Should we support SDES in WebRTC?

draft-ohlsson-rtcweb-sdes-support-00

author: Oscar Ohlsson

Presenter: Hadriel Kaplan

The Question

- We *are* going to mandate DTLS-SRTP be implemented, and *used* much of the time
  - Whenever the far-end does DTLS-SRTP
  - Whenever the Web connection is not HTTPS
- So the question is: should we *also* support SDES?

Photo: Rebecca (Becky/Bex)

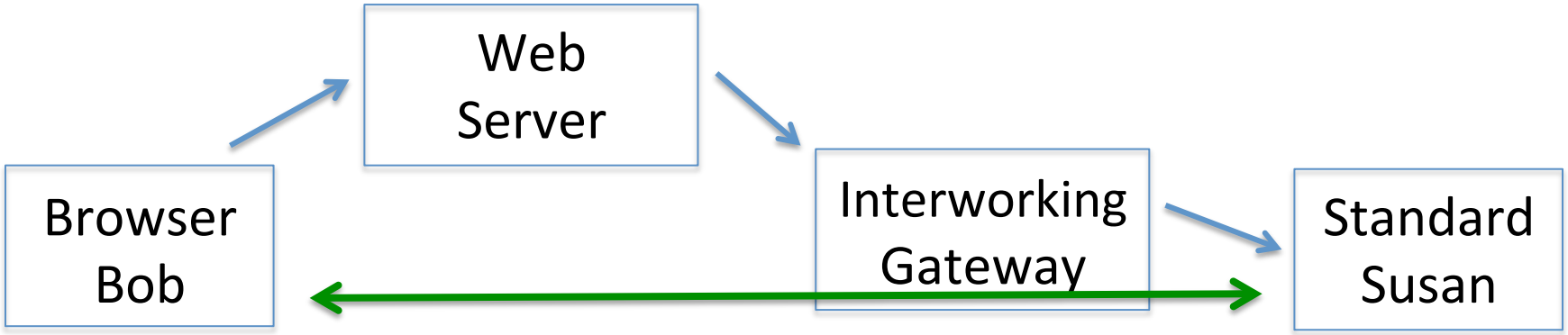
Why bother?

1. Reduced complexity for interworking gateways (i.e., less cost/perf-overhead)
2. Allows end-end SRTP for interworking cases
3. Reduced time-to-media
4. It is known to work and be interoperable
5. It's trivial additional complexity, assuming it does not truly degrade security
   - And that's the really big question

The concerns with including SDES

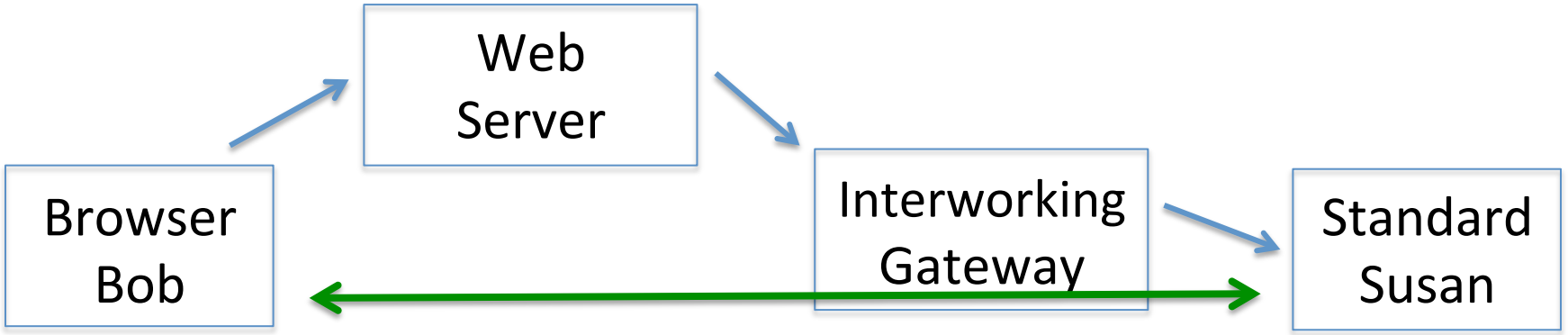
1. Enables eavesdropping
   - Enables malicious websites to snoop
2. No perfect-forward-secrecy
   - Logged SDES keys let one decrypt after-the-fact
3. Susceptible to downgrade attacks
   - Malicious website could force SDES every time
4. Unverifiable
   - Don't know if the media is secure end-end or not
5. Not complicated enough
   - Developers aren't challenged enough

Enables Eavesdropping



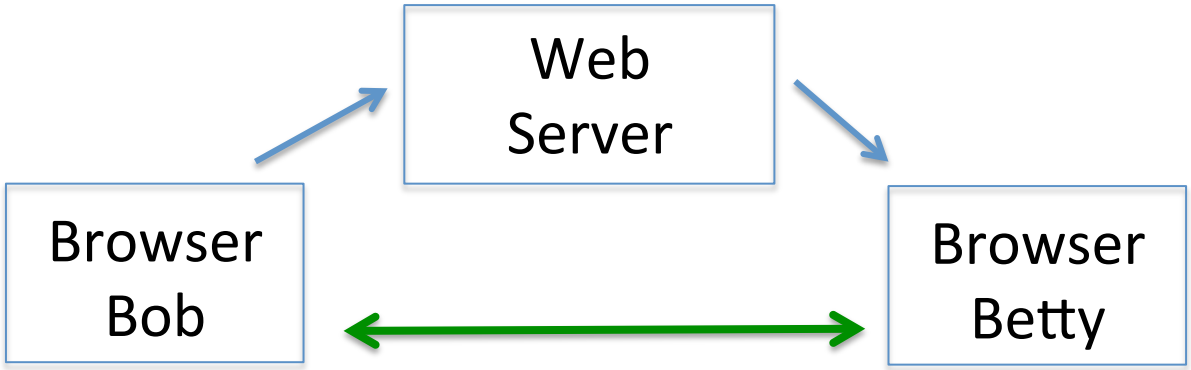
- With SDES, Web-server sees the key
  - IWF gateway sees the key even in DTLS-SRTP
- If evil Web-server force media to go through a path it can snoop, then it can decrypt media
- BUT, it can do that with DTLS-SRTP too
  - It just inserts itself as the IWF Gateway

No PFS



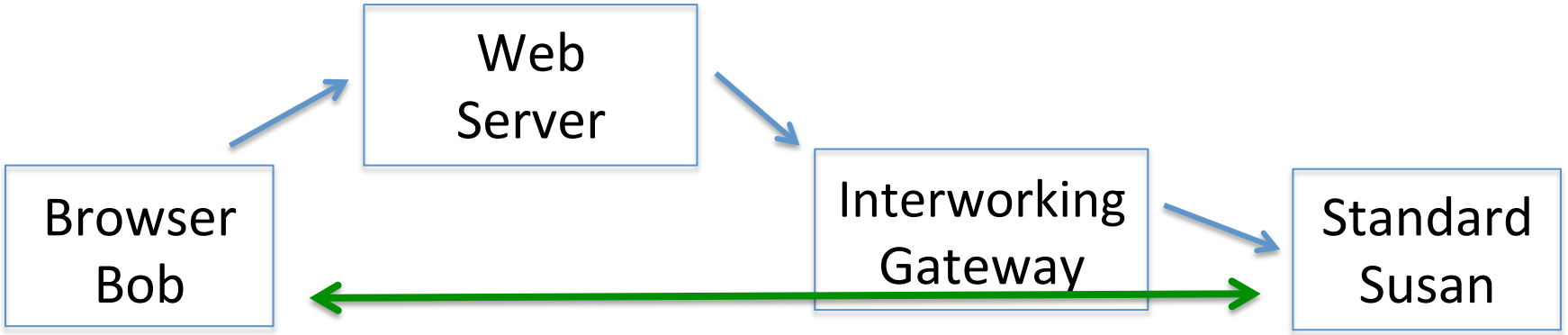
- With SDES, if a session key is ever discovered in the future, past media using it is compromised
  - Of course the media has to actually be available/ recorded
- BUT, DTLS-SRTP loses PFS with an IWF gateway too
  - Nothing prevents IWF from logging all keys

Susceptible to downgrade



- Evil Web-Server can make Bob think he's talking through an IWF Gateway, do SDES
- BUT, a Web-Server can already insert itself as a DTLS-SRTP B2BUA
  - It would be cheaper with SDES though

Unverifiable

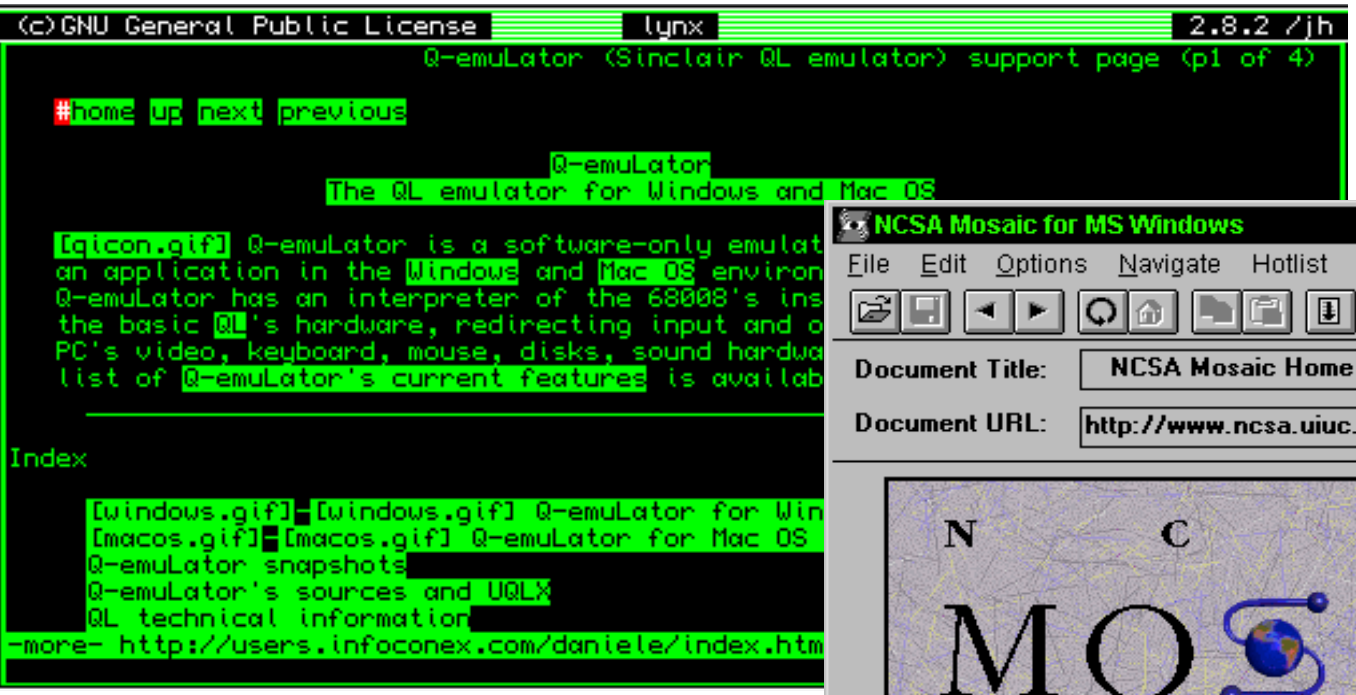


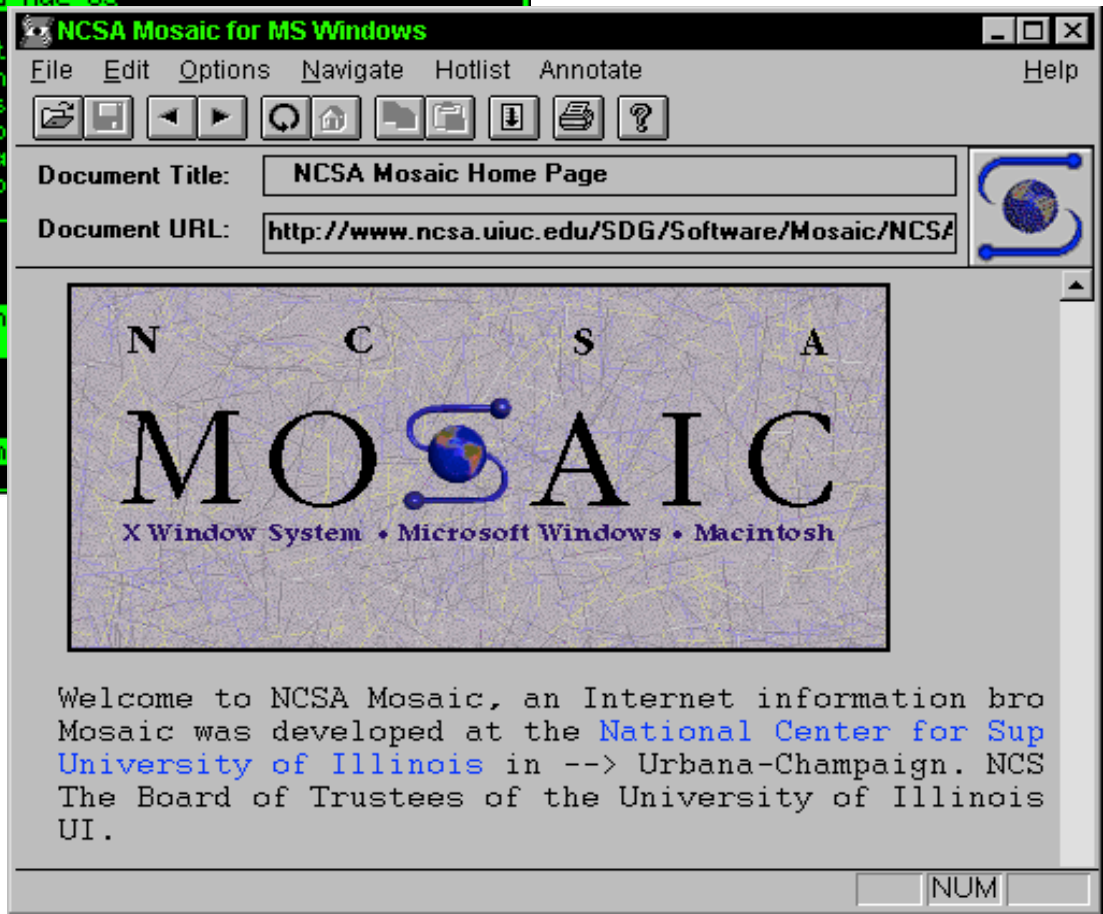
- With SDES, you don't know if the media is secure end-to-end
- The short answer is "yes you do know: it's NOT secure end-to-end"
  - DTLS-SRTP isn't either, until and unless you verify the keys or we have an IdP solution *deployed*
  - And for interworking cases DTLS-SRTP is not end-to-end secure, but who would know?

What if we're wrong?

- We'd remove/deprecate it
  - Security issues cause updates all the time

We've been told Browsers get upgraded often and quickly





Time for the Mic-line

- Let the circus begin...