Hardware Security Modules Unpacking a Black Box

Christian Colombo

L-Università ta' Malta

Imagine there were No Adversaries…

You could…

…trust everyone to be who they say they are

…just store your data

…communicate freely

…provide services

Imagine there were No Adversaries…

You could…

…trust everyone to be who they say they are

…just store your data

…communicate freely

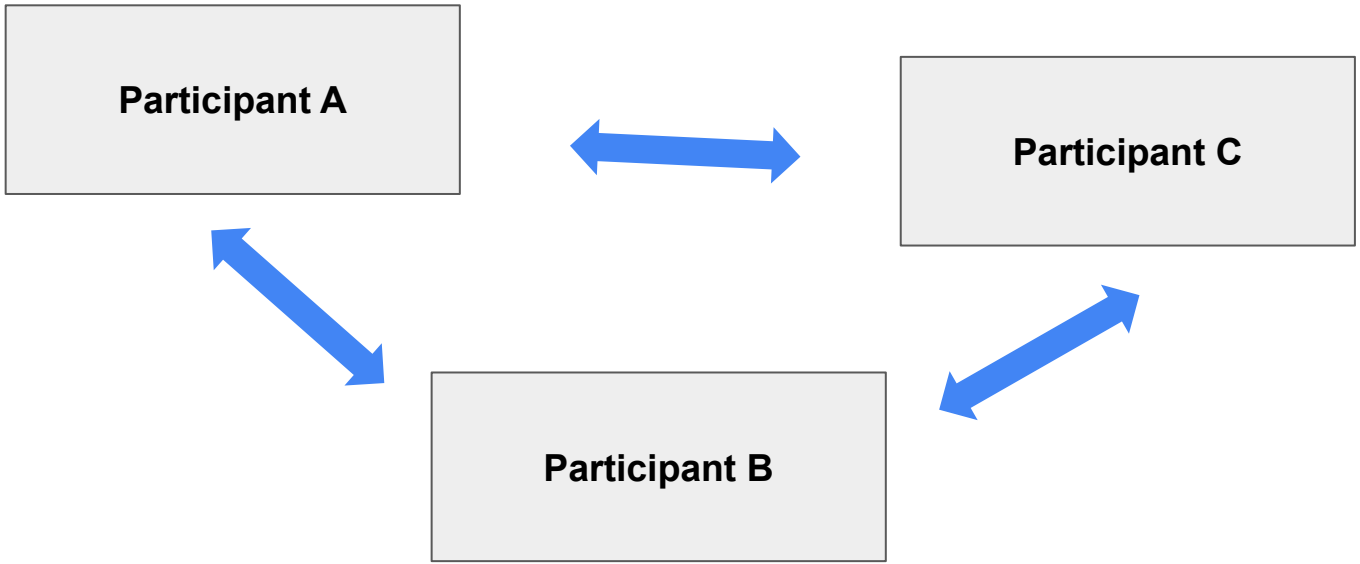
…provide services

Sadly cannot be further from the truth!

Imagine we could avoid interacting…

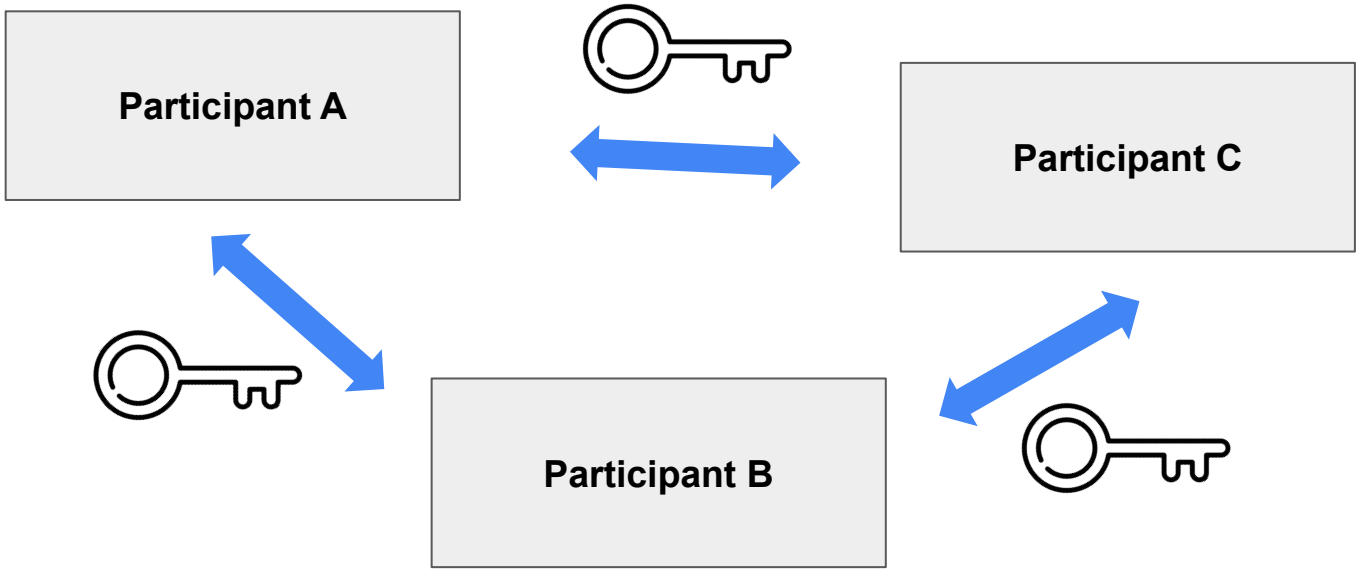
Imagine we could avoid interacting…

… but we do need to interact!
(in the presence of adversaries)

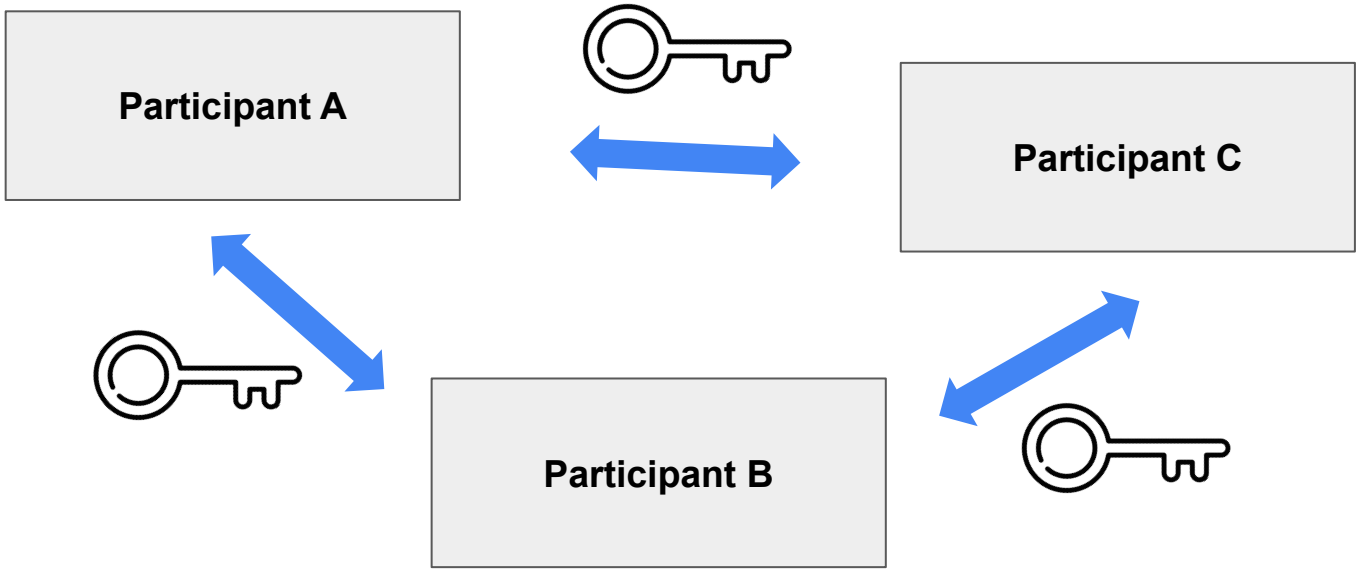


… but we do need to interact!
(in the presence of adversaries)

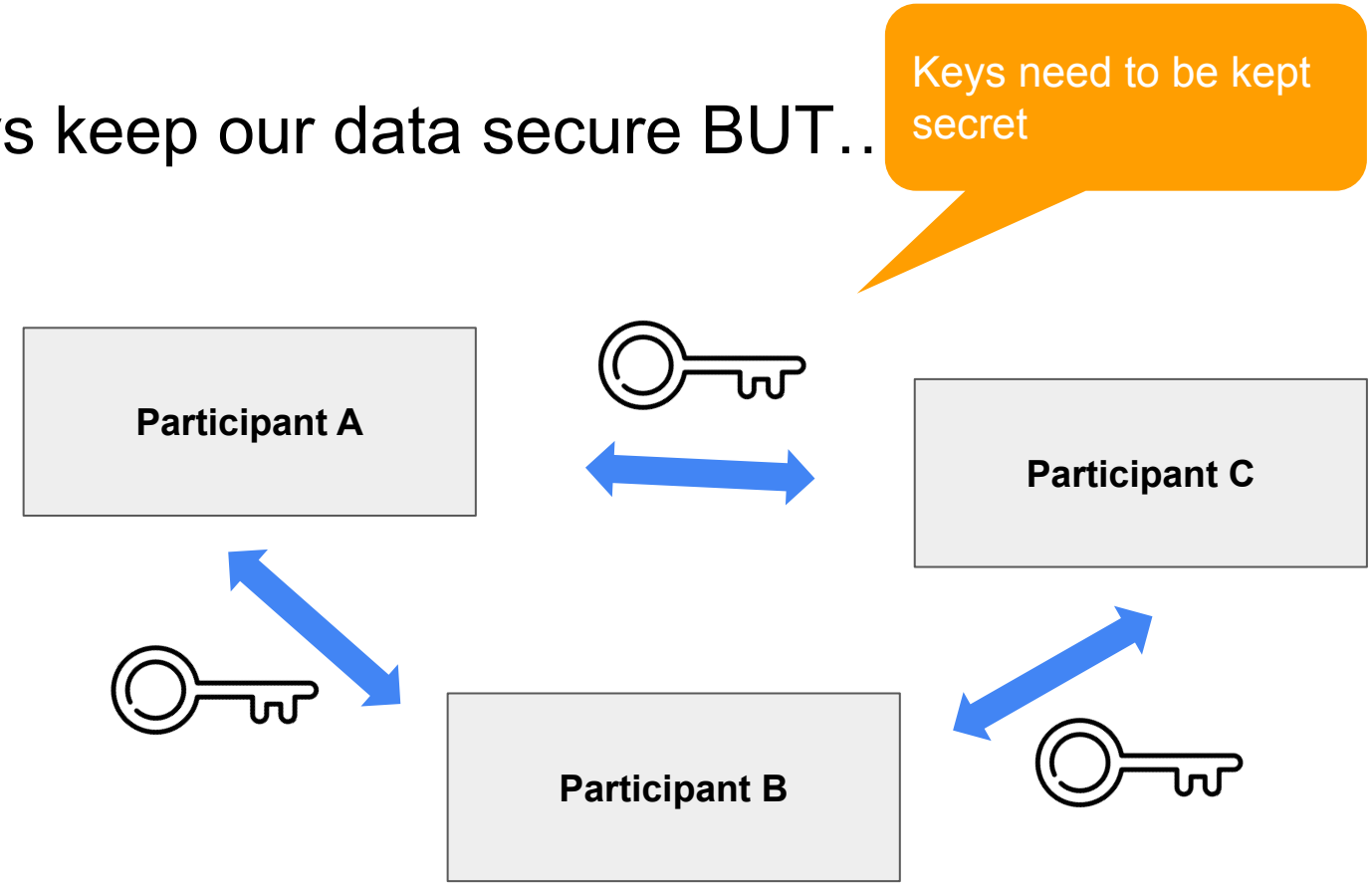
This is why we need cryptography!!



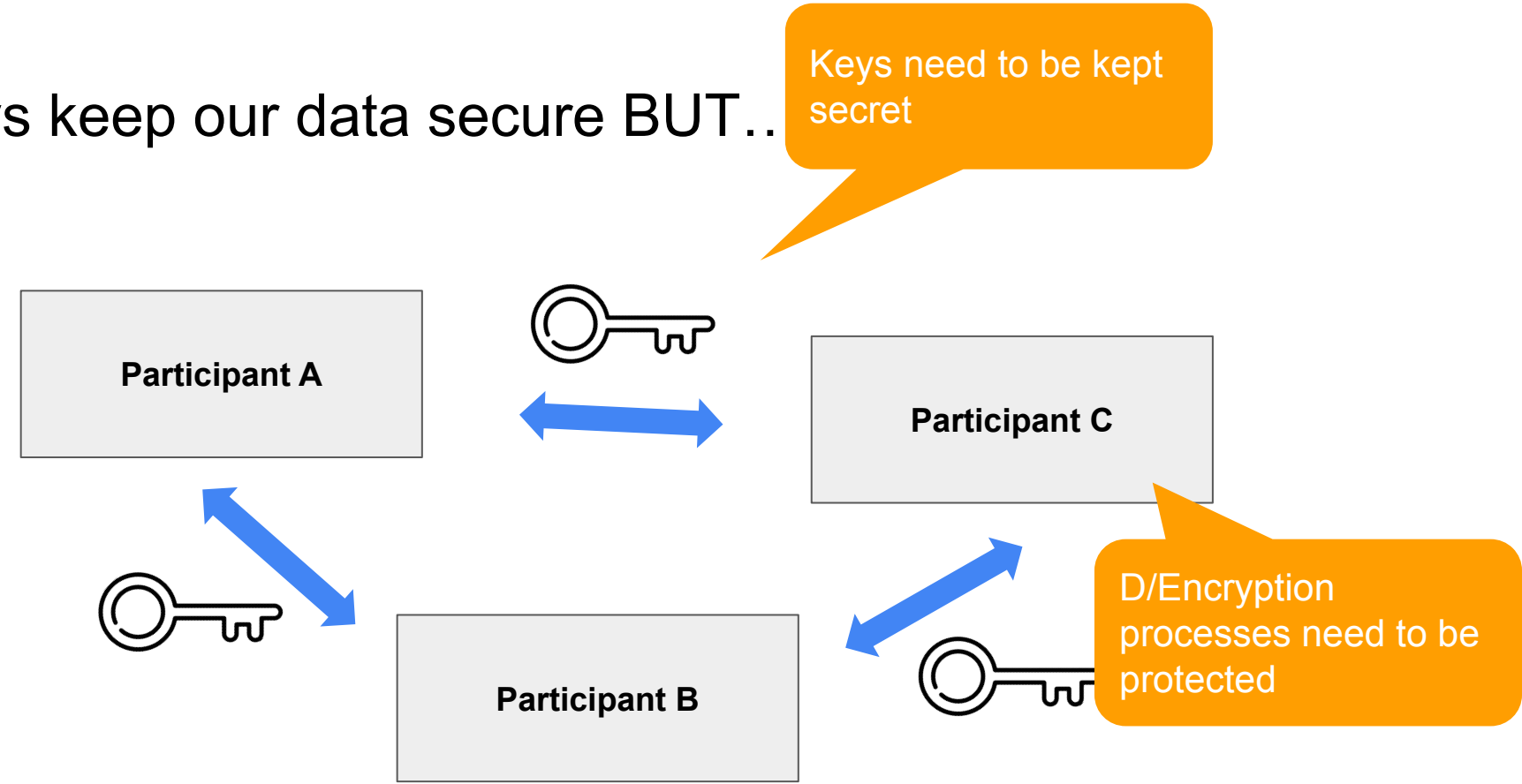
Keys keep our data secure BUT…



Keys keep our data secure BUT..



Keys keep our data secure BUT..



Where can things go wrong?

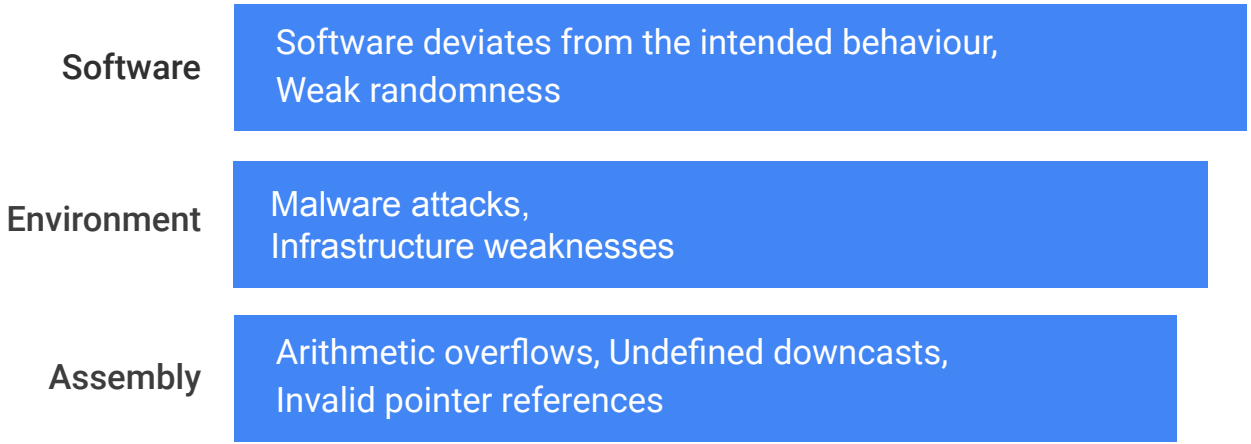
Many things can go wrong on many different levels

Software	Software deviates from the intended behaviour, Weak randomness

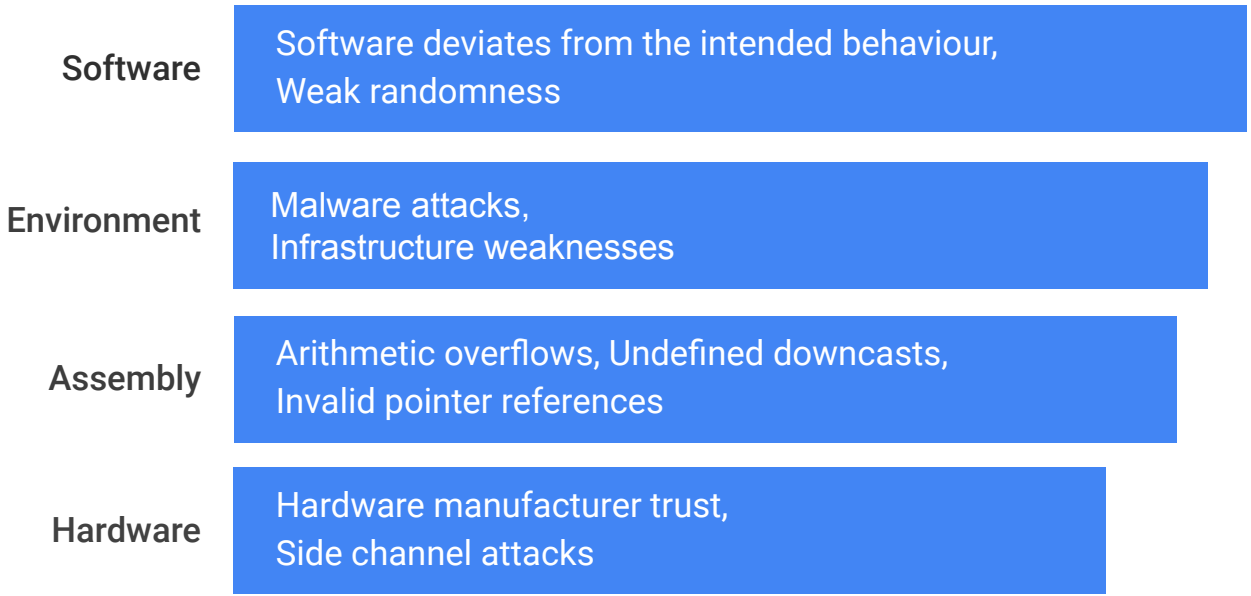
Many things can go wrong on many different levels

Software	Software deviates from the intended behaviour, Weak randomness
Environment	Malware attacks, Infrastructure weaknesses

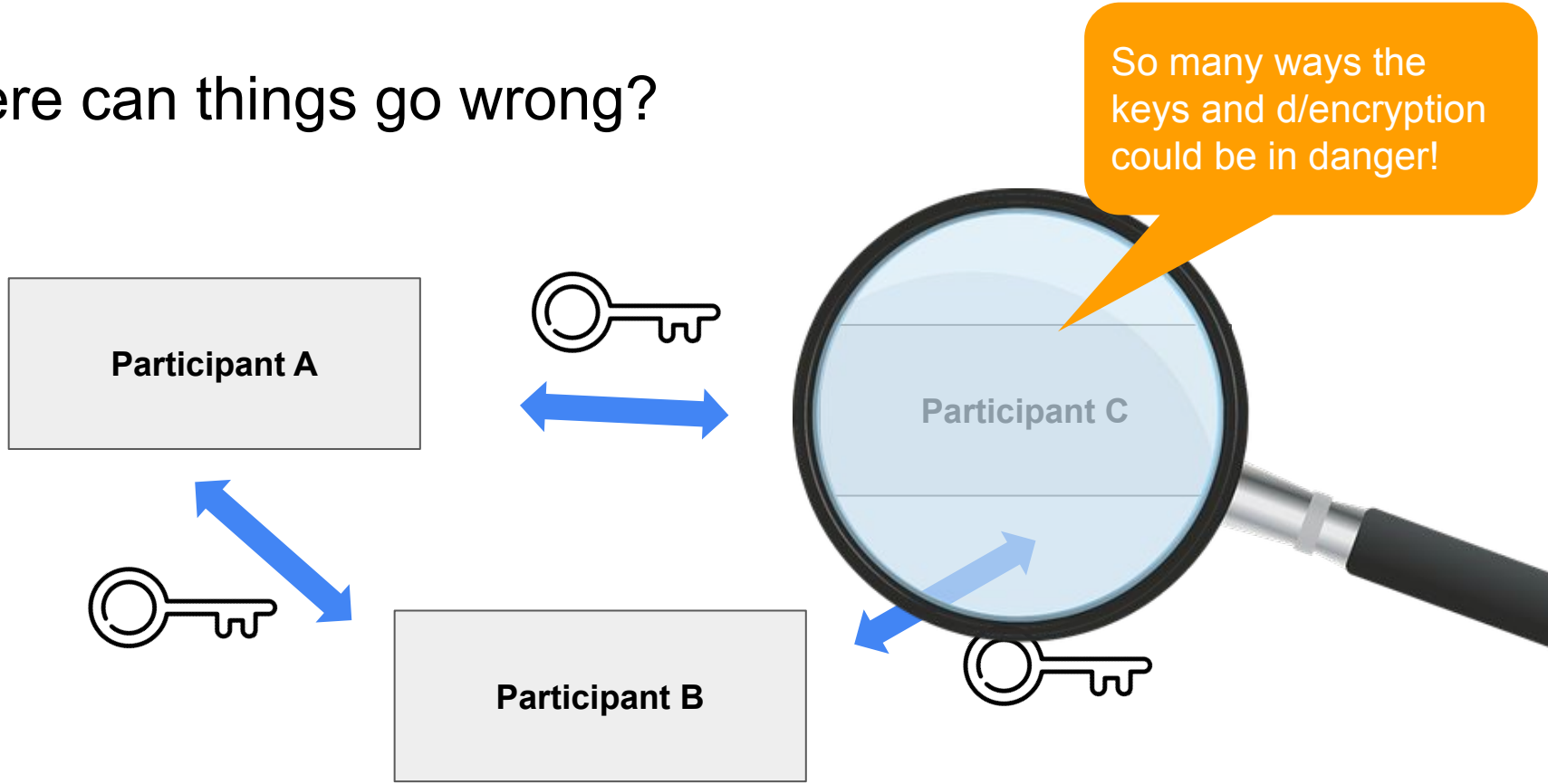
Many things can go wrong on many different levels



Many things can go wrong on many different levels



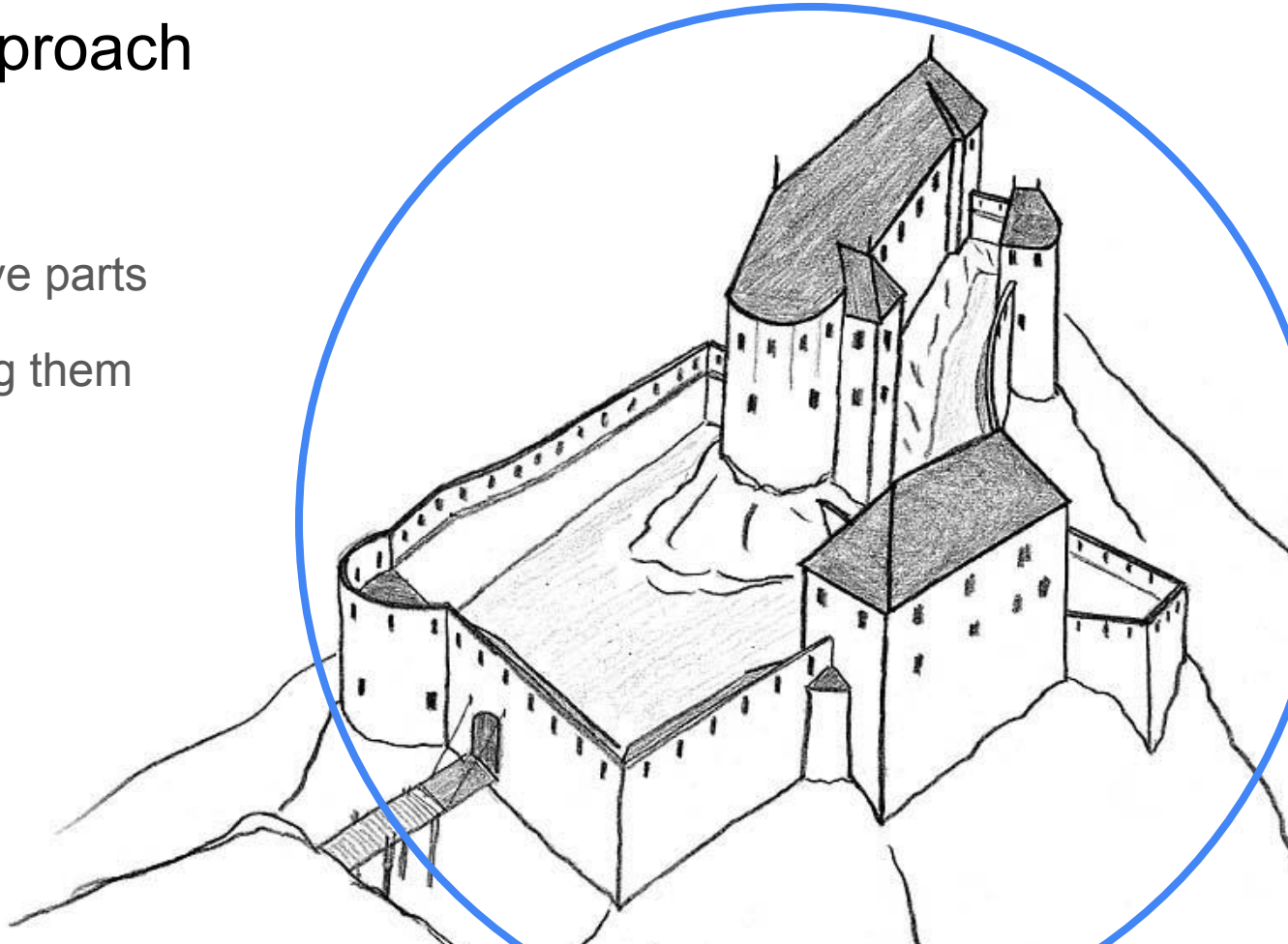
Where can things go wrong?



A Risk-Based Approach

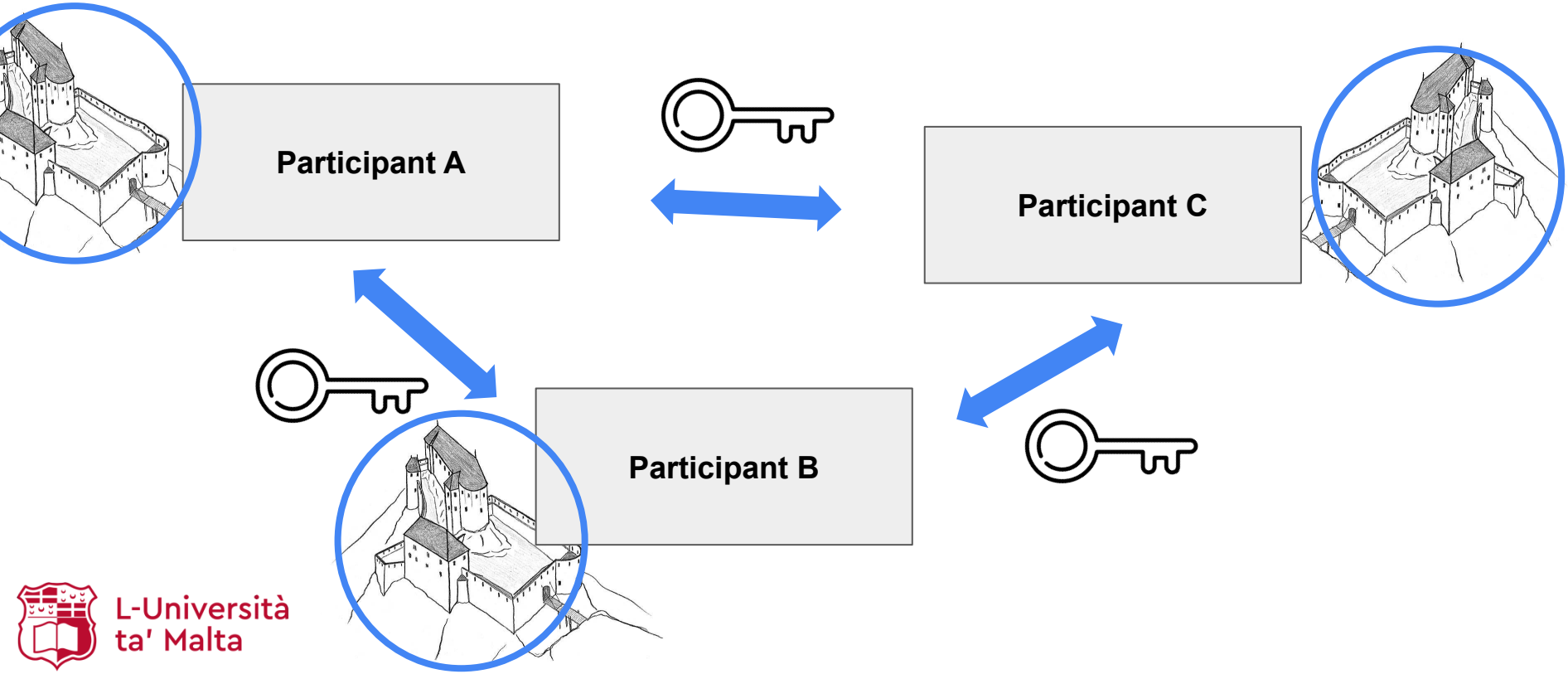
Isolate the most sensitive parts

Spend more on securing them



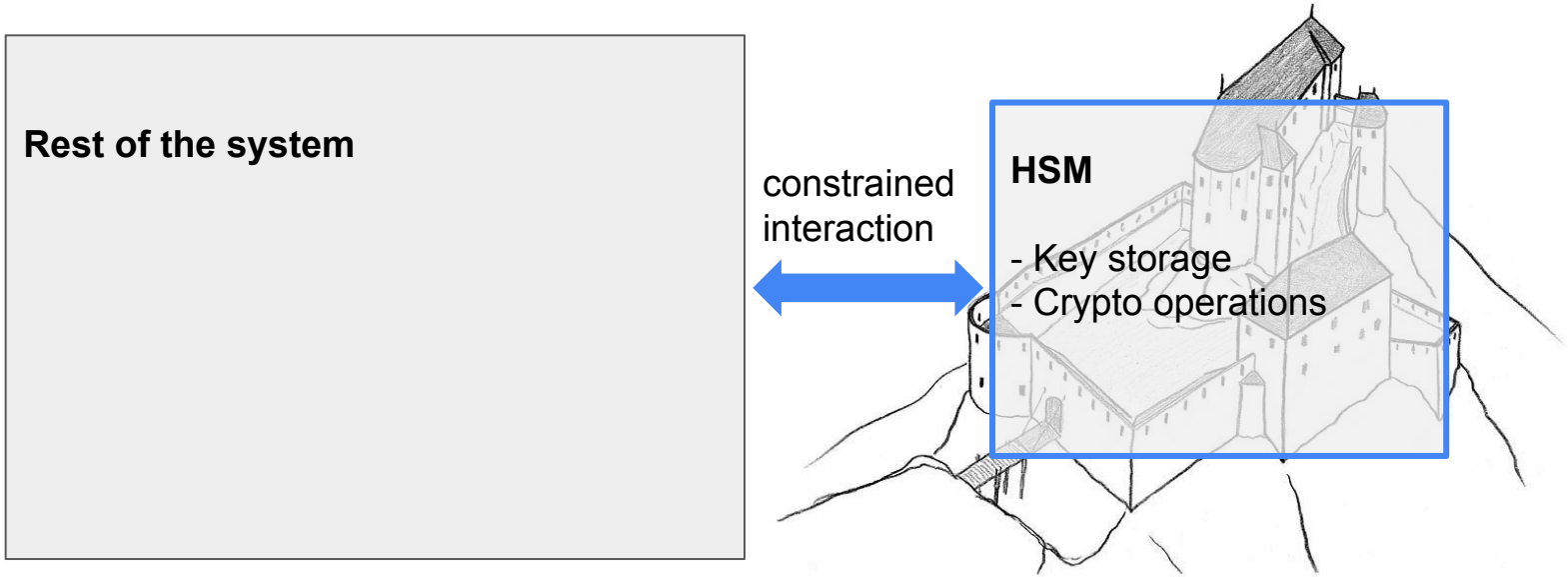
L-Università ta' Malta

Isolating the most sensitive parts

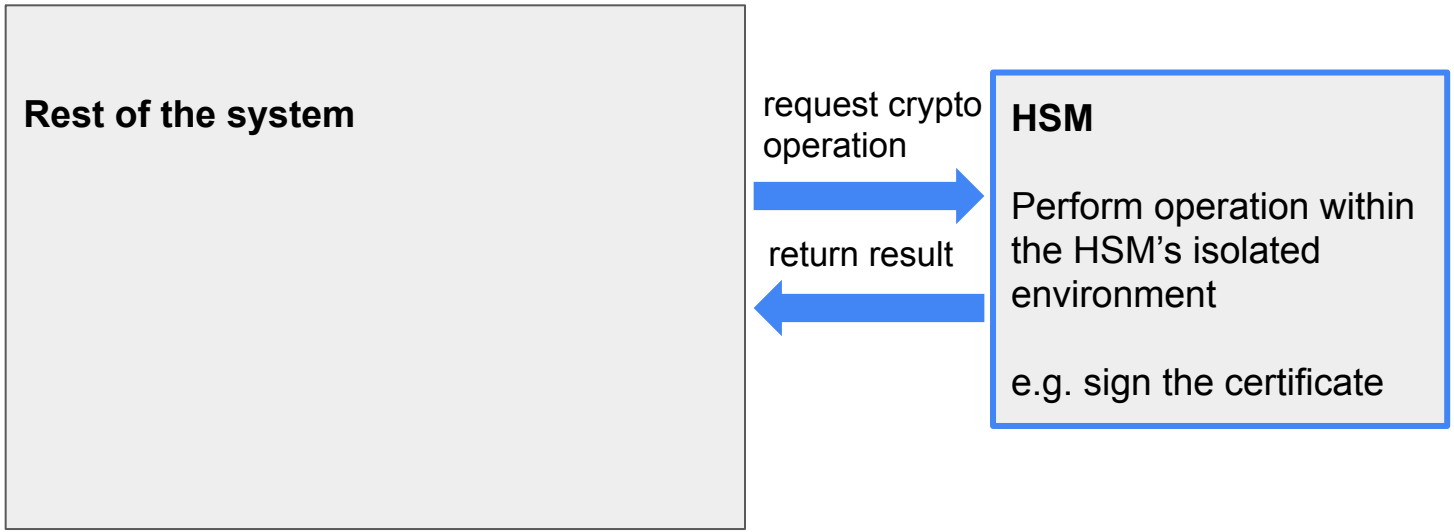


L-Università ta' Malta

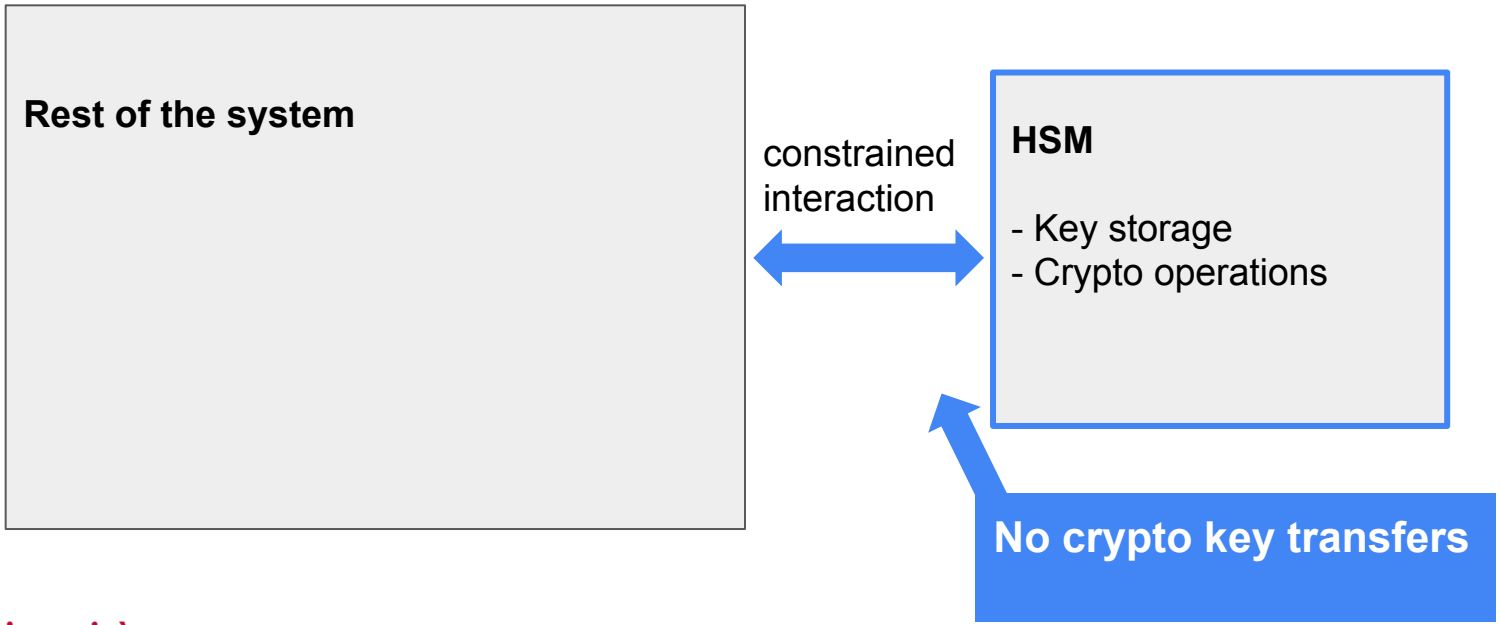
Hardware Security Module (HSM) to the Rescue



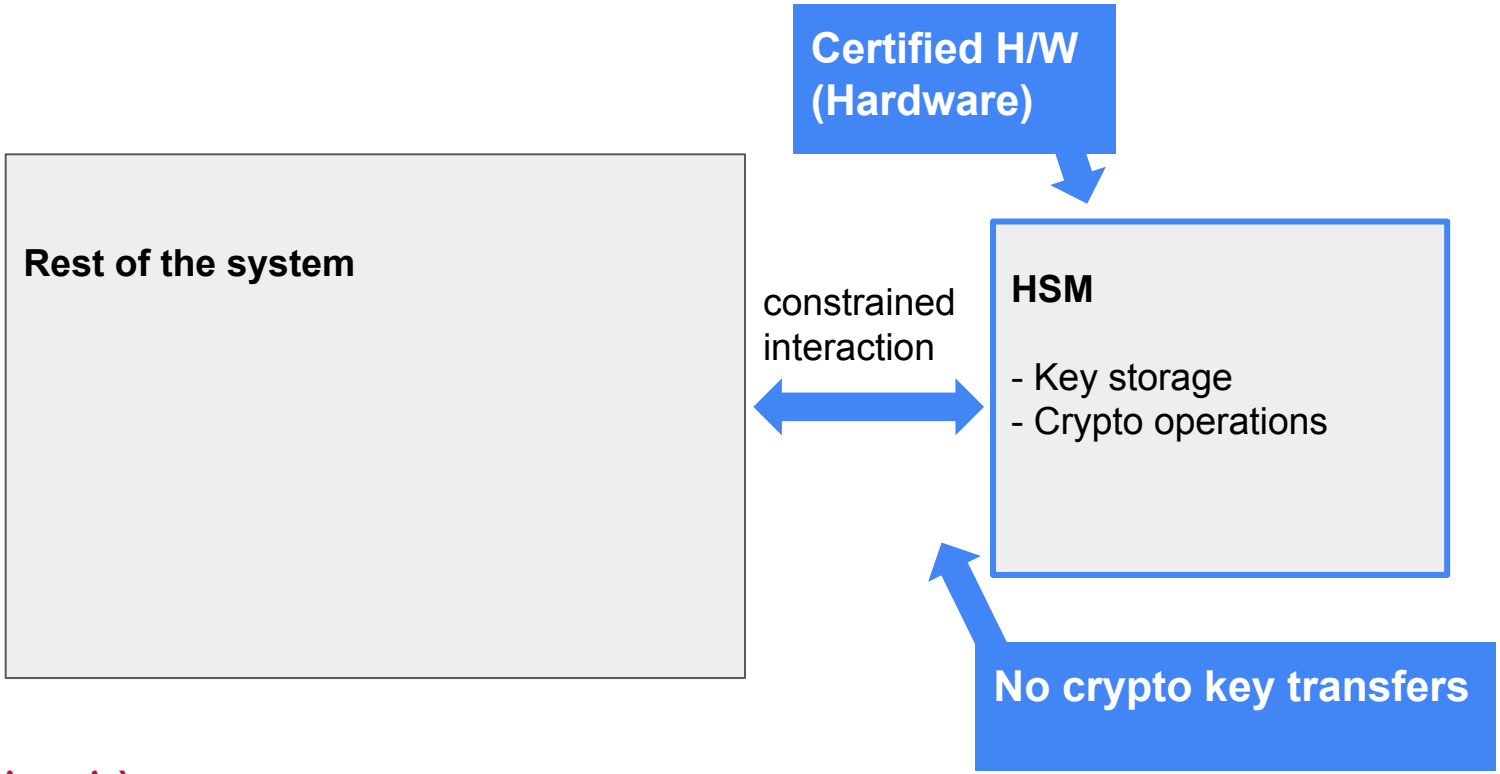
Example - Signing a Digital Certificate



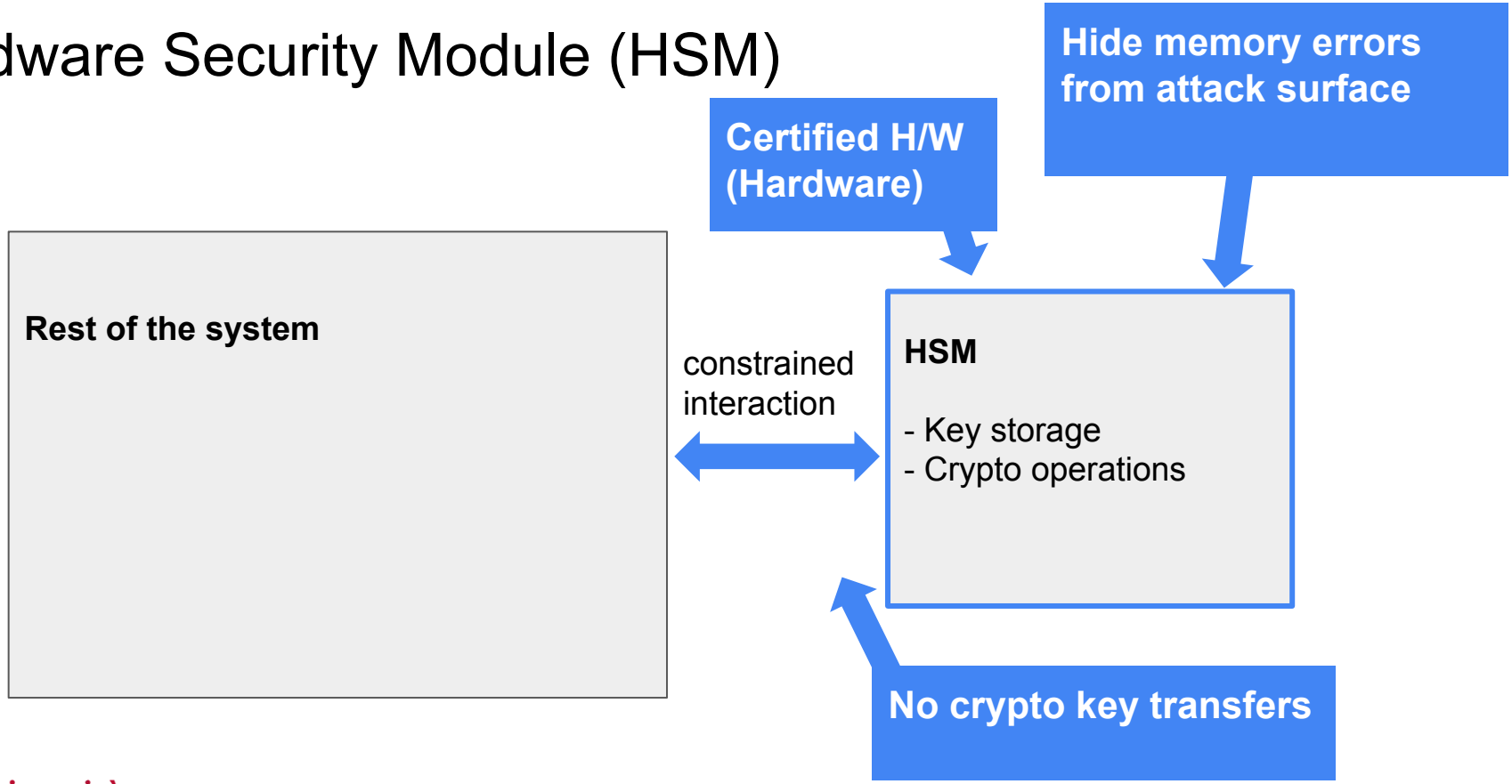
Hardware Security Module (HSM)



Hardware Security Module (HSM)



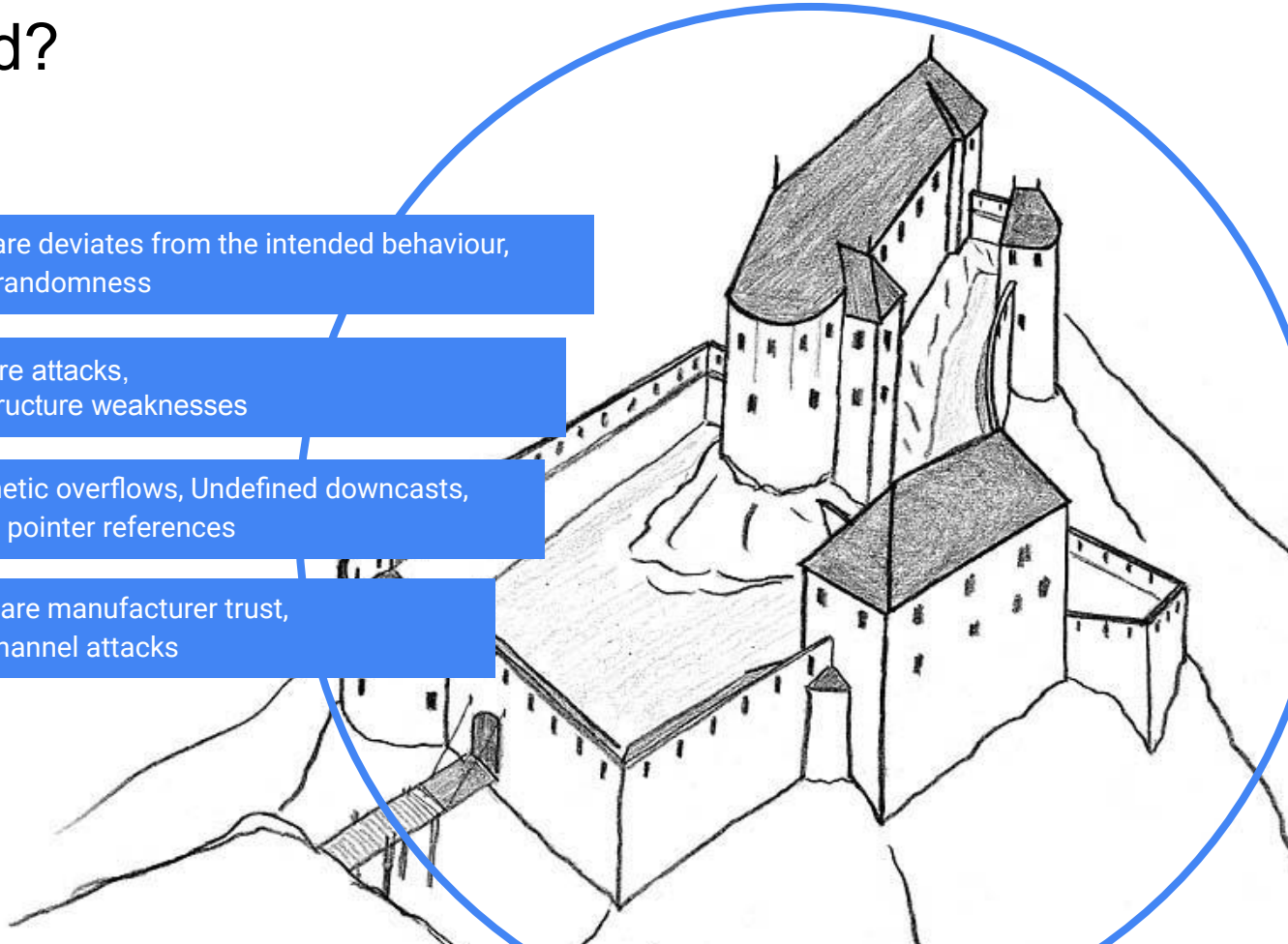
Hardware Security Module (HSM)



What is Protected?

Limit attack surface

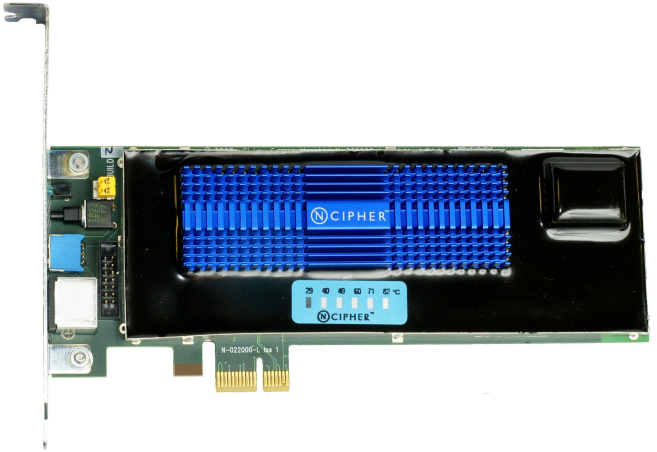
Software	Software deviates from the intended behaviour, Weak randomness
Environment	Malware attacks, Infrastructure weaknesses
Assembly	Arithmetic overflows, Undefined downcasts, Invalid pointer references
Hardware	Hardware manufacturer trust, Side channel attacks



L-Università ta' Malta

What is an HSM?

What is an HSM?



L-Università ta' Malta

What is Special about an HSM?

Tamper resistance + evidence

Certified CC EAL 4+ / FIPS 140-3 (3+)

Hardware-based entropy generator

Hardware-accelerated crypto operations

HSM raises alerts / becomes inoperable / deletes keys upon tamper detection

What is Special about an HSM?

Tamper resistance + evidence

Certified CC EAL 4+ / FIPS 140-3 (3+)

Hardware-based entropy generator

Hardware-accelerated crypto operations

Common Criteria - CC
(Evaluation Assurance Level)

Federal Information Processing Standards - FIPS
(Security Level)

What is Special about an HSM?

Tamper resistance + evidence

Certified CC EAL 4+ / FIPS 140-3 (3+)

Hardware-based entropy generator

Hardware-accelerated crypto operations

Enabling high quality random number generation (central to crypto operations)

What is Special about an HSM?

Tamper resistance + evidence

Certified CC EAL 4+ / FIPS 140-3 (3+)

Hardware-based entropy generator

Hardware-accelerated crypto operations

Offloads particular operations to specialized hardware - resulting in time savings and resource/power efficiency

Ecosystem of Hardware-Supported Security

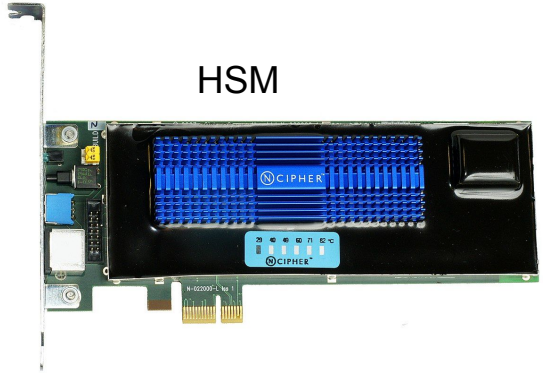
HSM

L-Università ta' Malta

Ecosystem of Hardware-Supported Security

Smart card/Secure element

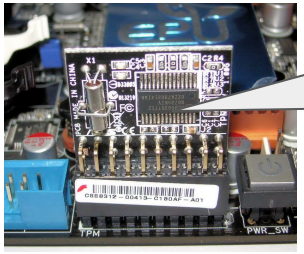
HSM



Ecosystem of Hardware-Supported Security

Smart card/Secure element

HSM



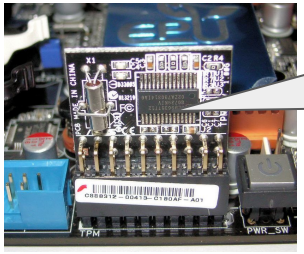
Secure element on the motherboard

Trusted Platform Module (TPM)
[ISO11889 standard]

Ecosystem of Hardware-Supported Security

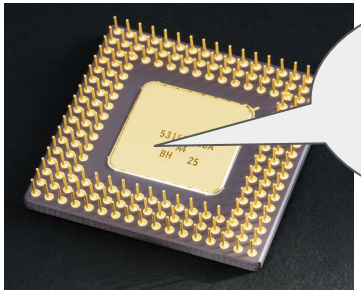
Smart card/Secure element

HSM



Secure element on the motherboard

Trusted Platform Module (TPM) [ISO11889 standard]



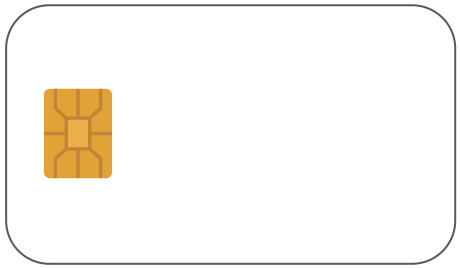
Secure element within the CPU

Trusted Execution Environment

L-Università ta' Malta

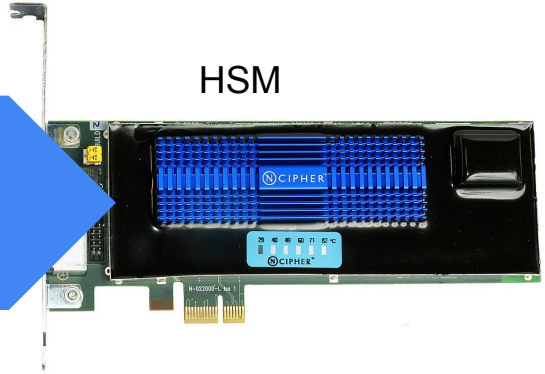
Ecosystem of Hardware-Supported Security

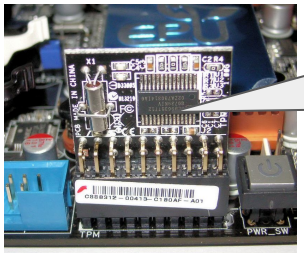
Smart card/Secure element



More processing power and flexibility

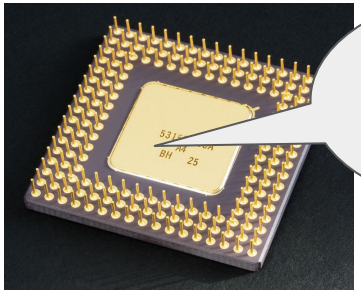
HSM





Secure element on the motherboard

Trusted Platform Module (TPM) [ISO11889 standard]

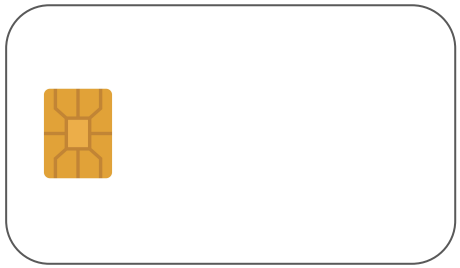


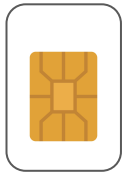
Secure element within the CPU

Trusted Execution Environment

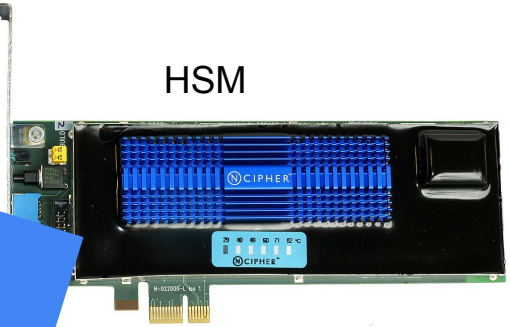
Ecosystem of Hardware-Supported Security

Smart card/Secure element

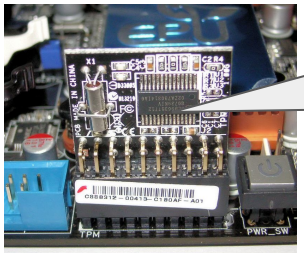




HSM

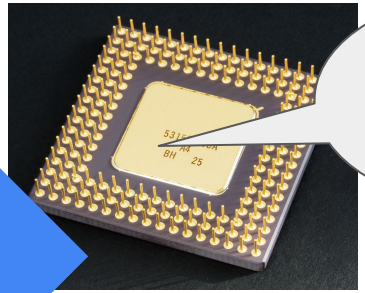


More independence from host



Trusted Platform Module (TP
[ISO11889 standard]

Closer to host



Secure element within the CPU

ed Execution Environment

L-Università ta' Malta

Choosing your HSM

Variety of HSMs

- **USB** - Low-volume transaction environments
- **PCIe** - High speed on-premises solution
- **Network** - Easier to share across applications
- **Cloud** - Could be outsourced

Variety of HSMs

- **USB** - Low-volume transaction environments
- **PCIe** - High speed on-premises solution
- **Network** - Easier to share across applications
- **Cloud** - Could be outsourced

What are the implications of outsourcing your HSM?

What are the Implications of Outsourcing your HSM?

Who Owns, Controls, Uses, Possesses **Y**our **O**wn **K**ey?

L-Università
ta' Malta

What are the Implications of Outsourcing your HSM?

Who Owns, Controls, Uses, Possesses **Y**our **O**wn **K**ey?

- **BringYOK** - client provides the key to the cloud provider

What are the Implications of Outsourcing your HSM?

Who Owns, Controls, Uses, Possesses **Y**our **O**wn **K**ey?

- **BringYOK** - client provides the key to the cloud provider
- **ControlYOK** - client controls all the key's life cycle

What are the Implications of Outsourcing your HSM?

Who Owns, Controls, Uses, Possesses **Y**our **O**wn **K**ey?

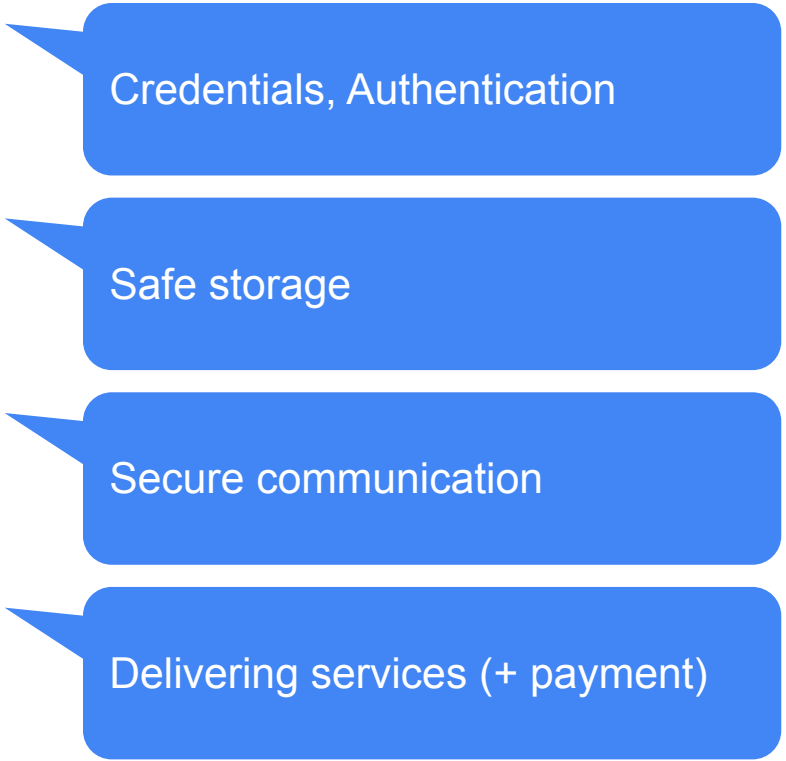
- **BringYOK** - client provides the key to the cloud provider
- **ControlYOK** - client controls all the key's life cycle
- **HoldYOK** - key never leaves the clients infrastructure (cloud provider only deals with encrypted data)

Use Cases of HSMs

- Credential management
- Authentication
- Database/cloud encryption
- Document signing
- Secure communication
- Payments processing
- Application level encryption

Use Cases of HSMs

- Credential management
- Authentication
- Database/cloud encryption
- Document signing
- Secure communication
- Payments processing
- Application level encryption



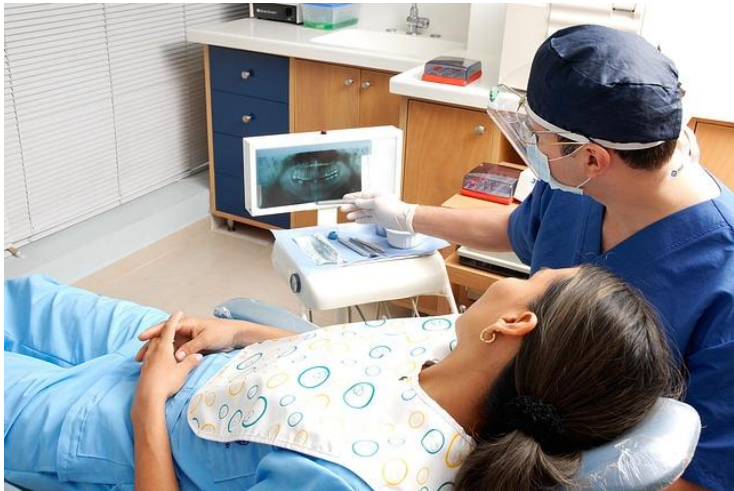
Automotive Industry

- Supply chain security through device attestation
- Secure firmware updates
- Digital car key
- Enable car parts to interact with each other securely
- Car communication with a charging station, road infrastructure
- In-car payments

L-Università ta' Malta

eHealth

- Digital authentication of patient’s health insurance card
- Security for on-premise eHealth data
- Security for cloud-stored eHealth data
- Internet of Medical Things (IoMT)
- Digital transmission of patients’ documents
- Remote healthcare services



L-Università ta' Malta

eGovernment

- Issuance of legal documents
- Security for on-premise eGovernment data
- Security for cloud-stored eGovernment data
- Connected public services
- Public warning solutions
- Offering digital services

L-Università ta' Malta

Banking/Financial Services

- Digital identity management
- Generate credit and debit cards
- CVC generation and validation
- Verify the user-entered PIN
- Secure CNP (Card Not Present) transactions
- POS (Point of Sale) security
- Digital currency transactions

Note: PCI HSM compliance certification

Telecoms

- eSIM provisioning
- Lawful interception solutions for network operators
- Trusted mobile networks
- Security for on-premise telecoms data
- Security for cloud-stored telecoms data
- Device identity protection
- Secure IoT communication
- Public warning
- Secure mobile payment

L-Università ta' Malta

Other Industries

- Manufacturing
- Retail
- Insurance
- Gaming
- Cloud services
- Energy and utilities
- Crypto currencies (ownership of private keys = ownership of account)

Choosing an HSM

- Different kinds of speed, volume required
- Hardware accelerated functions
- Readily available API functions
- Country of design/production
- Certification level

Other Considerations

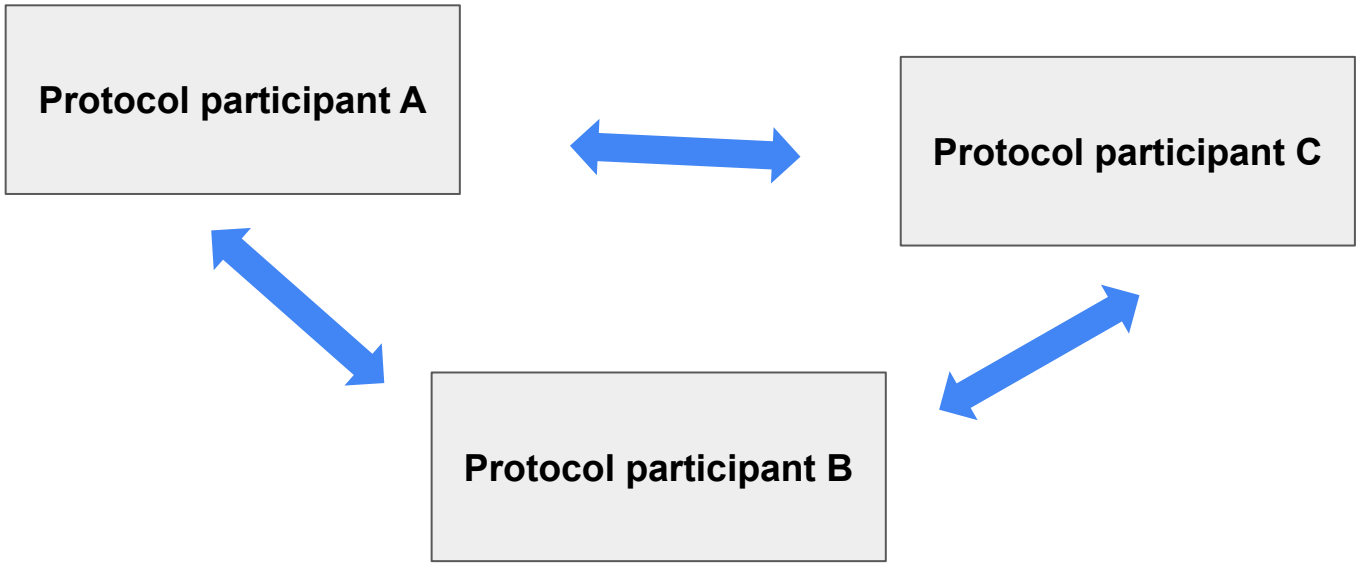
- Backup/failover system
- Future firmware updates
- Authentication options
  (quorum of authenticated users before allowing operations)
- Cost

Secure Communication in the Quantum Era

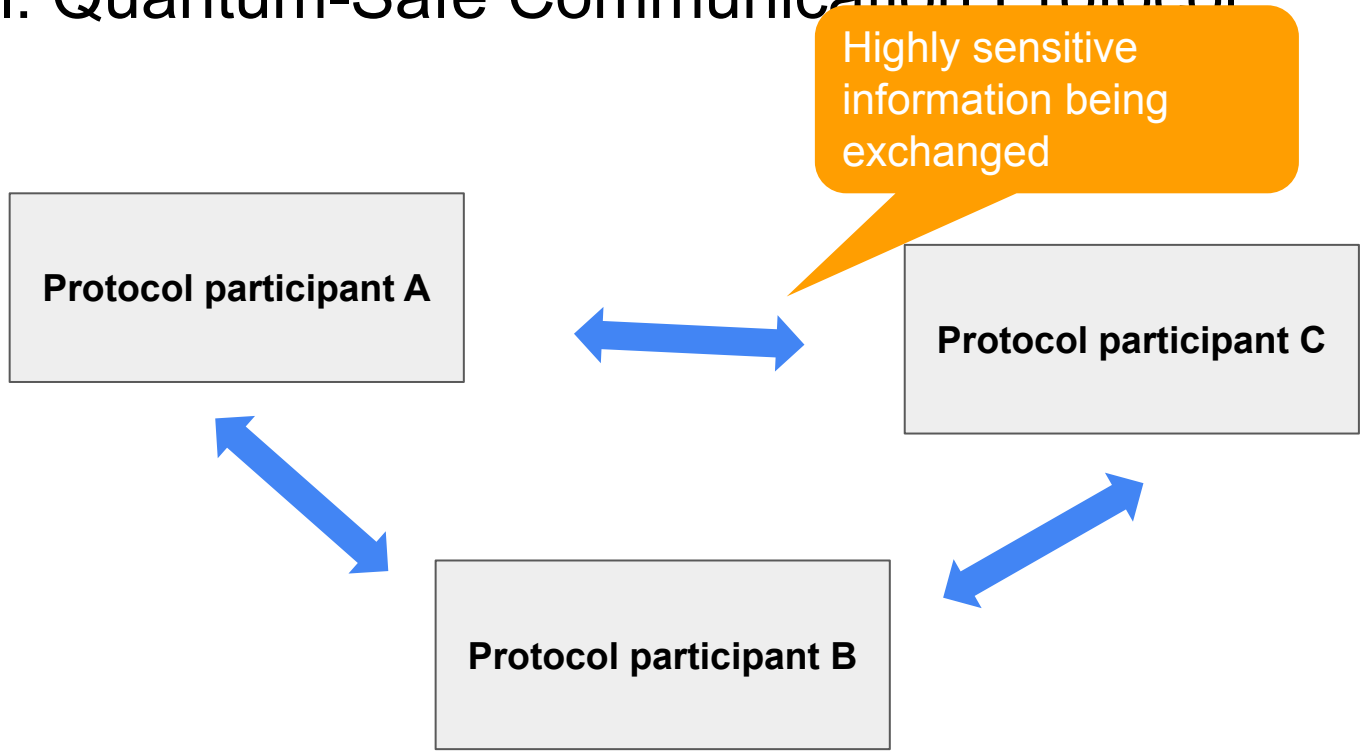
L-Università ta' Malta

NATO OTAN

Goal: Quantum-Safe Communication Protocol



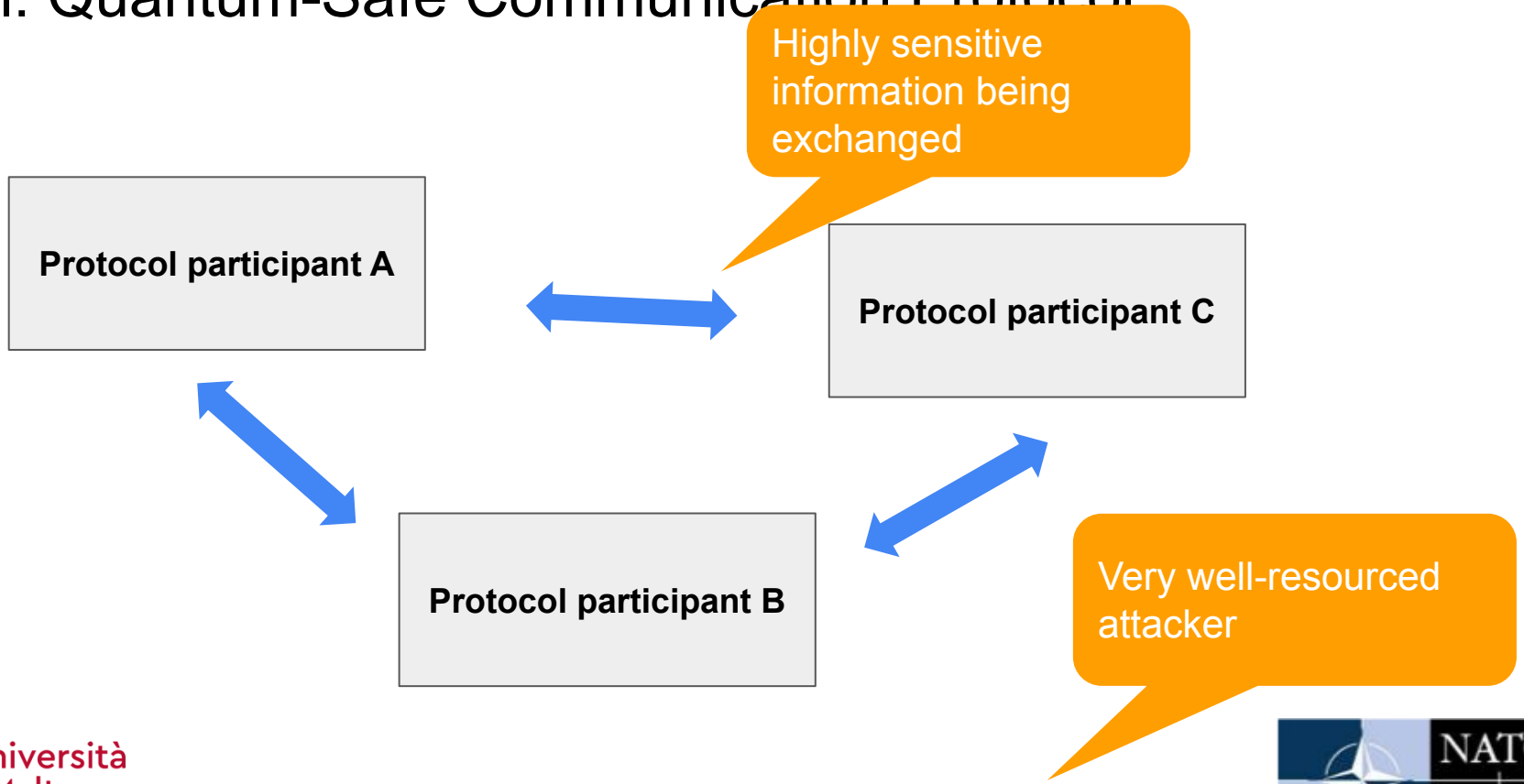
Goal: Quantum-Safe Communication Protocol



L-Università ta' Malta

NATO OTAN

Goal: Quantum-Safe Communication Protocol



L-Università ta' Malta

NATO OTAN

Protocol Design

- `Computation:`
  - For $0 \leq i \leq n, U_i$ chooses $\beta_i \leftarrow \mathbb{Z}_q$ and computes $g_i = \iota(\mathtt{pw})^{\beta_i}$. U_0 sets $M_0 := (g_0)$.
  - For $1 \leq i \leq n, U_i$ computes $(pk_i, sk_i) \leftarrow \mathcal{K}.\mathtt{KeyGen}(1^l)$, and sets $M_i := (pk_i, g_i)$.
- `Communication:`
  - For $0 \leq i \leq n$, U_i broadcasts M_i.

Round II.

- `Computation:`
  - Keying material:
    - U_0 chooses $k \leftarrow \{0,1\}^{p(l)}$, and for each $1 \leq j \leq n$ computes
      $$(c_j, k_j) \leftarrow \mathtt{Encaps}(pk_j, 1^l)$$
      and sets $d_j := k \oplus k_j$, $m_{0,j} := (d_j, c_j)$.
    - For $0 \leq i \leq n$, U_i sets $g_{i,j} := g_j^{\beta_i}$ for each $j \neq i$.
  - Tags:
    - U_0 computes $t_{0,j} := \mathtt{Tag}([g_{0,j}]_L, U_0 || m_{0,j} || M_0 || ... || M_n)$ for each $1 \leq j \leq n$.
    - For $1 \leq i \leq n$, U_i computes

L-Università ta' Malta

NATO OTAN

Protocol Design

- `Computation:`
  - For $0 \leq i \leq n, U_i$ chooses $\beta_i \leftarrow \mathbb{Z}_q$ and computes $g_i = \iota(\mathtt{pw})^{\beta_i}$. U_0 sets $M_0 := (g_0)$.
  - For $1 \leq i \leq n, U_i$ computes $(pk_i, sk_i) \leftarrow \mathcal{K}.\mathtt{KeyGen}(1^l)$, and sets $M_i := (pk_i, g_i)$.
- `Communication:`
  - For $0 \leq i \leq n$, U_i broadcasts M_i.

Round II.

- `Computation:`
  - Keying material:
    - U_0 chooses $k \leftarrow \{0,1\}^{p(l)}$, and for each $1 \leq j \leq n$ computes
      $$(c_j, k_j) \leftarrow \mathtt{Encaps}(pk_j, 1^l)$$
      and sets $d_j := k \oplus k_j$, $m_{0,j} := (d_j, c_j)$.
    - For $0 \leq i \leq n$, U_i sets $g_{i,j} := g_j^{\beta_i}$ for each $j \neq i$.
  - Tags:
    - U_0 computes $t_{0,j} := \mathtt{Tag}([g_{0,j}]_L, U_0 || m_{0,j} || M_0 || ... || M_n)$ for each $1 \leq j \leq n$.
    - For $1 \leq i \leq n$, U_i computes

Proven Correct

L-Università ta' Malta

NATO OTAN

Protocol Design

- **Computation:**
  - For $0 \leq i \leq n, U_i$ chooses $\beta_i \leftarrow \mathbb{Z}_q$ and computes $g_i = \iota(\mathtt{pw})^{\beta_i}$. U_0 sets $M_0 := (g_0)$.
  - For $1 \leq i \leq n, U_i$ computes $(pk_i, sk_i) \leftarrow \mathcal{K}.\mathtt{KeyG}$ $M_i := (pk_i, g_i)$.
- **Communication:**
  - For $0 \leq i \leq n$, U_i broadcasts M_i.

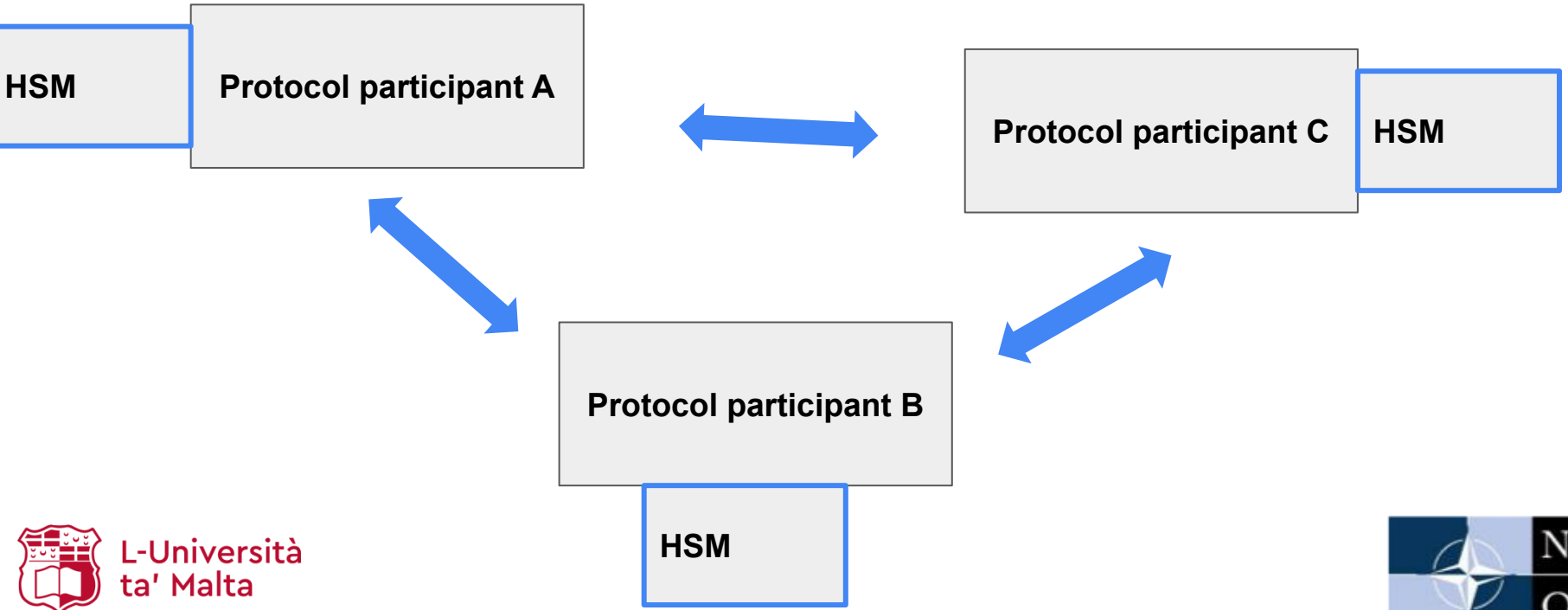
Round II.

- **Computation:**
  - Keying material:
    - U_0 chooses $k \leftarrow \{0,1\}^{p(l)}$, and for each $1 \leq j \leq n$ computes
      $$(c_j, k_j) \leftarrow \mathtt{Encaps}(pk_j, 1^l)$$
      and sets $d_j := k \oplus k_j$, $m_{0,j} := (d_j, c_j)$.
    - For $0 \leq i \leq n$, U_i sets $g_{i,j} := g_j^{\beta_i}$ for each $j \neq i$.
  - Tags:
    - U_0 computes $t_{0,j} := \mathtt{Tag}([g_{0,j}]_L, U_0 || m_{0,j} || M_0 || ... || M_n)$ for each $1 \leq j \leq n$.
    - For $1 \leq i \leq n$, U_i computes

Proven Correct

What does it really mean?

Using HSMs


HSM
Protocol participant A
Protocol participant C
HSM
Protocol participant B
HSM

Which HSM? - SECube

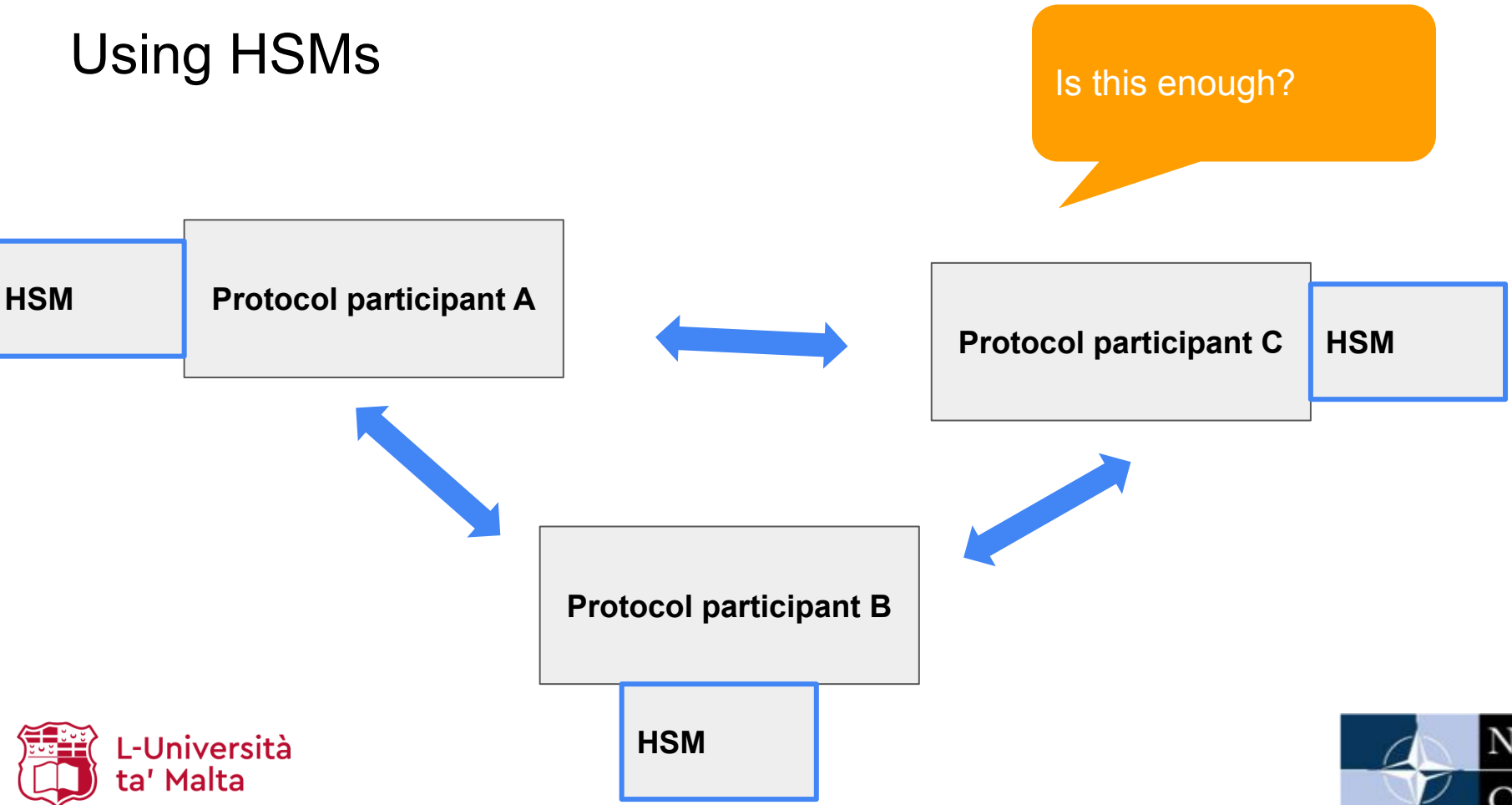
A small HSM (per protocol client)

Open source library, firmware

Three cores:

- ARM Cortex-M4 processor
- Field-Programmable-Gate-Array (FPGA)
- EAL5+ SmartCard

Using HSMs

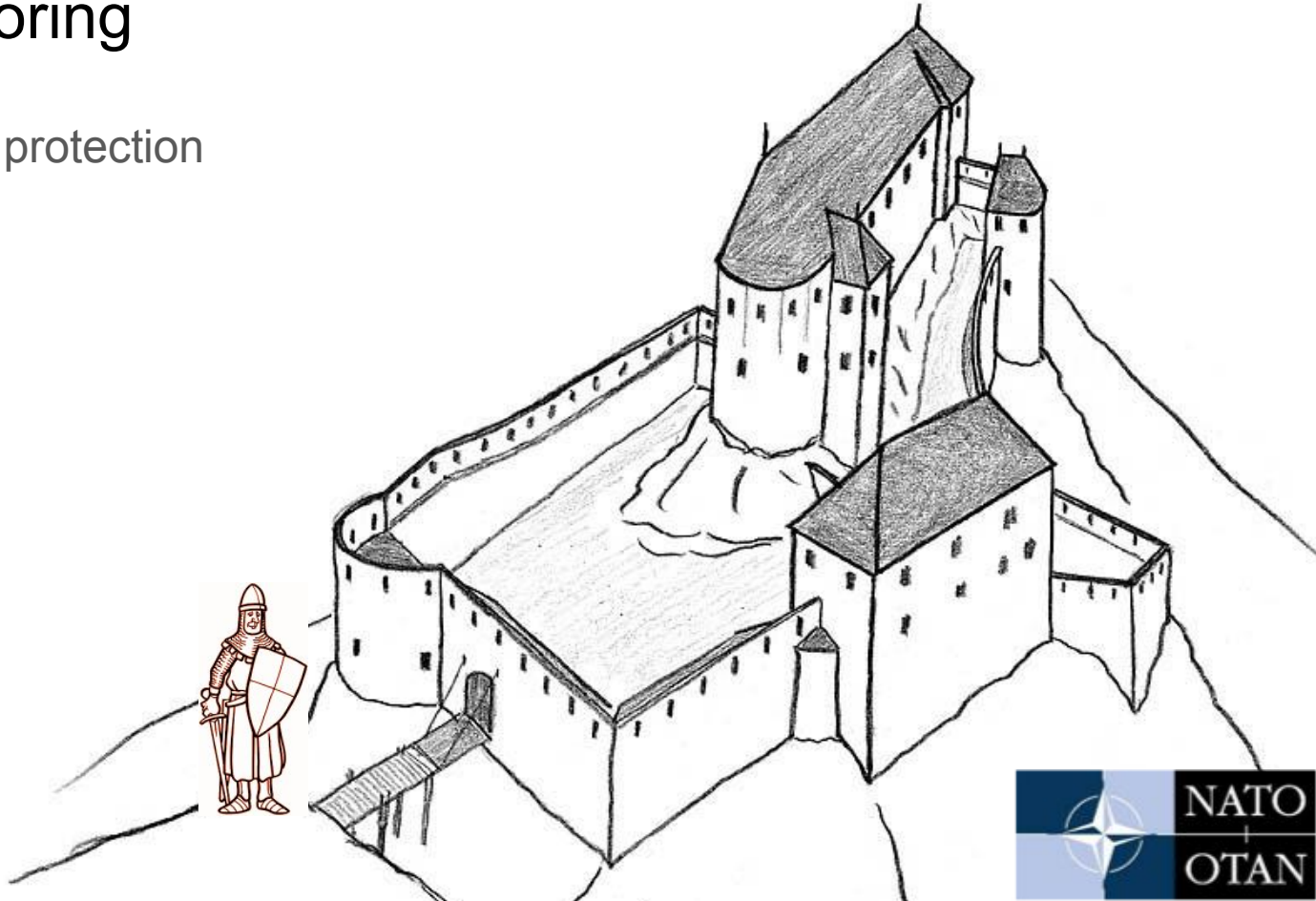


L-Università ta' Malta

NATO OTAN

Software Monitoring

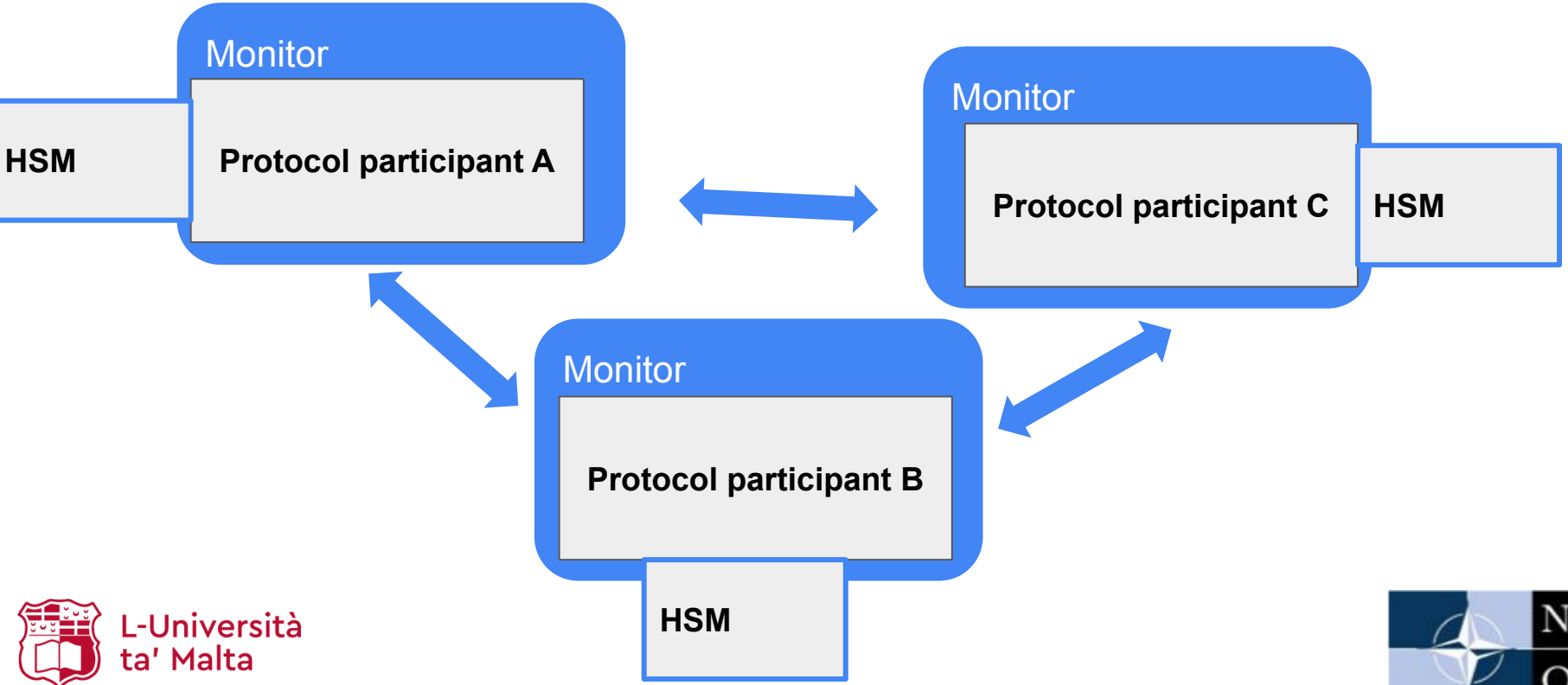
An additional layer of protection



L-Università ta' Malta

NATO OTAN

Introducing Monitors

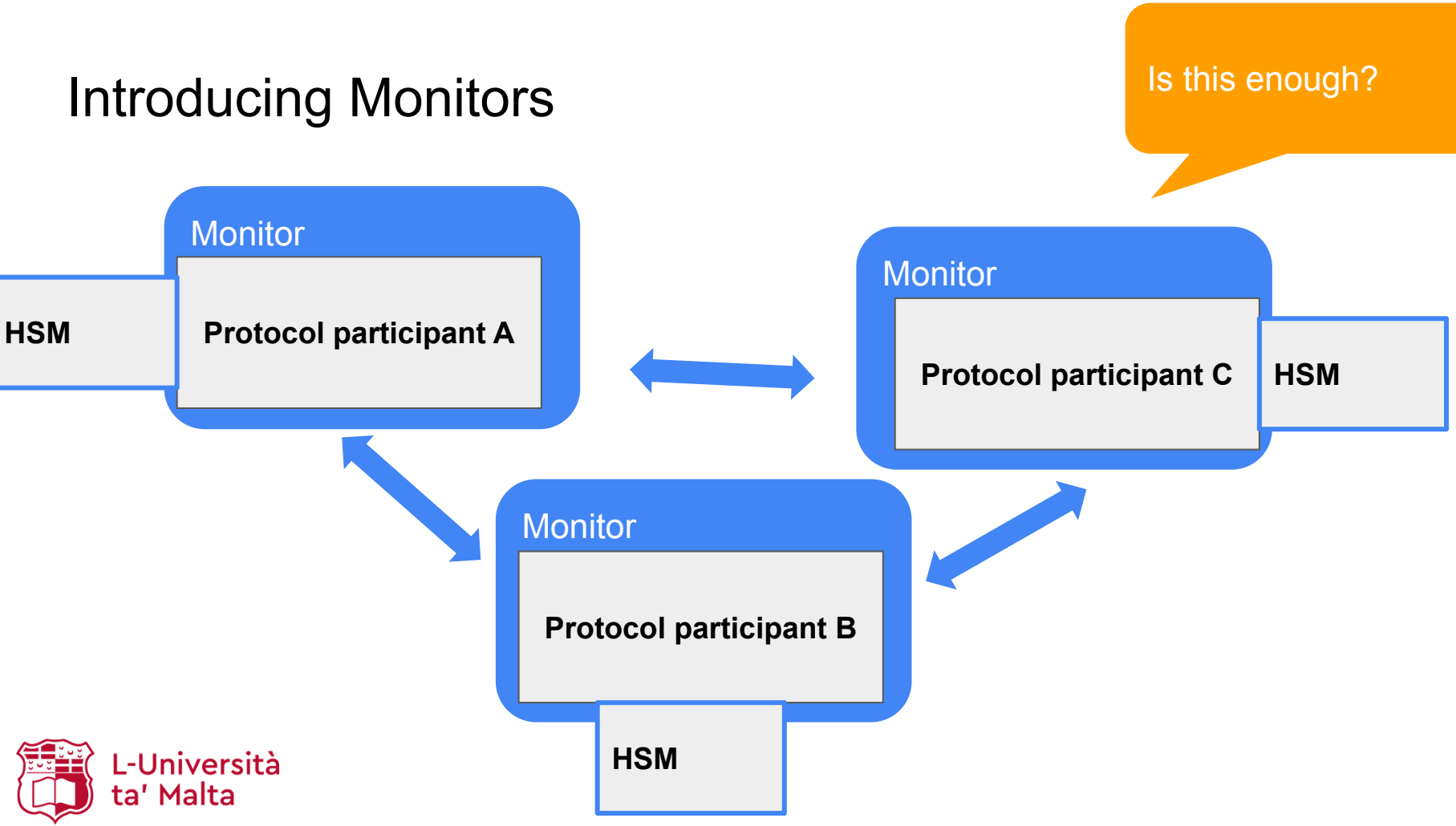


What did we Monitor?

- Function calls at different levels: app, library
- Parameter values
- Data entering and leaving the boundary of trust
- Quality of randomness

Looking Forward and Concluding

Introducing Monitors



L-Università ta' Malta

Safeguarding Monitors and Logs

- Isolating monitoring processes at OS level
- Making logs tamper-evident

Conclusion

We operate in a connected world with adversaries, hence cryptography

Endpoint security remains a big challenge!

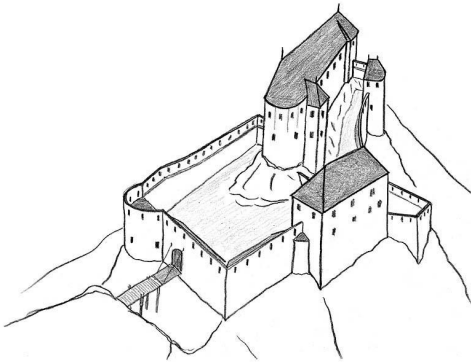
Conclusion

We operate in a connected world with adversaries, hence cryptography

Endpoint security remains a big challenge!

Crypto hardening is a **never-ending battle**

Adding hardware to the solution can up the stakes



L-Università ta' Malta

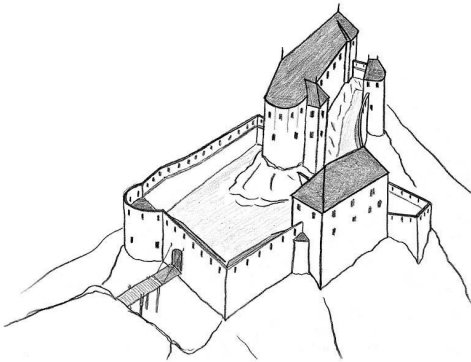
Conclusion

We operate in a connected world with adversaries, hence cryptography

Endpoint security remains a big challenge!

Crypto hardening is a **never-ending battle**

Adding hardware to the solution can up the stakes



Managing the impact of attacks (layered defences, unique keys)

Increase the cost of attack up to an accepted risk

L-Università ta' Malta

Questions

christian.colombo@um.edu.mt

End