Timed Release Encryption (TRE)

Protocols – Possible Applications – Additional properties
Current research + Open Tasks

Konstantinos Chalkias
(PhD candidate)
Computational Systems and Software Engineering (CSSE) Lab.,
Dept. of Applied Informatics
University of Macedonia
Thessaloniki, Greece

chalkias <ατ> java.uom.gr

"Emerging Topics in Cryptographic Design and Cryptanalysis", Ecrypt PhD Summer School 2007, Samos, Greece

Introduction

- The aim of TRE is to support applications where encrypted confidential data must not be decrypted by anyone, including the designated recipient(s), until a predetermined future time-instant.

Possible applications

- **Electronic voting**, which requires delayed opening of votes.
- **Sealed-bid auctions**, where bids must stay sealed until the bidding period is over.
- **Internet – based contests**, where participating teams must not access the challenge problem before the contest starts.
- **Online games, e-lotteries and card games**, where a player should be able to verify that the game is run honestly by the "house".

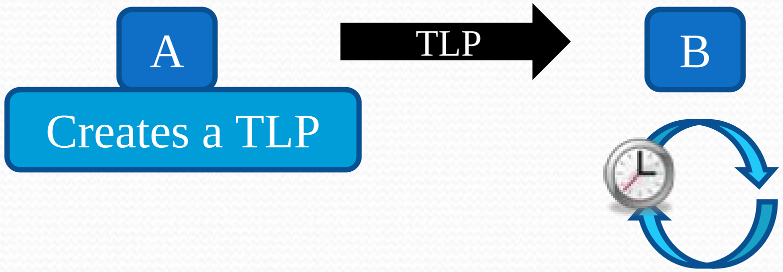
TRE approaches

$x = a^{2^t} \pmod{n},\ n = pq$

$k = 2^t \pmod{\varphi(n)}$
$x = a^k \pmod{n}$

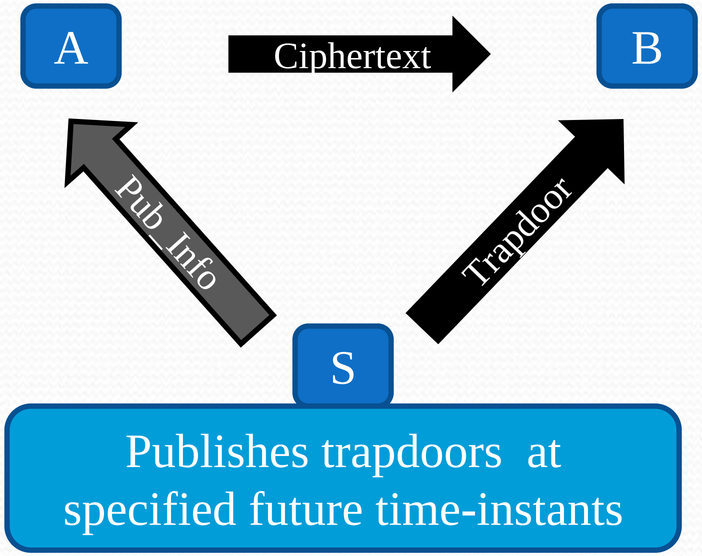
1. CPU-based (Time-Lock Puzzles)

- depend on the hardware characteristics
- cannot guarantee precise timing of information release
- impractical for many real-life settings

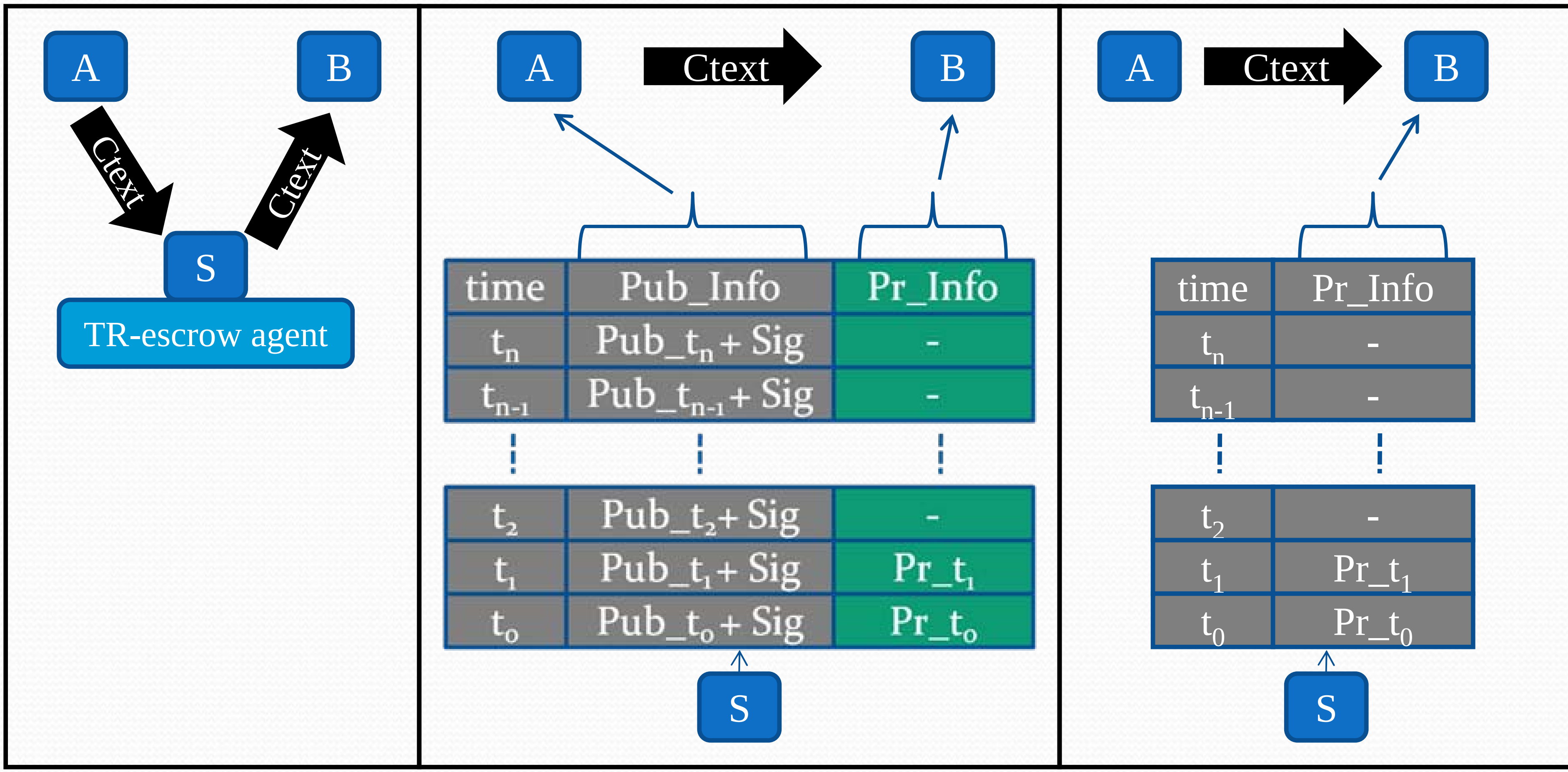


2. Agent-based (using Time-Servers)

- 3rd party based
- provide absolute timing



Agent-based TRE models

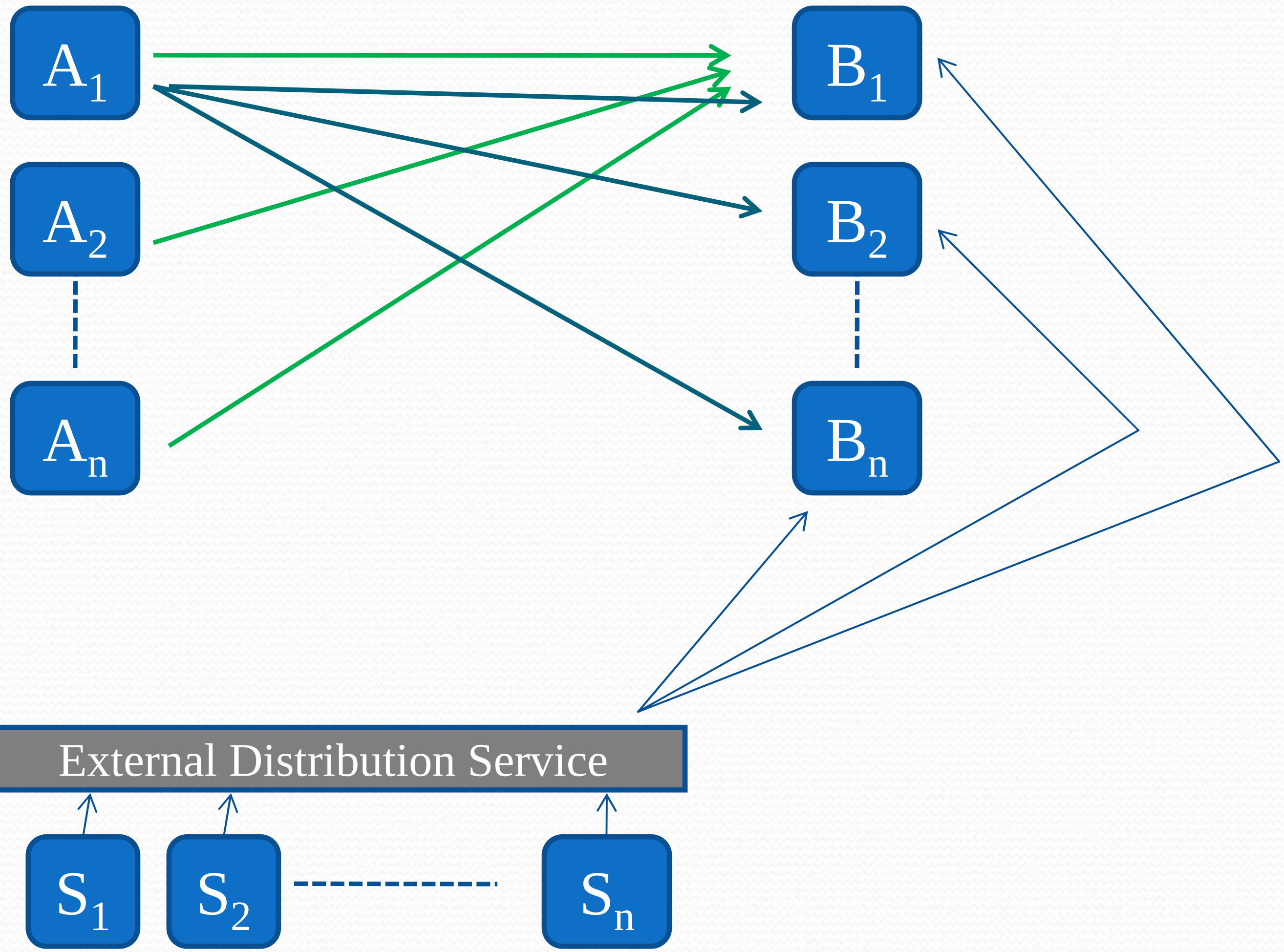


1993
(T. May)

1996
(Rivest et al.)

2003- …
(Mont et al. ,
Blake and Chan)

Ideal Agent-based TRE Protocol



Example TRE Protocol (1)

Modified version of [Blake and Chan ’04], also in [Cathalo et al. ’05]

Setup:

1. On input a security parameter security parameter k_0, generate parameters $\langle G_1, G_2, e, p \rangle$ where $e : G_1 \times G_1 \rightarrow G_2$ is a pairing on groups of prime order p
2. Select two hash functions $h_1 : \{0, 1\}^* \rightarrow G_1^*, \; h_2 : G_2 \rightarrow \{0, 1\}^n$, where n is the length of plaintext
3. Choose an arbitrary generator $P \in G_1$
4. Set Time-server’s (TS) key-pair $\langle s \xleftarrow{R} \mathbf{Z}_p^*, S = sP \rangle$
5. The ciphertext space is $C = G_1 \times \{0, 1\}^{n+\tau}$, the plaintext space is $M = \{0,1\}^n$, the public parameters are:

$$params = \{k_0, p, G_1, G_2, P, S, e, h_1, h_2, n, \tau, M, C\}$$

Example TRE Protocol (2)

TS-Release: Given $t \in \{0,1\}^{\tau}$, the server computes $T = h_1(t)$ and discloses a trapdoor $W = sT \in G_1$

User Keygen: on input $params$ return user's key-pair $\langle a \xleftarrow{R} \mathbf{Z}_p^*, A = aP \rangle$

Encryption: (Given A, S, send $m \in \{0,1\}^n$ to be decrypted at $t \in \{0,1\}^{\tau}$)

- select $r \xleftarrow{R} \mathbf{Z}_p^*$
- compute $T = h_1(t)$
- compute $c_1 = rA = raP$
- compute key $k = e(S,T)^r = e(P,T)^{rs}$
- compute $c_2 = m \oplus k$

$C : \langle c_1, c_2, t \rangle$

Decryption: Given S, C, a, W

- compute $T = h_1(t)$
- check server's trapdoor: $e(S,T) \stackrel{?}{=} e(P,W)$
- compute key as $k = e(c_1, W)^{1/a} = e(raP, sT)^{1/a} = e(P,T)^{rs}$
- retrieve message: $m = c_2 \oplus k$

Properties + Open Tasks

Desirable Properties:

- Confidentiality of release time
- Pre-open capability
- Easy construction of "old" trapdoors
- Multi - time-server support
- Multi - receiver support

Open Tasks:

- Design a "mixed" protocol that combines all of the desirable properties, improve in efficiency 1. use BB → short signatures instead of BLS, 2. use simple public key format VS. [BC04, CLQ05] $Pub_A : \langle aP, aS \rangle$
- Design protocols for specific applications (e.g., sealed-bid auctions).
- Implementation (protocols / time-servers).
- There are problems on proposed IB-TRE schemes (e.g., in order to improve in speed, [BC04] assume that the time-servers and TA's are the same entities).
- What about a Certificate-less TRE scheme?
- All of the modern TRE schemes are one-pass protocols! Why?? FS, UKS, KCI threats.

TRE References

[1] I. F. Blake and A. C.-F. Chan. Scalable, server-passive, user-anonymous timed release cryptography. In 25th IEEE Int'l. Conf. on Distributed Computing Systems, pp. 504-513. IEEE Computer Society, 2005.

[2] J. Cathalo, B. Libert, and J.-J. Quisquater. Efficient and non-interactive timed-release encryption. In Intl. Conf. on Information and Communications Security, LNCS 3783, pp. 291-303. Springer-Verlag, 2005.

[3] K. Chalkias and G. Stephanides. Timed release cryptography from bilinear pairings using hash chains. In 10th IFIP CMS, pp. 130-140. Springer-Verlag, 2006.

[4] A. W. Dent and Q. Tang. Revisiting the security model for timed-release public-key encryption with pre-open capability. In Cryptology ePrint Archive: Report 2006/306, 2006.

[5] Y. H. Hwang, D. H. Yum, and P. J. Lee. Timed-release encryption with pre-open capability and its application to certified e-mail system. In Information Security Conf., LNCS 3650, pp. 344-358. Springer-Verlag, 2005.

[6] M. C. Mont, K. Harrison, and M. Sadler. The hp time vault service: Innovating the way confidential information is disclosed at the right time. In Intl. World Wide Web Conf., pp. 160-169. ACM Press, 2003.

[7] D. Nali, C. Adams, and A. Miri. Time-based release of confidential information in hierarchical settings. In Information Security, LNCS 3650, pp. 29-43. Springer-Verlag, 2005.

[8] I. Osipkov, Y. Kim, and J.-H. Cheon. Timed-release public key based authenticated encryption. In http://eprint.iacr.org/2004/231, 2004.