# Securing Tango Control System: A brain storming

Sergi Blanch-Torné[1], Josep Maria Miret Biosca[2], Ramiro Moreno Chiral[2], Francesc Sebé Feixa[2]

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

October 2, 2013
github.Papers: 2013-10-02 (revision f9870bb2)

**Abstract.** [3]
Current use of Tango is mostly in Synchrotron and recently extending it into a neutron source, but industry has expressed a desire to participate in the community. This industry desire has been made with concern on security. Not a concern in IT environmental, that is institution/user choose, it was about the use of cryptology to mathematically protect the system.
The goal of ensure Tango must produce an outcome as similar as the *TLS* is for the web navigation. Must be possible to co-live with non secured access, but with a tendency to a complete transparent ensuring. Perhaps the migration process would be not as fast as we could want, specially due to the introduction of the certificates infrastructure, but as the Tango installations are contained in the institutions, and upgrade in this way would be like any other upgrade.
Also as web navigation did, Tango is used with instances running over different architectures and operating systems, from small embedded devices, up to very big computers. Then the objective in this ensuring process is that must work just as for the larger than for the tiny.
It is very important goal to have the Tango implementation as Free Software, as this paper cryptography outcomes must be to have public access with auditable algorithms and sources.
**Keywords:** Cryptography[4], Distributed Control Systems, Cryptography engineering.

## 1 Introduction

- Definition of an *Industrial Control System* (ICS). And definition of what is included under this definition, namely: *Programmable Logic Controllers* (PLC), *Supervisory Control And Data Acquisition* (SCADA) and *Distributed Control System* (DCS).
- Define *Distributed system* and *Middleware* [1].
- Tango definition as a DCS. Tango as a PLC wrapper. Describe the Tango Consortium and its (9) members. Remark its *Free software* licensing and repositories in *sourceforge*. Tango-core (the middlelayer) is LGPLv3 and the agents in the distributed system in the Tango-DeviceServer repository are GPLv3.
  - Define DeviceServer
  - Define DeviceClass
  - Define Device
- auxiliary pieces that with Tango extends their uses: Atk, Sardana and Taurus to build SCADAs. Mambo as data archiver.
- Tango as a middleware: Corba (*omniORB* 4.1 [2]) for the synchronous and asynchronous, and ∅mq (3.2 [3]) for the event based communication.
- View Tango middleware as the basic brick to have *transportation* layer (from the OSI schema) in the DCS. The *session* layer is almost uncovered with shiny skills in Atk, Sardana and Taurus.
- FIXME: "*Is the security within* Atk, Sardana, Taurus *and* Mambo, *included in this paper?*"
- TODO: "*Other complementary tools like* Lima*?*"

[3] Partially supported by grants MTM2010-21580-C02-01 (Spanish Ministerio de Ciencia e Innovación), 2009SGR-442 (Generalitat de Catalunya).
[4] This big keyword includes proposals over *Public key*, *Elliptic Curves*, *Symmetric algorithms*, *stream cyphers*, *secret sharing* and also *Homomorphic encryption* for databases.

- Last TANGO-meeting (the 27th) (aerospace) industry (Onera), beyond the scientific research currently using TANGO, request to improve the security skills of TANGO.
- There is a broad spectrum of potential stakeholders with a huge request on security, in terms of cryptography, like energy generation and delivery, home automation, car manufacturers, and many more.
  - TANGO needs the 's' as happened to https, stmps, imaps, telnet (ssh),...
- The competitors of TANGO, neither have much security skills.
-
-

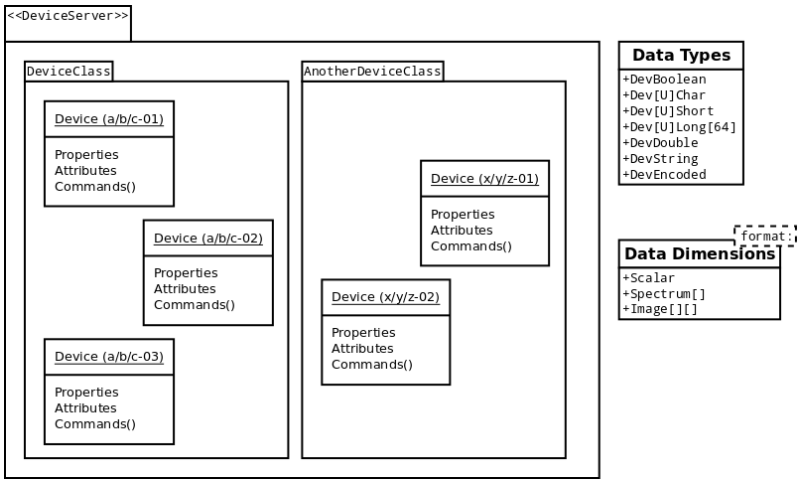


**Fig. 1.** Schematic view about the definition of the term DeviceServer, DeviceClass and Device, and Data information. TODO: "*This image must be improved*"

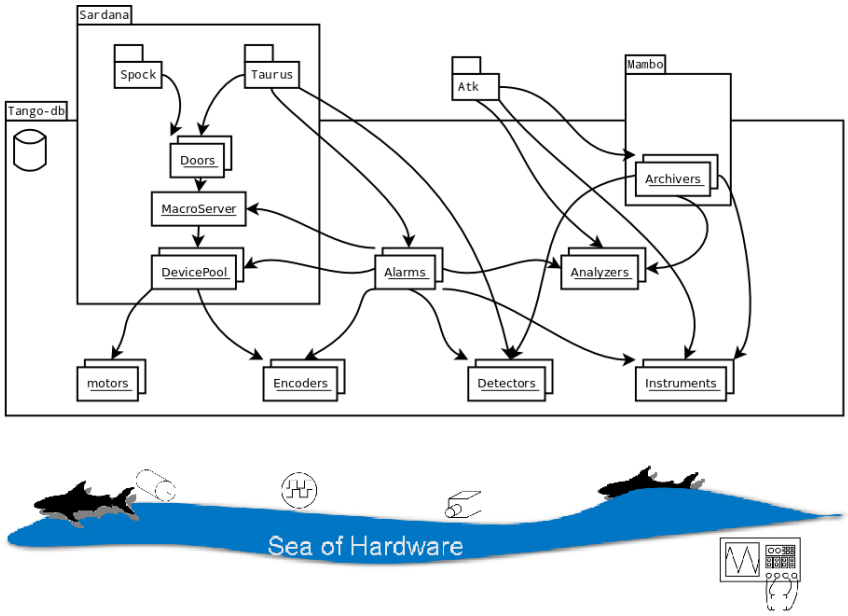


**Fig. 2.** Tango schematic layout TODO: "*This image must be improved significantly*"

### 1.1 Defining distributed system security

- What is the meaning of a secure system? What is security in a distributed system?

- The basics on *information security*
  - Confidentiality: Information must be disclosed only to the authorized.
  - integrity: Only authorized can set in the system.
  - Availability: Information must be accessible for those who are authorized.
  - Authenticity: Information must only be emitted by the authorized.
  - Non-repudiation: Forbid validity changes on the information emitters.
- Add another brick
  - Auditory: trace who access where (extremely useful for a security breach analysis).
-
-

### 1.2 Structure of the paper

- The structure of this paper starts with the definitions in the introduction (section 1)
- This is followed by a section to identify scenarios (section 2) that has a subsection with practical examples of the scenarios (2.1) and the definition of a security thread (section 2.2) over the identified scenarios.
- Next is to do a Brainstorming of the possible attacks (section 3) starting from the fundamental environment security (section 3.1) followed by the passive and active attacks (sections 3.2 and 3.3), non forgetting the big thread that means implementation issues known as side channel attacks (section 3.4). All attack study must contain countermeasures that are in section 3.5 with special emphasis on intrusion detection in section 3.5.1.
- Central part of the paper is the section 4 named "Cryptography Engineering".
  - First part with the application layering view from [1] in section 4.1 (presentation layer 4.1.1, domain layer 4.1.2 and data layer 4.1.3)
  - Followed by the proposed solutions in section 4.2 that are split in:
    - Authentication 4.2.1
      - Zero-knowledge proofs 4.2.1.1
    - Encryption 4.2.2
      - Public key (Elliptic curves) 4.2.2.1
        - Secret sharing 4.2.2.1
      - Symmetric ciphers 4.2.2.2
      - Stream ciphers 4.2.2.3
        - Secret splitting 4.2.2.3
    - Ordered cryptography 4.2.3
      -
- Conclusions section (5) with a further work subsection (5.1)
-
-

## 2 Identifying scenarios

- Securing communications between RFID cards and an authorized reader [4] would be not too different than communication between two agents in a distributed system or between and agent and the element in the presentation layer.
- From the view from [1] over the distributed system transparencies:
  - What is implemented in Tango and what is not? And why?
  - Is any of the "nots" necessary to ensure a quality service.
- In terms of security threads, what it's call "*thread modeling*" which is more representative from [5] for the current use case? Gather information also from [6]
  - Three may types: *Hospital*, *Bank*, *Military Base* (from where the security threads usually comes from).
  - Practical paranoia [7]:
  - Identify threads
  - Capability attack scenario
- Cryptosystem configuration, security levels and information classification. Section 2.2.1. Can be saw as the nowadays number of rotors from the times of the electro-mechanical machines of last century.

| | |
|---|---|
| Access | Hide differences in data representation and how a resource is accessed |
| Location | Hide where a resource is located |
| Migration | Hide that a resource may move to another location |
| Relocation | Hide that a resource may be moved to another location while in use |
| Replication | Hide that a resource is replicated |
| Concurrency | Hide that a resource may be shared by several competitive users |
| Failure | Hide a faulure and recovery of a resource |
| Persistence | Hide whether a (software) resource is in memory or on disk |

**Table 1.** Distributed systems transparencies from [1]

- Setup & Public-Key distribution protocols [5] sec.3.7.2
- Cryptosystem setup reset.
- Secret Shared schemas for (k,n)-to decrypt or (k,n)-signants. Section 4.2.2.1.
- multicast and events (ØMQ) can be scenarios of secret splitting. Section 4.2.2.3.
-
-

### 2.1 Practical examples

- The example of a laboratory use: Optics Lab *Nanometer Optical Measuring-Long Term Profile* (NOM-LTP) for mirrors surface characterization in the angströms ($1Å = 1 \times 10^{-10}m$) range.
  - 3 Hosts
  - 12 DServers
  - 19 DClasses
  - 28 Devices
- The example of a beamline use.
  - 7 Hosts
  - 34 DServers
  - 82 DClasses
  - 615 Devices
- The example of an accelerator use with all the subsystems working together.
  - 139 Hosts
  - 1426 DServers
  - 1551 DClasses
  - 4259 Devices
- Factories production lines
- Critical factories
- Traffic lights and tools
- Energy station

### 2.2 Defining security threads

- Security threads, policies and mechanisms. Section 2. 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.
  - In practice, what is filtered is an specific computer traffic, but this breaks many of the distributed system transparencies (section 2).
  - The example of the Beamlines (read) access to (a few but crucial) accelerator information is a great example of what means a security thread.
  - The industrial example of "do it fast" or "finish it now" more often than thought hides an insecure system or even worst a "bugged" system.

#### 2.2.1 Security levels

- Security levels: Open or unclassified, confidential, Secret, Top Secret. But an institution would like to define more than 4.
- Remember the TODO: "*German standard*" on this levelling, the European commission "*fiche 17*" [8]("Exchange of EU classified information"), FIPS 140-2 [9], Secure Sharing Suite S.3 TODO: "*document not yet found []*".
- Same security level would require an isolated environments. That is even if two subsystems have the same level, would be necessary to isolate threads between them.
-
-

## 3 Brainstorming attacks

- This has relation with the scenarios identified in section 2. Specially what concerns the security thread types [5].
- How mush work it takes to break the system? What's the value of the protected system?
-
-

### 3.1 Environmental IT Security

- The weakest brick: secure the transmission but store in a plain file system
- Human behaviour and psychology.
- ISO/IEC 27000-series
-
-

### 3.2 Passive attacks

- Eavesdropping
-
-

### 3.3 Active attacks

- Men-in-the-middle (active attacks) between agents
- Spoofing: mask and falsify data
- Noise-Interruption-Poisoning: Break the public face, web site or gui. Kill a vital agent.
- Modification/Fabrication: Supplant agents.
-
-

### 3.4 Side channel attacks

-
-

### 3.5 Attacks countermeasures

-
-

#### 3.5.1 Intrusion Detection

- Detection and recovery
-
-

# 4 Cryptography Engineering

-
-

## 4.1 Distributed system layering approach

-
-

### 4.1.1 Ensuring presentation layer

- Agent authentication in a distributed system.
  - Not very different than RFID systems, it has similarities and there is a German standard [10] for travel documents (passports).
- Ensuring communication between agents and between those agents with the user interfaces (ATK and TAURUS).
  - *Command*, *Attribute*, *Properties*: Authenticate who can do the *read* and *write* operations. Encrypted logging who did any change, with levels to grant access levelling.
- Deal with multicast can event subscription and emission.
-
-

### 4.1.2 Ensuring domain layer

- Trusted Computing and Hardware protections: is an agent allowed to run on this specific machine? (what about transparencies)
- Ensure logging system
-
-

### 4.1.3 Ensuring data layer

- TANGO data base as a centralized "phone guide".
- TANGO database access control
- Ensuring between instrumentation and the agents out of the scope of this paper, often is also out of the device server developer hands. This is a very dependant on the instrumentation manufacturers. From the iso layer level view, even if the access to the hardware is not networked, the agent communication to the instrumentation is *data link layer* and this paper is focus in *transport* and *session* layers.
- Homomorphic Encryption for Database access
-
-

## 4.2 Proposed solutions

### 4.2.1 Authentication

- ECDSA [11]
- SHA [12]
-
-

#### *4.2.1.1 Zero-knowledge proof for authentication*

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

### 4.2.2 Encryption

- 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 (section 4.2.2.1), generalized Rijndael (section 4.2.2.2) and stream cipher (section 4.2.2.3).*
- Public-key to agreed a season key as the usual hybrid systems. This session keys shall be used for symmetric or stream cyphering.
- Session keys refresh.
- 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
- RFCs: 6239 [13], 5647 [14]
-

#### *4.2.2.1 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 [15]. Together with the contribution work of [16], [17], [18].
- Capability to reset a curve setup on any of those secrecy levels (section 2.2.1) and the feature to have different setups between different subsystems to have separated environments between them.
- Standards about elliptic curves; International [19], [**?**], [20], [21], USA: [22], [23], German: [24],[25], Russian: [26]
-
-

*Secret Sharing*

- To allow some one access to some specific data, perhaps it can require the "grant" from more than one agent of the distributed system. That is, to give it the key may (k,n) must act to.
- Authorization units may be bigger than one agent. A (k,n)-signature to have only one to verify for all.
-
-

#### *4.2.2.2 Rijndael generalization for symmetric key*

- AES contest [27] and the book [28]
- How to decide the good parameters of Rijndael? (#rounds,#rows,#columns,wordsize of the block and the key) [29]
- Current AES has advantage on 32bit processor implementation, what about 64bits
- AESWrap [30]
- Secrecy levels (section 2.2.1)
-

#### *4.2.2.3 Stream ciphering*

- Key Derivation Functions?
- Rabbit (rfc4503)
- VEST
- Chacha20
-
-

*Secret Splitting*

- Multicast and event system. When a event is emitted, many would be subscribed, but encryption must be only made once.
-
-

### 4.2.3 Ordered cryptography

-
-

#### *4.2.3.1 Homomorphic Encryption*

- Introduce the meaning of the private database query system [31]
- TODO: "*Search for references from* Josep Domingo Ferrer *(from the Rovira i Virgili, PhD director of F.Sebé*"
-
-

## 5 Conclusions

- All those fields mention on this paper requires a much further detailed paper each.
-
-

### 5.1 Further work

- ATK/ TAURUS user authentication using PAM system (or equivalent in non unix-like systems). Any other user interface that can access tango.
- In all the algorithms on this paper this must be taken into account to minimize redesigns.
-
-

## References


1. A. S. Tanenbaum and M. van Steen, *Distributed systems, Principles and Paradigms*. Prentice Hall, 2002. International Edition.
2. D. Grisby, July 2009.
3. P. Hintjens, February 2013.
4. S. Martínez, *Protocolos de seguridad para sistemas de indentificación por radiofrecuencia*. PhD thesis, Universitat de Lleida, march 2011. Directed by: Concepció Roig and Magda Valls.
5. R. J. Anderson, *Security engineering - a guide to building dependable distributed systems (2. ed.)*. Wiley, 2008.
6. N. Ferguson, B. Schneier, and T. Kohno, *Cryptography Engineering: Design, principles and practical applications*. Wiley, 2010.
7. N. Ferguson and B. Schneier, *Practical Cryptography*. New York, NY, USA: John Wiley & Sons, Inc., 2003.
8. "Exchange of eu classified information," 2003.
9. "Fips pub 140-2, security requirements for cryptographic modules," 2002. U.S.Department of Commerce/National Institute of Standards and Technology.
10. "Bsi tr-03110: Advanced security mechanisms for machine readable travel documents."
11. P. Gallagher, D. D. Foreword, and C. F. Director, "Fips pub 186-3 federal information processing standards publication digital signature standard (dss)," 2009.
12. "Fips pub 180-2, secure hash standard (shs)," 2002. U.S.Department of Commerce/National Institute of Standards and Technology.
13. K. Igoe, "Suite B Cryptographic Suites for Secure Shell (SSH)." RFC 6239 (Informational), May 2011.
14. K. Igoe and J. Solinas, "AES Galois Counter Mode for the Secure Shell Transport Layer Protocol." RFC 5647 (Informational), Aug. 2009.
15. S. Blanch-Torné, R. Moreno, F. Sebé, and J. Valera, "Security risk associated with multiple users sharing the same elliptic curve." Draft.
16. J. Valera, "Volcales de $\ell$-isogenias de curvas elípticas," *Sistemas Informáticos. Escola Politècnica Superior. Universidad de Lleida*, Sept 2011. Directed by: Josep M. Miret.
17. R. Moreno, *Subgrupos de Sylow de las curva ellíupticas definidas sobre cuerpos finitos*. PhD thesis, Universitat Politècnica de Catalunya, 2005. Directed by: Anna Rio and Josep M. Miret.
18. R. Tomàs, *Volcans d'isogenies de corbes el·líptiques: Aplicacions criptogràfiques en targetes intel·ligents*. PhD thesis, Universitat de Lleida, march 2011. Directed by: Josep M. Miret and Daniel Sadornil.

19. A. Jivsov, "Elliptic Curve Cryptography (ECC) in OpenPGP." RFC 6637 (Proposed Standard), June 2012.
20. "Sec 1. standards for efficient cryptography group: Elliptic curve cryptography."
21. "Sec 2. standards for efficient cryptography group: Recommended elliptic curve domain parameters."
22. "Ieee p1363 standard specifications for public key cryptography," January 2000.
23. "Ansi x9.62, public key cryptography for the financial services industry: The elliptic curve digital signature algorithm (ecdsa)."
24. "Ecc brainpool standard curves and curve generation," October 2005.
25. "Bsi tr-03111: Elliptic curve cryptography, version 2.0."
26. "Information technology. cryptographic data security. signature and verification processes of digital signature.," 2001. Gosudarstvennyi, Standard of Russian Federation, Government Committee of the Russia for Standards.
27. "Specification for the advanced encryption standard (aes)." Federal Information Processin Standards Publication 197, 2001.
28. J. Daemen and V. Rijmen, *The Design of Rijndael*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2002.
29. S. Blanch-Torné, R. Moreno, F. Sebé, and M. Valls, "Generalised rijndael." Draft.
30. J. Schaad and R. Housley, "Advanced Encryption Standard (AES) Key Wrap Algorithm." RFC 3394 (Informational), Sept. 2002.
31. D. B. nad Craig Bentry, S. Halevi, F. Wang, and D. J. Wu, "Private database queries using somewhat homomorphic encryption," *International Association for Cryptologic Research*, June 2013.