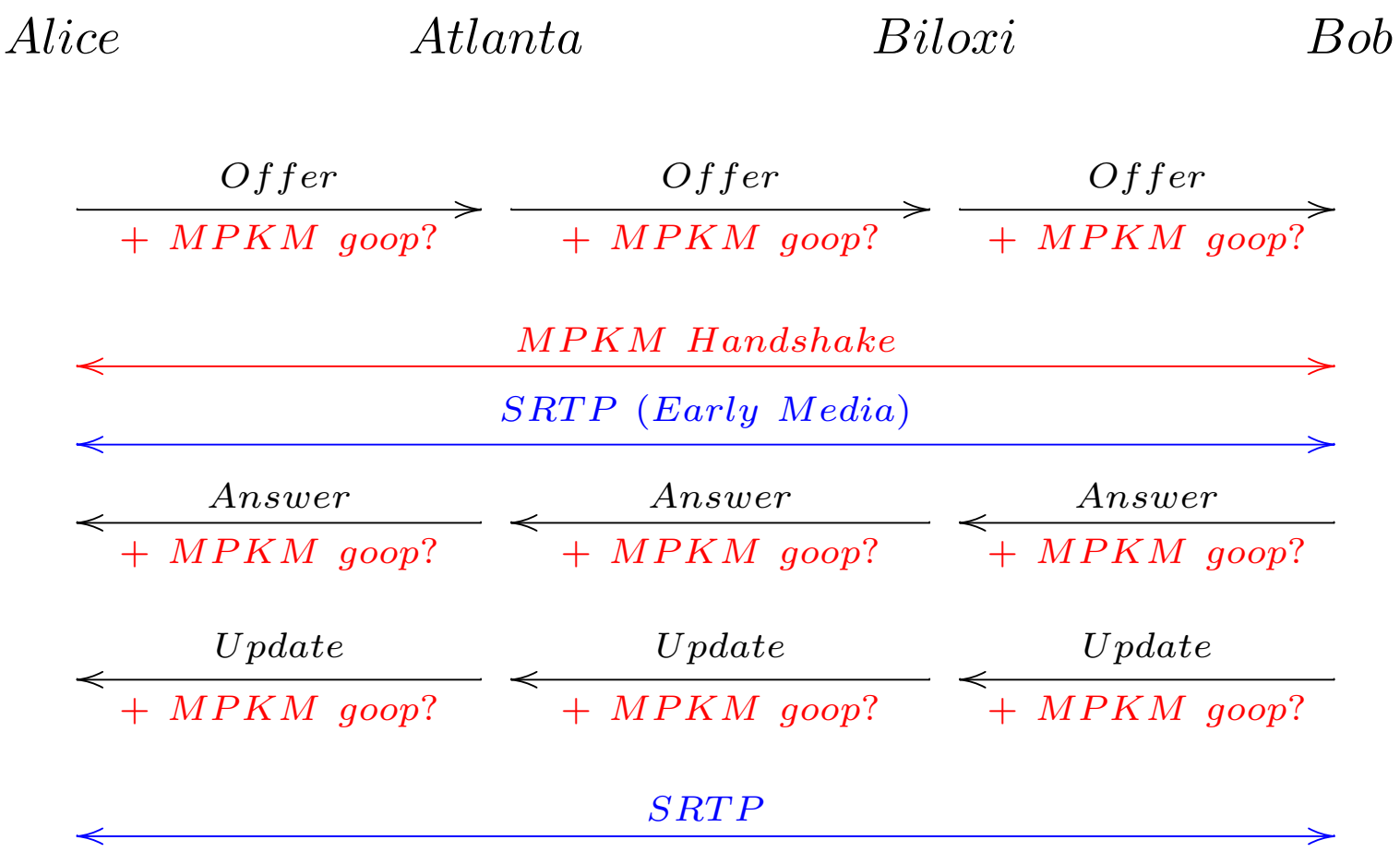
DTLS-SRTP

Eric Rescorla
David McGrew
Jason Fischl
Hannes Tschofenig

The Big Picture: Media-Plane Key Management



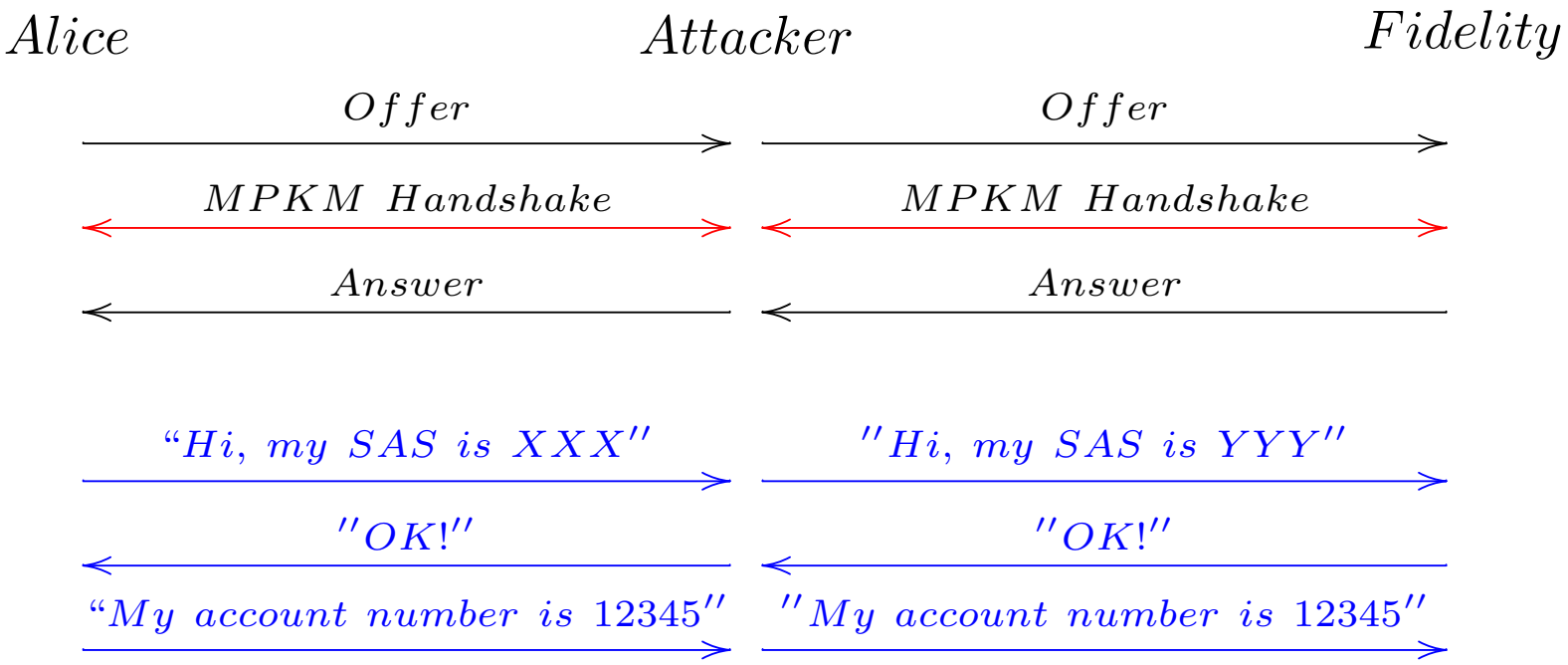
What is that goop in the signalling?

- Cryptographically bind signalling to the MPKM
  - To authenticate the endpoints?
- Indicate willingness to do security
  - and clue in signalling-path elements

Can you authenticate the endpoints without signalling?

- Perform a cryptographic handshake
- Authenticate the handshake over the voice channel
  - Fingerprint reading
  - Or a short authentication string (more convenient but no more secure)
- Unfortunately this isn't secure in many settings
  - Very susceptible to MITM attacks when calling people you don't know
  - Cut-and-paste attacks on the authentication string
  - This assumes people will read the authenticator anyway
- Plus it doesn't work when gatewaying to the PSTN

Impersonation attacks

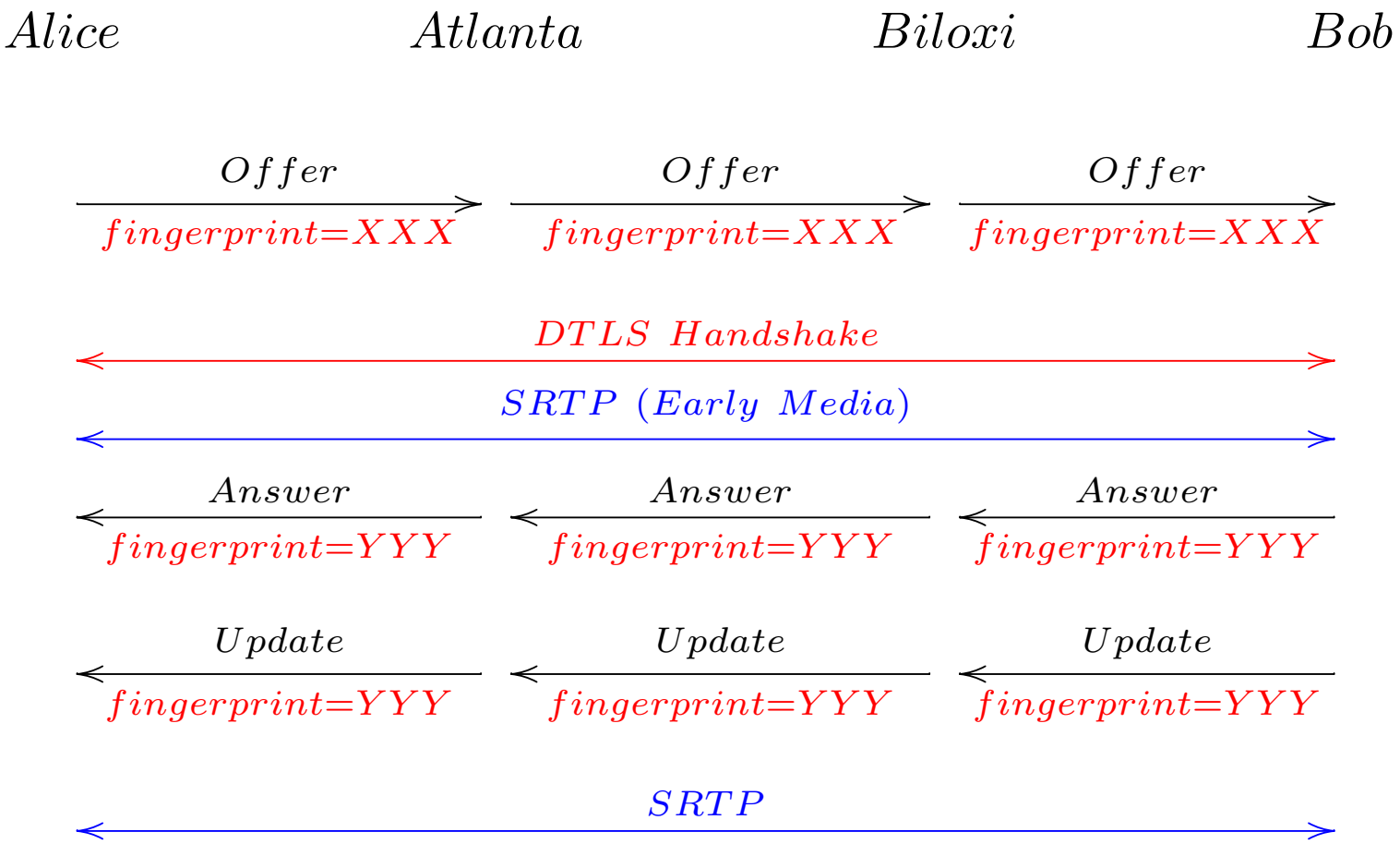


- No way to distinguish the attacker from a legitimate answerer
  - How do you know what Fidelity's CSR sounds like?
  - The voice sounds the same throughout the call!
  - Even easier to clone an IVR system
- This is a variant of the classic "mafia attack" [DGB87]

Cut-and-paste attacks

- SAS has a limited coding space (32 symbols)
- People will happily read their SAS to you
  - You get 4 symbols per call
  - 15 calls $\rightarrow$ 85% of symbols
  - 85% of symbols $\rightarrow$ 52% forgery probability
- Base-256 works better
  - But attack still possible
  - Especially on IVR...

Binding the signalling to DTLS-SRTP



- Fingerprints are protected via Identity/Connected Identity

No PKI required

- Yes, DTLS uses certificates (sort of)
  - What's being used is public keys (a la SSH)
  - Unfortunately DTLS won't carry raw public keys
  - So we pack them into certificates
- This is totally transparent to the user
  - Keys and certificates are automatically self-generated
  - The peer does not need to check them
    - Because the fingerprint is in the signalling
- But third-party certificates work seamlessly

Key Continuity

- What if I don't always have secure signalling?
- DTLS-SRTP includes a key continuity feature (a la SSH)
  - Cache the public key of each peer
    - Stored under the AOR
  - Signal a warning if the key changes
- What about multiple devices?
  - Option 1: All devices share one key (best)
  - Option 2: Each device has its own key
    - Peer has to store multiple keys under the AOR (not a big deal)
    - Leaks which device you're using (a big deal)

Indicating willingness to do security

How do you know that the other side will do security?

- Probing
  - Answerer sends DTLS ClientHello packets
  - If he gets a response he proceeds with handshake
  - Interaction with bandwidth reservation?
- Signal in the SDP
  - ... using SDP capability negotiation [And07]
  - Once you are cryptographically bound to the signalling capability negotiation makes sense
- DTLS-SRTP is written to use capability negotiation
  - But could use probing + SAS for ad hoc modes

So, what's new here?

- A DTLS ClientHello extension to negotiate SRTP transport
  - Indicates protection profiles
  - DTLS master_secret used to generate SRTP keys
  - DTLS authentication and key exchange untouched
- Some way to signal willingness to do DTLS-SRTP
  - Simple application of capability negotiation
  - Fingerprint already defined by RFC 4572 [Len06]
- An SAS mode for DTLS?
  - If the group thinks this is important
  - First cut at [MR07]
- That's it

Drafts

- draft-mcgrew-tls-srtp-02
- draft-fischl-sipping-media-dtls-02
- draft-fischl-mmusic-sdp-dtls-02

References

[And07] F. Andreasen. SDP Capability Negotiation. draft-ietf-mmusic-sdp-capability-negotiation-05.txt, March 2007.

[DGB87] Y. Desmedt, C. Goutier, and S. Bengio. Special uses and abuses of the Fiat Shamir passport protocol, 1987.

[Len06] J. Lennox. Connection-Oriented Media Transport over the Transport Layer Security (TLS) Protocol in the Session Description Protocol (SDP). RFC 4572, July 2006.

[MR07] David McGrew and Eric Rescorla. Short Authentication Strings for TLS. `https://svn.resiprocate.org/rep/ietf-drafts/ekr/draft-mcgrew-tls-sas.xml`, 2007.