Improving Encryption Expressivity with Multi-Locking

Cyrius NUGIER

Context: Increasingly complex regulation systems emerge with specific control access requirements from various parties.

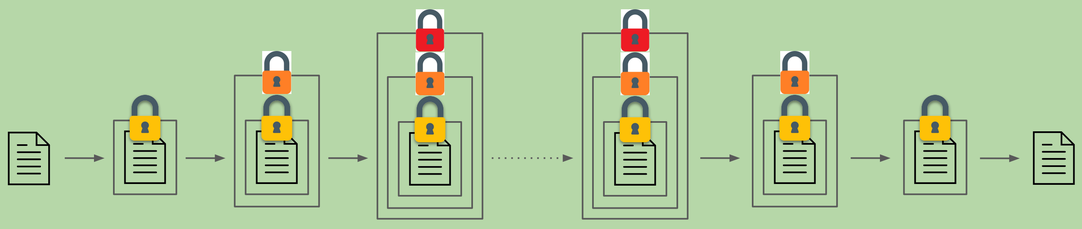
Objective: A secure way for various users apply control policies in a predetermined order.

Current solutions: Super-encryption, Identity[1] / Fuzzy-Identity[2] / Attribute Based Encryption [3]

Super Encryption

$Dec_{k1} \circ Dec_{k2} \circ \dots \circ Enc_{k2} \circ Enc_{k1}\ (m) = m$

Locks acts like a pile (FIFO)

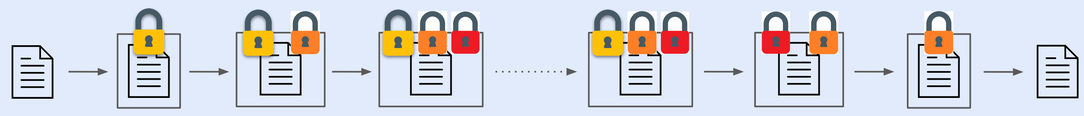


Multi-Locking

Super Encryption with Commutativity:

$Dec_i \circ Enc_j = Enc_j \circ Dec_i$

Locks acts like a set



The question: Which access policies can be enforced with Multi-Locking but not with (non commutative) Super Encryption ?

Sequential Circuits

Determines the order of application of access policies.

Binary operators AND, OR and THEN(→) linking users.

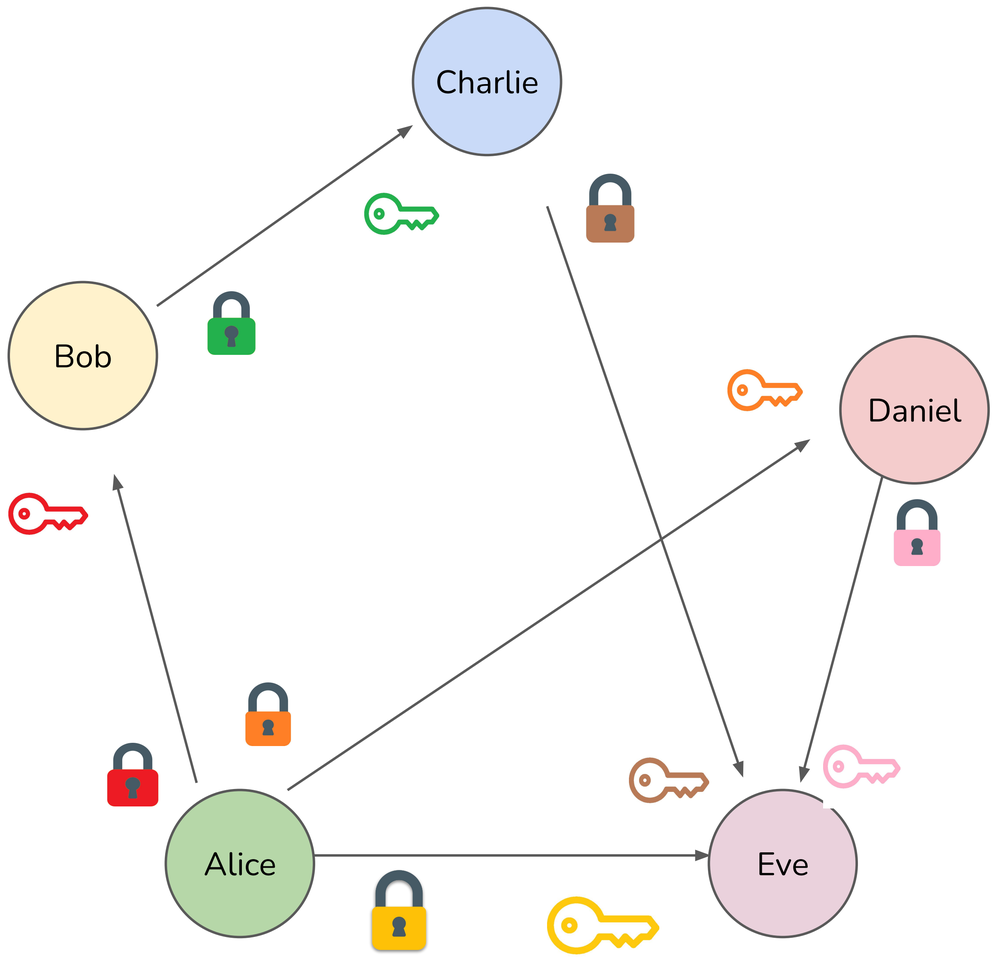
Alice → (Bob → Charlie) OR Daniel → Eve

Alice → Bob OR Charlie → Daniel → Eve

Alice → (Bob AND Charlie) → Daniel

Lock-Key Graphs

Represent the users and the control policies
The start of an edge is an encryption and the end is a decryption with the corresponding key



Our Results

Proof 1 — A Lock-Key graph enforces a Sequential circuit if all valid sequences of users are paths in the graph.

Proof 2 — For all Sequential Circuits, there exist at least a Lock-Key graph that enforces it (a fast algorithm finds it).

Proof 3 — The security of the message is at least the one of the scheme used for the encryption between the first and the last user.

Proof 4 — Super Encryption schemes are all planar

Methodology

- Determine a Sequential Circuit following specifications
- Find a Lock-key graph enforcing it
- Add edges for extra access control required by specifications
- Chose a Multi-Locking Family (with interesting crypto schemes)
- Chose a scheme for each edge and distribute keys

Multi-Locking Families

Sets of schemes with commutativity between encryptions and decryptions

- Xor / One time pad
- Modular Multiplication
  - ✗ RSA : not secure since the factorisation is shared across schemes for commutativity[4]
  - ✓ No secret shared between schemes Include pairing-based crypto such as Identity / Fuzzy Identity / Attribute Based Encryption

References:

[1] A. Shamir, “Identity-based cryptosystems and signature schemes,” Advances in Cryptology pp. 47–53, 1985.
[2] W. B. Sahai A., “Fuzzy identity-based encryption,” Cramer R. (eds) Advances in Cryptology – EUROCRYPT 2005. EUROCRYPT 2005. Lecture Notes in Computer Science, vol3494. Springer, Berlin, Heidelberg, 2005
[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” Proceedings of the ACM Conference on Computer and Communications Security pp. 89–98, 01 2006.
[4] D. Boneh et al. , “Twenty years of attacks on the rsa cryptosystem,” Notices of the AMS , vol. 46, no. 2, pp. 203–213, 1998.
[5] Icons made by Freepik, Pixel perfect from www.flaticon.com

LAAS-CNRS
/ Laboratoire d’analyse et d’architecture des systèmes du CNRS

Laboratoire conventionné avec l’Université Fédérale Toulouse Midi-Pyrénées