BTI 4202: From Secure Channels to Key Management

Christian Grothoff

Berner Fachhochschule

7.5.2021
Learning Objectives

Asynchronous secure channels

Key Management: An Example for Architecture vulnerabilities

Introduction to GnuPG

Introduction to p≡p

Example Vulnerability: The Insecurity of WEP

References
1. Attack against Otway-Rees protocol
2. Compromise of long term keys
3. Known session-key attacks: Kerberos and Otway-Rees
4. Attacking synchronized clock protocols: Kerberos
5. Man in the middle attack on DH
Otway-Rees protocol

1: $M, A, B, \{N_a, M, A, B\}_{K_{as}}$

2: $M, A, B, \{N_a, M, A, B\}_{K_{as}}, \{N_b, M, A, B\}_{K_{bs}}$

3: $M, \{N_a, K_{ab}\}_{K_{as}}, \{N_b, K_{ab}\}_{K_{bs}}$

4: $M, \{N_a, K_{ab}\}_{K_{as}}$
Part I: Asynchronous Secure Channels
Reminder: Forward secrecy

What happens if your private key is compromised to your past communication data?
Asynchronous forward secrecy: SCIMP

Idea of Silence Circle’s SCIMP:

Replace key with its own hash.

- New key in zero round trips!
- Forward secrecy!
Suppose your regain control over your system. What happens with your *future* communication data?
Axolotl / Signal Protocol
Securing unidirectional communication

- Alice knows Bob’s public key $B$
- Alice wants to send $M$ to Bob
- Alice cannot receive messages from Bob (possibly ever)
Securing unidirectional communication

- Alice knows Bob’s public key $B$
- Alice wants to send $M$ to Bob
- Alice cannot receive messages from Bob (possibly ever)

Suggestion:

$$K := DH(T_A^{priv}, B)$$  \hspace{1cm} (1)

$$C := E_K(S_A(T_A^{pub}, A, B)||M)$$  \hspace{1cm} (2)

With Curve25519, cryptography has 92–128 bytes overhead:

- one or two 32 byte public keys
- one 64 byte EdDSA signature
- (plus HMAC)

What are the security properties we get here?
Part II: Trust Issues in X.509
Guiding questions “SSL and the Future of Authenticity”

- What is fundamentally wrong with the current CA model?
- What is the idea of “trust agility”, and is it reasonable?
- Understand the notion of “perspectives”. Evaluate strengths and weaknesses of the perspective model.
Interlude: SSL and the Future of Authenticity

BlackHat 2011
Break
Part III: Introduction to GnuPG
GnuPG

- Free version of PGP, with library (libgcrypt)
- Provides common cryptographic primitives
- Provides implementation of OpenPGP (RFC 2440)
- Commonly used for secure E-mail
- Provides web of trust
Using GnuPG

$ gpg --gen-key
$ gpg --export
$ gpg --import FILENAME
$ gpg --edit-key EMAIL; > fpr > sign > trust
$ gpg --clearsign FILENAME
The Web of Trust

Problem:

- Alice has certified many of her contacts and flagged some as trusted to check keys well.
- Bob has been certified by many of his contacts.
- Alice has not yet certified Bob, but wants to securely communicate with him.

Solution:

- Find paths in the certification graph from Alice to Bob.
- If sufficient number of short paths exist certifying the same key, trust it.
The Web of Trust

Problem:
- Alice has certified many of her contacts and flagged some as trusted to check keys well.
- Bob has been certified by many of his contacts.
- Alice has not yet certified Bob, but wants to securely communicate with him.

Solution:
- Find paths in the certification graph from Alice to Bob.
- If sufficient number of short paths exist certifying the same key, trust it.
Excercise: Explore

https://pgp.mit.edu
Break
Part IV: Introduction to $p\equiv p$
Please join Big Blue Button

https://meet.pep.security/b/chr-8ym-6a5-zim
Part V: Insecurity of WEP
Homework: WEP Insecurity

Read the article “Intercepting Mobile Communications: The Insecurity of 802.11” until section 4.2. For each of the attacks, decryption (section 3), message modification (section 4.1) and message injection (section 4.2) explain:

▶ How does the attack work?
▶ Why does it work (i.e., what are the flaws that make the attack possible)?