Decentralized Public Key Infrastructure for Autonomous Embedded Systems

Arthur Baudet, Oum-El-Kheir Aktouf, Annabelle Mercier, Philippe Elbaz-Vincent

arthur.baudet@lcis.grenoble-inp.fr

C&ESAR — 2022-11-16

This work is supported by the French National Research Agency in the framework of the “Investissements d’avenir” program (ANR-15-IDEX-02). Reproduction prohibited without written permission of the authors.
This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program (ANR-15-IDEX-02).
Reshaping technologies and social sciences

This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program (ANR-15-IDEX-02).

Reproduction prohibited without written permission of the authors.

---

### Autonomous Embedded Systems

**Agent**
- Physical or software entity
- Autonomous (proactive or/and reactive)

**Embedded agent**
- Resources limitations
- Communication limitation
- Mobility

**Multi-Agent Key Infrastructure**
- > 2 agents
- Decentralized
- Global problem divided in smaller problems
- Cooperation between agents
- Open
- Dynamic
- Heterogeneous

---

Russell et al., Wooldridge et al. [1, 2]
Wildfire Monitoring

Drones monitoring a wildfire

Context and Case study
Drones monitoring a wildfire and intruders
State of the Art

What is done to secure MEAS?

Availability
Communication Integrity
State of the Art

What is done to secure MEAS?

- Availability
- Communication Integrity
- Trust Management System
- Cryptography
State of the Art

What is done to secure MEAS?

Availability
Communication Integrity

Trust Management System

Cryptography
State of the Art

What is done to secure MEAS?

- Pre-loaded certificate & key
- Third-party

Availability
Communication Integrity
Trust Management System
Cryptography

Context and Case study 5/19
Arthur Baudet
State of the Art

What is done to secure MEAS?

- Availability
- Communication Integrity
- Trust Management System
- Cryptography

- Pre-loaded certificate & key
- Third-party

Conflict with Autonomy & Heterogeneity
Attacker Model

“Insider attack”: Similar resources as nodes

Control over communication medium: Tampering, replay, etc.
Attacker Model

“Insider attack”: Similar resources as nodes

Control over communication medium: Tampering, replay, etc.

- Secure Communication
- Integrity
- Authenticity
Attacker Model

“Insider attack”: Similar resources as nodes

Control over communication medium: Tampering, replay, etc.

- Secure Communication
- Integrity
- Authenticity
- No Authentication
- Heterogeneity
- Decentralization

Objective
Attacker Model

“Insider attack”: Similar resources as nodes

Control over communication medium: Tampering, replay, etc.

- Secure Communication
- Integrity
- Authenticity
- No Authentication
- Accountability
- Trust Management System

Heterogeneity
Decentralization
Objective

Attacker Model

“Insider attack”: Similar resources as nodes

Control over communication medium: Tampering, replay, etc.

Secure Communication

Heterogeneity
Decentralization

Public Key Encryption

Integrity

Authentication

No Authentication

Accountability

Trust Management System
Multi-Agent Key Infrastructure (MAKI)
Public Key Infrastructure for Multi-Agent System
Multi-Agent Key Infrastructure (MAKI)
Public Key Infrastructure for Multi-Agent System

Main Rule
Messages must be signed with a key linked to a valid certificate.

Hypotheses
1. Standard cryptography is secured
2. Basic routing exists
3. An adequate Trust Management System (TMS) is running
Identity $\leftrightarrow$ Public Key
Reshaping technologies and social sciences

This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program (ANR-15-IDEX-02).

Reproduction prohibited without written permission of the authors.

Organization

Identity $\leftrightarrow$ Public Key

Role: None
- Default
- Require a CA to get a certificate
- Share its certificate

Role: Certification Authority (CA)
- Deliver, store and revoke certificates
- Self-signed or cross-certified
- Share its certificate
Reshaping technologies and social sciences

This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program (ANR-15-IDEX-02).

Reproduction prohibited without written permission of the authors.

Organization

Identity $\leftrightarrow$ Public Key

Role: None
- Default
- Require a CA to get a certificate
- Share its certificate

Role: Certification Authority (CA)
- Deliver, store and revoke certificates
- Self-signed or cross-certified
- Share its certificate

Revocation
- Certificate Revocation List
- Short-lived certificate
Self-Organization

Global Rules

- At least one CA is required
- > 1 CA is advisable to prevent single-point-of-failure situations
- Agents choose their roles
- Any agent can become a CA
Global Rules

- At least one CA is required
- > 1 CA is advisable to prevent single-point-of-failure situations
- Agents choose their roles
- Any agent can become a CA

Role self-assignment flowchart
3 Thresholds
- Low
- Moderate
- High
Trust Management

Interaction Rules

**Certificate Authority**
- Delivering a certificate: Moderate or None
- Revoking a certificate: Moderate
- Requesting a cross-certification: High
- Accepting a cross-certification request: High

**None**
- Requesting a certificate: Moderate or None

3 Thresholds
- Low
- Moderate
- High
Trust Management

Interaction Rules

**Certificate Authority**
- Delivering a certificate: Moderate or None
- Revoking a certificate: Moderate
- Requesting a cross-certification: High
- Accepting a cross-certification request: High

**None**
- Requesting a certificate: Moderate or None

3 Thresholds
- Low
- Moderate
- High

Delivering certificate: \(\uparrow\) trust
Being cross-certified: \(\uparrow\) trust
Ignoring requests: \(\downarrow\) trust
Trust Modelling

\[ f : \mathbb{N} \rightarrow \mathbb{N}_+ \]
\[ x \mapsto \frac{x}{x + 10} \]

- Slow increase
- Fast decrease
Agent architecture

Agent architecture without MAKI
Agent architecture with MAKI
Wildfire Monitoring

Drones monitoring a wildfire executing MAKI
Q1: Does the TMS really benefits from MAKI?

Q2: Does the self-organization leads to the correct organization?
Q1: Does the TMS really benefit from MAKI?

Q2: Does the self-organization lead to the correct organization?

Simulation
Yet Another Multi-Agent Systems Simulator (YAMASS)
- In-house simulator
- Ongoing work
- Based on Mesa [3]
- Allows easy reproducibility
- Enforces agents positioning
Setup

Simulation of a MEAS executing MAKI in YAMASS
Red: CA, Black: None

- 10 agents
  - $\leftrightarrow$ 4 possible CAs: $A_6, A_7, A_8, A_9$
  - $\leftrightarrow$ 3 CAs: $A_6, A_8, A_9$
- All in communication range
- Application is emulated
Scenario I — None agent intrusion

None agent ($A_0$) is malicious

Trust variation without revocation

Trust variation with revocation
Scenario I — None agent intrusion

None agent ($A_0$) is malicious

Q1: Yes
Revocation → more effective exclusion

Trust variation without revocation

Trust variation with revocation

Proof-of-Concept
Scenario II — Malicious CAs coalition

All the CAs ($A_6$, $A_8$, $A_9$) are malicious

State of the system before the detection
Red: CA, Black: None
Scenario II — Malicious CAs coalition

All the CAs ($A_6$, $A_8$, $A_9$) are malicious

<table>
<thead>
<tr>
<th>Step: 701</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent 7: Certificate (Issuer: 7, Subject: 7, ...)</td>
</tr>
<tr>
<td>Agent 0: Certificate Request (Source: 0, Destination: 7)</td>
</tr>
<tr>
<td>Agent 1: Certificate Request (Source: 1, Destination: 7)</td>
</tr>
<tr>
<td>Agent 3: Certificate Request (Source: 3, Destination: 7)</td>
</tr>
<tr>
<td>Agent 2: Certificate Request (Source: 2, Destination: 7)</td>
</tr>
<tr>
<td>Agent 4: Certificate Request (Source: 4, Destination: 7)</td>
</tr>
<tr>
<td>Agent 5: Certificate Request (Source: 5, Destination: 7)</td>
</tr>
</tbody>
</table>

State of the system after the detection
Red: CA, Black: None

Simplified excerpt of the execution trace
Scenario II — Malicious CAs coalition

**All the CAs (A_6, A_8, A_9) are malicious**

- **State of the system after the detection**
  - Red: CA, Black: None

- **System overtaken** → **Self-organization:** A_7 becomes a CA

- **Simplified excerpt of the execution trace**
  - step:701
  - agent:7:<>:CertAdvert(Certificate(issuer: 7, subject: 7, ...))
  - agent:0:=>:CertReq(src: 0, dest: 7)
  - agent:1:=>:CertReq(src: 1, dest: 7)
  - agent:3:=>:CertReq(src: 3, dest: 7)
  - agent:2:=>:CertReq(src: 2, dest: 7)
  - agent:4:=>:CertReq(src: 4, dest: 7)
  - agent:5:=>:CertReq(src: 5, dest: 7)

Q2: Yes

Proof-of-Concept 17/19

Arthur Baudet
Takeaway

- No standard for key management in decentralized autonomous systems
- MAKI: Multi-Agent Key Infrastructure is PKI for decentralized autonomous systems
  → Stronger hypotheses can be taken

- Slow start
- Few strong assurances
Future work

→ Self-Organization validation: Model checking
→ Proof-of-Concept: State of the art trust model
→ Sharing certificates: Blockchain-based solution
Thank you for your attention

Questions?


