Ensuring TANGO Control System

Sergi Blanch-Torné[1], Ramiro Moreno Chiral[2]

[1] Escola Politècnica Superior, Universitat de Lleida. Spain.
sblanch@alumnes.udl.es
[2] Departament de Matemàtica. Universitat de Lleida. Spain.
ramiro@matematica.udl.es

June 15, 2013
github.Papers: 2013-06-15 (revision 36fadc7)

Abstract. [3]

- embedded cryptography
- Ensuring TANGO must be like http*s*. Transparent as possible from the current usage.
-
-

Keywords: Cryptography, Elliptic Curves, Distributed Systems, SCADA, Controls system, Synchrotron

1 Introduction

- What is TANGO ?
- What is the meaning of a secure system? What is security in a distributed system?
- Security threads, policies and mechanisms.
- TANGO as a Supervisory Control And Data Acquisition (SCADA) and/or Industrial Control System (ICS)
- Distributed systems transparencies [1] that TANGO complains, and which are not
- Go further that the Locking/Access control
- Why to secure it? Trust in a peripheral firewalls is not enough. Often communications between tango installations (different tango-db) requires firewall rules to allow it, but this doesn't allow to filter by agent or by who is allowed to access the information.
- Embedded in instrumentation, limited calculation capacity (it must behave indistinguishable if it's a huge server or an embedded board), limited bandwidth (Don't increase the current needs significantly): *very good candidate for elliptic curves, generalized Rijndael and stream cipher.*
- The price of the information and the balance between the cost to ensure and the value of the ensured goods. Security levels: Open, confidential, Secret, Top Secret. (remember the German standard on this levelling).
-
-

[3] Partially supported by grants MTM2010-21580-C02-01 (Spanish Ministerio de Ciencia e Innovación), 2009SGR-442 (Generalitat de Catalunya).

2 Identifying scenarios

- Confidentiality (encryption and authentication): information must be disclosed only *to* the authorized and only *by* the authorized),
- Integrity (authorization): only authorized can set information.
- Auditory: trace who access where (extremely useful for a security breach analysis).
- In terms of security threads, which is more representative from [2] for the current use case? Hospital, Bank, Military Base. Practical paranoia [3]
- Key distribution protocols [2] sec.3.7.2
-
-

2.1 Ensuring presentation layer

- Agent authentication in a distributed system
- Ensuring communication between agents and between those agents with the user interfaces. *Command*, *read* and *write* operations; *Properties* modifications and changes application. This can be compared with *RFID* communication between card and readers, but adding communication in between the agents
- Atk / Taurus user authentication using PAM system (or equivalent in non unix-like systems). Any other user interface that can access tango.
-
-

2.2 Ensuring domain layer

- Trusted Computing and Hardware protections
- multicast, events and the other features that must be secured. Perhaps secret sharing? Secret splitting?
- Ensure logging system
-

2.3 Ensuring data layer

- Tango database access control
- Ensuring between instrumentation and the agents out of the scope of this paper
- Homomorphic Encryption for Database access
-
-

3 Brainstorming attacks

-
-

3.1 Passive attacks

- Eavesdropping (Passive attacks) and Men-in-the-middle (active attacks) between agents.
- Noise to block an alarm transmission
-
-

3.2 Active attacks

- Interruption: Break the public face, web site or gui. Kill a vital agent.
- Modification/Fabrication: Supplant agents.
-
-

3.3 Side channel attacks

-
-

4 Attacks countermeasures

-
-

4.1 Intrusion Detection

- Detection and recovery
-
-

5 Communication hybrid schema

- Pubkey to agreed a season key as the usual hybrid systems
- Use the Symmetric key to seed a shared PseudoRandomGenerator as a key for a stream cipher of transmitted data and listened data between talkers
- *PseudoRandomGenerator* (PRG), can be use the KeyDerivationFunction (KDF) of the Rijndael or better other possible alternatives
-
-

5.1 Elliptic curves for public key

- Set institution set of curves with different sizes for different level of secrecy (or even different curves for a separable sets in the same secrecy level). Isogeny volcanoes [4].
- Capability to reset a curve setup on any of those secrecy levels
-
-

5.2 Rijndael generalization for symmetric key

- How to decide the good parameters of Rijndael? (#rounds,#rows,#columns,wordsize of the block and the key) [5]
- Current AES has advantage on 32bit processor implementation, what about 64bits
- AESWrap [6]
-
-

5.3 Key Derivation Functions for stream ciphering

-
-

6 Zero-knowledge proof for authentication

- The agents in the distributed system must be authenticated to be sure that they hasn't been supplanted
-
-

7 Protocols

- protocol layers [7]
- Trust ring vs. trust tree (institution CA until the leaves)
-
-

8 Environmental IT Security

- The weakest brick: secure the transmission but store in a plain file system
- Human behaviour and psychology.
-
-

9 Conclusions

-
-

References

1. A. S. Tanenbaum and M. van Steen, *Distributed systems, Principles and Paradigms.* Prentice Hall, 2002. International Edition.
2. R. J. Anderson, *Security engineering - a guide to building dependable distributed systems (2. ed.).* Wiley, 2008.
3. N. Ferguson and B. Schneier, *Practical Cryptography.* New York, NY, USA: John Wiley & Sons, Inc., 2003.
4. S. Blanch and R. Moreno, "Security risk associated with multiple users sharing the same elliptic curve."
5. S. Blanch and R. Moreno, "Generalised rijndael."
6. J. Schaad and R. Housley, "Advanced Encryption Standard (AES) Key Wrap Algorithm." RFC 3394 (Informational), Sept. 2002.
7. B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C.* New York, NY, USA: John Wiley & Sons, Inc., 2nd ed., 1995.