NTNU

Norwegian University of Science and Technology

# Modelling the Security of Key Exchange

Colin Boyd
including joint work with Janaka Alawatugoda,
Juan Gonzalez Nieto

Department of Telematics, NTNU

Workshop on Tools and Techniques for Security Analysis
December 2016

NTNU

# Outline


Key Exchange Models Before eCK
- Introduction
- Bellare–Rogaway Model Evolution
- Canetti–Krawczyk Model Evolution

eCK Model and Beyond
- eCK Model
- Forward Secrecy
- Models including Functional Queries

Summary and Conclusion

# Diffie–Hellman key exchange

| A | | B |
|---|---|---|
| $r_A \in_R \mathbb{Z}_q$<br>$t_A = g^{r_A}$ | $\xrightarrow{t_A}$ | |
| | | $r_B \in_R \mathbb{Z}_q$<br>$t_B = g^{x_B}$ |
| | $\xleftarrow{t_B}$ | |
| $Z_{AB} = t_B^{r_A}$ | | $Z_{BA} = t_A^{r_B}$ |

— $r_A$ and $r_B$ are *ephemeral secrets*

— $Z_{AB}$ is the *shared secret*

# HMQV protocol

| A | | B |
|---|---|---|
| $r_A \in_R \mathbb{Z}_q$ | | |
| $t_A = g^{r_A}$ | $\xrightarrow{t_A}$ | $r_B \in_R \mathbb{Z}_q$ |
| | | $t_B = g^{r_B}$ |
| | $\xleftarrow{t_B}$ | |
| $S_A = r_A + dx_A \bmod q$ | | $S_B = r_B + ex_B \bmod q$ |
| $Z_{AB} = (t_B y_B^e)^{S_A}$ | | $Z_{BA} = (t_A y_A^d)^{S_B}$ |

- $r_A$ and $r_B$ are *ephemeral secrets*
- $x_A$ and $x_B$ are *long-term secrets*
- $y_A = g^{x_A}$ and $y_B = g^{x_B}$ are *public keys*
- $d = H(t_A, ID_B)$ , $e = H(t_B, ID_A)$

# NAXOS protocol

| A | | B |
|---|---|---|
| $r_A \in_R \mathbb{Z}_q$<br>$h_A = H_1(x_A, r_A)$<br>$t_A = g^{h_A}$ | $\xrightarrow{t_A}$ | $r_B \in_R \mathbb{Z}_q$<br>$h_B = H_1(x_B, r_B)$<br>$t_B = g^{h_B}$ |
| | $\xleftarrow{t_B}$ | |
| $K_{AB} = H_2(t_B^{x_A}, y_B^{h_A}, t_B^{h_A}, ID_A, ID_B)$ | | $K_{AB} = H_2(y_A^{h_B}, t_A^{x_B}, t_A^{h_B}, ID_A, ID_B)$ |

- $r_A$ and $r_B$ are *ephemeral secrets*
- $x_A$ and $x_B$ are *long-term secrets*
- $y_A = g^{x_A}$ and $y_B = g^{x_B}$ are *public keys*
- $K_{AB}$ is the *session key*

# Jeong–Katz–Lee protocol TS3

| A | | B |
|---|---|---|
| $r_A \in_R \mathbb{Z}_q$ | | |
| $t_A = g^{r_A}$ | $t_A, \mathrm{MAC}_{K_M}(ID_A, ID_B, t_A) \longrightarrow$ | |
| | | $r_B \in_R \mathbb{Z}_q$ |
| | | $t_B = g^{r_B}$ |
| | $t_B, \mathrm{MAC}_{K_M}(ID_B, ID_A, t_B) \longleftarrow$ | |
| $Z_{AB} = t_B^{r_A}$ | | $Z_{BA} = t_A^{r_B}$ |

- $r_A$ and $r_B$ are *ephemeral secrets*
- $x_A$ and $x_B$ are *long-term secrets*
- $y_A = g^{x_A}$ and $y_B = g^{x_B}$ are *public keys*
- $K_M$ is MAC key derived from static Diffie–Hellman $g^{x_A x_B}$


# Need for formal modelling

- History of failed protocol designs and 1980s and 1990s
- What is a valid attack?
- Obtain proofs of security
- Analysis of real world protocols

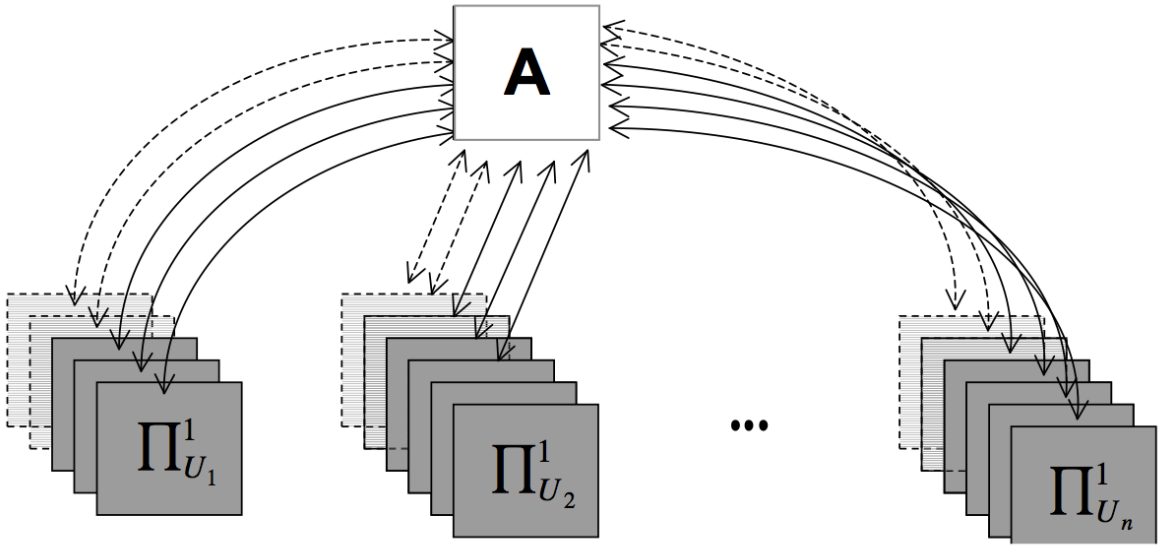
NTNU

# Bellare and Rogaway's security model

- First computational model, ACM CCS 1993
- Adversary controls the security game by querying a set of *sessions* at a party
- A session $\Pi_U^s$ represents the actions of party $U$ in the protocol run indexed by integer $s$
- Long-term keys are initialised using a key generation algorithm

NTNU

— The adversary *A* is computationally bounded to probabilistic polynomial time



# Adversarial queries in BR model

| Query | Inputs | Outputs |
|---|---|---|
| send | session + input message | output message |
| reveal | session | accepted session key |
| corrupt | party | long-term key |
| test | fresh session | session key / random |

— To win the security game the adversary must correctly decide the bit used in the answer to the test query

NTNU

# Freshness

— The test query may only be used for a *fresh* session
— An session is said to be *fresh* when:
  - it has accepted a session key, and
  - neither itself nor its *partner* have had a corrupt or reveal query
— The way of defining partners has varied in different models
— Original BR93 model defines partners to be sessions with *matching conversations*

NTNU

# BR model versions

| Model | Setting | Partnering mechanism |
|---|---|---|
| BR93 | 2-party shared key | Matching conversations |
| BR95 | Server-based | Partner function |
| SR96 | Smart card | Partner function |
| BWM97 | Public key | Matching conversations |
| BWJM97 | Key agreement | Matching conversations |
| BPR00 | Password-based | Session identifiers |

# Which elements are available to the adversary?

— Different models allow different combinations

| | |
|---|---|
| Actor | owner of the test session |
| Peer | (intended) partner of the test session |
| ✓ | element is available (leaked or chosen) |
| (✓) | element may be available |
| $\mathcal{F}$ | a (restricted) function of the element is available |

— Table shows only test session

— Usually all elements are available for non-test sessions

# Which elements are available to the adversary?

— BR model

| | Before test session | After test session |
|---|---|---|
| Actor long-term | | |
| Actor ephemeral | | |
| Peer long-term | | |
| Peer ephemeral | (✓) | |

For some protocols (such as HMQV) an active adversary can choose ephemeral key of peer session


# Modelling forward secrecy

- A protocol provides forward secrecy if adversary cannot distinguish session key from a random string even given the long-term keys after test session is complete
- Allow adversary to obtain long-term keys, *after* test session is complete
- Widely seen as desirable real-world property today
- Introduces *timing* into the model

NTNU

# Which elements are available to the adversary?

— BPR00 model with forward secrecy

| | Before test session | After test session |
|---|---|---|
| Actor long-term | | ✓ |
| Actor ephemeral | | |
| Peer long-term | | ✓ |
| Peer ephemeral | (✓) | |

— Which protocols provide forward secrecy?


# Canetti–Krawczyk (CK01) model

— Similar basic idea to Bellare–Rogaway models
— Two main motivations:
  - build secure channels for sessions
  - a modular design approach using *authenticators*
— Allows *session state* to be revealed

NTNU

# HMQV model

- Enhancement of CK01 model used to analyse HMQV protocol
- Session state query reveals ephemeral private key
- Key compromise impersonation (KCI) attacks are captured by allowing adversary to obtain private key of the owner of the test session

# Which elements are available to the adversary?

— HMQV model capturing KCI attack

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | | |
| Peer ephemeral | (✓) | |

— Which protocols provide KCI resistance?


# Common elements of all models

- Adversary controls network
- Some mechanism identifies partners of sessions
- Adversary can obtain session key from sessions other than test session and its partner (if it exists)
- Adversary wins by distinguishing session key of test session from random string

NTNU

# The eCK model

— Proposed at Provsec 2007 by LaMacchia, Lauter and Mityagin, now widely referred to as eCK model

— Tackles directly some limitations in the CK and BR models. Specific advantages are:

- the adversary can obtain ephemeral secrets which belong to the test session;
- the adversary can obtain the long-term key of the test session and of its partner even before the session is completed.

# Which elements are available to the adversary?

— eCK model

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | | |
| Peer ephemeral | ✓ | ✓ |

or

| | | |
|---|---|---|
| Actor long-term | | |
| Actor ephemeral | ✓ | ✓ |
| Peer long-term | | |
| Peer ephemeral | ✓ | ✓ |

or . . .


# Which elements are available to the adversary?

— eCK model if adversary is passive in test session

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | ✓ | ✓ |
| Peer ephemeral | | |

or

| | | |
|---|---|---|
| Actor long-term | | |
| Actor ephemeral | ✓ | ✓ |
| Peer long-term | ✓ | ✓ |
| Peer ephemeral | | |

— NAXOS protocol is secure in eCK model

NTNU

# Strong and weak forward secrecy

## Strong forward secrecy (sFS)

- Adversary takes an active part in the session under attack
- Victim executes session with the adversary

## Weak forward secrecy (wFS)

- Adversary is prevented from taking an active part in the session under attack
- Victim executes the session with a legitimate party

- eCK model *cannot* capture strong forward secrecy since it does not consider timing

# Which elements are available to the adversary?

— eCK-PFS (Cremers–Feltz, 2012)

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | | ✓ |
| Peer ephemeral | ✓ | ✓ |

or,

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | ✓ | ✓ |
| Peer ephemeral | | |

or . . .

NTNU

# Leakage resilient key exchange

- Aims to capture side channel attacks
- Adversary gets access to a chosen function of the long-term secret with some restrictions
  - Leakage can be continuous or bounded
  - Leakage can be restricted to before the test session occurs
- First results by Moriyama and Okamoto, 2011 – assume *before-the-fact* leakage
- ASB 2015 achieve *continuous, after the fact leakage* (CAFL) security in an eCK type model

# Which elements are available to the adversary?

— Leakage resilient model (CAFL-eCK)

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | $\mathcal{F}$ | $\mathcal{F}$ |
| Peer ephemeral | ✓ | ✓ |

or

| | | |
|---|---|---|
| Actor long-term | $\mathcal{F}$ | $\mathcal{F}$ |
| Actor ephemeral | ✓ | ✓ |
| Peer long-term | ✓ | ✓ |
| Peer ephemeral | | |

or . . .

— $\mathcal{F}$ is restricted function of long-term secret


# Post-compromise security

— Analysed by Cohn-Gordon, Cremers and Garratt, IEEE Security and Privacy 2016

— Adversary can obtain (partial) information about long-term key *before* test session

— Models temporary loss of long-term secrets

# Which elements are available to the adversary?

— Post-compromise security - weak compromise

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | $\mathcal{F}$ | |
| Peer ephemeral | ✓ | ✓ |

— $\mathcal{F}$ is interface to long-term secret, for example *HSM*

— $\mathcal{F}$ queries can be added to adversary queries for test session before completed

— Seems similar to CAFL-eCK but restrictions on $\mathcal{F}$ are different

# Which elements are available to the adversary?

— Post-compromise security - full compromise

| | Before test session | After test session |
|---|---|---|
| Actor long-term | ✓ | ✓ |
| Actor ephemeral | | |
| Peer long-term | ✓ then ✘ | ✓ |
| Peer ephemeral | | |

— Can only be satisfied using *stateful* protocols

— Long-term keys evolve over time (ratcheting)

# Which elements are available to the adversary?

— Mass surveillance model?

| | Before test session | After test session |
|---|---|---|
| Actor long-term | | ✓ |
| Actor ephemeral | | ✓ |
| Peer long-term | | ✓ |
| Peer ephemeral | | ✓ |

— Adversary is passive before test session

— Adversary can learn secrets after test session

— No stateless protocol is secure in this model

# Which elements are available to the adversary?

— Weaker mass surveillance model?

| | Before test session | After test session |
|---|---|---|
| Actor long-term | | |
| Actor ephemeral | | ✓ |
| Peer long-term | | |
| Peer ephemeral | | ✓ |

— Adversary is passive before test session

— Adversary can learn secrets after test session

— No TLS 1.2 or TLS 1.3 variant is secure in this model

NTNU

Key Exchange Models Before eCK

- Introduction
- Bellare–Rogaway Model Evolution
- Canetti–Krawczyk Model Evolution

eCK Model and Beyond

- eCK Model
- Forward Secrecy
- Models including Functional Queries

Summary and Conclusion


# Types of models

- Security goal is session key indistinguishability:
  - original BR and CK models
  - added adversary queries: eCK, eCK-PFS
  - added functional access to long-term key: CAFL, PCS
- Security goal is channel security:
  - ACCE for authenticated encryption
  - different authentication levels
  - general functional test (Krawczyk, CCM 2016)
  - different adversary queries could be added
- Is indistinguishability the right definition for real-world key exchange?


# Current and future challenges

- Security against ephemeral key leakage for real-world protocols
- Post-quantum security
- Taming complexity . . . with automation?
- Classifying and unifying models . . . stateful protocols, functional security, . . .
- More real-world protocols: DTLS, ZRTP, . . .
- Modelling humans
- All of the above in the group setting


# Key References


— Entity authentication and key distribution, M Bellare, P Rogaway - ACM CCS 1993

— Analysis of key-exchange protocols and their use for building secure channels, R Canetti, H Krawczyk - Crypto 2001

— Stronger security of authenticated key exchange, B LaMacchia, K Lauter, A Mityagin - International Conference on Provable Security, 2007

— On forward secrecy in one-round key exchange, C Boyd, JG Nieto - IMA Conference on Cryptography and Coding, 2011

— Examining indistinguishability-based security models for key exchange protocols: the case of CK, CK-HMQV, and eCK, C Cremers, ASIACCS 2011


# Key References


- Leakage resilient eCK-secure key exchange protocol without random oracles D Moriyama and T Okamoto, ASIACCS 2011
- Beyond eCK: perfect forward secrecy under actor compromise and ephemeral-key reveal, C Cremers, M Feltz - Designs, Codes and Cryptography, 2015
- Continuous after-the-fact leakage-resilient eCK-secure key exchange, J Alawatugoda, D Stebila, C Boyd - IMA Conference on Cryptography and Coding, 2015
- On Post-Compromise Security, K Cohn-Gordon, C Cremers, L Garratt - IEEE Security and Privacy Symposium 2016
- A Unilateral-to-Mutual Authentication Compiler for Key Exchange (with Applications to Client Authentication in TLS 1.3), Hugo Krawczyk - ACM CCS 2016

NTNU

Norwegian University of Science and Technology

# Modelling the Security of Key Exchange

Colin Boyd
including joint work with Janaka Alawatugoda,
Juan Gonzalez Nieto

Department of Telematics, NTNU

Workshop on Tools and Techniques for Security Analysis
December 2016