Security Protocols and Verification

Attack of Cryptographic Protocols

Garance Frolla
Ely Marthouret
Ewan Decima

Team: **ASKO OM8464A2**

September / November 2025

Contents

1 Attack on Chronos V2 **2**

2 Attack Description **2**

2.1 Assumptions 2
2.2 Attack Flow 3
2.3 Attack Results 4

1 Attack on Chronos V2

Its actually the same attack as before but with the ids.

1. $A \to I(S) : ID_A, \{|K, M|\}_{K_{AS}}, \{|B|\}_{K_{AS}}$
2. $I(A) \to S : ID_A, \{|K, M|\}_{K_{AS}}, \{|B|\}_{K_{AS}}$
3. $S \to I(B) : ID_S, \{|K|\}_{K_{BS}}, \{|A|\}_{K_{BS}}$
4. $I \to S : ID_I, \{|K_I, M_I|\}_{K_{IS}}, \{|B|\}_{K_{IS}}$
5. $S \to I(B) : ID_S, \{|K_I|\}_{K_{BS}}, \{|I|\}_{K_{BS}}, \{M_I\}_{pub(B)}$
6. $I(S) \to B : ID_S, \{|K_I|\}_{K_{BS}}, \{|A|\}_{K_{BS}}\{M\}_{pub(B)}$
7. $B \to I(A) : ID_B, \{|M+1|\}_{K_I}$
8. $I(A) \to S : ID_A, \{|K, M|\}_{K_{AS}}, \{|I|\}_{K_{AS}}$
9. $S \to I : ID_S, \{|K|\}_{K_{IS}}, \{|I|\}_{K_{IS}}$
10. $I(B) \to A : ID_B, \{|M+1|\}_K$

2 Attack Description

2.1 Assumptions

Assumption: the attack relies solely on the intruder I possessing $\{|I|\}_{K_{AS}}$. This is a plausible assumption within the considered threat model. In order to get its I can do :

1. $I \to S : ID_S, \{|K, M|\}_{K_{IS}}, \{|A|\}_{K_{IS}}$
2. $S \to I(A) : ID_S, \{|K|\}_{K_{AS}}, \{|I|\}_{K_{AS}}$

intercepts the message that S sends to A and stop the communication.

The intruder I can easily learn the identities of A, B and S: agent identifiers are not secret and can be obtained simply by initiating legitimate protocol runs. For example, I can engage A to obtain ID_A as follows

1. $I \to S : ID_I, \{|K, M|\}_{K_{IS}}, \{|B|\}_{K_{IS}}$.
2. $S \to A : ID_S, \{|K|\}_{K_{AS}}, \{|I|\}_{K_{AS}}, \{|M|\}_{pub(A)}$.
3. $A \to I : ID_A, \{|M+1|\}_K$.

By the same procedure I can obtain ID_B (replace $A$with B in the above exchange).

2.2 Attack Flow

- **Message 1:** A initiates the protocol, wanting to establish a secure session with B through the trusted server S. However, the intruder I intercepts this message while impersonating the server S. I stores $\{|K, M|\}_{K_{AS}}$ for later use. $\mathcal{K}_1 = \Big\{K(ID_A), K(ID_B), K(ID_S)$ $K\big(\{|K, M|\}_{K_{AS}}\big), K\big(\{|B|\}_{K_{AS}}\big), K\big(\{|I|\}_{K_{AS}}\big)\Big\}$

- **Message 2:** The intruder I forwards A's original message to the real server S, maintaining the deception. I impersonates A to S, making S believe the request is legitimate. S decrypts the message using K_{AS} and learns that A wants to communicate with B using session key K.

- **Message 3:** Server S, believing the request is legitimate, prepares to forward the session key to B. However, I intercepts this message while impersonating B. I cannot directly read this message intended for B. $\mathcal{K}_3 = \mathcal{K}_1 \cup \Big\{K\big(\{|K|\}_{K_{BS}}\big), K\big(\{|A|\}_{K_{BS}}\big)\Big\}$

- **Message 4:** This is where the intruder launches a parallel session. I initiates their own session with server S, generating their own session key K_I and nonce M_I. I sends these to S encrypted with K_{IS}, requesting a session with B. This parallel session runs alongside the original A-to-B session.

- **Message 5:** Server S responds to I's request by sending the session key K_I to what it believes is B, encrypted with K_{BS}. S includes I's identity encrypted with K_{BS}, and the nonce M_I encrypted with B's public key. The intruder I intercepts this message.

- **Message 6:** This is the critical substitution attack. The intruder I crafts a fraudulent message to B by combining elements from different sessions. I takes the session key K_I (from message 5, which I knows) encrypted with K_{BS}, but substitutes I's identity with A's identity from message 3. I also replaces $\{M_I\}_{pub(B)}$ with the original encrypted nonce $\{M\}_{pub(B)}$ from A's session. B receives this message, decrypts it, and believes that A wants to establish a session using key K_I (which B thinks is the key from A, but is actually the intruder's key).

- **Message 7:** B, believing they are responding to A with the correct session key, computes $M + 1$ and encrypts it with what they think is A's session key: K_I. However, B actually uses K_I (the intruder's key).

I intercepts this message and can decrypt it because I possesses K_I. $\mathcal{K}_7 = \mathcal{K}_3 \cup \{K(M)\}$

- **Message 8:** The intruder I, still maintaining the parallel session, sends another message to server S while impersonating A. I sends the original $\{|K, M|\}_{K_{AS}}$ from A's initial request, but changes the intended recipient from B to I. (*see 2.1*).

- **Message 9:** Server S responds to what it believes is A's request to establish a session with I. S sends back K encrypted with K_{IS}, along with I's identity. The intruder I receives this message and can decrypt it, confirming access to the session key K. $\mathcal{K}_9 = \mathcal{K}_7 \cup \{K(K)\}$

- **Message 10:** Finally, I forwards B's authentication response ($M + 1$) to A, but encrypted with the original session key K instead of K_I. Since I knows both K (from messages 8-9) and the plaintext $M + 1$ (decrypted from message 7 using K_I), I can re-encrypt the message appropriately. A receives $\{|M+1|\}_K$ and believes that B has successfully authenticated and possesses the session key K.

2.3 Attack Results

At the conclusion of this attack, A and B believe they have successfully established a secure session with each other, but they are actually using different session keys:

- A **believes:** They completed the protocol with B and that the session key K (which A generated) is shared only with B and server S.

- B **believes:** They completed the protocol with A and that the session key K_I (which B thinks came from A) is shared only with A and server S.

- **In reality:** The intruder I knows both session keys K and K_I. I can act as a man-in-the-middle, decrypting all messages from both A and B, reading them, and potentially modifying them before re-encrypting and forwarding them to maintain the illusion of direct communication.