Dealing with Key Compromise in CryptoVerif

Bruno Blanchet

INRIA Paris
Bruno.Blanchet@inria.fr

March 2023

CryptoVerif, http://cryptoverif.inria.fr/

CryptoVerif is a mechanized prover that:

- works in the computational model.
- generates proofs by sequences of games.
- proves secrecy, correspondence, and indistinguishability properties.
- provides a generic method for specifying properties of cryptographic primitives.
- works for N sessions (polynomial in the security parameter), with an active adversary.
- gives a bound on the probability of an attack (exact security).
- has automatic and interactive modes.

Extensions

- Proof of **secrecy**, when part of an array is secret, and part is public.
- New commands and game transformations:
  - **focus** $q_1, \dots, q_m$ tells CryptoVerif to prove only the properties $q_1, \dots, q_m$.
  - **success simplify** removes parts of the game such that the adversary cannot break the desired properties when they are executed.
  - **guess** the tested session, the value of a variable, which branch of a test is taken.

General strategy for dealing with key compromise

1. Insert events e_i executed when some authentication properties are broken (and the key is not compromised).
2. **focus** on proving **event**$(e_i) \Rightarrow$ **false**.
3. **success simplify** removes the compromise of the key.
4. We prove queries **event**$(e_i) \Rightarrow$ **false**.
5. We go back to before **focus** and prove the other properties (implicitly using the authentication properties already proved).

Applications

- Forward secrecy with respect to the compromise of the pre-shared key in TLS 1.3 and WireGuard.
- PRF-ODH with compromise of Diffie-Hellman exponents, illustrated on Noise NK.
- Forward secrecy for OEKE.
- Grouping compromise scenarios in WireGuard, by guessing which branch is taken.