FinTech R&D Innovation Conference

SnT securityandtrust.lu

Secure Authentication over Insecure Channels

Dr Jean Lancrenon

Research associate, SnT

ApSIA

FinTech R&D Innovation Conference
January 19th, 2016
Chambre de Commerce, Luxembourg

uni.lu UNIVERSITÉ DU LUXEMBOURG

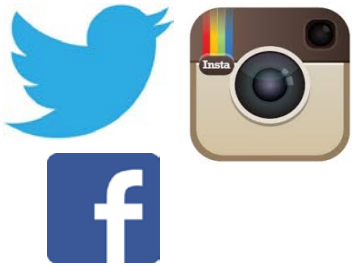
FinTech R&D Innovation Conference

Bank of America

BGL BNP PARIBAS

CRÉDIT AGRICOLE

SnT securityandtrust.lu



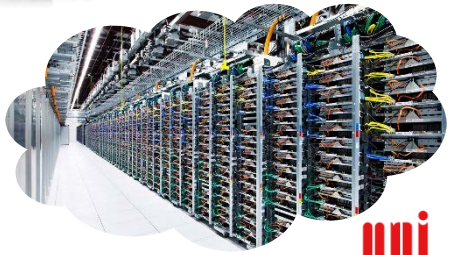
Services

The e-world today

Hardware

JPMorganChase

DigiNotar Internet Trust Services

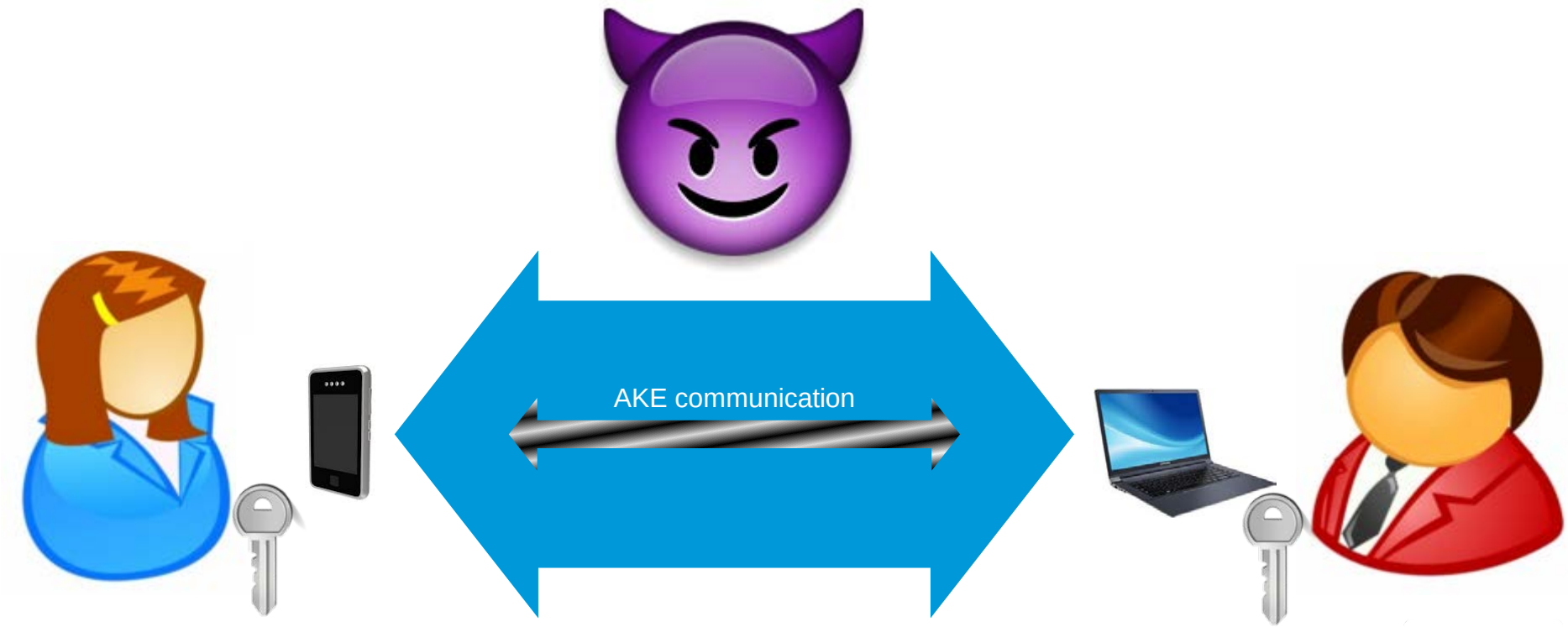


Breaks!!!

uni.lu Université du Luxembourg

- The Internet has to be properly secured
- How?
- A fundamental starting point: Secure communications
- A crypto solution: ***Authenticated Key Exchange*** protocols

AKE

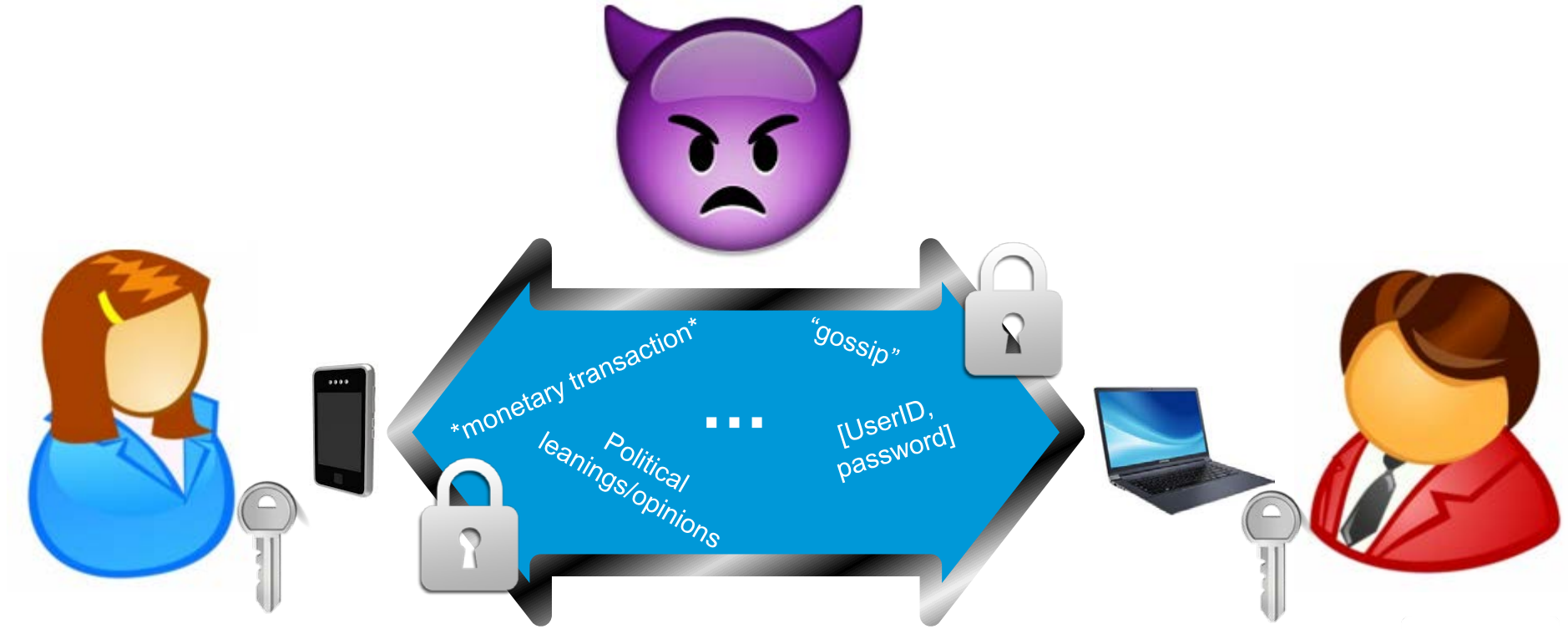


- Both parties have the guarantee that  is known **only** by the correct opposing party
- Long-term authentication material must be in place (e.g. passwords, PKI…)

FinTech R&D Innovation Conference

AKE

SnT securityandtrust.lu



uni.lu Université du Luxembourg

AKE

- Not new technology (dates back to 1976, with Diffie and Hellman)

- Everybody in the room has used it at least once:

  - You're using it anytime you see this:

AKE

What is the mathematical study of AKE security?

1) Create a mathematical model of the adversary
2) Define “breaking” the protocol
3) Prove mathematically that the adversary cannot break the protocol

Why is it important?

1) Eliminates a large class of adversaries
2) Provides new insight into protocol design and desirable security properties
3) Secure systems start with fundamentally secure crypto

What kind of research?

1) The modeling is not yet fully understood
2) Models need refinement and enrichment

AKE

- initialize user instance $(\mathcal{U}, i, role^i_{\mathcal{U}}, pid^i_{\mathcal{U}})$: $\mathcal{A}$ activates an instance $\mathcal{U}^i$ of initialized user $\mathcal{U}$, where $i \in \mathbb{N}$. $\mathcal{A}$ specifies $\mathcal{U}^i$'s $role^i_{\mathcal{U}} \in \{open, connect\}$ and partner identity $pid^i_{\mathcal{U}}$. We convene that the instance with the role *open* sends the last protocol message, while the instance with the role *connect* receives the last protocol message. We require that $pid^i_{\mathcal{U}}$ be the identity of some other initialized user $\mathcal{V}$. Furthermore, we must have $role_{\mathcal{U}} = client$ and $role_{\mathcal{V}} = server$ or vice-versa.
- send $(\mathcal{U}, i, InMsg)$: $\mathcal{A}$ sends message $InMsg$ to instance $\mathcal{U}^i$. As a result, $\mathcal{U}^i$ processes $InMsg$ according to the protocol, and outputs a reply $OutMsg$ and its $status^i_{\mathcal{U}} \in \{continue, abort,$
  $accept, terminate\}$ to $\mathcal{A}$.
  Depending on the value of $status^i_{\mathcal{U}}$, the following happens:
  If $status^i_{\mathcal{U}} = continue$, $\mathcal{U}^i$ is simply waiting for another protocol message;
  If $status^i_{\mathcal{U}} = accept$, $\mathcal{U}^i$ must have $role^i_{\mathcal{U}} = connect$. $\mathcal{U}^i$ has generated a session identity $sid^i_{\mathcal{U}} \neq \varepsilon$, and may be waiting for another protocol message. The session identity $sid^i_{\mathcal{U}}$ is handed to $\mathcal{A}$. This status indicates that the instance believes it has enough information to compute a good session key;
  If $status^i_{\mathcal{U}} = terminate$, $\mathcal{U}^i$ has generated a session key $sk^i_{\mathcal{U}}$ and a session identity $sid^i_{\mathcal{U}}$ and will no longer produce or receive messages. The session identity $sid^i_{\mathcal{U}}$ is handed to $\mathcal{A}$. The session key $sk^i_{\mathcal{U}}$ is not.
  If $status^i_{\mathcal{U}} = abort$, $\mathcal{U}^i$ halts with $sk^i_{\mathcal{U}} = \perp$ and will no longer produce or receive messages.
- execute $(\mathcal{C}, i, \mathcal{S}, j)$: This causes an honest protocol execution between instances $\mathcal{C}^i$ and $\mathcal{S}^j$. The complete set $ExchMsg$ of exchanged messages is then handed to $\mathcal{A}$. Both instances $\mathcal{C}^i$ and $\mathcal{S}^j$ must have been initialized but otherwise unused, and at the end of the query we have $status^i_{\mathcal{C}} = status^j_{\mathcal{S}} = terminate$.
- ideal oracle (F, x): $\mathcal{CH}$ returns the value of $v = F(x)$ (where $F \in \{H, E, D\}$) to $\mathcal{A}$.

$$T := \frac{6n_{se}}{N} + \frac{4(n_{se} + n_{ex})(2n_{se} + n_{ex} + n_{h1})}{q^2} + \frac{n_{h0}^2 + 2n_{h1}}{q} + \frac{n_{h1}^2 + 2n_{se}}{2^k} +$$
$$2n_{h1}(1 + {n_{se}}^2) \times Succ^{cdh}_{PW,\mathbb{G}}(\mathcal{B}) + 4n_{h0}^3 \times \left(Adv^{didh}_{g,\mathbb{G}}(\mathcal{D}) + \frac{n_{h1}^3 + 3n_{se}}{q} \right) \quad (5)$$

Theoretical AKE security is absolutely necessary but

CERTAINLY NOT SUFFICIENT

FinTech R&D Innovation Conference

SnT securityandtrust.lu

TAKEAWAY POINTS

- AKE (and secure channel) protocols are ***CENTRAL*** to having a secure, trustworthy platform for online communication for anybody online

- AKE protocol design and deployment starts (but certainly does not end) with good security theory

- AKE security theory is not yet fully understood, warranting further research

uni.lu Université du Luxembourg

FinTech R&D Innovation Conference

SnT securityandtrust.lu

Fonds National de la Recherche Luxembourg

uni.lu Université du Luxembourg

(The speaker is supported by the Fonds National de la Recherche, Luxembourg)

FinTech R&D Innovation Conference

SnT securityandtrust.lu

for questions contact:
jean.lancrenon@uni.lu

uni.lu Université du Luxembourg