Automated Analysis of TLS 1.3

0-RTT, Resumption and Delayed Authentication

IEEE S&P, 24/05/2016

Cas Cremers

Marko Horvat

Sam Scott

Thyla van der Merwe

mozilla

Where our work fits in

Where our work fits in

Dowling et al. [draft-05]
Kohlweiss et al. [draft-05]
Krawczyk and Wee [OPTLS]
Dowling et al. [draft-10]

We are here!

Motivation and approach

TLS 1.3 designed to be more efficient than TLS 1.2:

- 0-RTT handshake mode.
- PSK mode.
- Delayed client authentication.

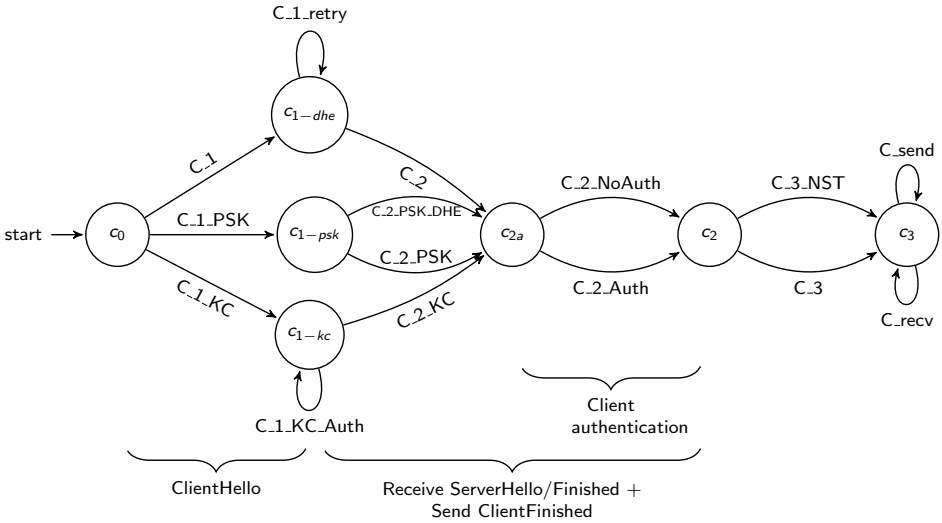
Our goal

Improve the security of TLS 1.3 by analysing the specification using state-of-the-art formal analysis methods.

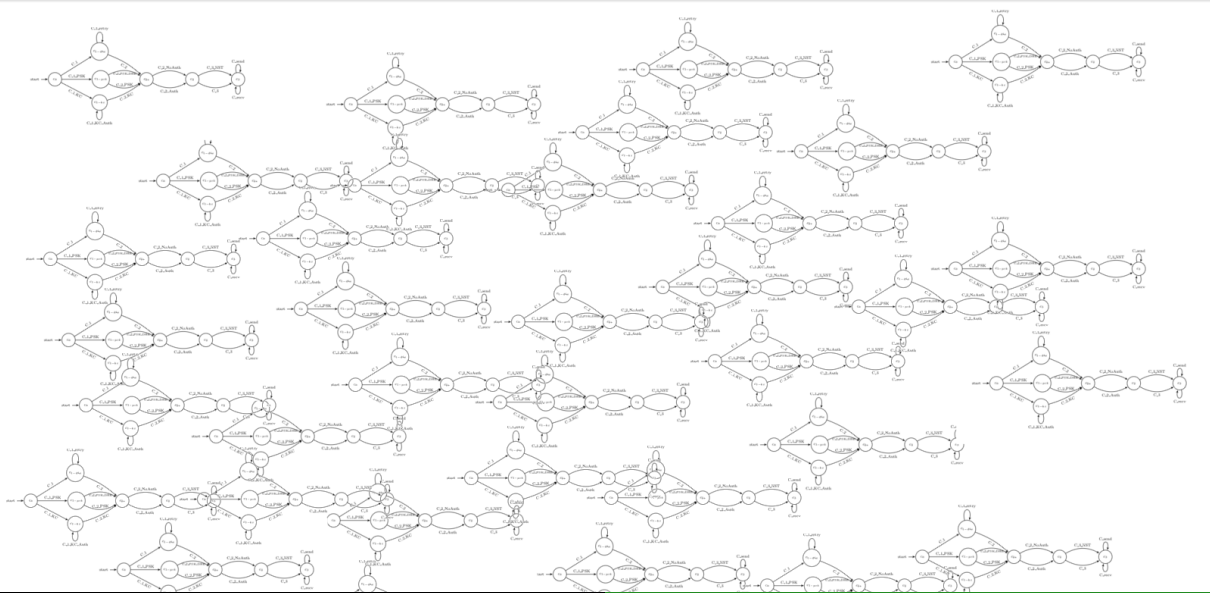
We focus on interaction attacks:

- Perfect cryptography assumption.
- Dolev-Yao attacker.

Building a model. Client state machine



Unbounded number of concurrent sessions



Tamarin

We encoded our model for use in the Tamarin prover:

- State-of-the-art tool for automated protocol analysis.
  - Loops.
  - Branches.
  - Symbolic Diffie-Hellman.
- However, requires considerable user interaction for very complex models.

Verification

We verified the core properties of TLS 1.3 revision 10 as an authenticated key exchange protocol:

- Secrecy of session keys.
  - Forward secrecy included.
- Unilateral and mutual authentication.
- Integrity of handshake messages.

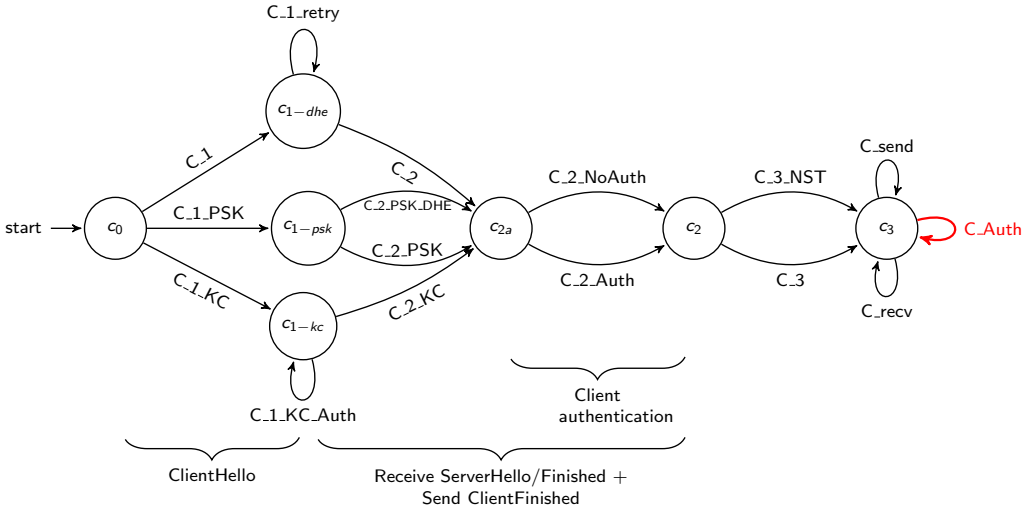
Verification

We verified the core properties of TLS 1.3 revision 10 as an authenticated key exchange protocol:

- Secrecy of session keys.
  - Forward secrecy included.
- Unilateral and mutual authentication.
- Integrity of handshake messages.

Is it safe to include delayed client authentication in revision 10?

Attacking client authentication

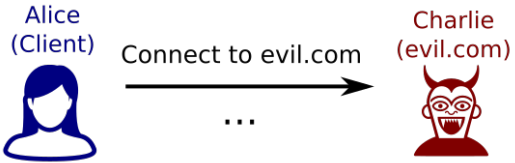


Attacking client authentication

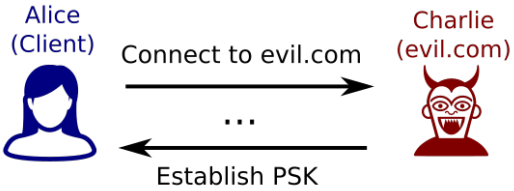
Alice
(Client)

Charlie
(evil.com)

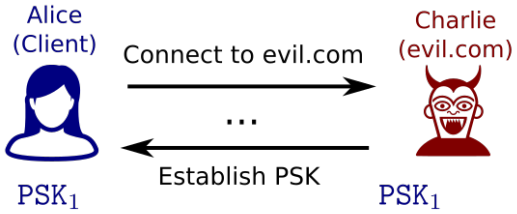
Attacking client authentication



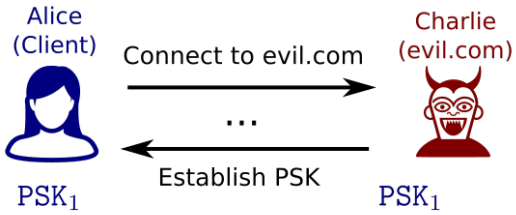
Attacking client authentication



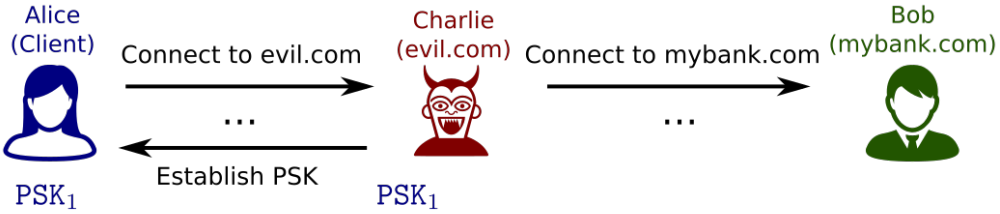
Attacking client authentication



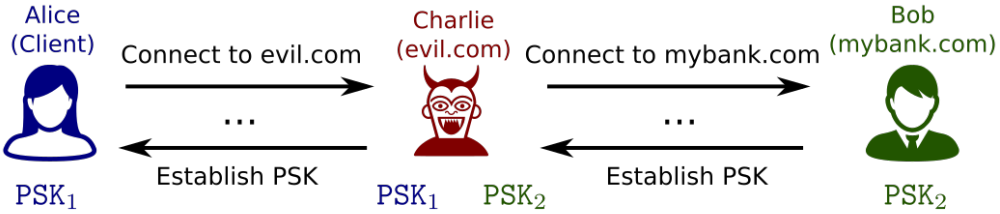
Attacking client authentication



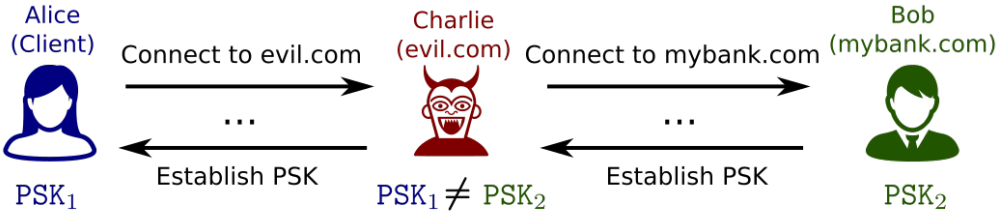
Attacking client authentication



Attacking client authentication



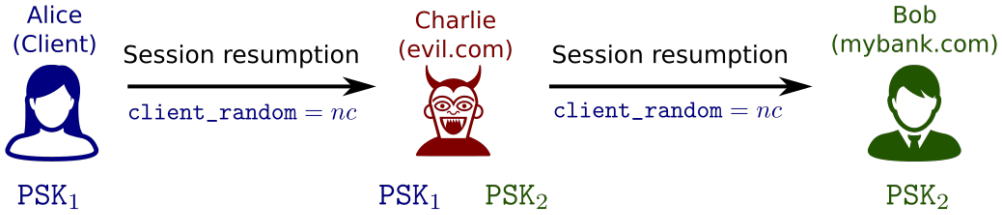
Attacking client authentication



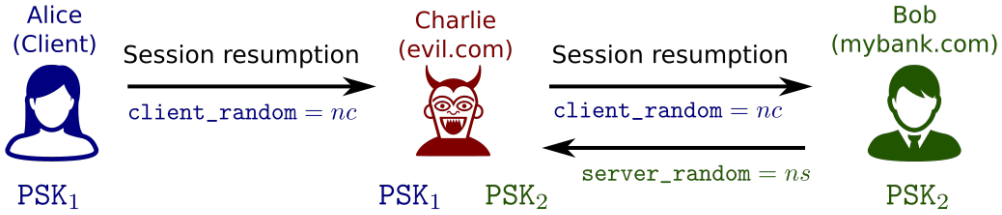
Attacking client authentication

$$\texttt{session\_hash} = H(nc \qquad)$$

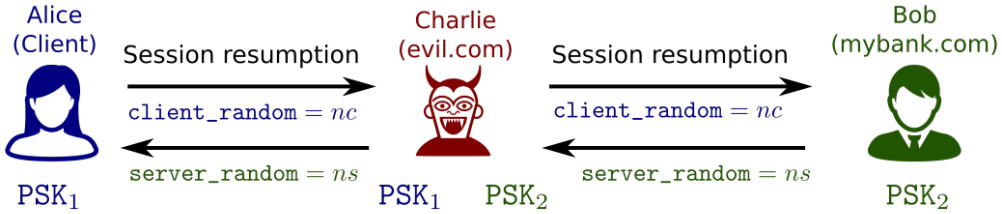
Attacking client authentication



Attacking client authentication



Attacking client authentication



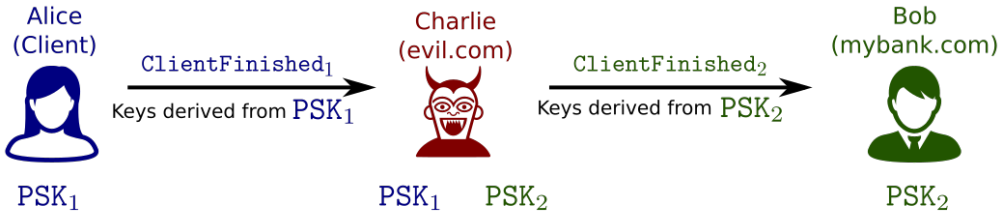
session_hash = $H(nc\ \ ns\ \ \dots)$

session_hash = $H(nc\ \ ns\ \ \dots)$

Attacking client authentication

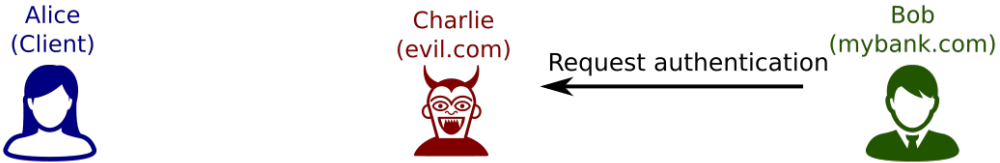
$\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$ $\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$

Attacking client authentication



$\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$ $\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$

Attacking client authentication



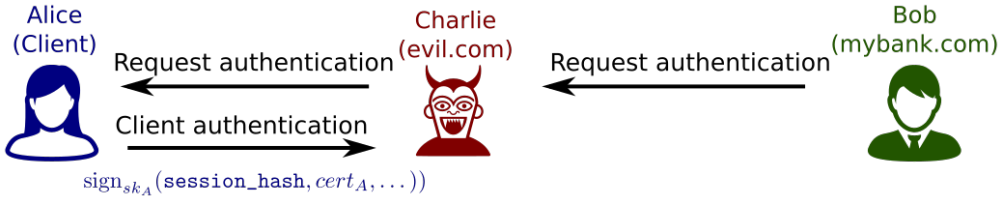
$\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$

$\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$

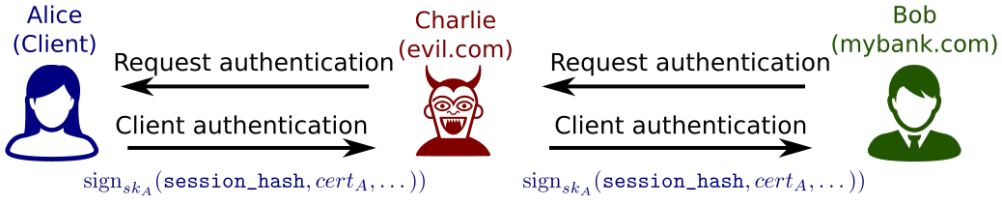
Attacking client authentication

$\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$ $\texttt{session\_hash} = H(nc\ \ ns\ \ \dots)$

Attacking client authentication



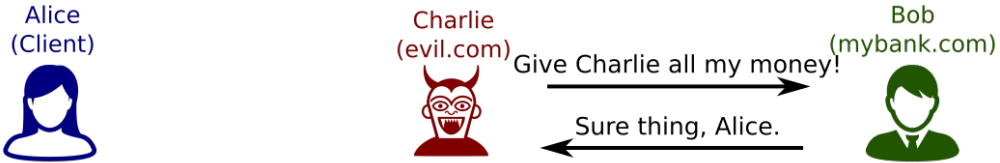
Attacking client authentication



$\texttt{session\_hash} = H(nc \;\; ns \;\; \ldots)$

$\texttt{session\_hash} = H(nc \;\; ns \;\; \ldots)$

Attacking client authentication



Disclosure

- We posted analysis results and attack to TLS mailing list end-October 2015.

  "Nice analysis! I think that the composition of different mechanisms in the protocol is likely to be where many subtle issues lie, and analyses like this one support that concern."

Disclosure

- We posted analysis results and attack to TLS mailing list end-October 2015.

  "Nice analysis! I think that the composition of different mechanisms in the protocol is likely to be where many subtle issues lie, and analyses like this one support that concern."

  "Thanks for posting this. It's great to see people doing real formal analysis of the TLS 1.3 draft; this is really helpful in guiding the design."

Disclosure

- We posted analysis results and attack to TLS mailing list end-October 2015.

  "Nice analysis! I think that the composition of different mechanisms in the protocol is likely to be where many subtle issues lie, and analyses like this one support that concern."

  "Thanks for posting this. It's great to see people doing real formal analysis of the TLS 1.3 draft; this is really helpful in guiding the design."

  "This result motivates and confirms the need to modify the handshake hashes to contain the server Finished when we add post-handshake authentication..."

Attack cause and mitigation

- Attack shows initial proposal for delayed client authentication incomplete.
- Highlights strict necessity of binding client signatures to server certificate.
- Working group proposed to include transcript to bind them to *sessions*.
- This proposal was merged in revision 11, which prevents our attack.

Conclusions

- First comprehensive analysis of the new TLS 1.3 modes and their interaction.
- This story has a happy ending:
  - Revision 10 was successfully verified.
  - Tamarin was used to find an interaction attack on delayed authentication.
  - Proposed fix verified and included in revision 11.
- Future work: Update model and verify revision 13.
- Our work is part of the larger, concerted effort of different approaches to hardening TLS 1.3.

Authors:

Cas Cremers
cas.cremers@cs.ox.ac.uk

Marko Horvat
mhorvat@mpi-sws.org

Sam Scott
sam.scott.2012@live.rhul.ac.uk

Thyla van der Merwe
thyla.vandermerwe.2012@live.rhul.ac.uk