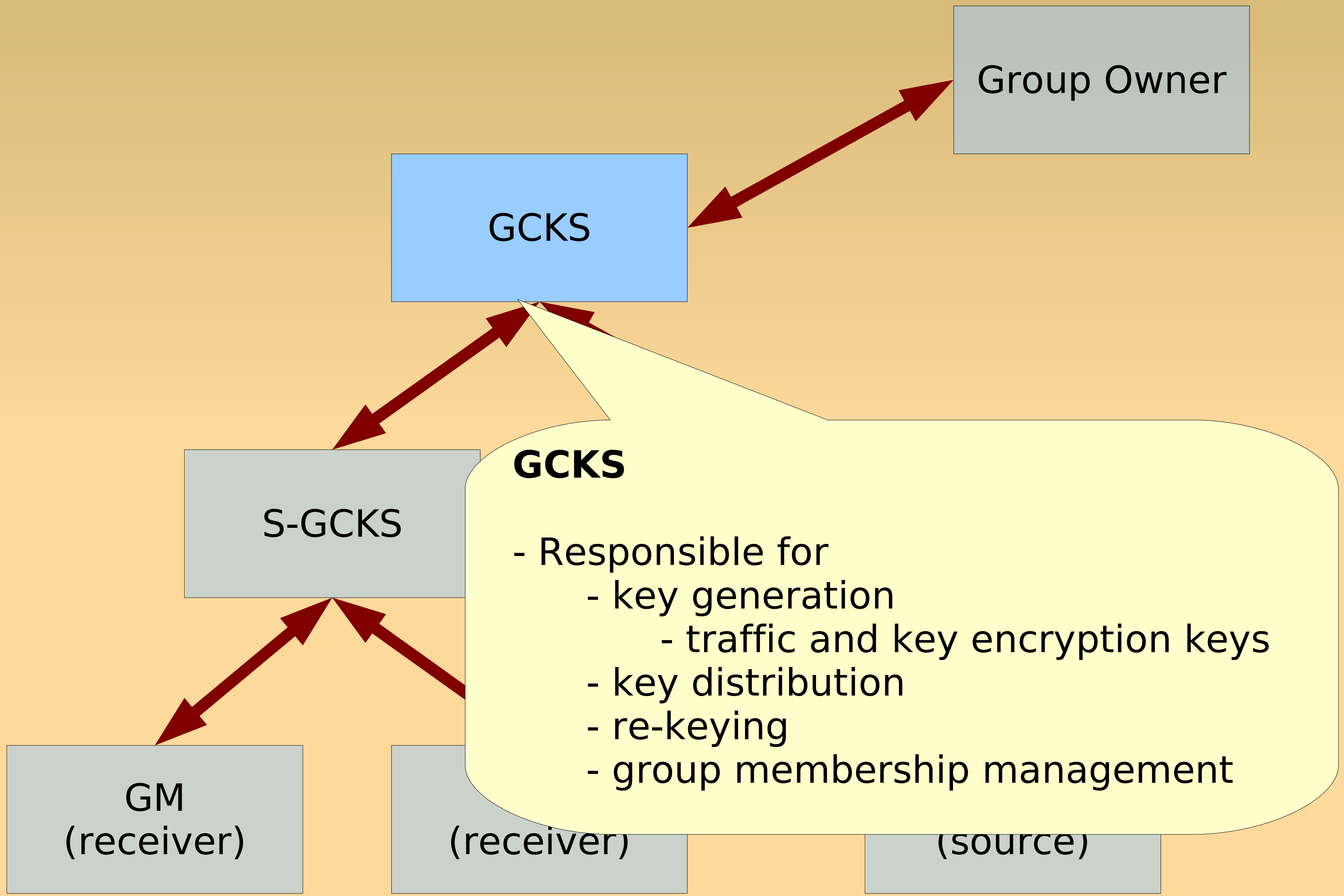
GSAKMP Trust Model



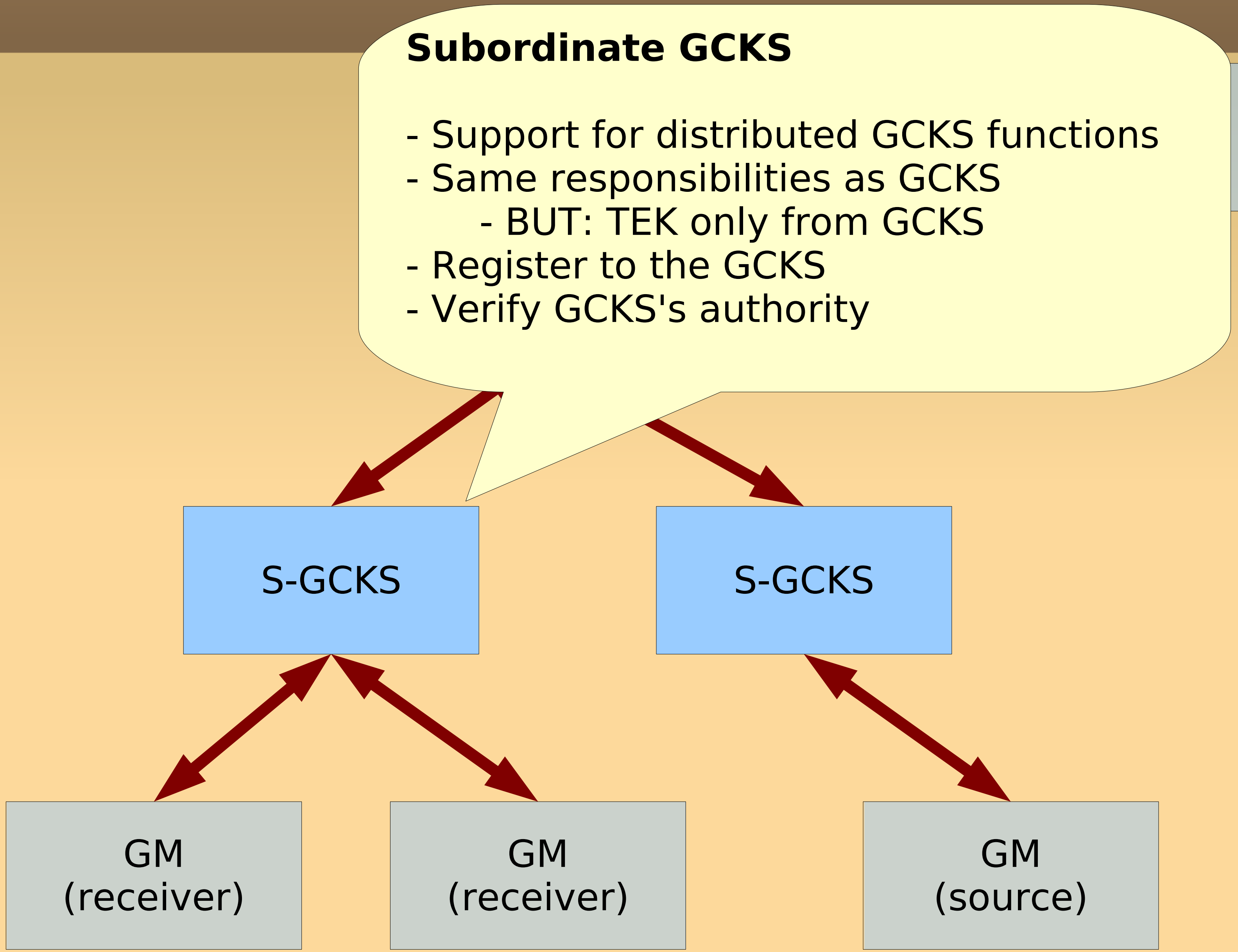
GSAKMP Trust Model



GSAKMP Trust Model



GSAKMP Trust Model



GSAKMP Trust Model

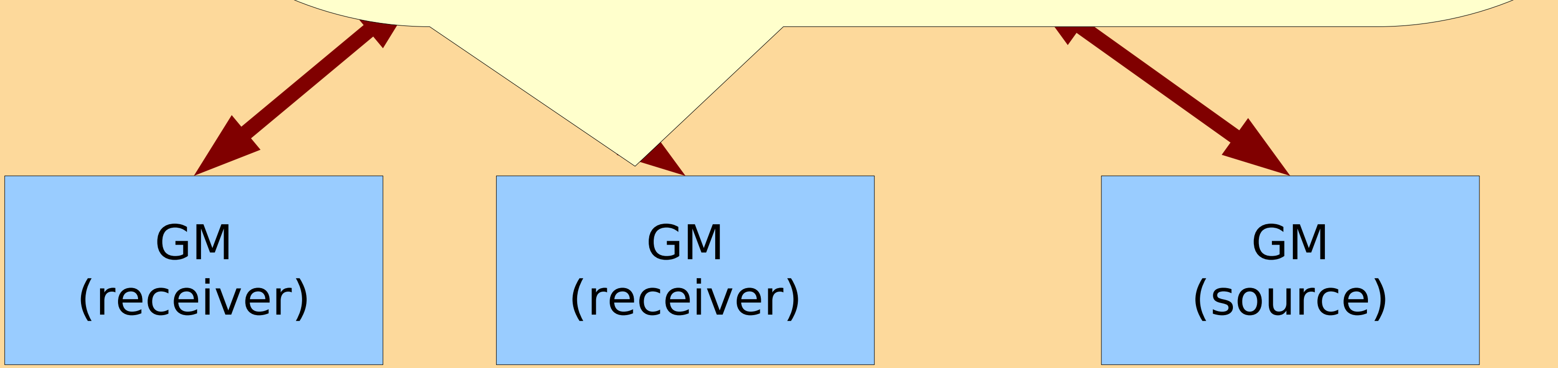
Group Member (GM)

GM has to:

- Verify that all security related actions are authorized
- Use group keys properly

GSAKMP cannot control who sends data to the group
-> Multicast protocol and application issue
Senders authorized in PT

Sender configurations: 1, subset of GMs, or all

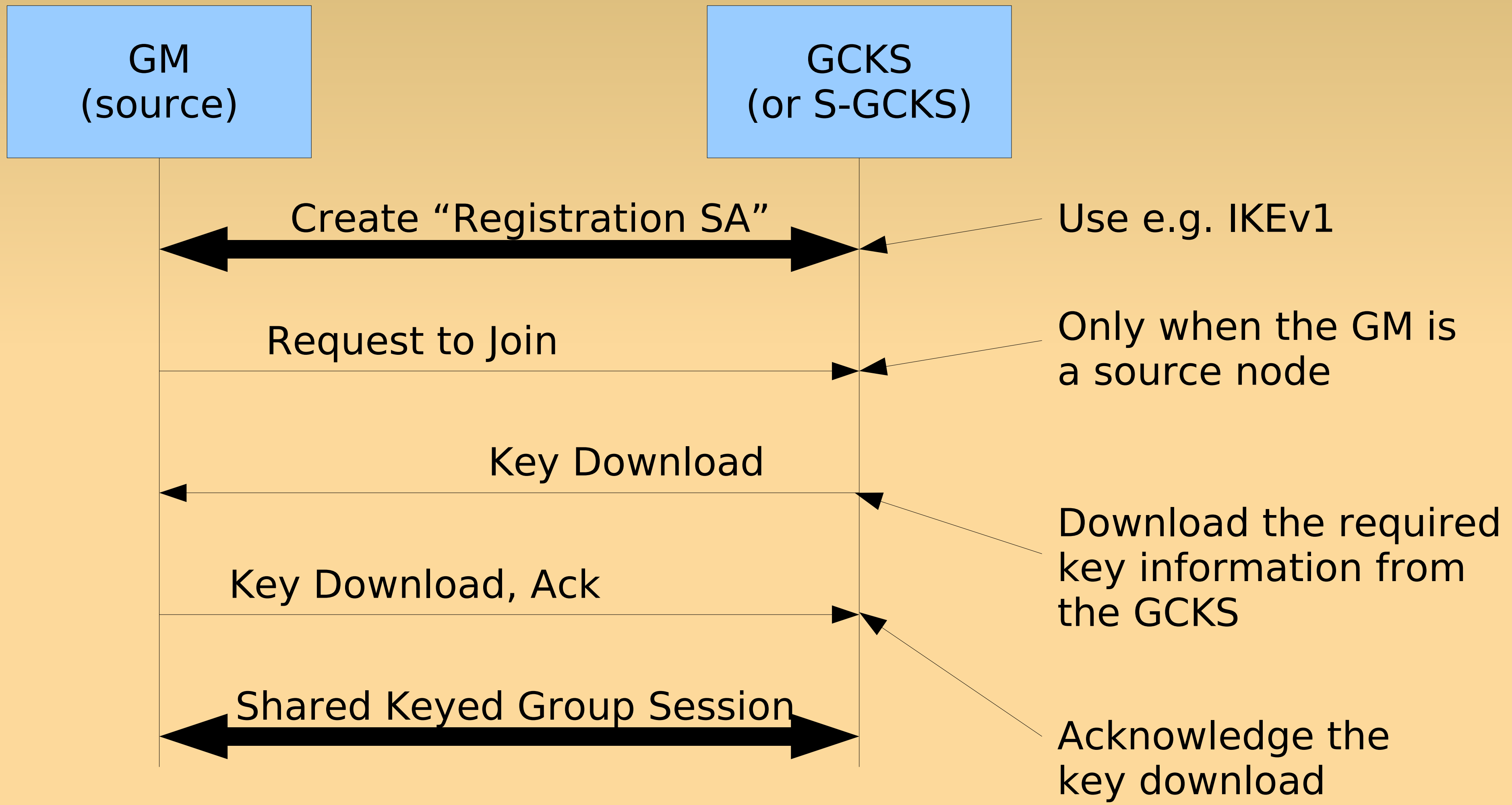


GSAKMP Assumptions

- GCKS or GO <u>never</u> compromized
- PKI is trustworthy (for cert validation)
- Compromized GM reported to GO
- No precise time dependency (in security related actions)
- Compromized GM cannot decrypt further traffic
- Confidentiality, integrity, multicast source authentication, and anti-replay protection for GSAKMP messages

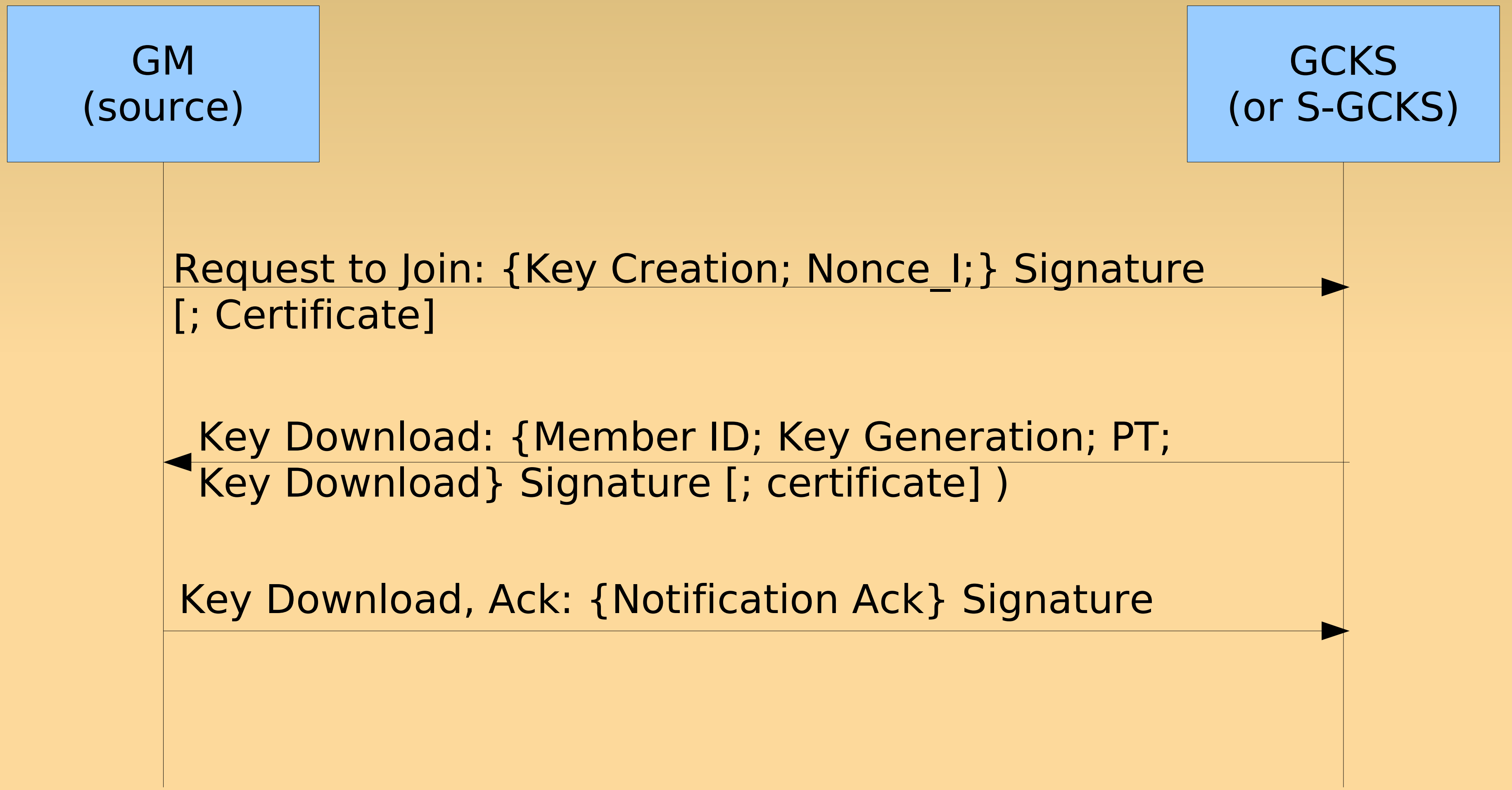
GSAKMP

Message Exchange



GSAKMP

Message Exchange



GSAKMP

- Diffie-Hellman used for key generation
  - protecting further downloads from the GCKS
- GM leaves the group
  - LKH MAY be used for re-keying
  - “Many times it is best to rebuild the group”
    - Problem: This doesn't work with large groups

Multimedia Internet Keying

MIKEY

Multimedia Internet Key Exchange

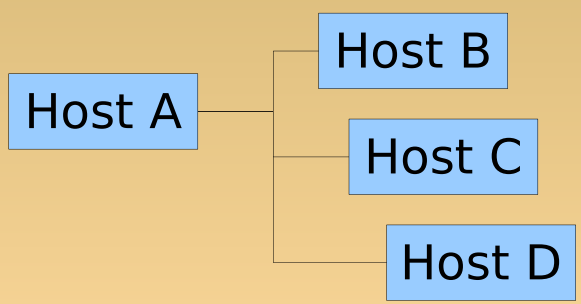
- Originally designed for real-time applications
  - Secure RTP
- Issues
  - Lower latency
  - heterogeneous networks
  - better performance for small, interactive groups

MIKEY

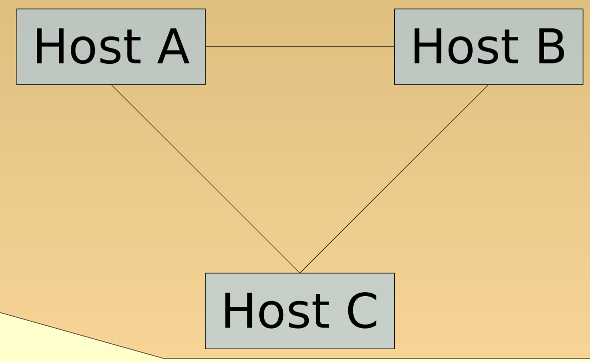
Multimedia Internet Key Exchange

- Source handles GCKS functions (usually)
- No actual re-keying
  - Changes in groups handled by setting up a new connection
  - Cannot efficiently support big and unstable groups
  - MBMS (3GPP) defines re-keying

MIKEY - scenarios

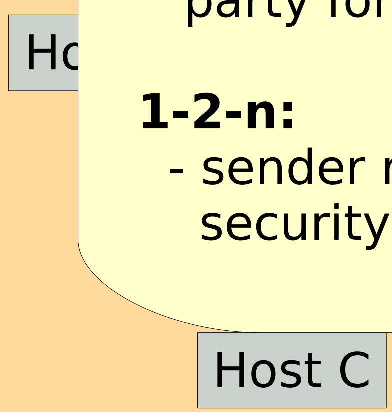


peer-to-peer /
one-to-many



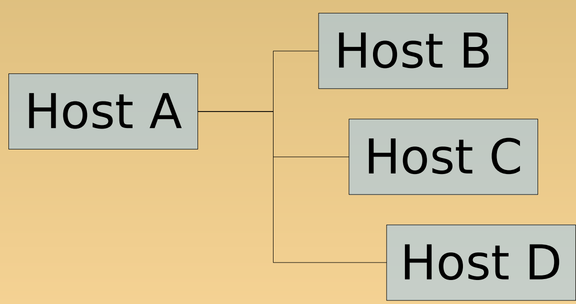
p-2-p: e.g. SIP call
-(mutual security agreement or each party for own outgoing)

1-2-n:
- sender responsible for setting up security parameters

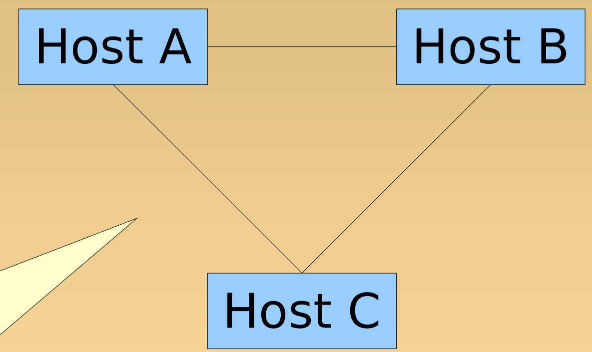


many-to-many
(centralized)

MIKEY - scenarios



peer-to-peer /



many-to-many
(distributed)

Small size group

1) Initiator acts as a GC (MIKEY)

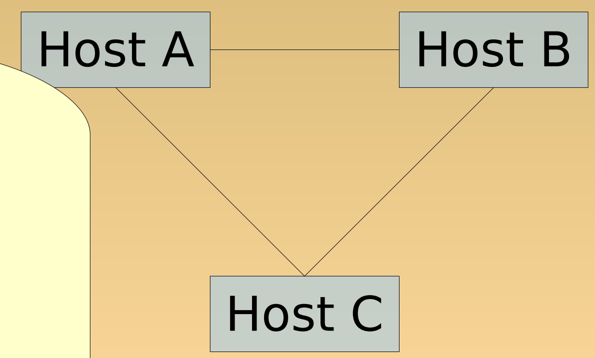
2) Authorization information is delegated to other participants (not defined in MIKEY)

many
(centralized)

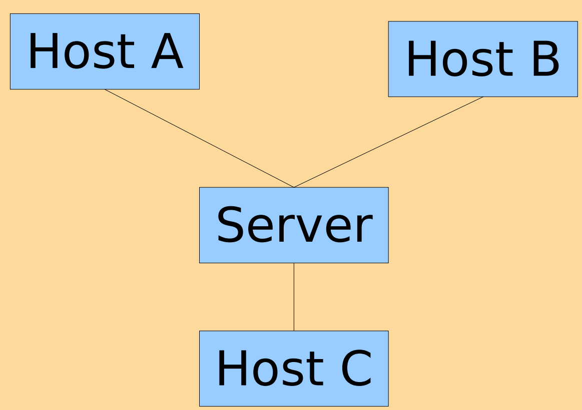
MIKEY - scenarios

Larger groups

- GCKS responsible for setting up security parameters
- Not main focus in MIKEY



many-to-many
(distributed)

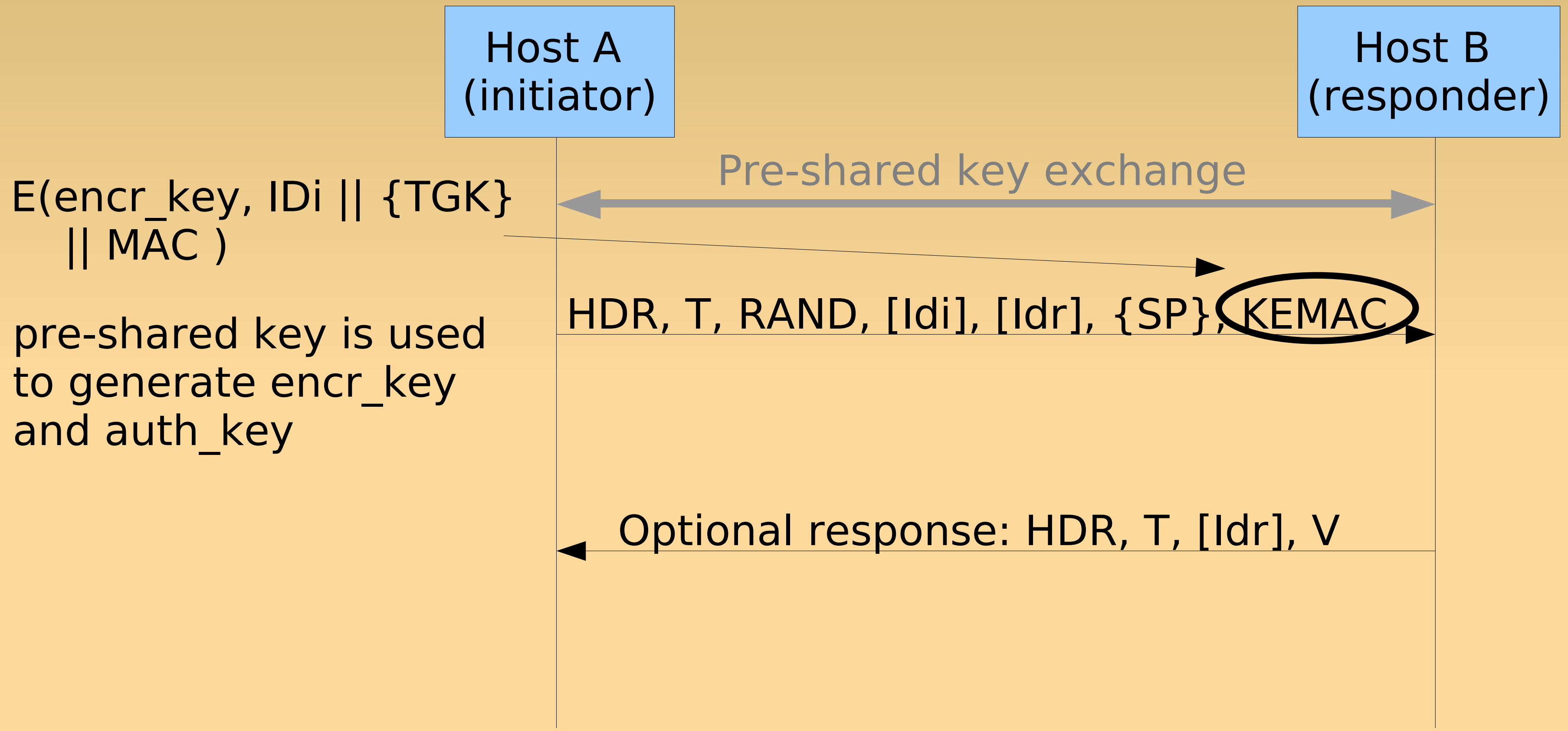


many-to-many
(centralized)

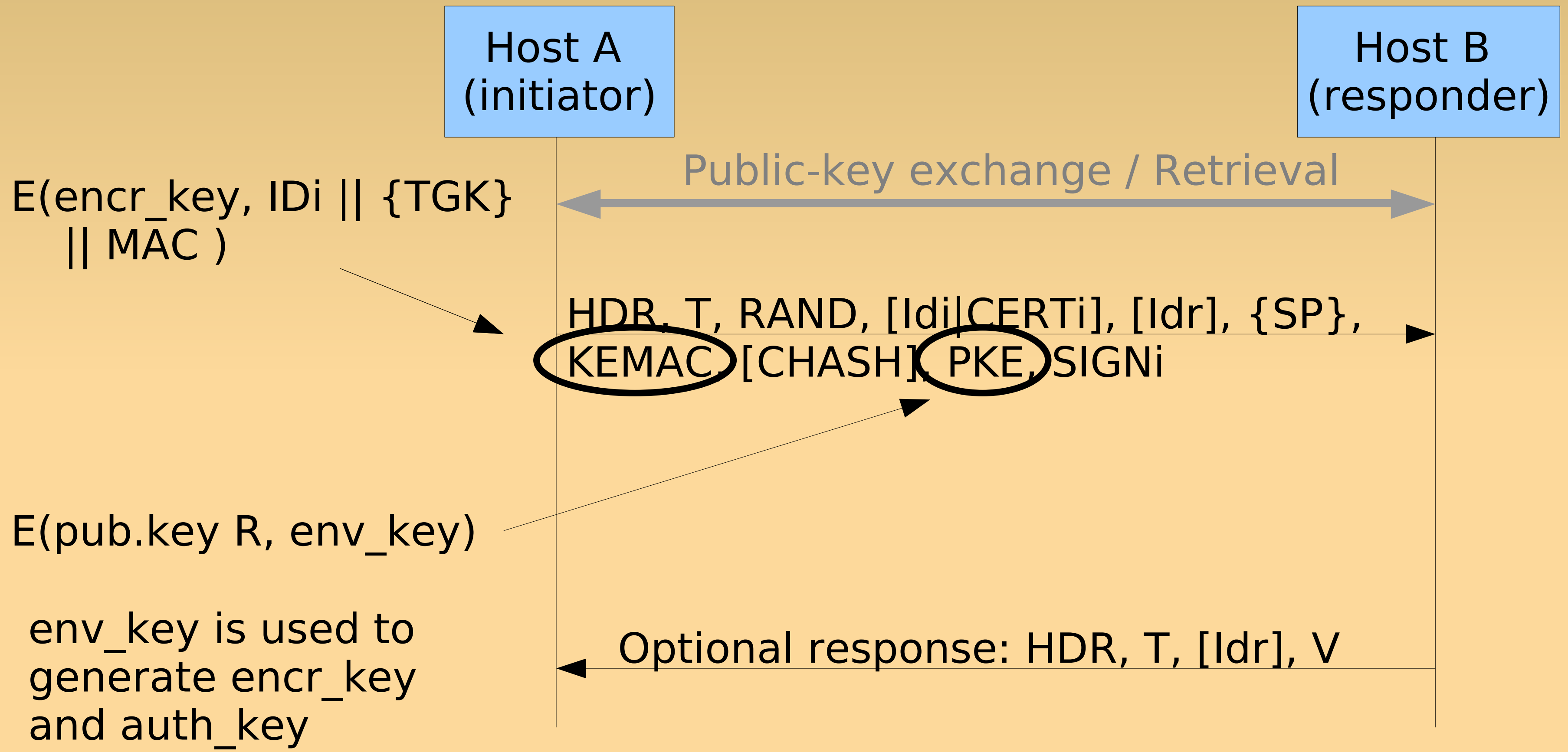
MIKEY – Generating a TGK

- TGK = TEK Generation Key
- Three methods
  - Pre-shared key
    - TGK transferred using the pre-shared key
    - Efficient but not scalable
  - Public-key based method
    - PKI needed for distributing public keys
  - Diffie-Hellman key exchange
    - For peer-to-peer case

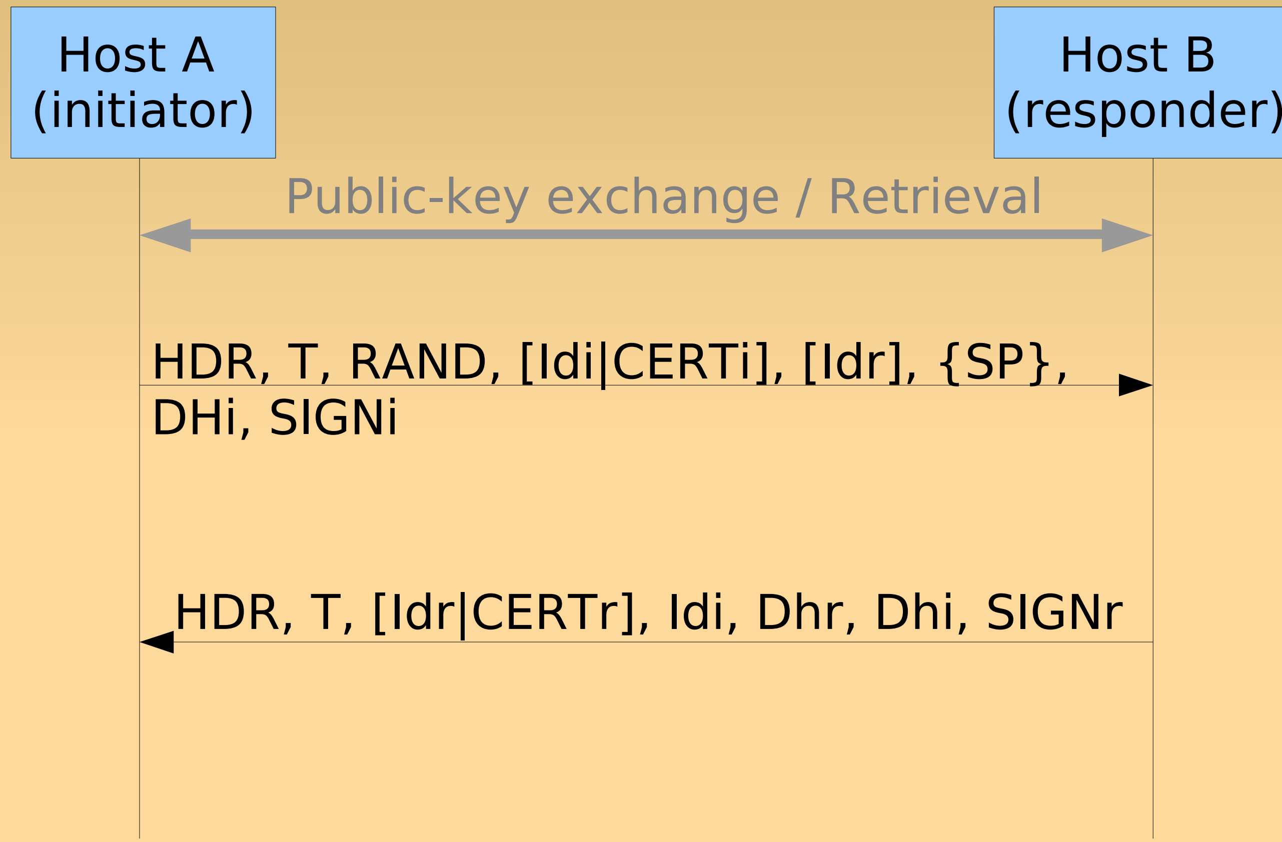
MIKEY: pre-shared key



MIKEY: public keys



MIKEY: Diffie-Hellman



TGK is the shared secret calculated from DH values

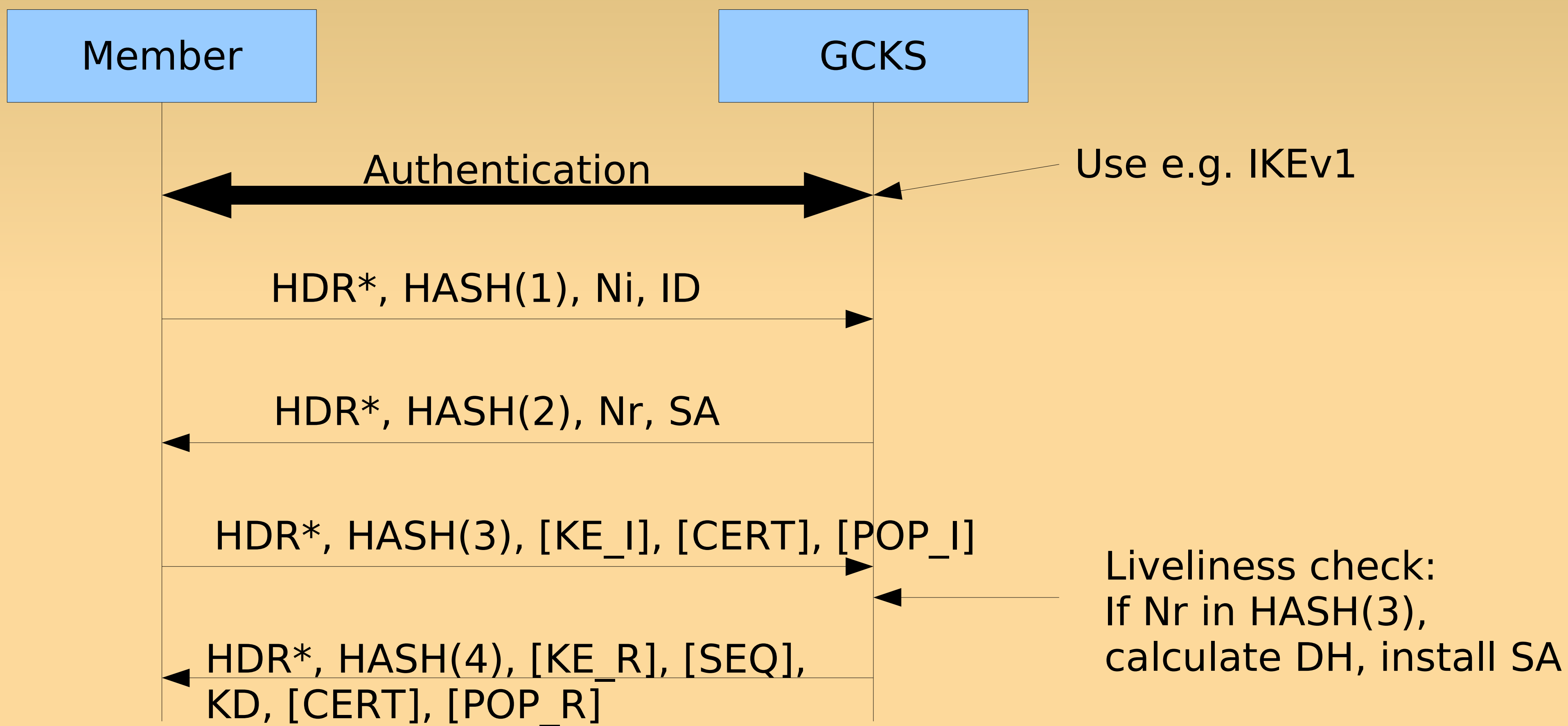
Group Domain Of Interpretation

GDOI

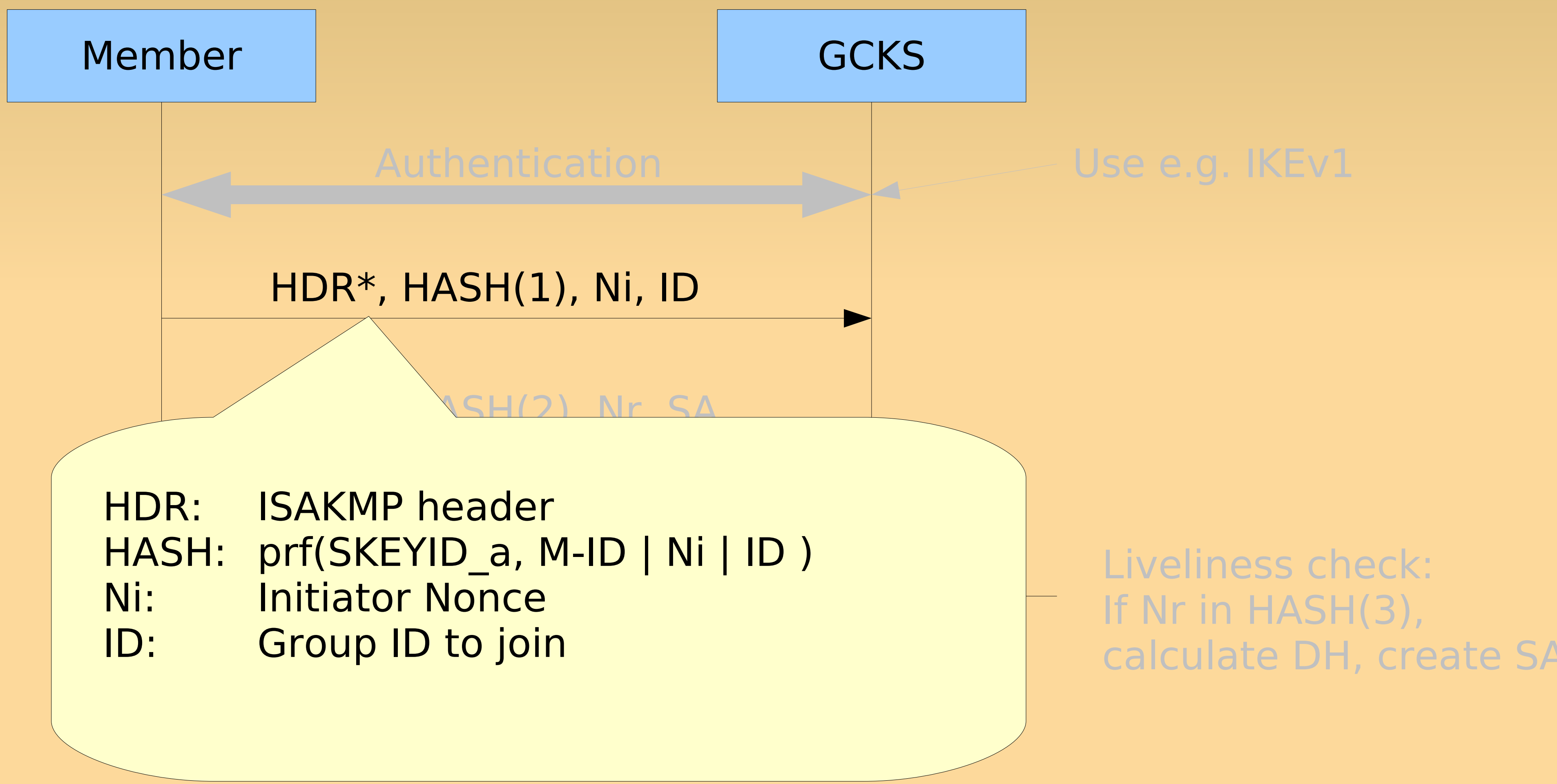
The Group Domain of Interpretation

- *Registration association* with ISAKMP phase 1
- GDOI defines
  - *Re-key association* setup
  - *Data association* setup
- TEK & KEK key transfer
  - GROUPKEY_PULL: initiated by the member
  - GROUPKEY_PUSH: initiated by the GCKS

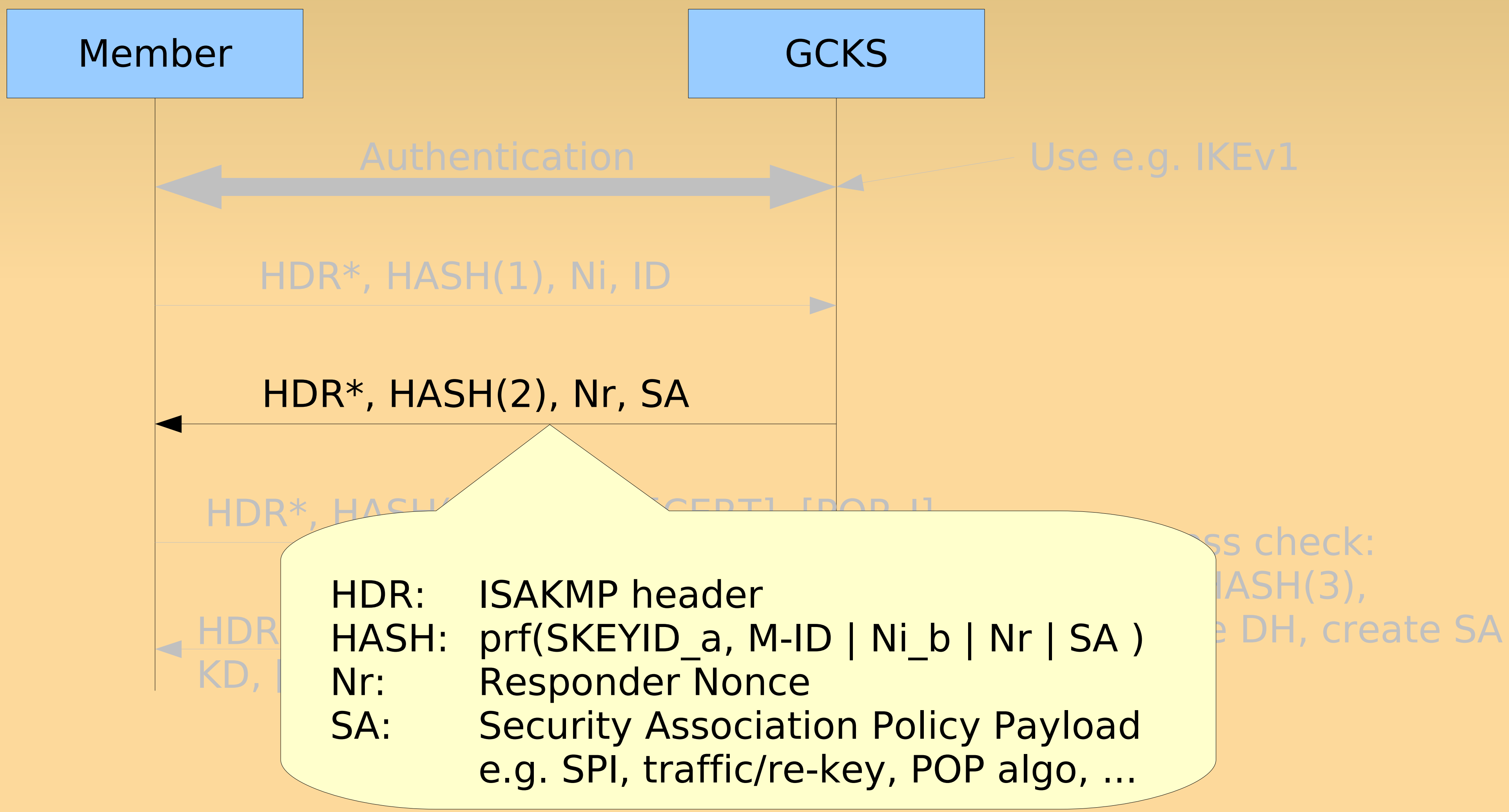
GDOI: GROUPKEY_PULL



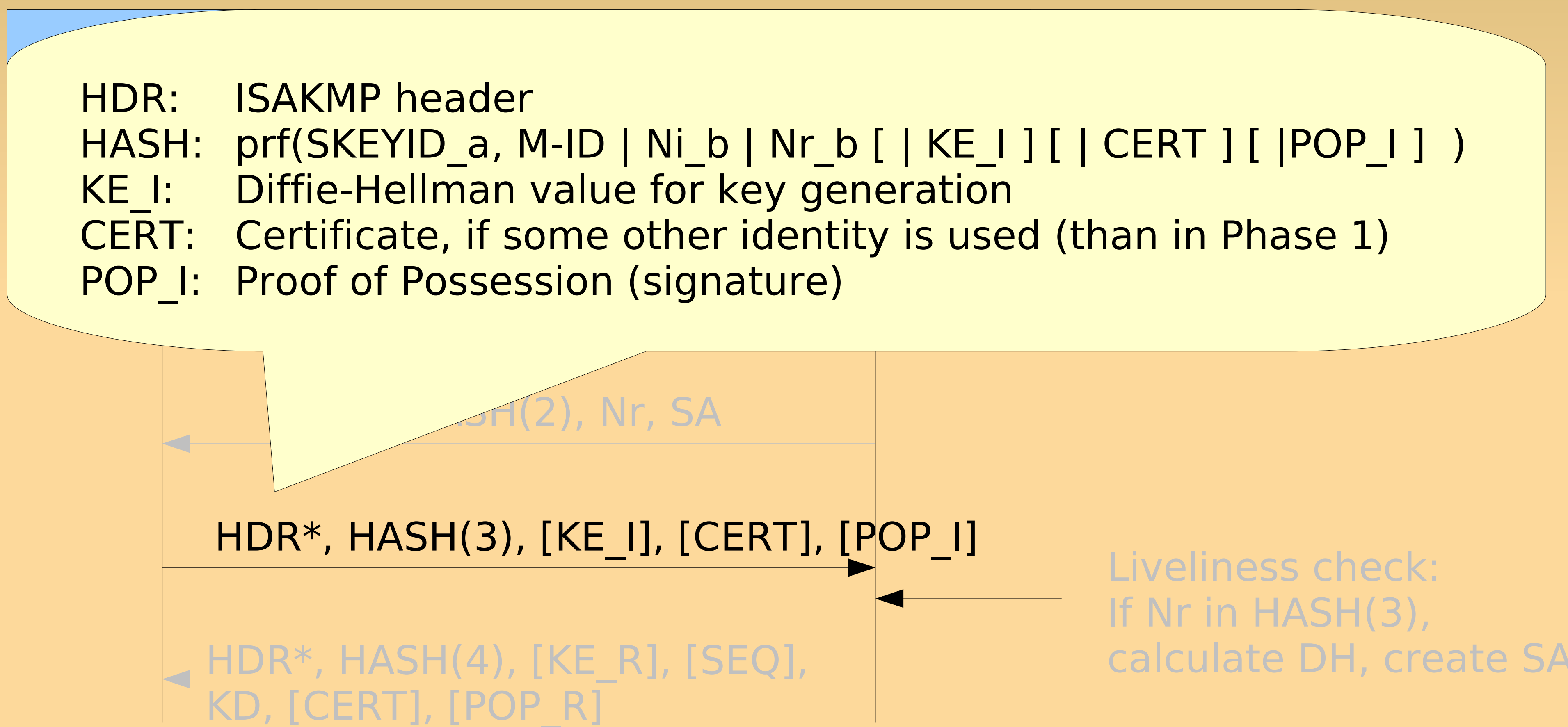
GDOI: GROUPKEY_PULL



GDOI: GROUPKEY_PULL



GDOI: GROUPKEY_PULL



GDOI: GROUPKEY_PULL

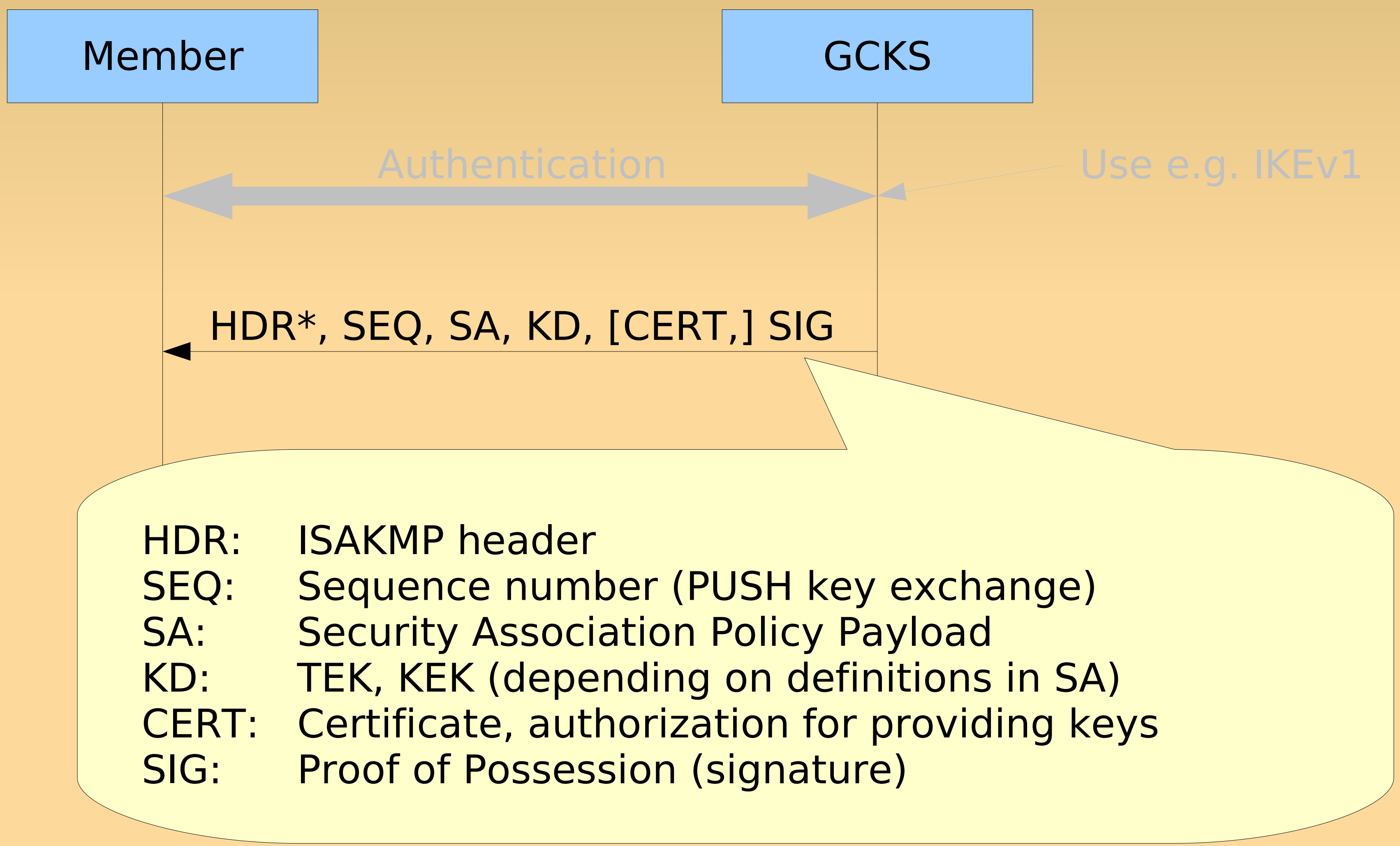
HDR: ISAKMP header
HASH: prf(SKEYID_a, M-ID | Ni_b | Nr_b [| KE_R] [| SEQ] | KD [| CERT] [| POP_R])
KE_R: Diffie-Hellman value for key generation
SEQ: Sequence number (PULL key exchange)
KD: TEK, KEK (depending on definitions in SA)
CERT: Certificate, authorization for providing keys
POP_R: Proof of Possession (signature)

HDR*, HASH(3), [KE_I], [CERT], [PO

Liveliness check:
If Nr in HASH(3),
calculate DH, create SA

HDR*, HASH(4), [KE_R], [SEQ],
KD, [CERT], [POP_R]

GDOI: GROUPKEY_PUSH



Host Identity Protocol

Host Identity Protocol

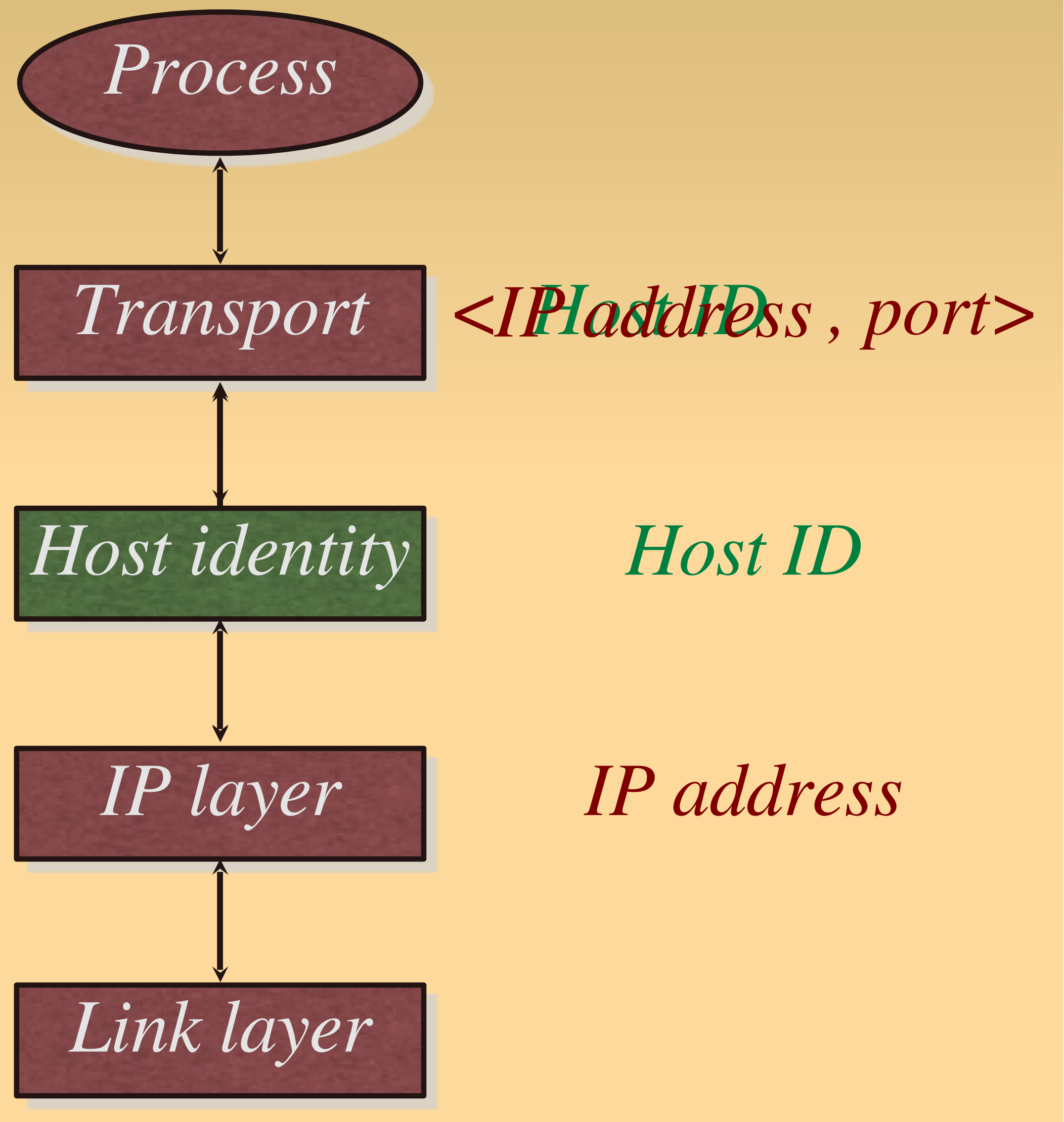
- IP address roles currently
  - Locator: describes the host's topological location in the network
  - Identifier: identifies the host
- Problems
  - How to know who is at the other end – IP address is not enough
  - Mobility difficult

HIP: Host Identities

- Host Identity (HI): public key of a key pair
  - Hosts can authenticate each other
- Secure binding between HI and IP address
- Locator is used only for data routing
  - IP address not needed once the packet arrives
  - ESP mandatory (currently)
    - SPI used to find a correct ESP SA
    - HITs are mapped to the SA
    - Checksums using HITs

A new layer

- New layer
- IP <-> HI mapping
- Sockets bound to HIs, not IPs
- Transparent to applications



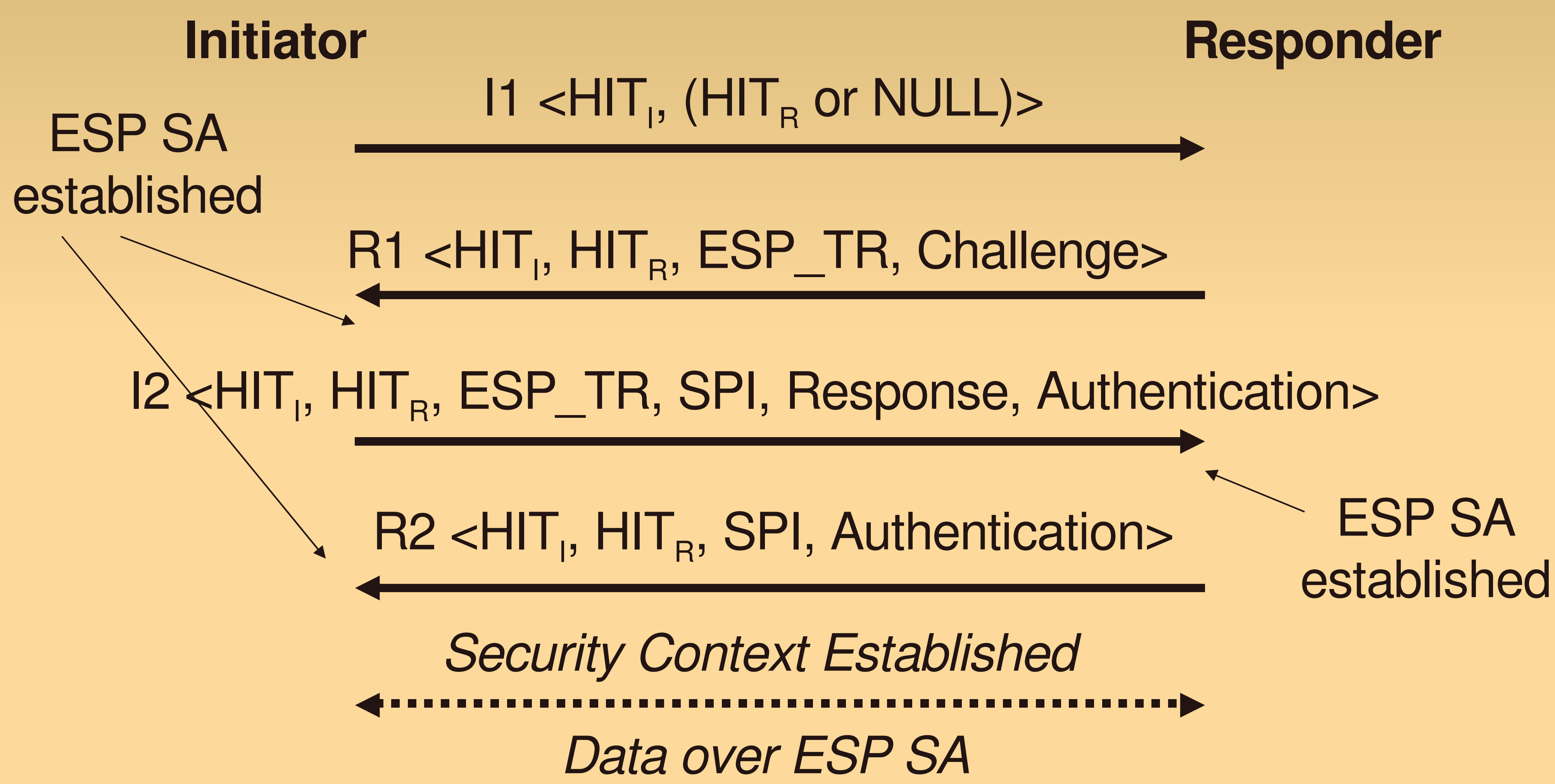
HIP: negotiation

- 4-way message exchange
  - Base Exchange (BEX)
  - Host authentication: public and private keys
  - Diffie-Hellman: common keying material
  - Creates HIP association
- Data traffic protection
  - ESP currently mandatory
  - ESP SA setup during BEX
  - Other protocols may be defined later

Other HIP features

- HI long => HIT (IPv6), LSI (IPv4)
- IPv4/v6 interoperability
  - mobility between v4 and v6 networks
  - v4 and v6 applications can communicate
    - Some limitations due to applications
- Easy mobility
  - Dynamic IP – HIT mapping
  - invisible to applications
- Multihoming support (based on mobility)
  - Independent of access technology

HIP Base Exchange

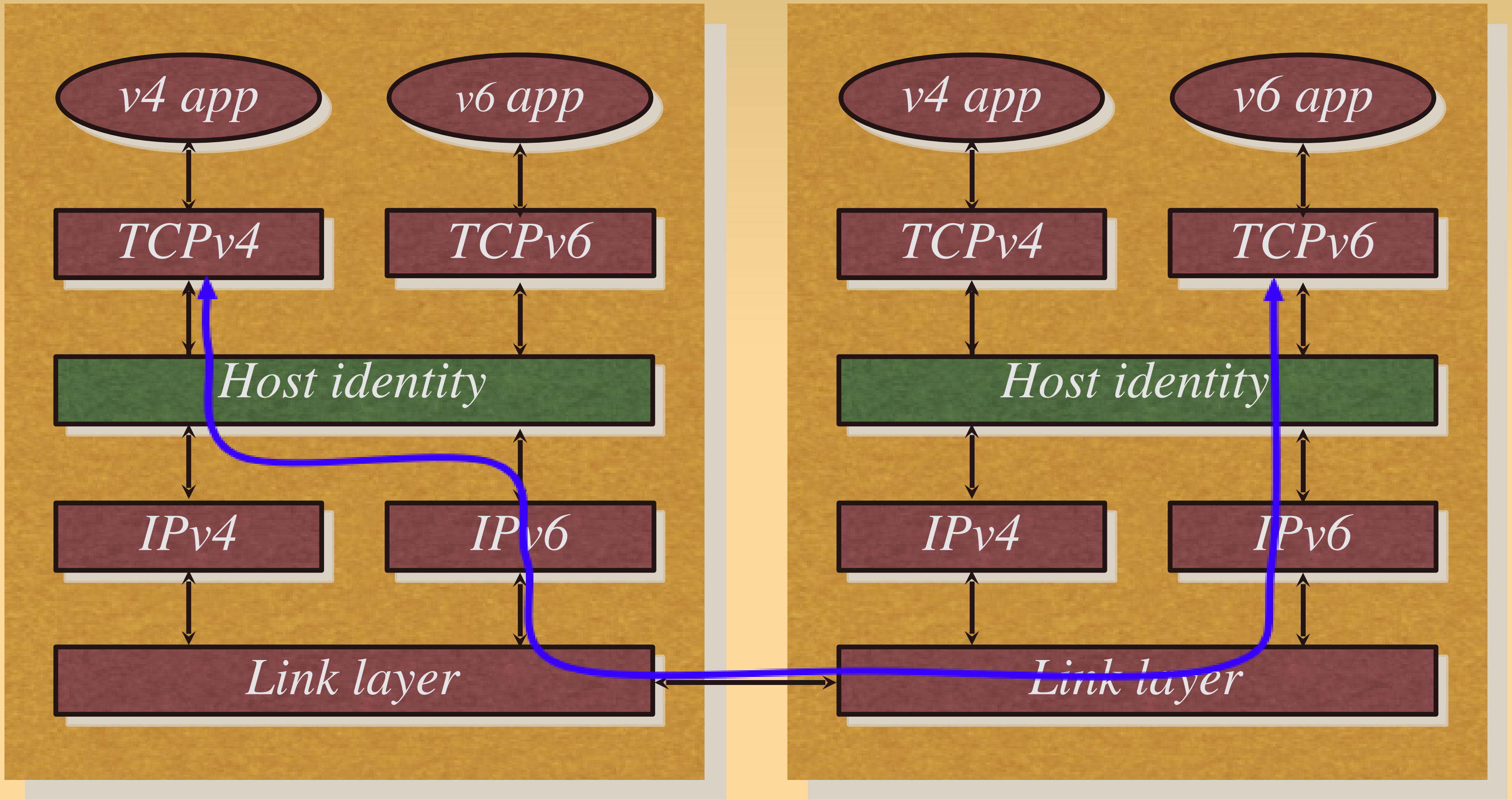


HIP: v4 and v6 interoperability

HI

HIT / LSI

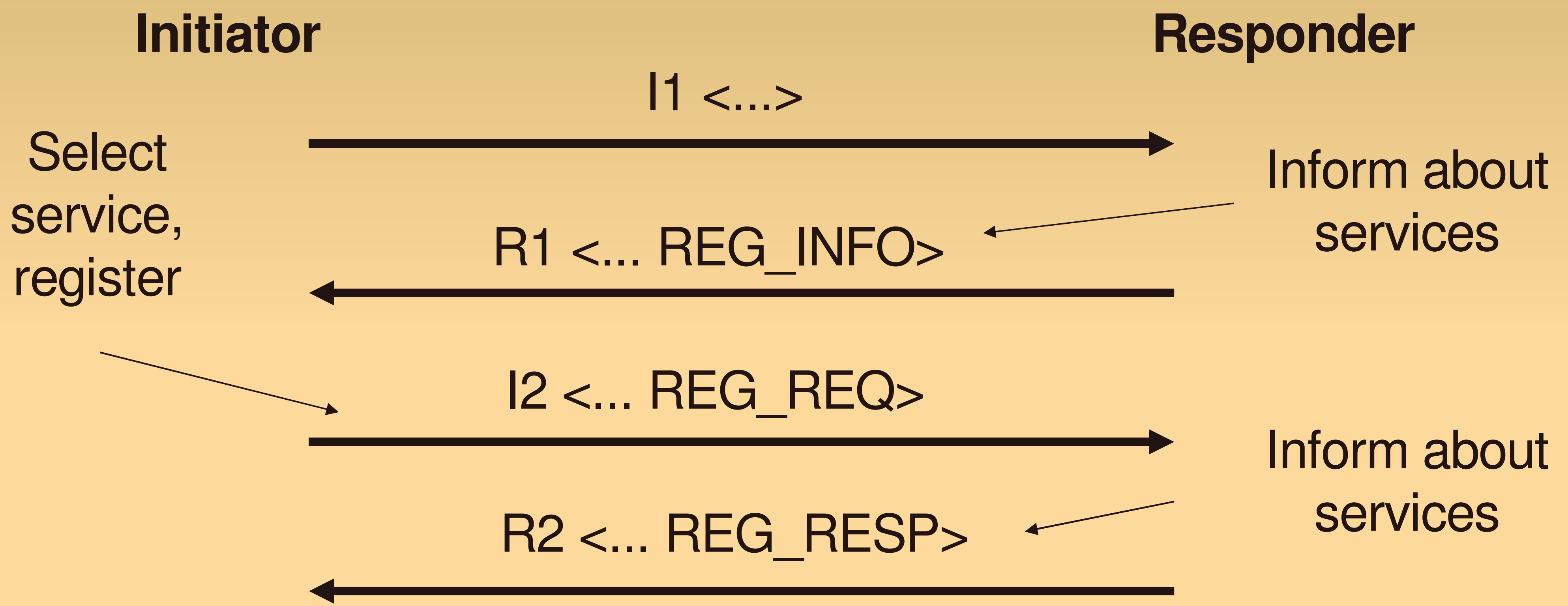
IP



HIP and current solutions

- IPsec: considered ~hard to configure
- Mobile IP large and complex
- Mobile IPv4 and IPv6 do not work together
- No simple solution for multihoming
- LOC: >100.000 vs. ~20.000

HIP Registration Protocol

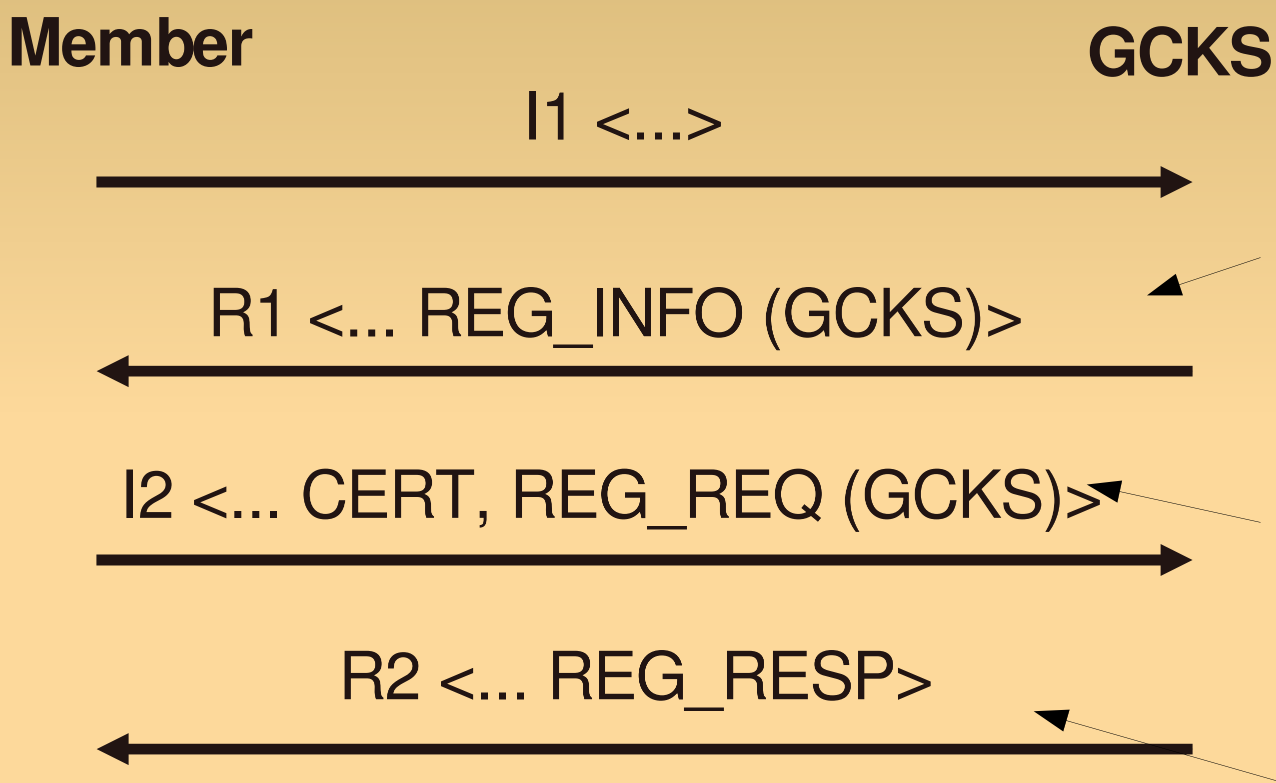


Merging HIP and GDOI

GDOI and HIP

- GDOI: two phases
- 1) replace phase 1 with HIP
  - *Registration association*
  - New “service” needed (GCKS)
- 2) Group Key Exchange
  - For now, use the GDOI phase 2
    - SKEYID_a (for hashes) from the negotiated keying material
  - In the future; HIP has UPDATE mechanism, define multicast key transfer in UPDATE

HIP “Phase 1: registration”



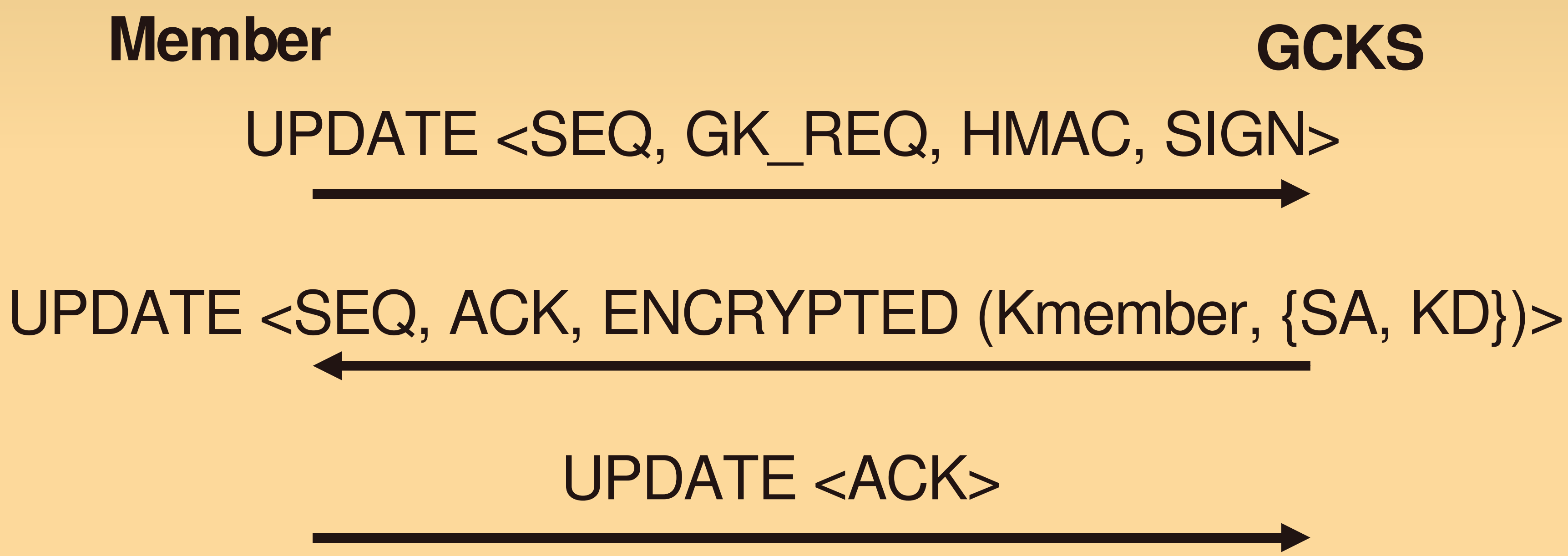
Inform about GCKS,
challenge,
D-H parameters,
GCKS authentication

Member authentication,
Authorization (cert),
D-H params, challenge
solution, SPI

GCKS Authorization (cert),
SPI (registration SA),

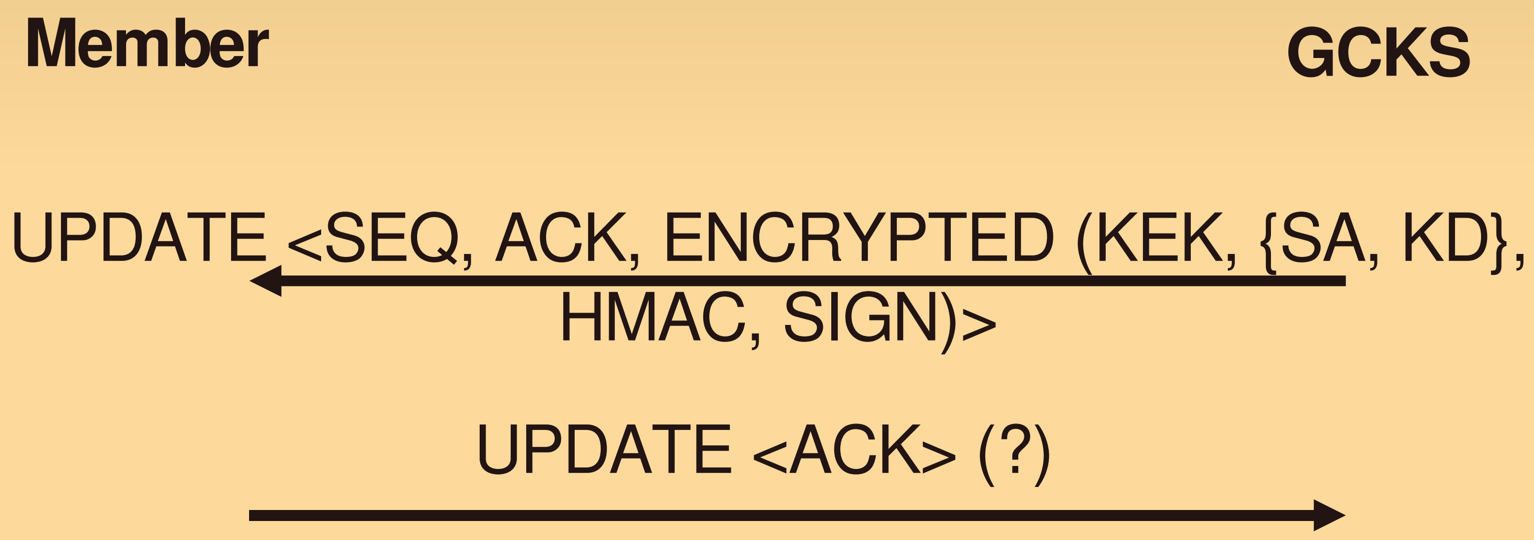
HIP “Phase 2: group keys (PULL)”

UPDATE messages are not encrypted, key information has to be inside ENCRYPTED parameter.



HIP “Phase 2: group keys (PUSH)”

UPDATE messages are not encrypted, key information has to be inside ENCRYPTED parameter.



Advantages / disadvantages

- For HIP hosts
  - Small updates to existing HIP implementations
  - No need for other types of security negotiations
- Mobility management
  - Mobile Member updates location to the GCKS
  - Does not solve the IP multicast (“data connection”) mobility
- Future work
  - Further optimization: Group UPDATE