Smals
ICT for society

Gestion des certificats digitaux et méthodes alternatives de chiffrement

Mai 2011

Julien Cathalo
Section Recherches

Cryptographie à clé publique

- Invention du concept : 1976 (Diffie, Hellman)
- Premier système concret : 1978 (Rivest, Shamir, Adleman), RSA
- Technologie mature
  - Nombreux standards
  - Nombreuses applications avec impact pour Smals et ses membres
    - eID, TLS/SSL

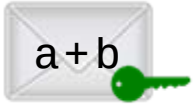
Cryptographie à clé publique

1. Contrainte : certificats
   - Gestion lourde
   - Risque d'incidents
2. Chiffrement alternatif
   - Nouvelles fonctionnalités
     - Éviter les certificats
     - Faire des calculs sur des données chiffrées
     - Répartir le déchiffrement entre plusieurs personnes
   - Des centaines d'articles sur le sujet
   - Impact pour Smals ?

Plan

1. Gestion des certificats digitaux

2. Méthodes alternatives de chiffrement



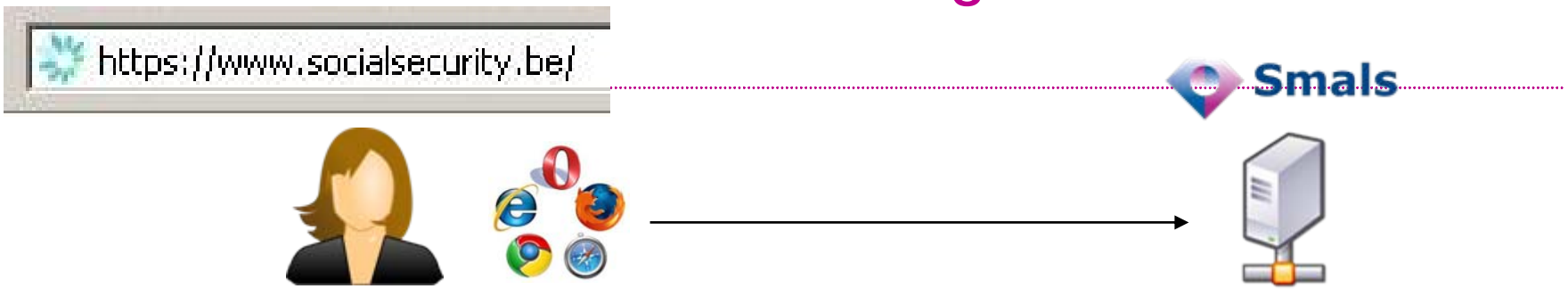
Partie 1 : Certificats Digitaux

Motivation

- Etude Venafi :

  "*78% des organisations ont connu des problèmes de disponibilité et des pertes financières à cause de certificats*"
- Chez Smals :
  - Des centaines de certificats à gérer
  - Quelques incidents
- Demande à la section Recherches
- Etude de la méthodologie
- Etude d'outils

Partie 1 : Certificats Digitaux



- Besoins de sécurité
  - Authentification
    - Vérifier que le site avec lequel elle est connectée est bien www.socialsecurity.be
    - Evite "url spoofing"
  - Chiffrement
    - Chiffrer les échanges entre le navigateur d'Alice et le site
    - Protège la confidentialité des informations échangées

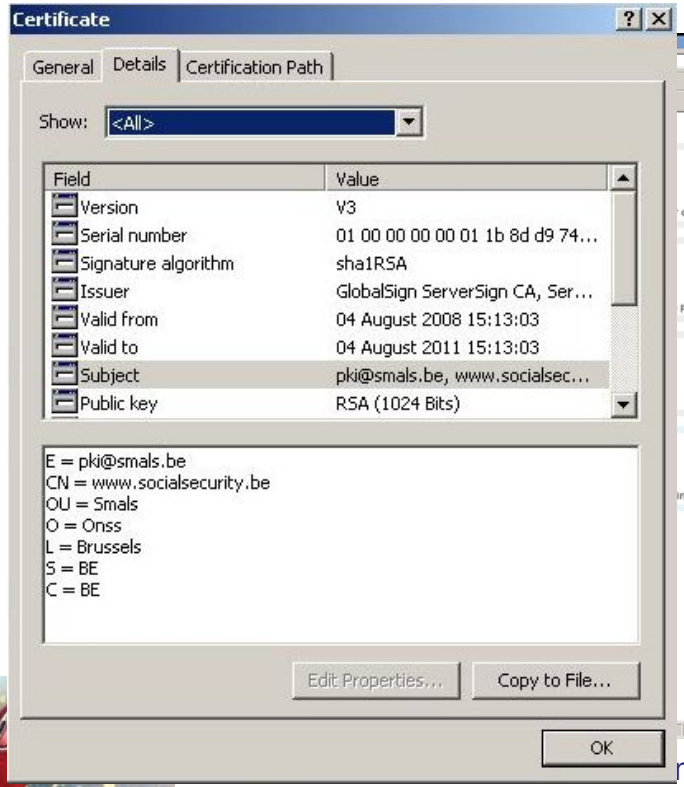
socialsecurity.be https://www.socialsecurity.be/

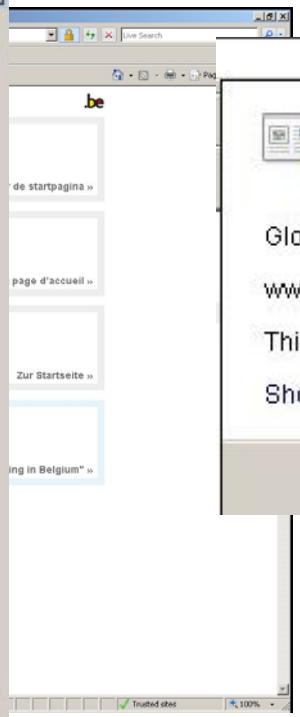
https://www.socialsecurity.be

ertificats Digitaux

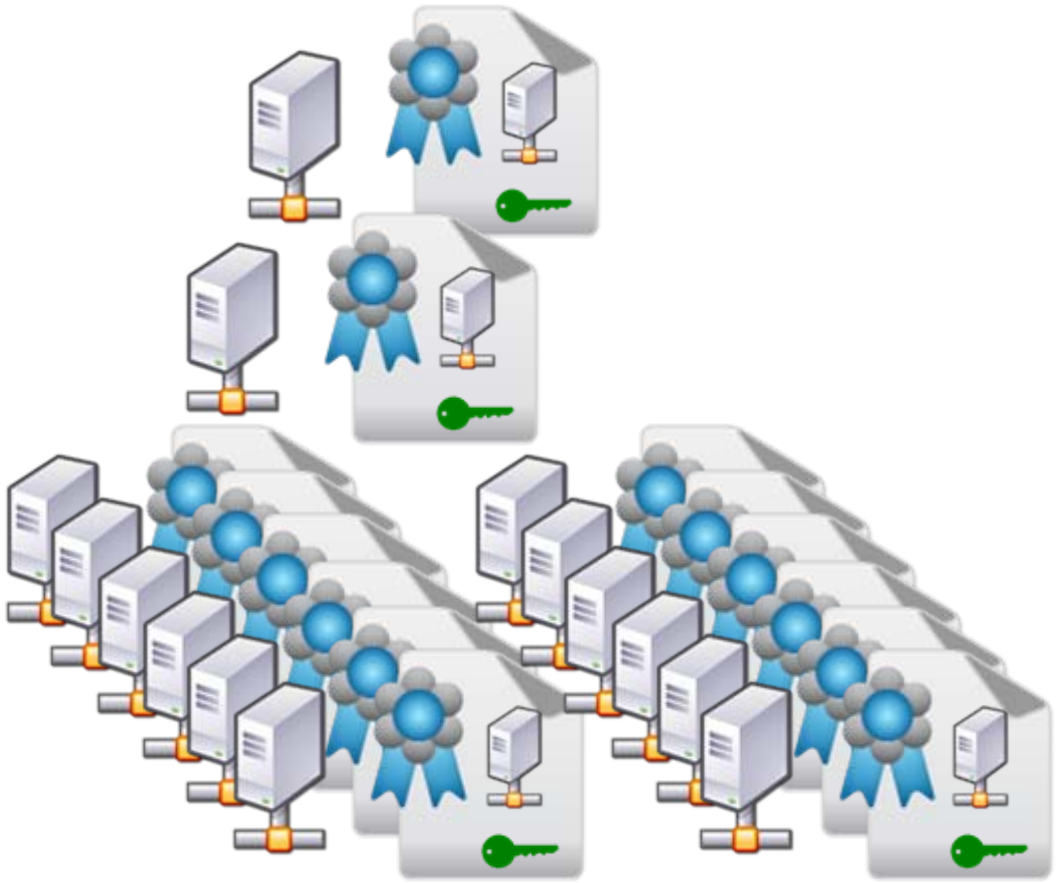
https://www.socialsecurity.be/

Smals





Partie 1 : Certificats Digitaux



Partie 1 : Certificats Digitaux

1. Introduction aux certificats
   - Cryptographie à clé publique
   - TLS / SSL
   - Certificats et PKI
2. Gestion de certificats
   - Méthodologie
   - Outils

Partie 1 : Certificats Digitaux

1. Introduction aux certificats
   - Cryptographie à clé publique
   - TLS / SSL
   - Certificats et PKI
2. Gestion de certificats
   - Méthodologie
   - Outils

La cryptographie

- Cryptographie : boîte à outils (algorithmes, protocoles) qui servent à sécuriser des informations
- Applications (entre autres) :
  - Chiffrement
  - Signature digitale
  - Authentification

Clé secrète ou clé publique ?

- Cryptographie à clé secrète (ou cryptographie symétrique)
  - Chiffrement : la même clé pour chiffrer ou déchiffrer
  - Algorithmes très rapides
- Cryptographie à clé publique (ou cryptographie asymétrique)

  - Deux clés par utilisateur
  - Chiffrement : une clé pour chiffrer, une autre pour déchiffrer
  - Signature digitale
  - Algorithmes plus lents

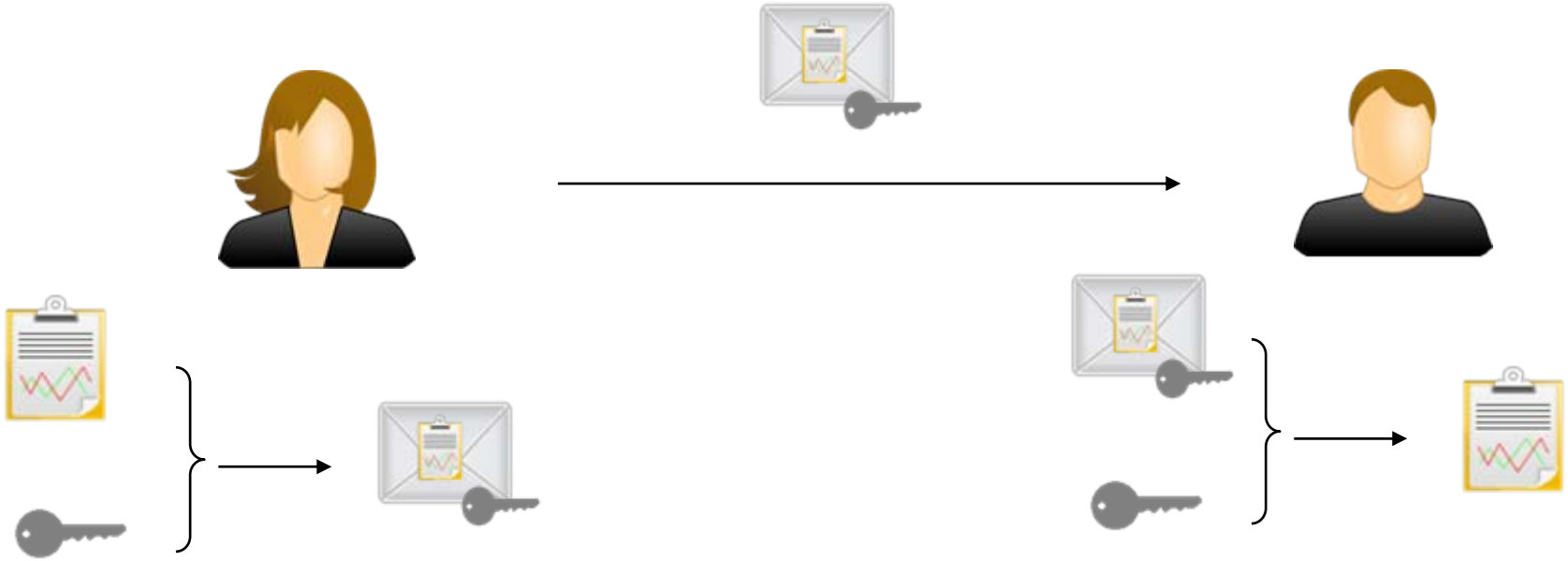
Notations

- Message / document : 
- Clé : 
- Message chiffré avec la clé : 
- Signature du message avec la clé : 

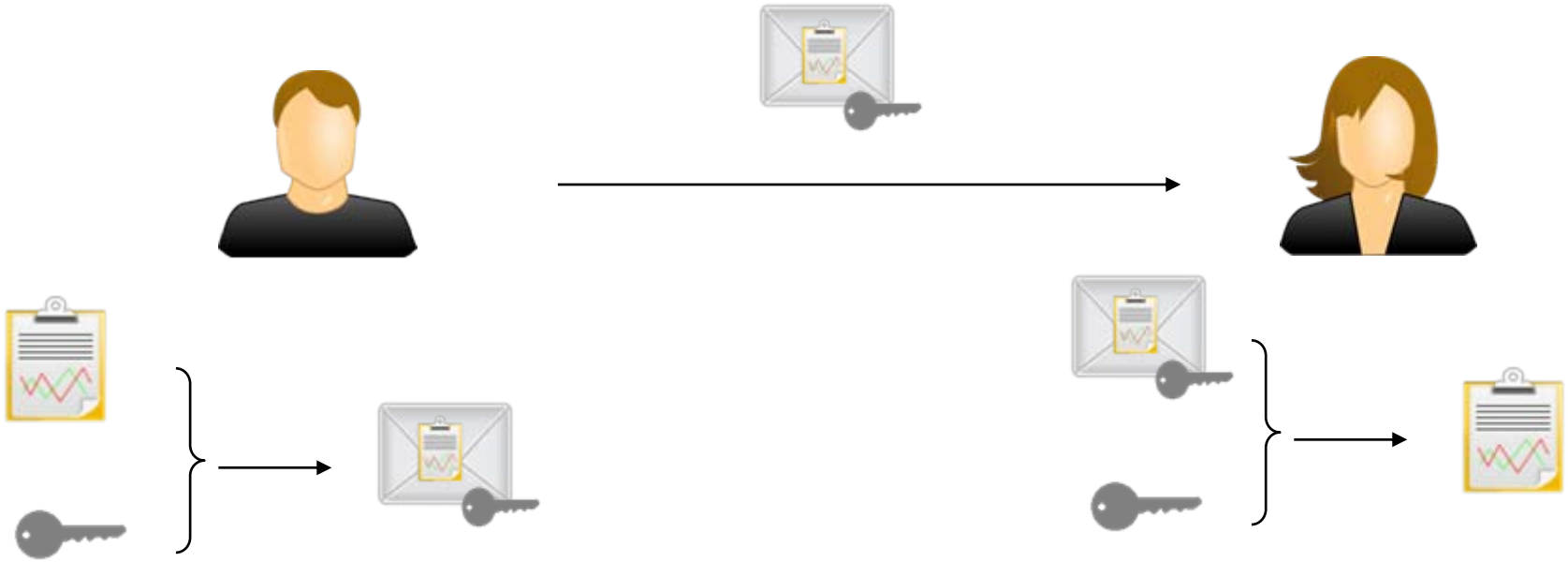
Chiffrement à clé secrète

- Famille : cryptographie à clé secrète
- Alice et Bob se mettent d'accord sur une donnée, la clé secrète 
- Alice veut chiffrer un message pour Bob

Chiffrement à clé secrète : Alice vers Bob



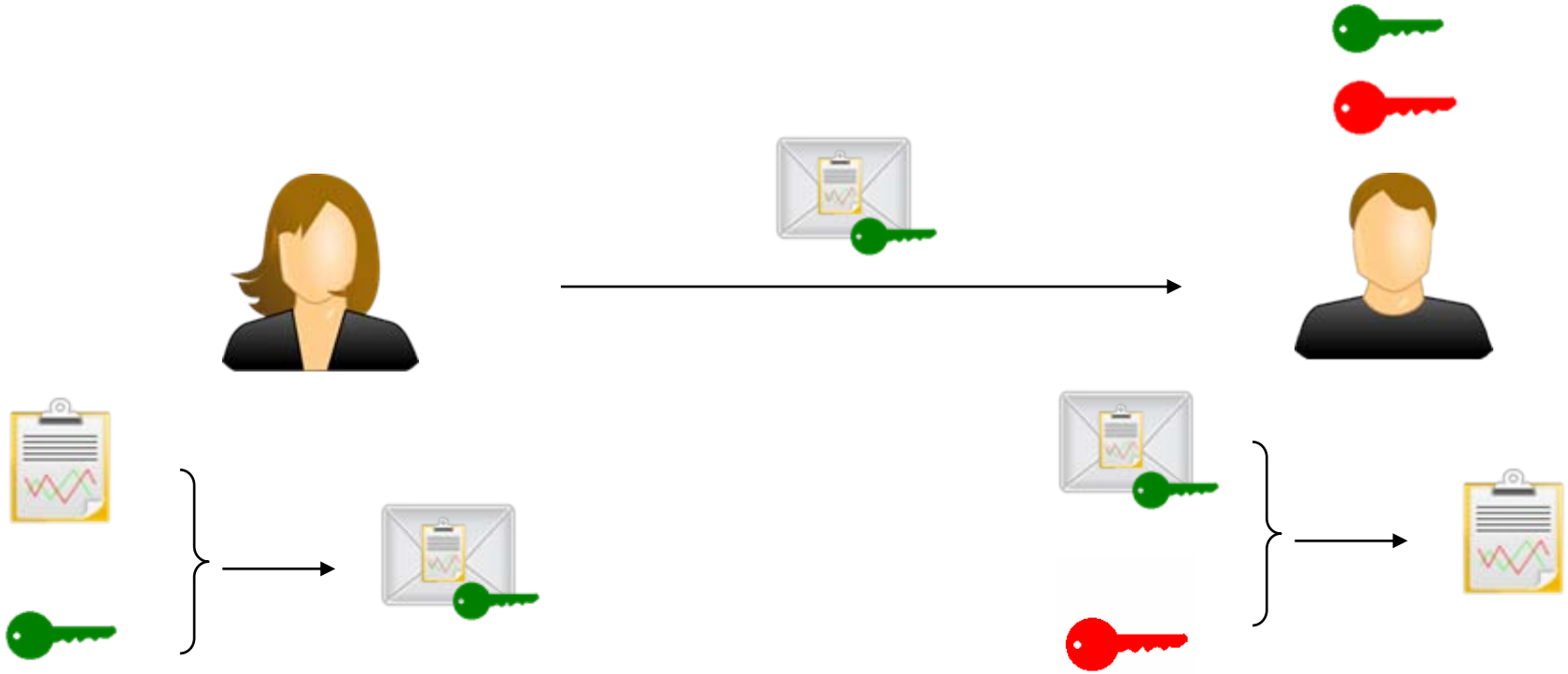
Chiffrement à clé secrète : Bob vers Alice



Chiffrement à clé publique

- Famille : cryptographie à clé publique
- Chaque utilisateur a deux clés
  - Clé privée 
  - Clé publique 
- Chacun peut générer sa paire de clés
-  doit être gardée secrète !
-  peut être rendue publique
- Propriétés mathématiques : impossible en pratique de calculer  à partir de 

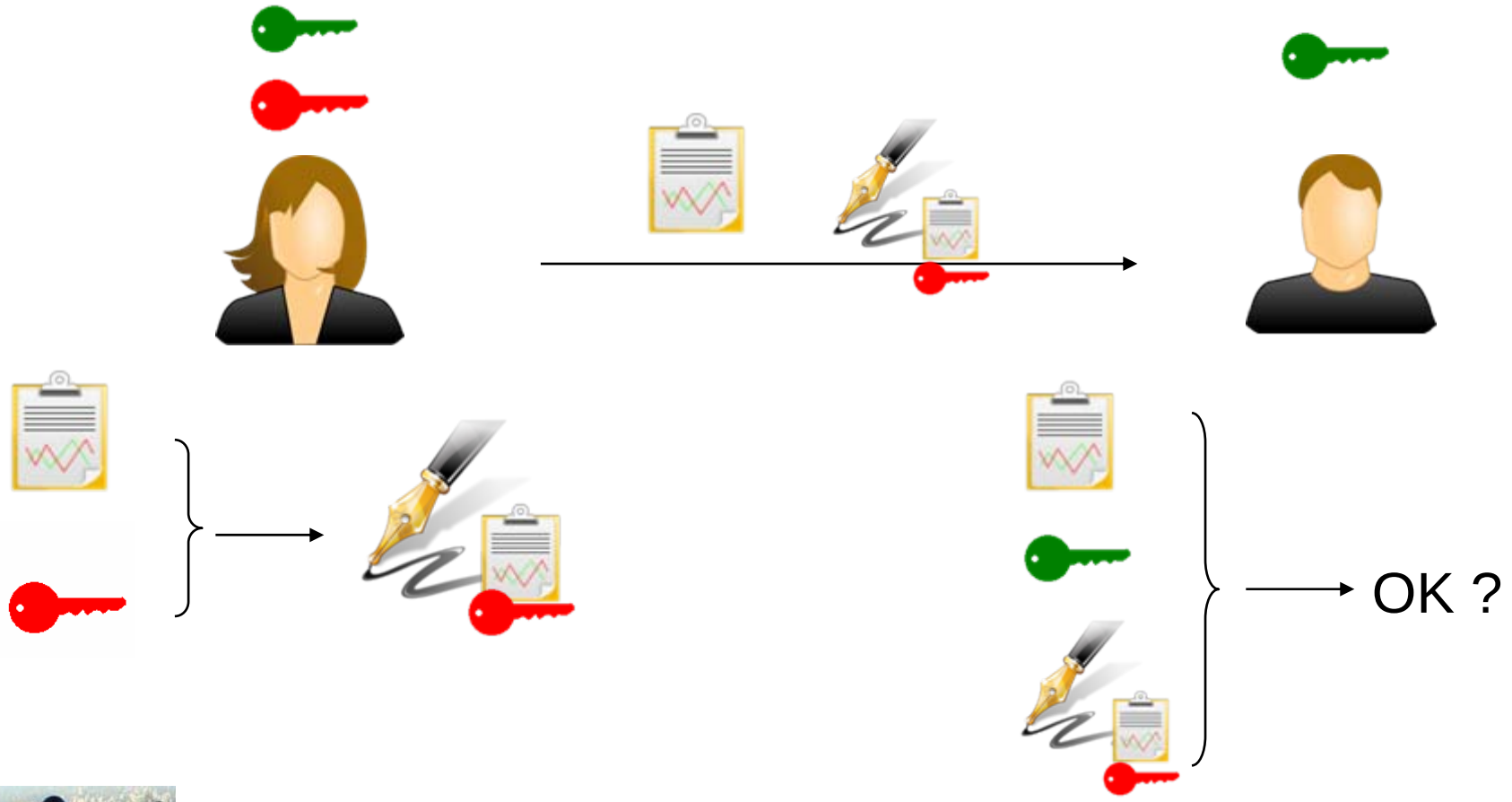
Chiffrement Alice vers Bob



Signature digitale

- Famille : cryptographie à clé publique
- Chaque utilisateur a deux clés
  - Clé privée 
  - Clé publique 

Signature par Alice et vérification par Bob



Partie 1 : Certificats Digitaux

1. Introduction aux certificats
   - Cryptographie à clé publique
   - TLS / SSL
   - Certificats et PKI
2. Gestion de certificats
   - Méthodologie
   - Outils

TLS / SSL

- TLS (Transport Layer Security)
  - Successeur de SSL (Secure Sockets Layer)
  - Protocole qui permet d'établir une connexion sécurisée client / serveur au niveau de la couche transport
  - Url en "https"
- Assure :
  - L'authentification du serveur
  - Le chiffrement et l'intégrité des messages échangés entre client et serveur
  - (optionnellement, l'authentification du client)

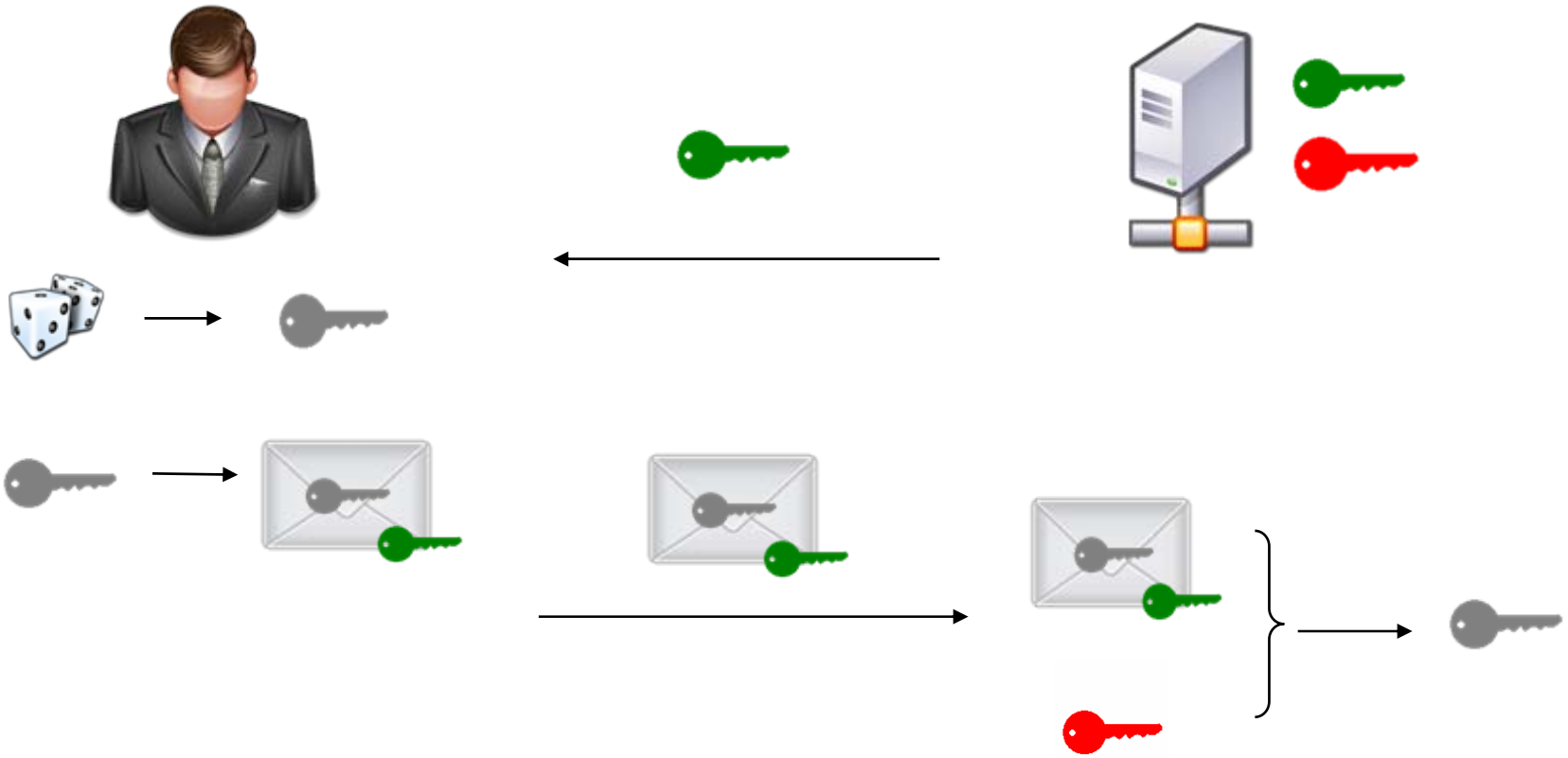
TLS / SSL

Deux étapes :

1. Simple TLS Handshake
    1. Etablit une clé de session
    2. Le client vérifie l'identité du serveur

2. Session sécurisée
    1. Utilise la clé de session pour chiffrer et authentifier les données échangées

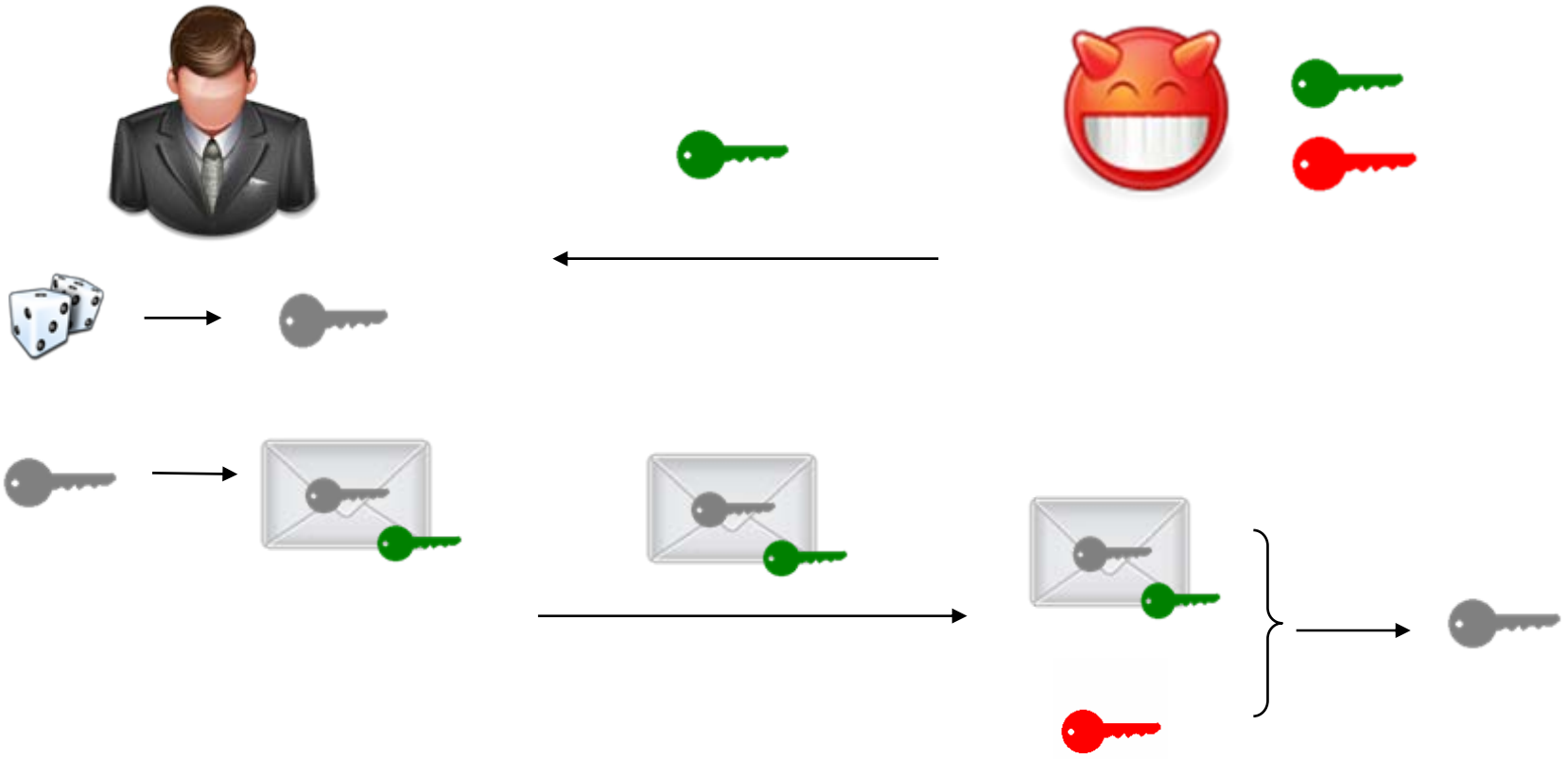
TLS / SSL avec Simple TLS Handshake



TLS / SSL avec Simple TLS Handshake

- Le client vérifie implicitement l'identité du serveur
  - Lui seul a pu déchiffrer  grâce à 
- Le client et au serveur de disposer d'une clé secrète connue d'eux seuls (); cette clé sert pour le reste de la session

Sans vérification de la clé publique...



Sans vérification de la clé publique...

- Le client doit pouvoir distinguer la clé publique du serveur authentique et celle de l'attaquant
- Le client doit vérifier le certificat digital du serveur

Partie 1 : Certificats Digitaux

1. Introduction aux certificats
   - Cryptographie à clé publique
   - TLS / SSL
   - Certificats et PKI
2. Gestion de certificats
   - Méthodologie
   - Outils

Certificat : qu'est-ce que c'est ?

- Un petit fichier (1 ou 2 KB)
- Dans un format normalisé (X.509)
- Qui sert à attester du lien entre

1. Une entité (personne morale, personne physique, serveur, application...)

et

2. Une clé publique

- Généré par une autorité de certification (CA) qui signe le contenu avec sa propre clé privée

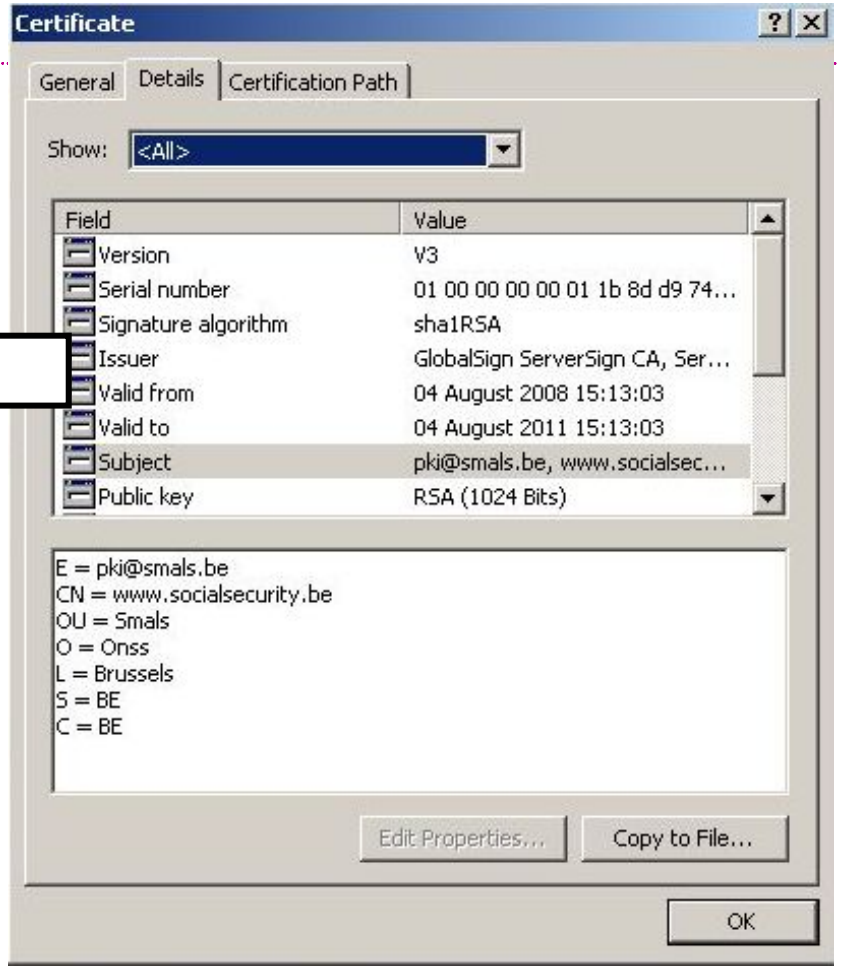
Champs (X.509 v3)

- Certificate
  - Version
  - Serial Number
  - Algorithm ID
  - Issuer
  - Validity
    - Not Before
    - Not After
  - Subject
  - Subject Pu
    - Public Key Algorithm
    - Subject Public Key
  - Issuer Unique Identifier (optional)
  - Subject Unique Identifier (optional)
  - Extensions (optional)
    - ...
- Certificate Signature Algorithm
- Certificate Signature

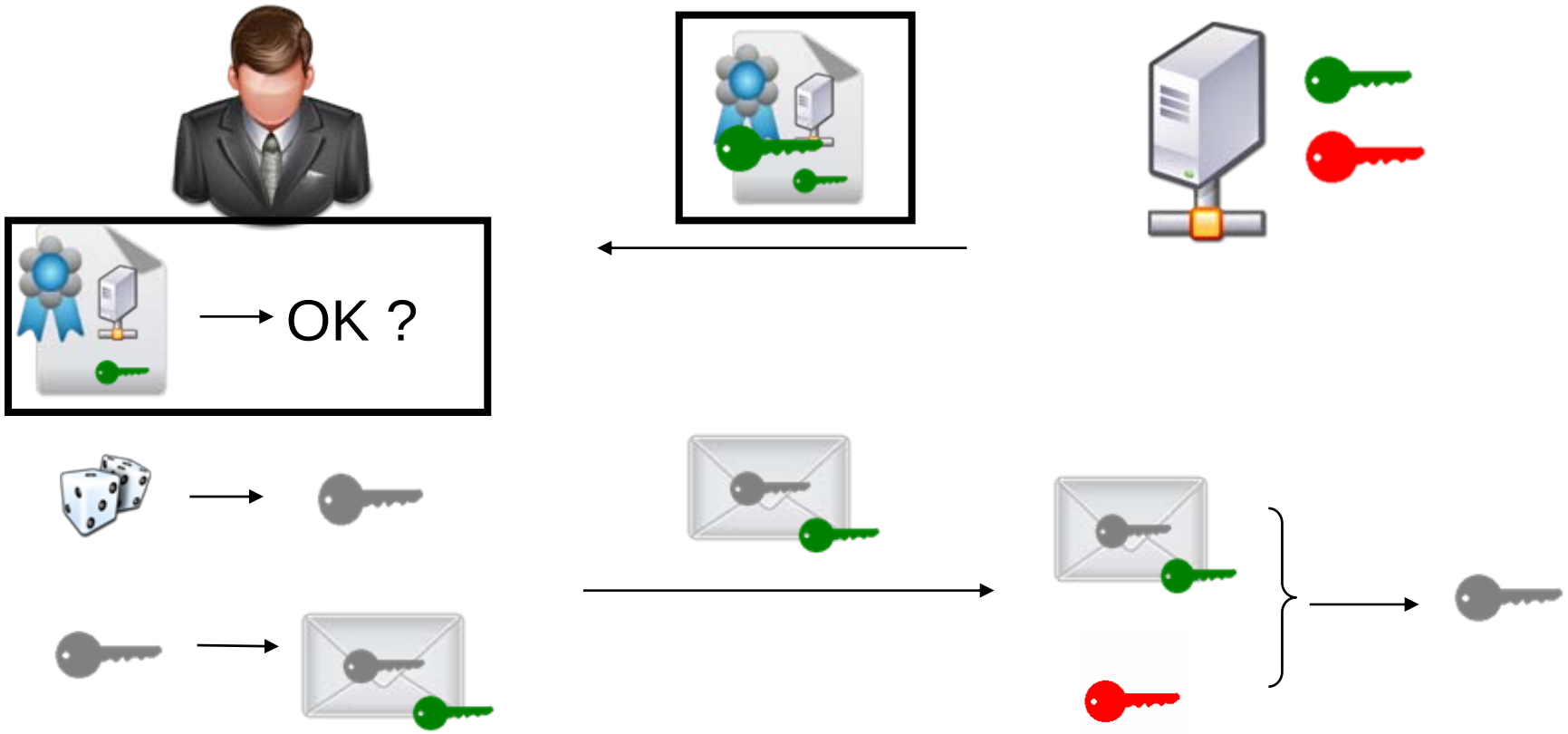
Qui a émis le certificat ?

Quand expire-t-il ?

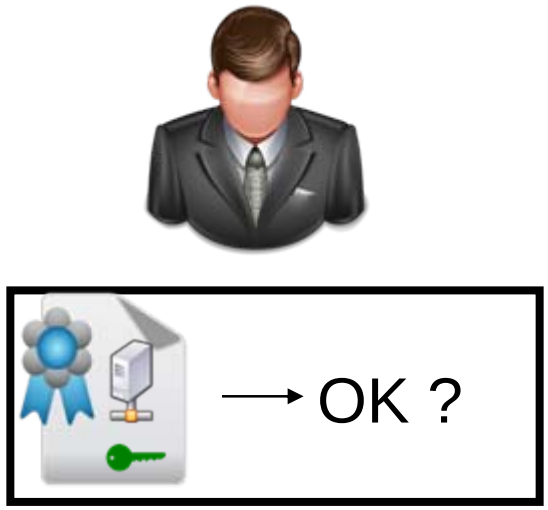
Pour qui ?



TLS / SSL avec Simple TLS Handshake



Vérification du certificat

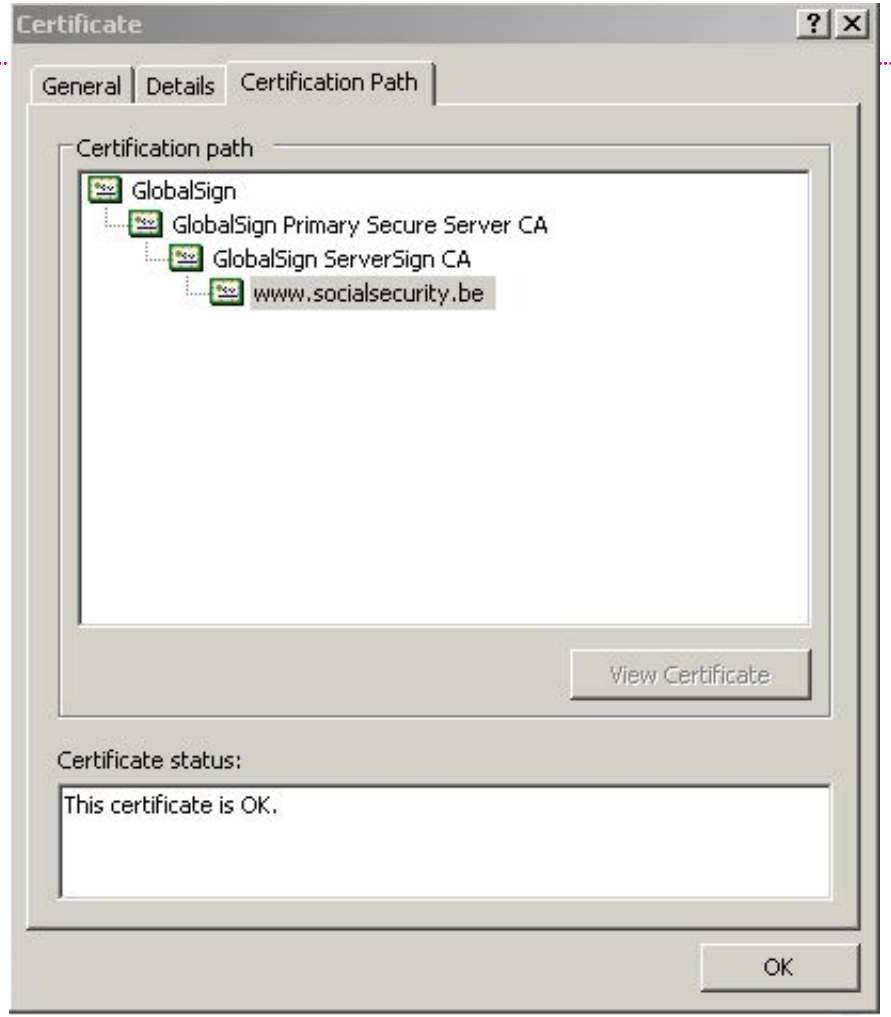


- Contenu du certificat
  - Subject
  - Validity
    - Not Before
    - Not After
- Vérifier la validité de la signature contenue dans le champ "Certificate Signature"
- En utilisant le Certificate Store et la chaîne de certification

Certificate Store

- Chaque navigateur contient une liste de certificats
- Par défaut, le navigateur a *confiance* dans les CAs qui ont émis ces certificats
- Pour vérifier un certificat qui n'est pas dans cette liste :
  - Chaîne de certification

Chaîne de certification



Chaîne de certification

Certificat contenu dans le CS, contient

Navigateur a confiance en

Chaîne de certification

Contient
signé avec
vérifié avec

Navigateur a confiance en 

Chaîne de certification

Contient 3

signé avec 2

vérifié avec 2

Navigateur a confiance en 1 2 3

Chaîne de certification

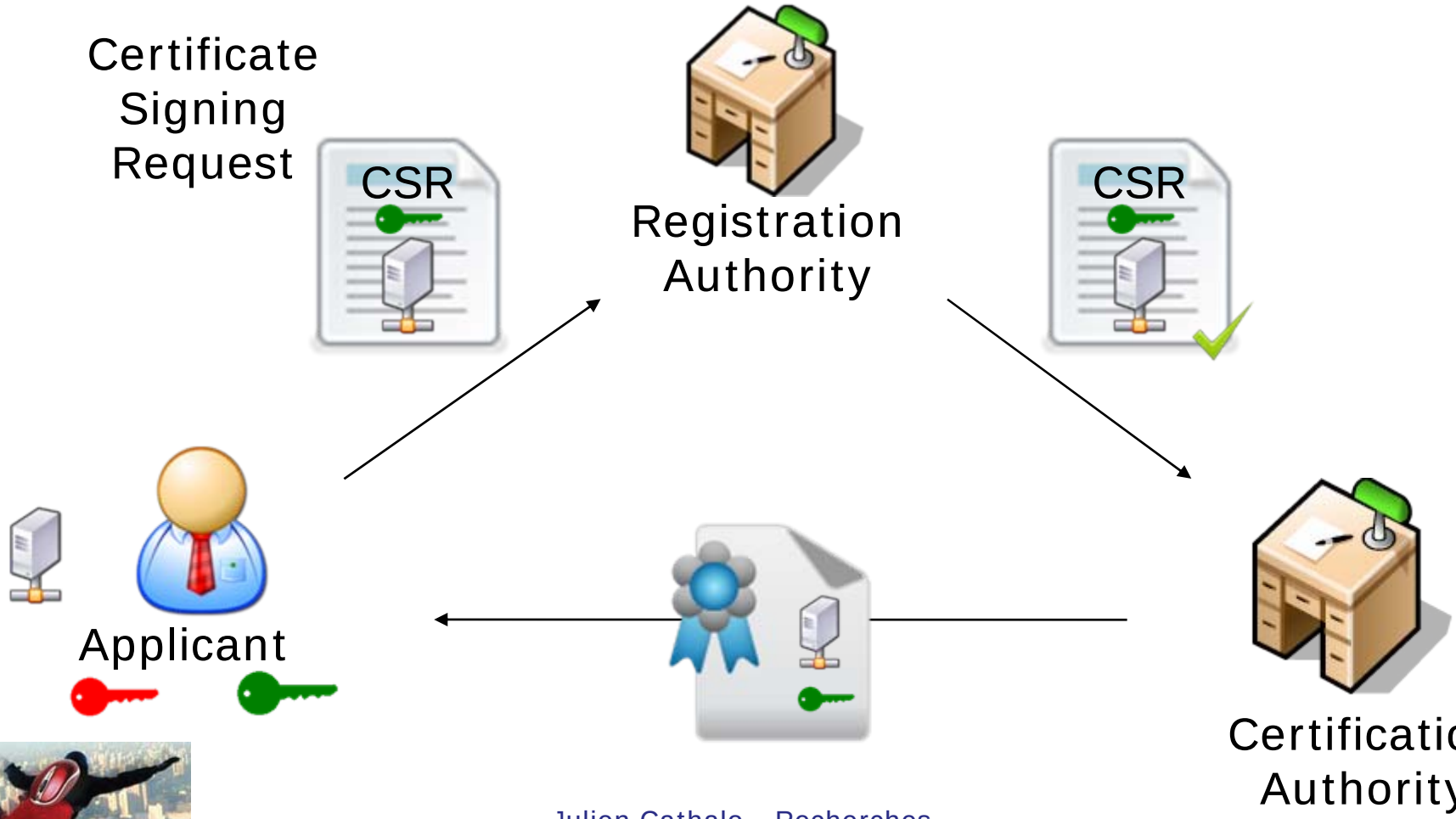
Contient 4

signé avec 3

vérifié avec 3

Navigateur a confiance en 1 2 3 4

Public Key Infrastructure (PKI) (Simplifiée)



TLS / SSL : Un système parfait ?

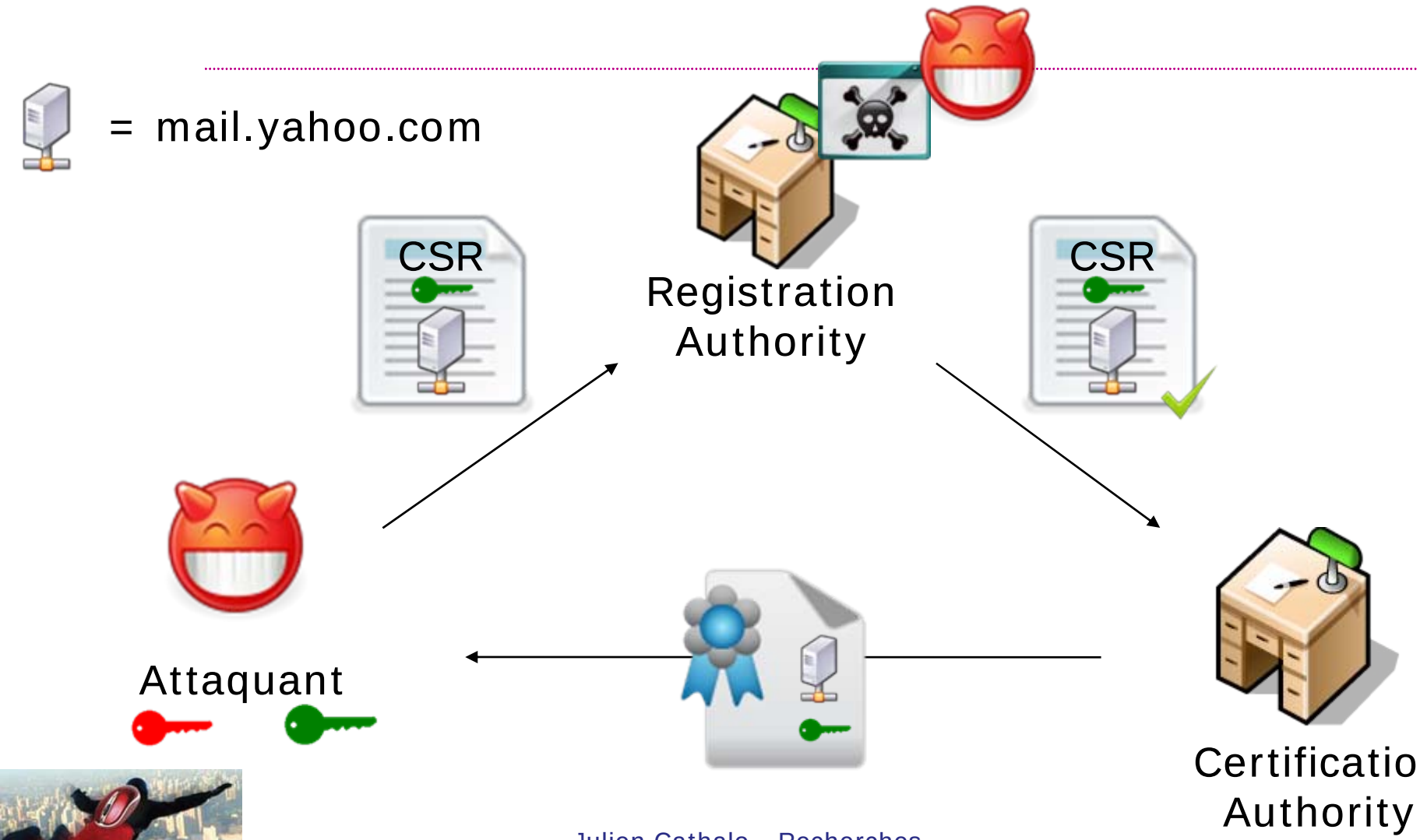
- Mal compris des internautes
  - Certificat invalide : l'internaute accepte la connexion malgré l'avertissement
  - Certificat valide : illusion de sécurité
    - TLS / SSL : Fourgon blindé entre PC et serveur
    - Mais si le PC est une passoire...
- Quelques failles
  - Dans le protocole lui-même
  - Dans ses implémentations

Disfonctionnements : le cas de Comodo

- Mars 2011
- Comodo (Certificate Authority, USA)
- Un attaquant a obtenu 9 faux certificats
  - mail.google.com
  - www.google.com
  - login.yahoo.com
  - login.skype.com
  - Etc...

Disfonctionnements : le cas de Comodo

= mail.yahoo.com



Disfonctionnements : le cas de Comodo

- Origine de l'attaque :
  - IP iranienne (pas une preuve !)
- Conséquences (en théorie)
  - Attaquant crée une fausse page web
  - Dirige des internautes vers cette page
  - Considérée comme authentique par le navigateur
  - Vol de logins / mots de passe d'utilisateurs, vol de données, etc...

Disfonctionnements : le cas de Comodo

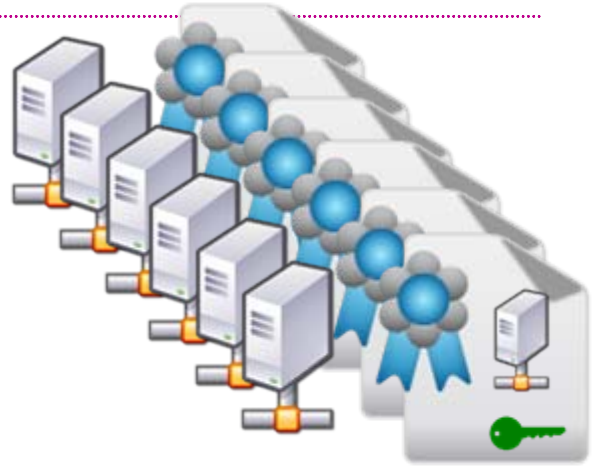
- Pas d'impact connu
  - Attaque détectée rapidement
  - Faux certificats
    - révoqués rapidement
    - a priori pas utilisés

Partie 1 : Certificats Digitaux

1. Introduction aux certificats
   - Cryptographie à clé publique
   - TLS / SSL
   - Certificats et PKI
2. Gestion de certificats
   - Méthodologie
   - Outils

Gestion de certificats

- Organisation comme Smals :
  - Des centaines de serveurs
  - Plusieurs environnements
  - Plusieurs CAs externes
- Principal risque :
  - Certificat expiré



Certificat expiré

- Causes possibles
  - Oubli de demande de renouvellement
  - Le CA n'a pas envoyé le certificat à temps
  - Erreur (typo) dans la CSR
  - Erreur dans le traitement de la CSR par le CA

Certificat expiré

- Conséquences
  - Arrêt du service
  - TLS / SSL : Internaute reçoit message d'erreur
  - Problème d'image
- Pistes pour diminuer ce risque
  - Méthodologie
    - Inventaire
    - Process
  - Outils

Méthodologie : inventaire

- Avoir un inventaire des certificats qui contient
  - Service responsable de l'installation
  - Champs du certificat
    - Date d'expiration

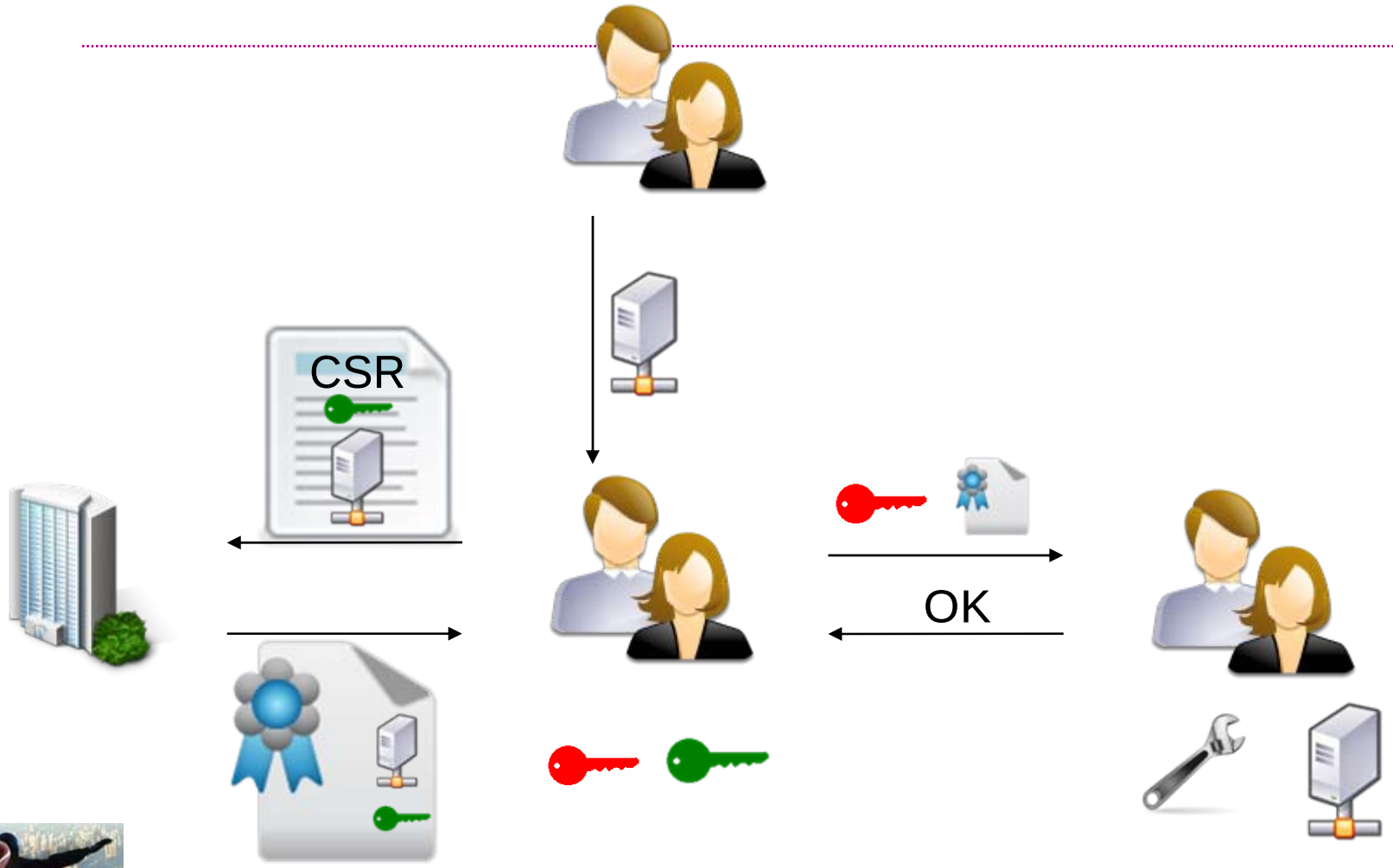
Méthodologie : inventaire

- Permet d'anticiper les expirations
- Forme ?
  - Tableur
  - CMDB (Configuration Management DataBase)
  - Outil dédié

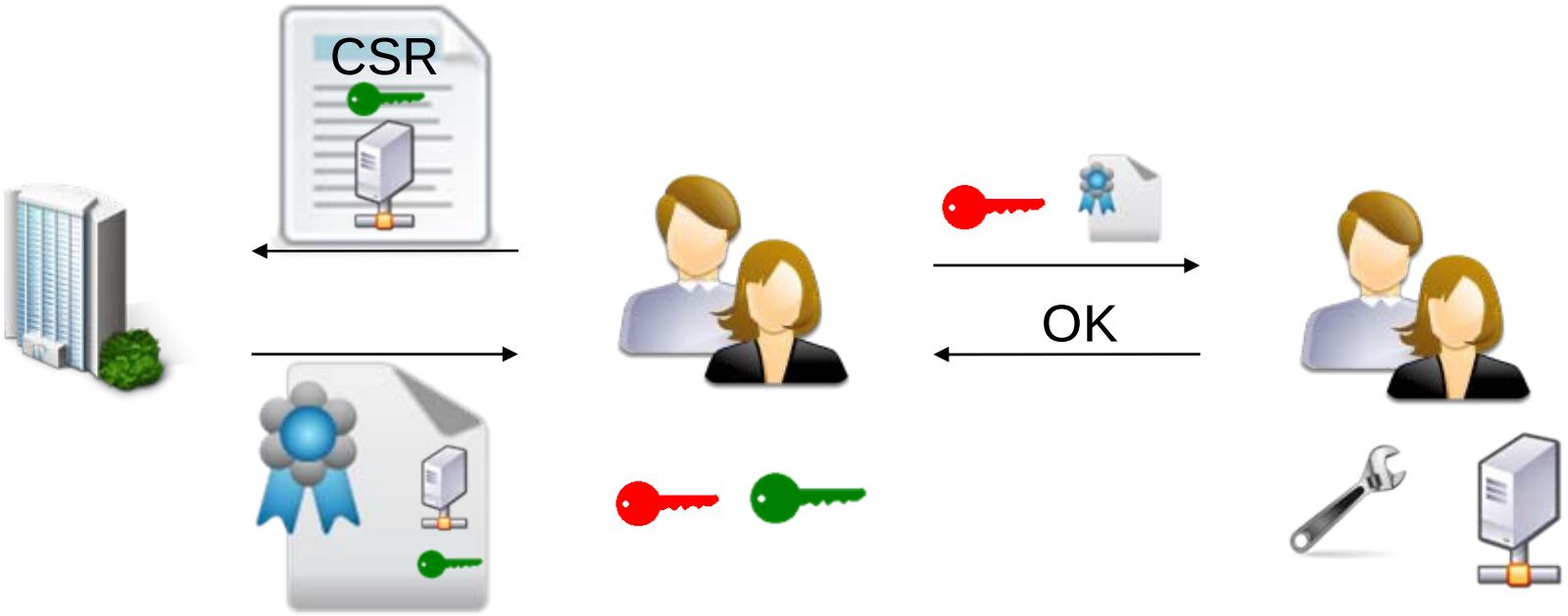
Méthodologie : processus

- Processus par nature complexe
- Intervenants :
  - Le service qui demande un nouveau certificat
  - Le service qui commande le certificat auprès d'un CA externe
  - Le CA externe
  - Le service qui installe le certificat

Demande d'un nouveau certificat



Renouvellement d'un certificat



Partie 1 : Certificats Digitaux

1. Introduction aux certificats
   - Cryptographie à clé publique
   - TLS / SSL
   - Certificats et PKI
2. Gestion de certificats
   - Méthodologie
   - Outils

Pourquoi un outil ?

- Découverte de certificats dans le réseau
  - In : Liste d'adresses IP, de ports
  - Out : Liste de certificats découverts

- Inventaire
  - Liste des certificats gérés, accès à leurs champs

- Monitoring
  - Vérifie à intervalles données la présence du certificat

Pourquoi un outil ?

- Notifications
  - Ex : Envoi d'email 6 semaines avant expiration
  - Envoi périodique d'un rapport pdf
- Automatisation de tâches :
  - Génération de CSR
  - Envoi de CSR au CA
  - Workflows
  - Installation des certificats

Outils : Aperçu du marché

- Trois outils commerciaux
  - Kousec Server Certificate Manager
  - Trustwave Certificate Lifecycle Manager
  - Venafi Encryption Director
- Tests de Kousec et Venafi

Kousec : Discovery

Discover Certificates - Port and Certificate Scan

Scan Set: scanset_14297 IP Set: ipset_196 Exec Status:Not Started Remaining time:

Parallelism: 1 Default HTTPS port only **Calc Expected Time** Start Scan

IP	10.1.0.1	**Open Ports**: - **Certificates**: -
IP	10.1.0.2	**Open Ports**: - **Certificates**: -
IP	10.1.0.3	**Open Ports**: - **Certificates**: -
IP	10.1.0.5	**Open Ports**: - **Certificates**: -
IP	10.1.0.6	**Open Ports**: - **Certificates**: -
IP	10.1.0.7	**Open Ports**: - **Certificates**: -
IP	10.1.0.110	**Open Ports**: - **Certificates**: -
IP	10.1.0.123	**Open Ports**: - **Certificates**: -

Kousec : Discovery

Kousec Server Certificate Manager

Application
User: admin
Logout
Settings

Manage Certificates
Cert Definitions
Cert Requests
Acquisition Processes
Certificates
Deploy Processes
CA Contracts
Trusted CAs
Monitor Control
Cert Discovery
Private CA
Private Keys
Sent Emails
Provider Accounts

Discover Certificates - IP Scan

IPSet: ipset_196 [Start IP Scan] Estimated : 90 seconds

IP Ranges	Address Count	Found Hosts
10.1.0.0-10.1.0.99	100	6
10.1.0.100-10.1.0.126 10.1.0.128-10.1.0.200	100	2
10.1.0.201-10.1.0.255	55	0

[NEXT: Ports and Certificates Scan] [Save Discovered Hosts as File]

Kousec Software

Kousec : Discovery

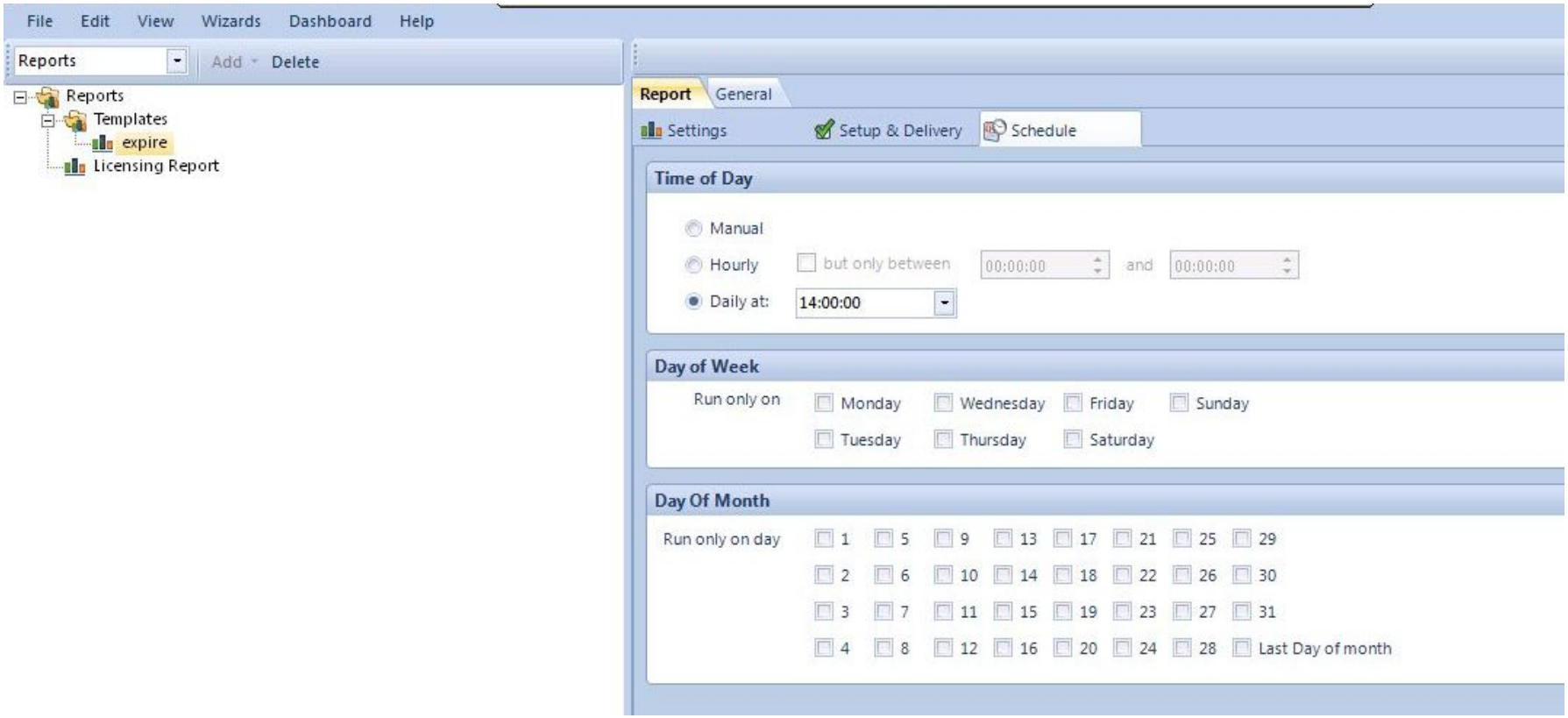
Discover Certificates - Analysis Done

Summary of Server Certificates Found

Out of 3 discovered certificates,
0 are valid,
3 are self-signed,
0 are untrusted,
0 are revoked,
0 are expired,
0 are expiring soon.

[Open Report in a new window] [Go to Start Page]

Venafi : Rapports périodiques



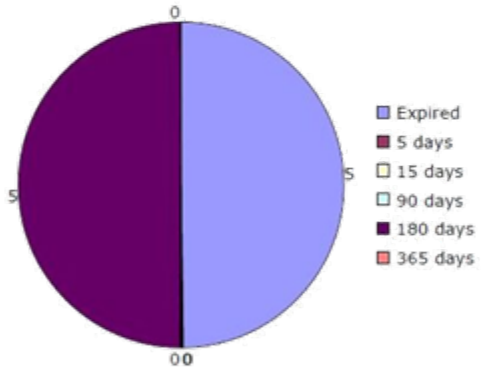
Venafi : Rapports périodiques

Expiration Summary Report — Venafi, Inc.

VENAFI

Report: Expiration Report
Prepared: Thursday, March 17, 2011

This is your certificate expiration summary report.



Generated by Venafi Encryption Director — Page 1 of 2

Outils : Avantages

- Évite les incidents
  - En avertissant à l'avance de l'expiration
- Facilite la communication entre les équipes
  - Gestion de workflows
- Gain de temps
  - Automatisation de tâches auparavant manuelles

Outils : Inconvénients

- Redondance avec autres outils
  - Workflow de l'outil / workflow ITIL
- Sécurité
  - Ouverture de flux
  - CSR générée avec l'outil = clé privée connue par l'outil
- Coût
  - Modèle de prix : x € / certificat
- Ne remplace pas la méthodologie

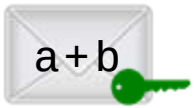
Conclusion Partie 1

- Certificats :
  - Process assez lourd
  - Très infréquent (durée de validité : 1,2,3 ans)
  - Doit être renouvelé à temps
- Éléments essentiels d'une bonne gestion :
  - Inventaire
  - Procédures précises et documentées
  - Communication interne
- Élément optionnel :
  - Outil dédié

Questions ?

Partie 2 : Alternatives au chiffrement à clé publique classique

- Motivations
- Identity-Based Encryption 

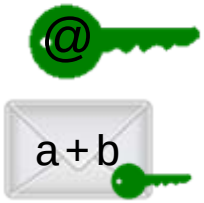
- Homomorphic Encryption 

- Threshold Encryption 
- Conclusion

Partie 2 : Alternatives au chiffrement à clé publique classique

- Motivations
- Identity-Based Encryption
- Homomorphic Encryption
- Threshold Encryption
- Conclusion

Motivations

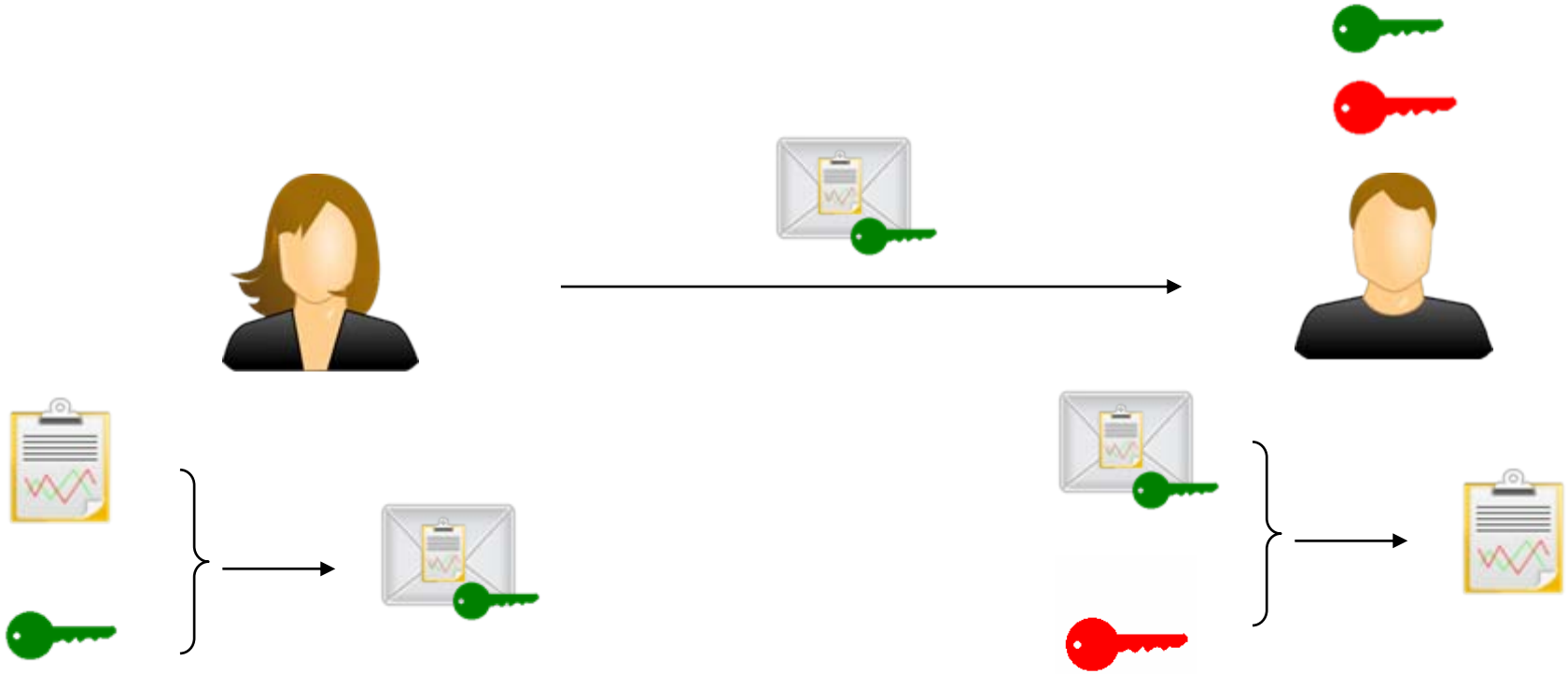


- Beaucoup d'activité dans le monde de la recherche en cryptographie
  - Nombreux articles sur des chiffrements "alternatifs"
  - Concepts théoriques ou opportunité ?
- Demande à la section Recherches
  - Concevoir un système sécurisé d'accès à des données confidentielles
  - Exigences fortes
  - Un système de chiffrement classique ne convenait pas

Chiffrement à clé publique classique

- Canal peu sûr
- Problème de confidentialité
- Pas de secret partagé entre Alice et Bob

Chiffrement Alice vers Bob



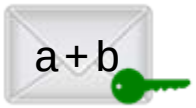
Contraintes et limites du chiffrement classique

- Voir Partie 1 :
  - Lourdeur de la gestion de certificats
- Il existe des variantes qui permettent
  - D'effectuer des calculs sur des données chiffrées
  - De chiffrer des messages pour un groupe de personnes

  et encore beaucoup d'autres...

Identity-Based Encryption

Homomorphic Encryption



Threshold Encryption

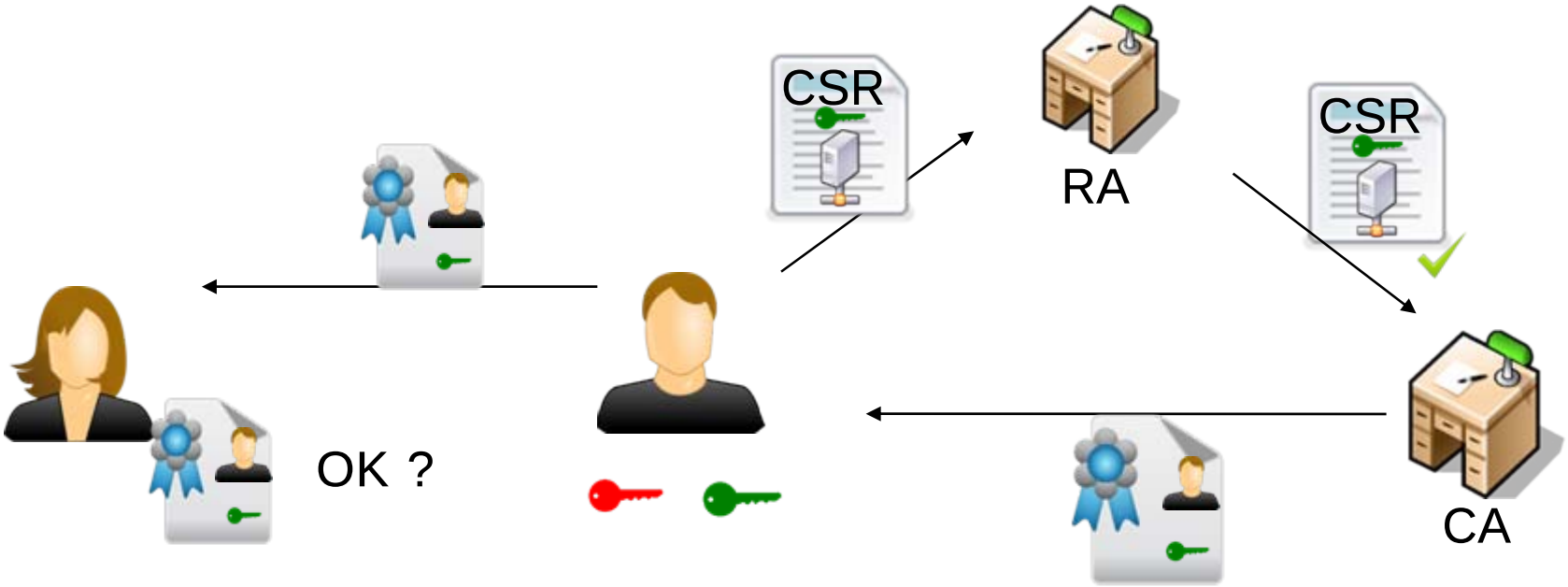
Partie 2 : Alternatives au chiffrement à clé publique classique

- Motivations
- Identity-Based Encryption 

- Homomorphic Encryption
- Threshold Encryption
- Conclusion

Chiffrement classique avec PKI

Alice veut envoyer un message chiffré à Bob



Une idée pour éviter les certificats

- Le certificat sert à attester du lien entre

1. Une entité (personne morale, personne physique, serveur, application…)

et

2. Une clé publique

- Idée :  = l'identité de Bob !

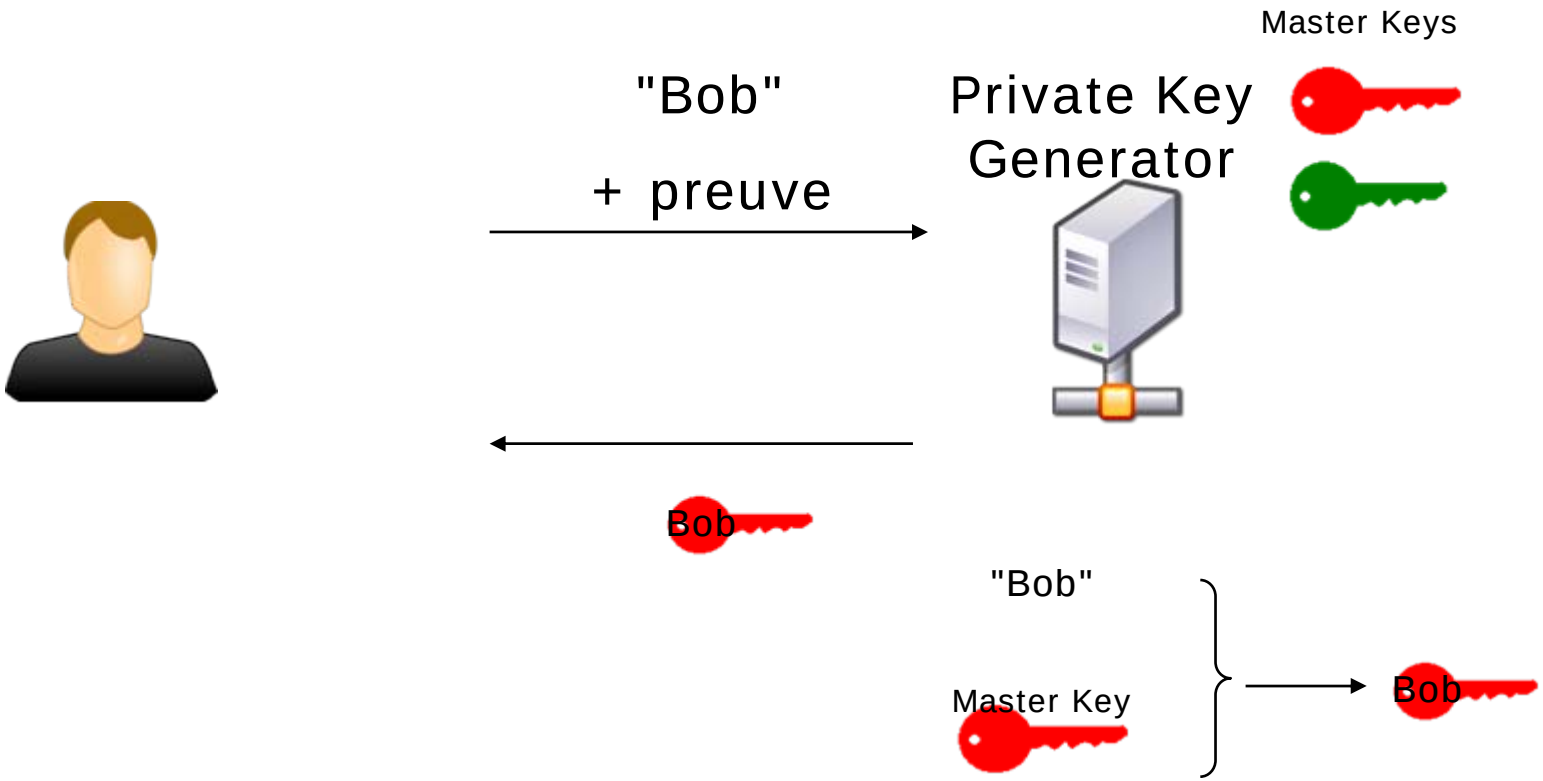
Identity-Based Encryption (IBE)

- Idée principale
  - Utiliser n'importe quelle chaîne de caractères comme clé publique
  - Exemple :
    - Chaîne de caractères qui représente l'*identité* d'une personne
    - Adresse e-mail

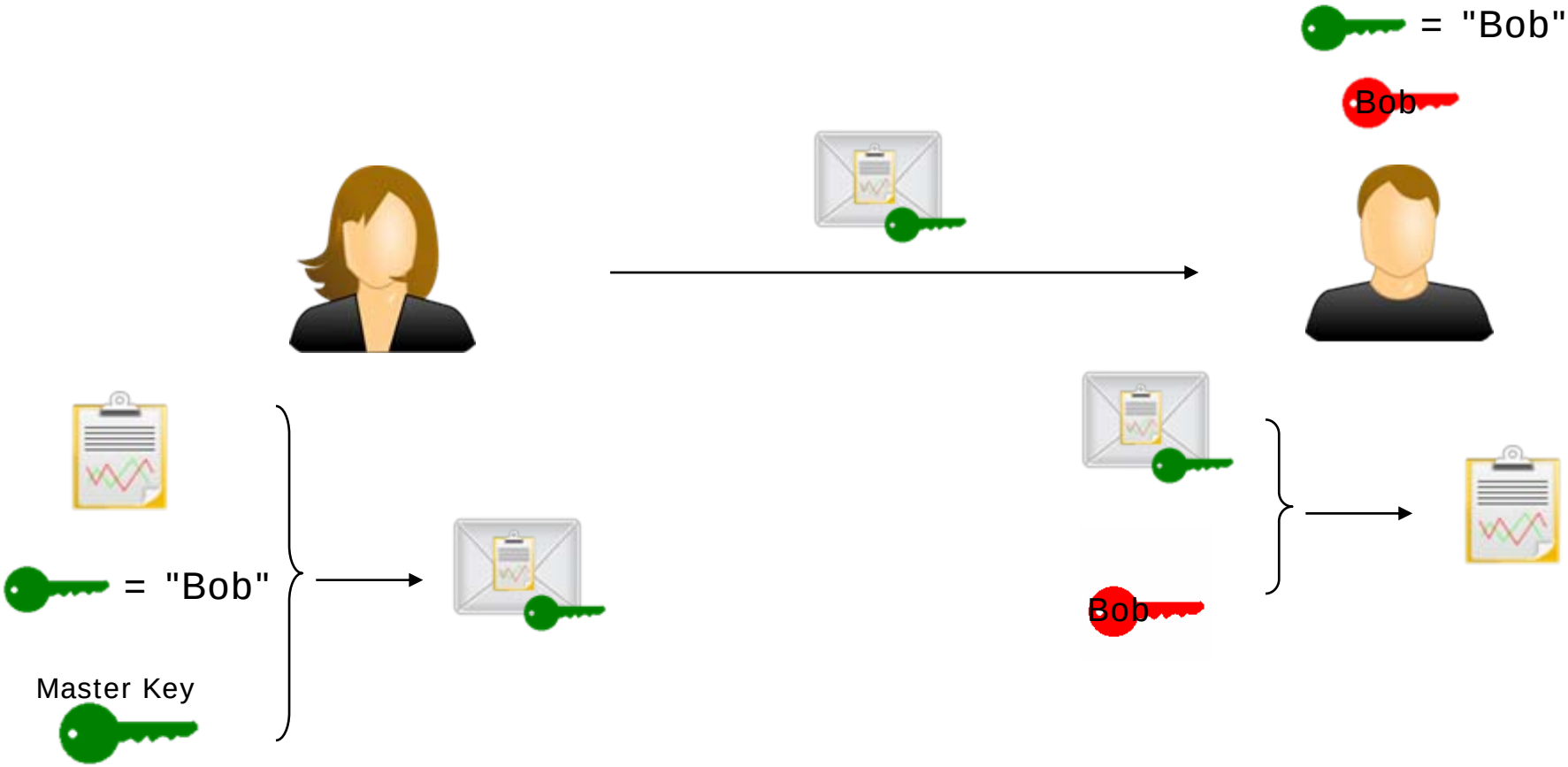
      = julien.cathalo@smals.be

    - Plus besoin de certificat

IBE : Bob s'inscrit dans le système



IBE : Alice envoie un message à Bob



IBE : Un exemple (mail Voltage)

- Bob s'inscrit via une interface web
- Sa clé publique : "bob@mail.com"
- Reçoit un mail avec un lien
- Le PKG génère la clé privée de Bob
- Une page web s'ouvre (TLS/SSL)
  et contient
- Alice utilise le logiciel Voltage pour chiffrer son document pour Bob
- Bob utilise le logiciel Voltage pour déchiffrer le document avec sa clé

PKI vs. IBE

PKI

- Alice et Bob doivent faire confiance au CA
- Alice doit obtenir 
- Bob doit être inscrit au préalable

IBE

- Alice et Bob doivent faire confiance au PKG
- Alice connaît implicitement 
- Bob peut s'inscrire après l'envoi du message

PKI vs. IBE

PKI

- Bob peut générer
- Le CA ne connaît pas
- Le CA ne peut pas lire les messages chiffrés par Alice
- Bob doit demander au CA

IBE

- Le PKG connaît
- Le PKG peut lire les messages chiffrés par Alice
- Le PKG doit transmettre de manière sécurisée à Bob

IBE : Quelle maturité ?

- Recherche :
  - Idée (théorique) : 1984 (Shamir)
  - Solution efficace : 2001 (Boneh-Franklin)
  - Nombreuses publications
- Produits :
  - Voltage Security
  - Trend Micro Encryption

IBE : Quelle maturité ?

- Standards :
  - IEEE : Projet P1363 "Standard Specifications For Public-Key Cryptography"
  - IETF : 3 RFCs
    - RFC 5091 (2007) : aspects mathématiques
    - RFC 5408 (2009) : algorithmes, formats de données
    - RFC 5409 (2009) : standards e-mail
- Librairies
  - C : PBC (Pairing-Based Crypto)
  - Java : jPBC

IBE : Opinion

- Le problème des certificats n'est pas réglé mais déplacé
- Gestion simplifiée pour Alice
- Trop risqué dans beaucoup de cas
  - Besoin d'une confiance absolue dans le PKG
- Applications prometteuses :
  1. Solution IBE pour email
  2. Composant d'architectures de sécurité

Solution IBE pour email

- Exemple :
  - PKG hébergé chez Smals
  - Chaque membre du personnel de Smals peut demander une clé privée
  - Une personne externe peut utiliser le système pour chiffrer des mails pour un membre du personnel
  - Simplement besoin d'un logiciel
  - Protège contre les attaques externes à Smals

IBE comme composant d'architectures de sécurité

- Certaines architectures utilisent un séquestre de clés (key escrow) :
  - Base de données qui contient des clés
  - Exemple : clés symétriques de déchiffrement

User	Key
Alice	2b7e151628aed2a6abf7158809cf4f3c
Bob	6bc1bee22e409f96e93d7e117393172a
Charlie	ae2d8a571e03ac9c9eb76fac45af8e51

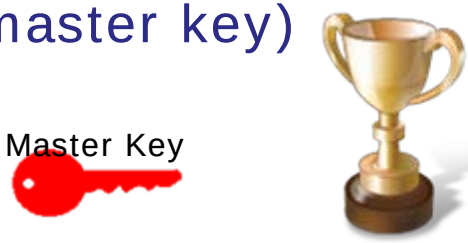
Key Escrow vs. IBE

Key Escrow

- L'admin du Key Escrow peut déchiffrer tous les messages
- Lourde gestion (une clé par utilisateur)

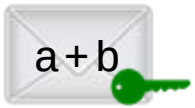
IBE

- L'admin du PKG peut déchiffrer tous les messages
- Simple gestion (une seule master key)



Partie 2 : Alternatives au chiffrement à clé publique classique

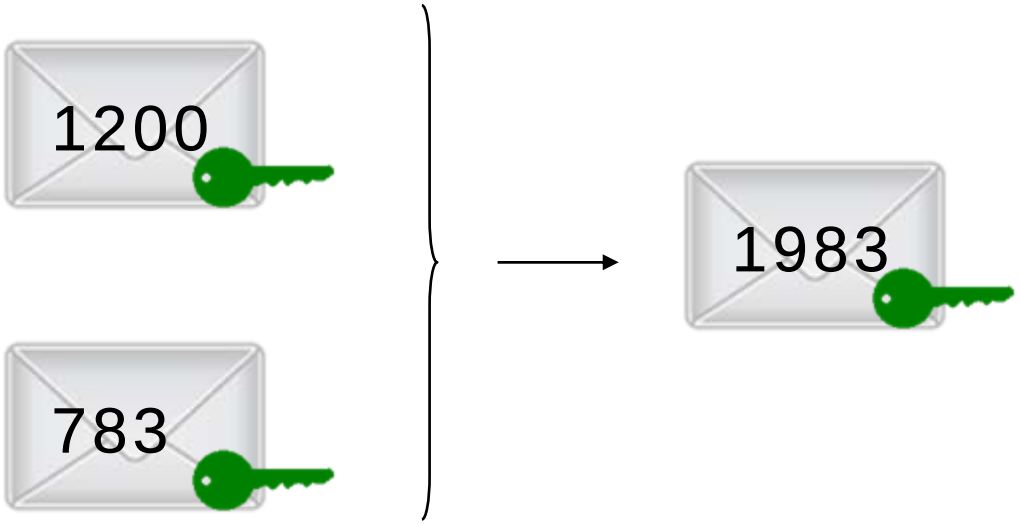
- Motivations
- Identity-Based Encryption
- Homomorphic Encryption



- Threshold Encryption
- Conclusion

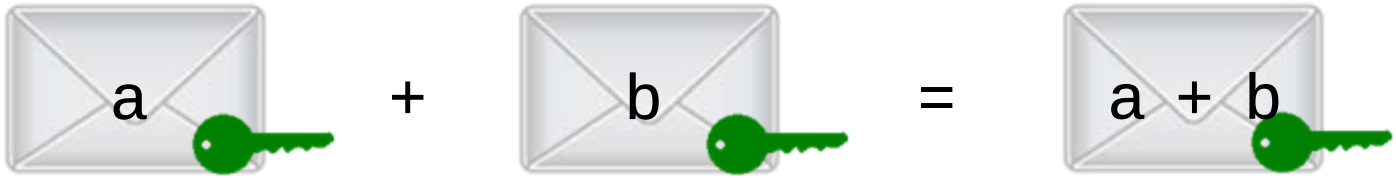
Homomorphic Encryption

- Faire des maths avec des données chiffrées ?



Homomorphic Encryption

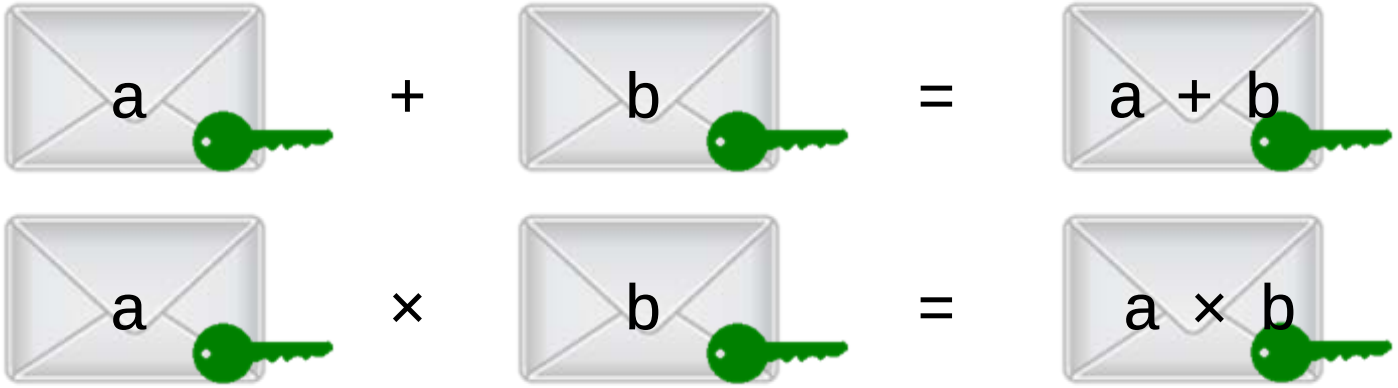
- Connu depuis les années 70/80
- On peut faire une opération sur les données chiffrées



- De nombreux algorithmes existent

Fully Homomorphic Encryption

- On peut faire deux opérations



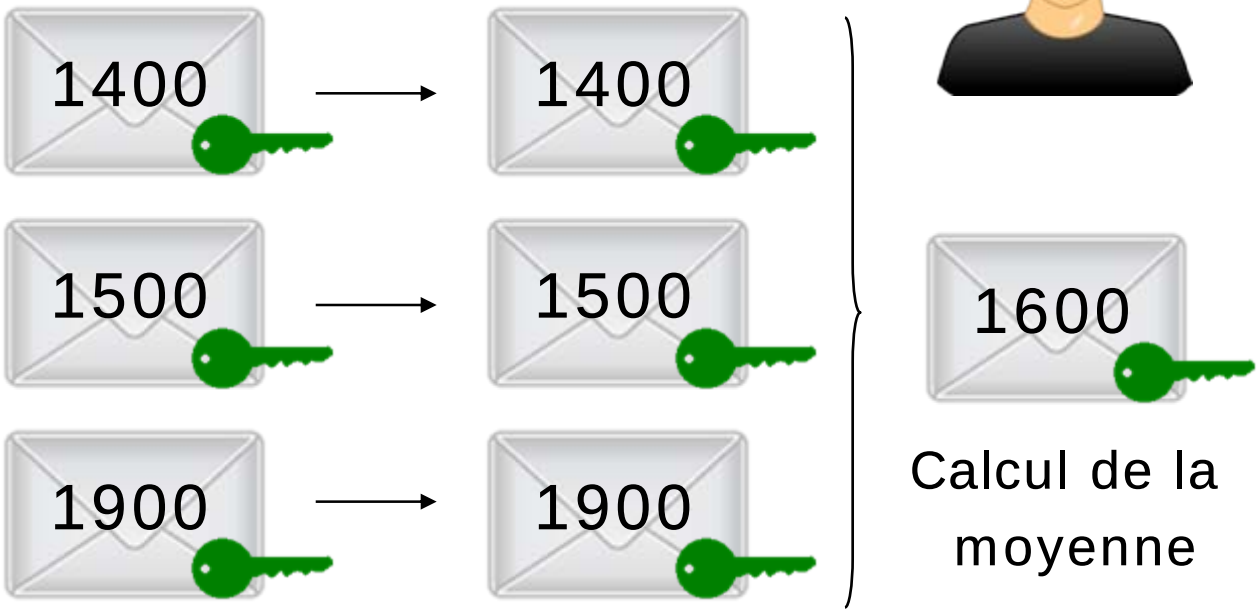
- Publication : 2009, Craig Gentry (IBM)
- Première implémentation : 2011

Fully Homomorphic Encryption : Applications

- Permet de déléguer à un tiers des opérations sur des données confidentielles
- Applications :
  - Calcul dans le cloud
  - Outsourcer des opérations
    - Comptabilité

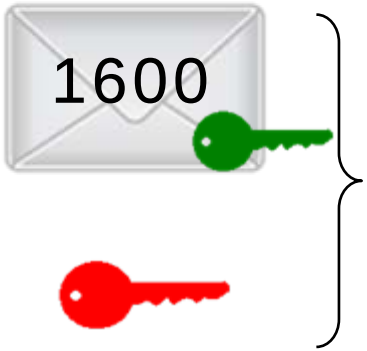
Fully Homomorphic Encryption : Applications

Employé	Salaire
Employé 1	1400€
Employé 2	1500€
Employé 3	1900€



Fully Homomorphic Encryption : Applications

Employé	Salaire
Employé 1	1400€
Employé 2	1500€
Employé 3	1900€



1600

Salaire Moyen	1600€

Fully Homomorphic Encryption : Opinion

- Concept prometteur
- Pas encore de maturité
- IBM communique beaucoup
- Beaucoup de contraintes techniques
  - Algorithmes complexes
  - Taille des paramètres

Partie 2 : Alternatives au chiffrement à clé publique classique

- Motivations
- Identity-Based Encryption
- Homomorphic Encryption
- Threshold Encryption 
- Conclusion

Motivation

- Demande à la section Recherches
- Concevoir un "coffre-fort"
- Business Case :
  - Dossiers médicaux de patients
  - Accès par acteurs de soins de santé
  - Fortes exigences de sécurité

Motivation

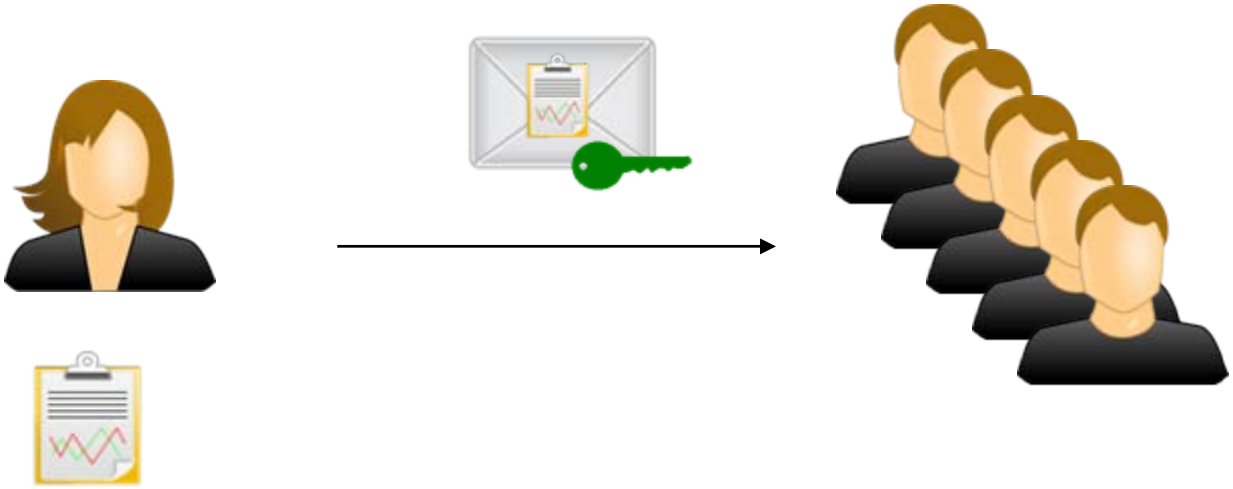
- Exigences de sécurité :
  - Contrôle d'accès à forte granularité
  - Chiffrement non adressé (pas de destinataire connu)
  - Aucune donnée qui transite en clair
  - Répartir la confiance entre plusieurs TTPs (Trusted Third Parties)
- Chiffrement PKI classique ?
  - Chiffrement adressé
  - Un seul TTP
  - Ne convient pas

Motivation

- Solution : Threshold Encryption
  - Inventé par Yvo Desmedt et Yair Frankel en 1989
- Travail de la section Recherches
  - Proposer un système pour le coffre-fort basé sur Threshold Encryption
    - Système générique (pas uniquement données médicales)
    - Répond aux exigences de sécurité
    - Permet le stockage dans le cloud
  - Étude de faisabilité et de performances

Threshold Encryption

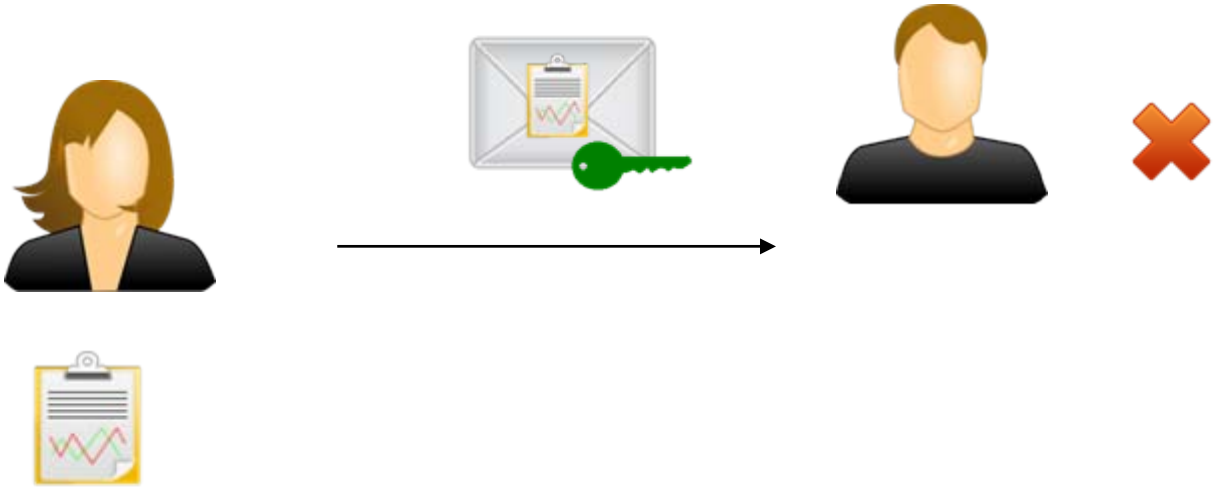
- Alice envoie un document à 5 personnes



Seuil = 3

Threshold Encryption

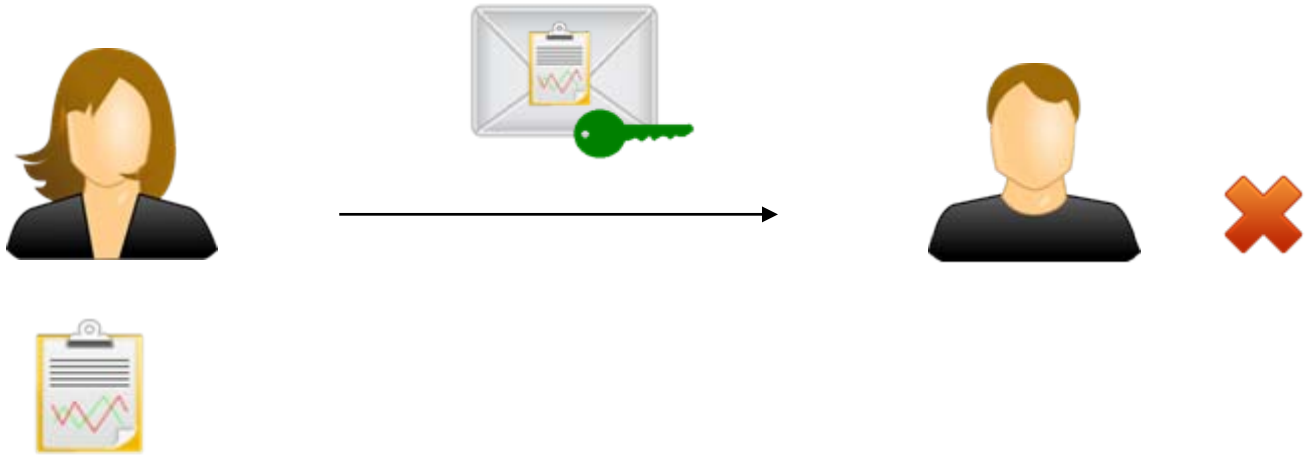
- 1 destinataire seul ne peut pas déchiffrer



Seuil = 3

Threshold Encryption

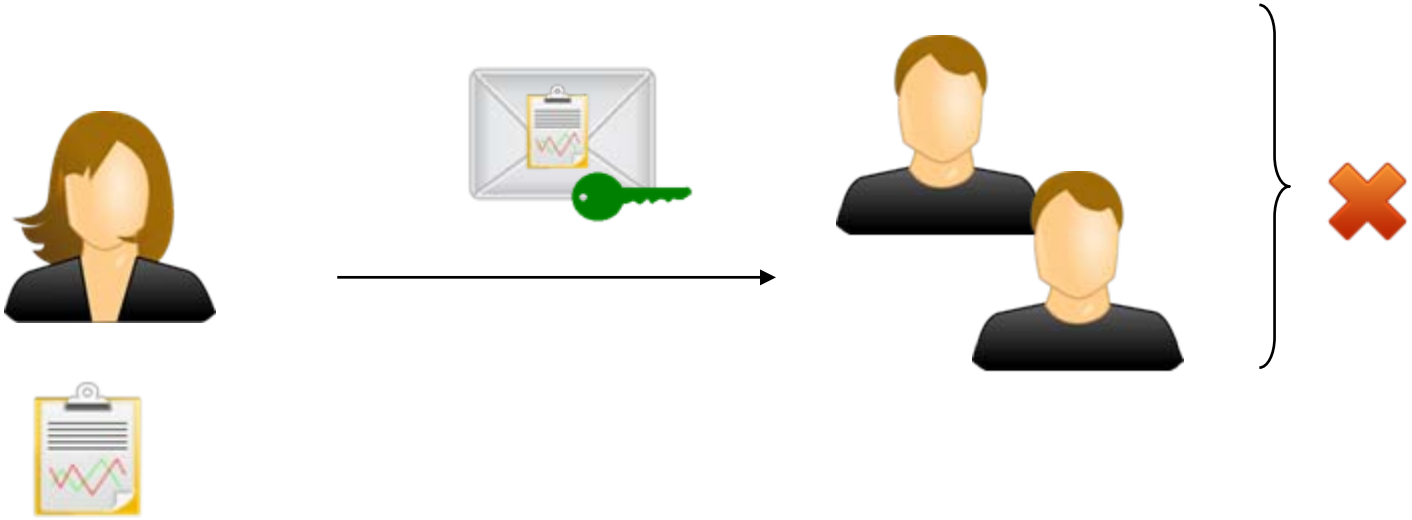
- 1 destinataire seul ne peut pas déchiffrer



Seuil = 3

Threshold Encryption

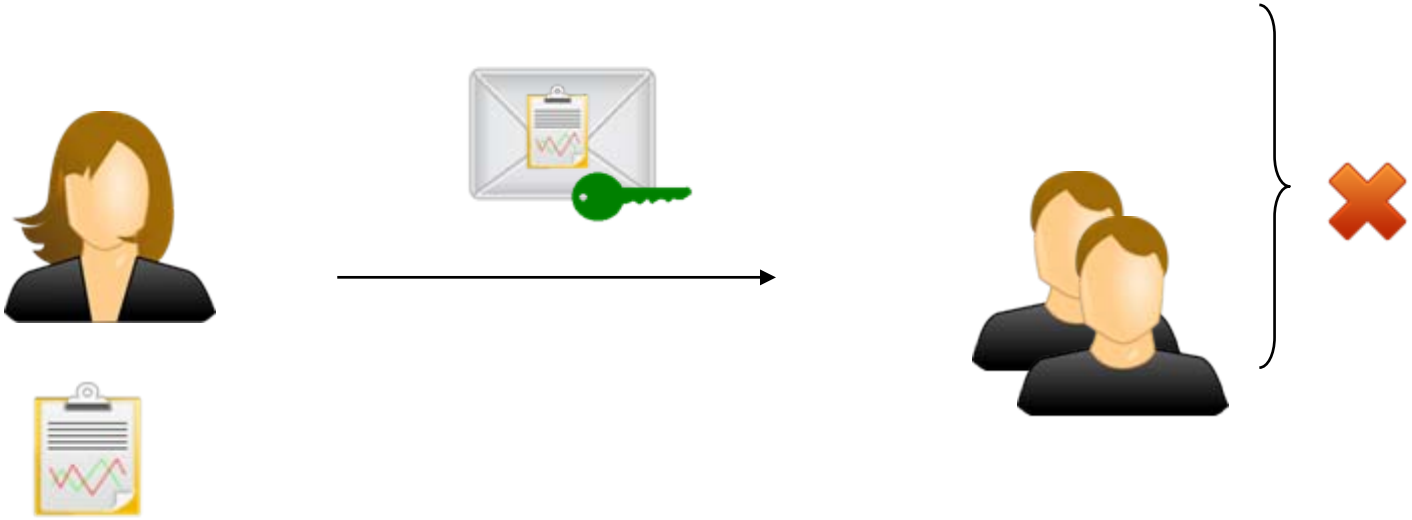
- 2 destinataires ensemble ne peuvent pas déchiffrer



Seuil = 3

Threshold Encryption

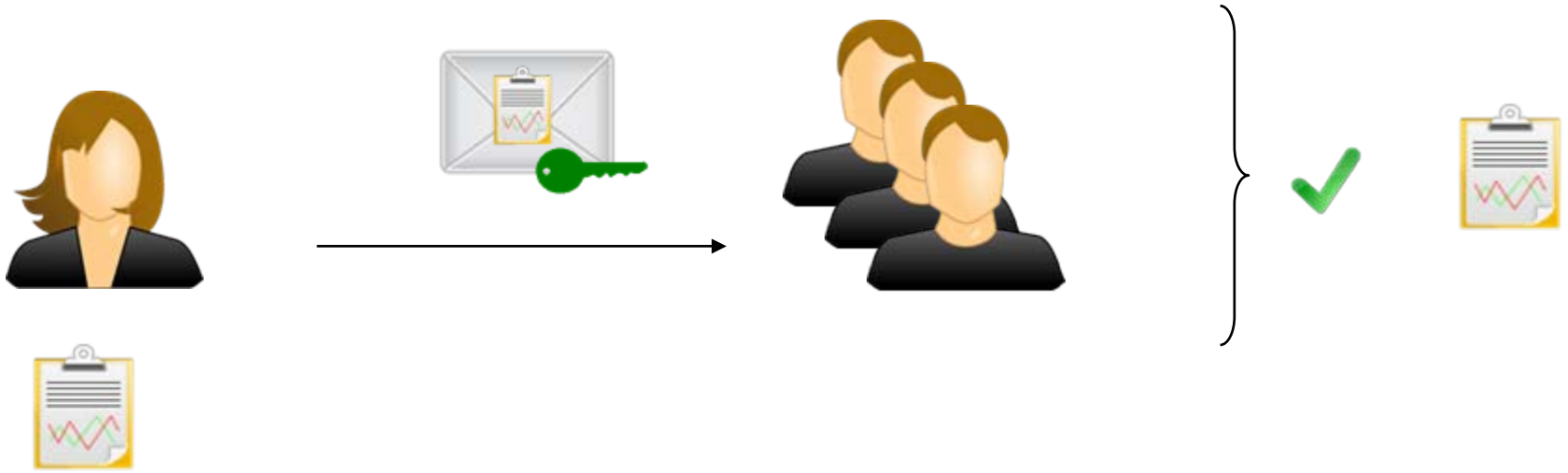
- 2 destinataires ensemble ne peuvent pas déchiffrer



Seuil = 3

Threshold Encryption

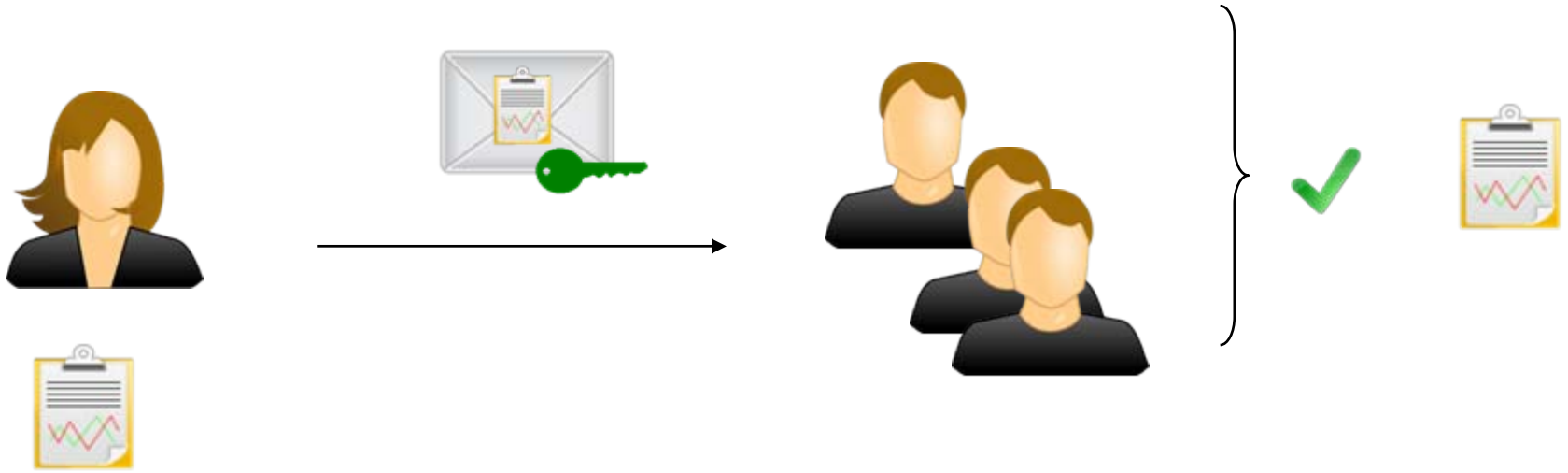
- 3 destinataires ensemble peuvent déchiffrer



Seuil = 3

Threshold Encryption

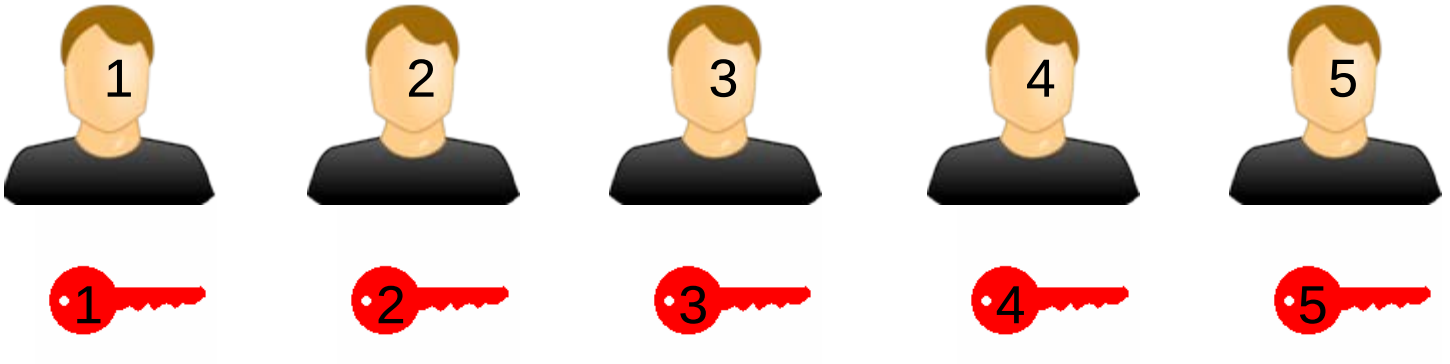
- 3 destinataires ensemble peuvent déchiffrer



Seuil = 3

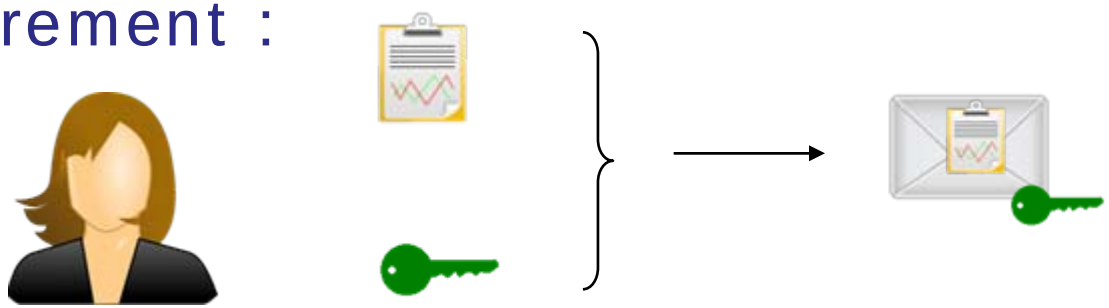
Threshold Encryption : clés

- Une seule clé publique 
- Une clé privée par utilisateur

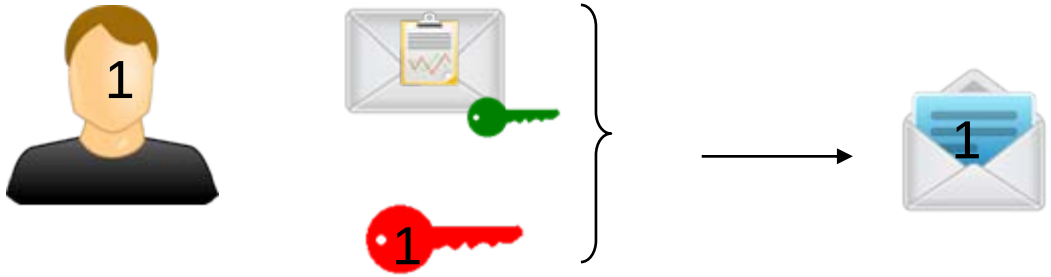


Threshold Encryption

- Chiffrement :

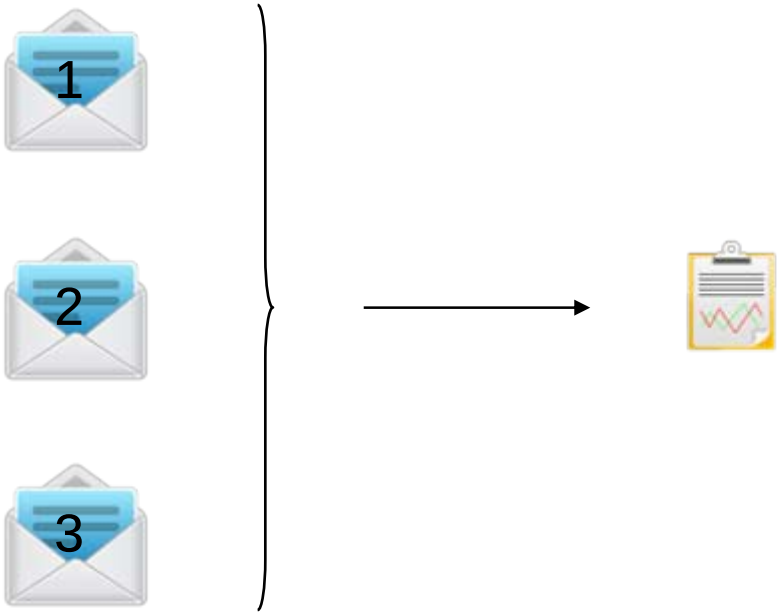


- Déchiffrement partiel :



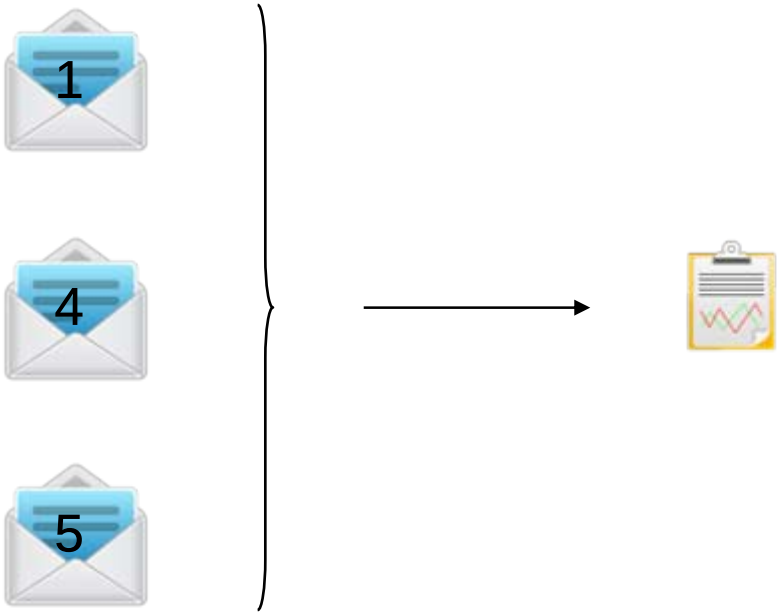
Threshold Encryption

- Combinaison de déchiffrements partiels :



Threshold Encryption

- Combinaison de déchiffrements partiels :



Pas besoin
de clé...

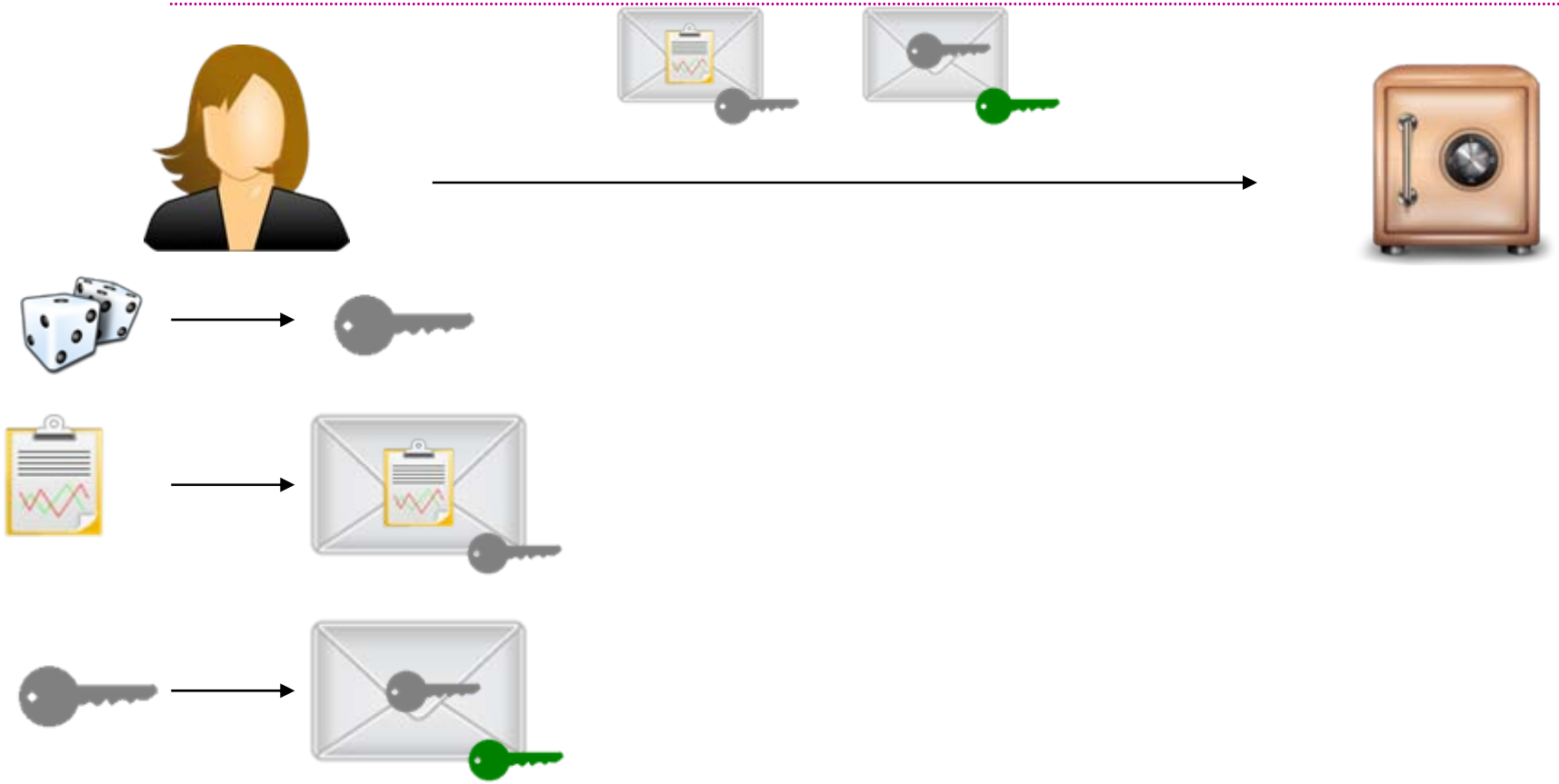
Coffre-fort

- Système sécurisé proposé par Smals qui permet
  - Écriture
  - Lecture
  - Stockage

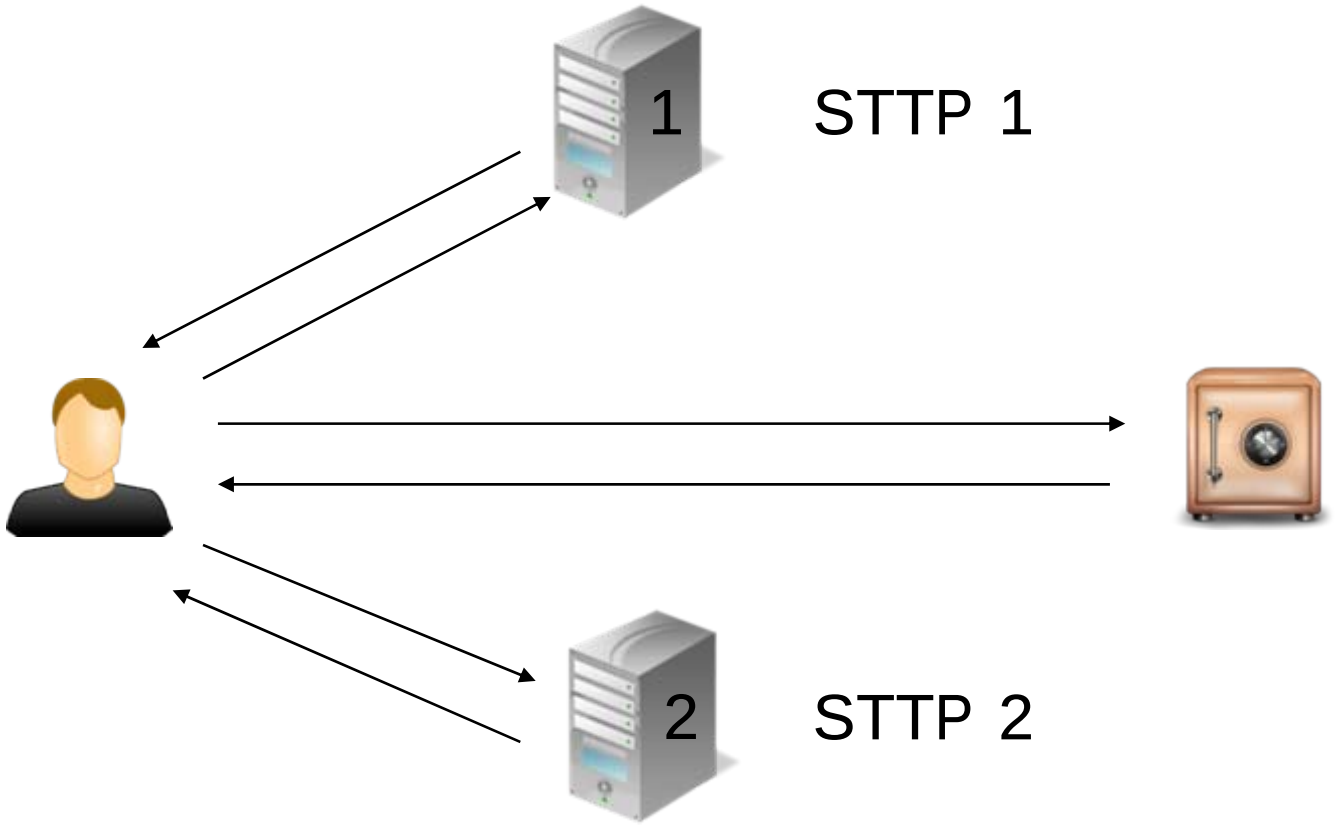
  dans une base de données

- Deux Semi-Trusted Third Parties (STTP)
- Réduit l'impact d'un TTP compromis
- Utilise Threshold Encryption avec un seuil de 2

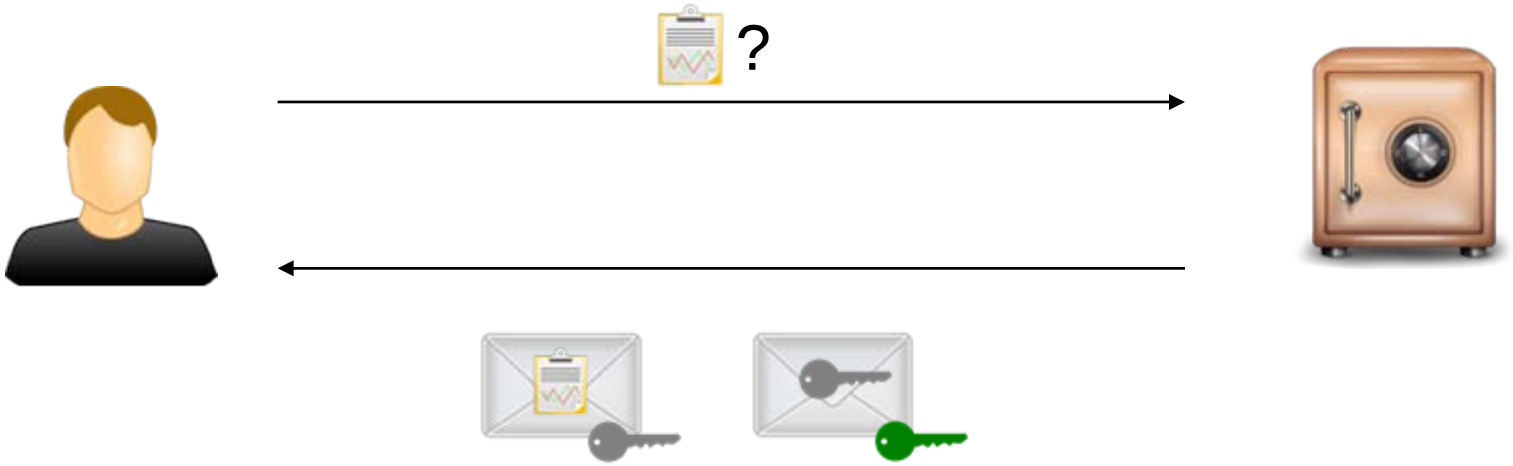
Coffre-fort : écriture



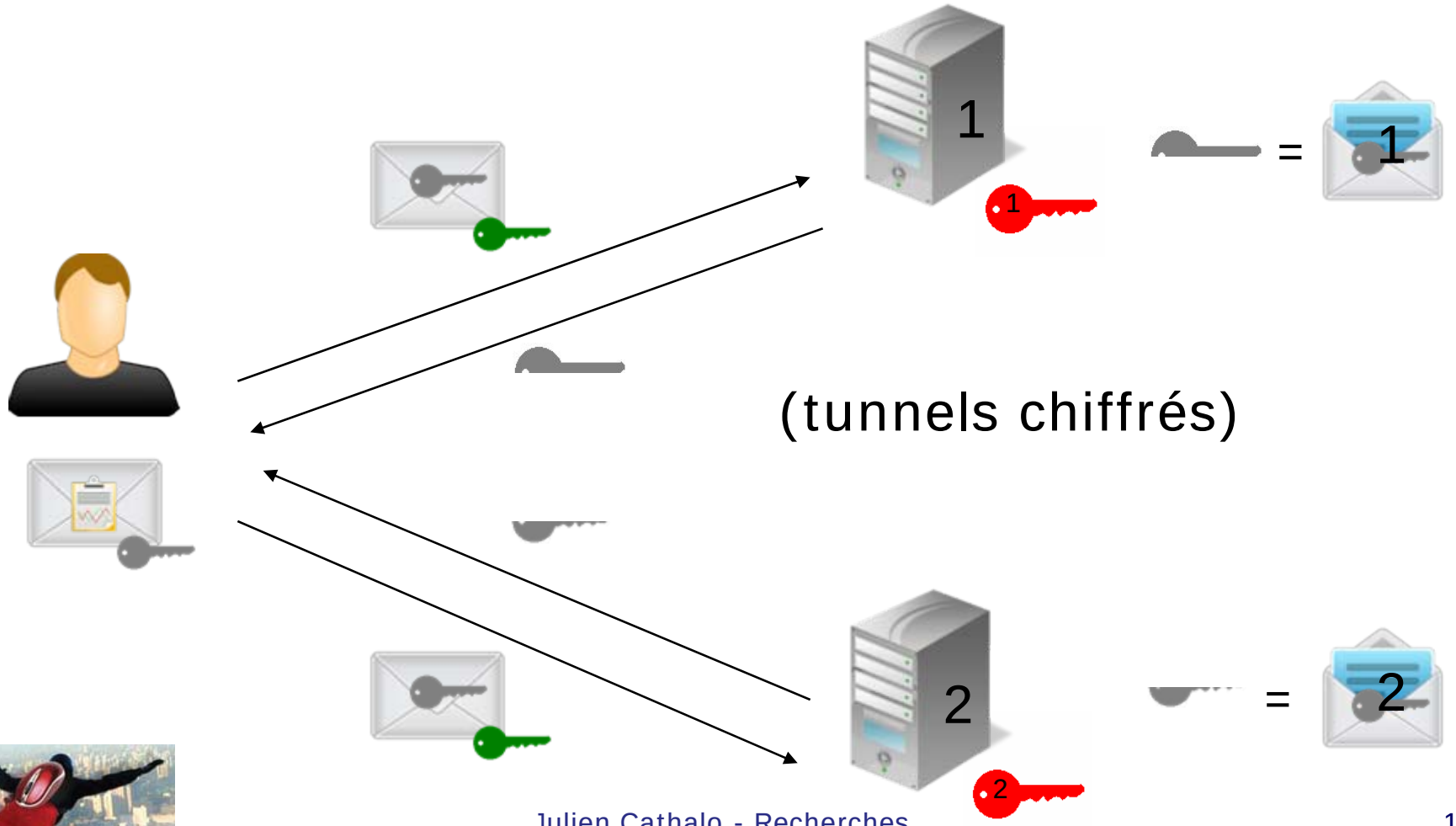
Coffre-fort : lecture (vue globale)



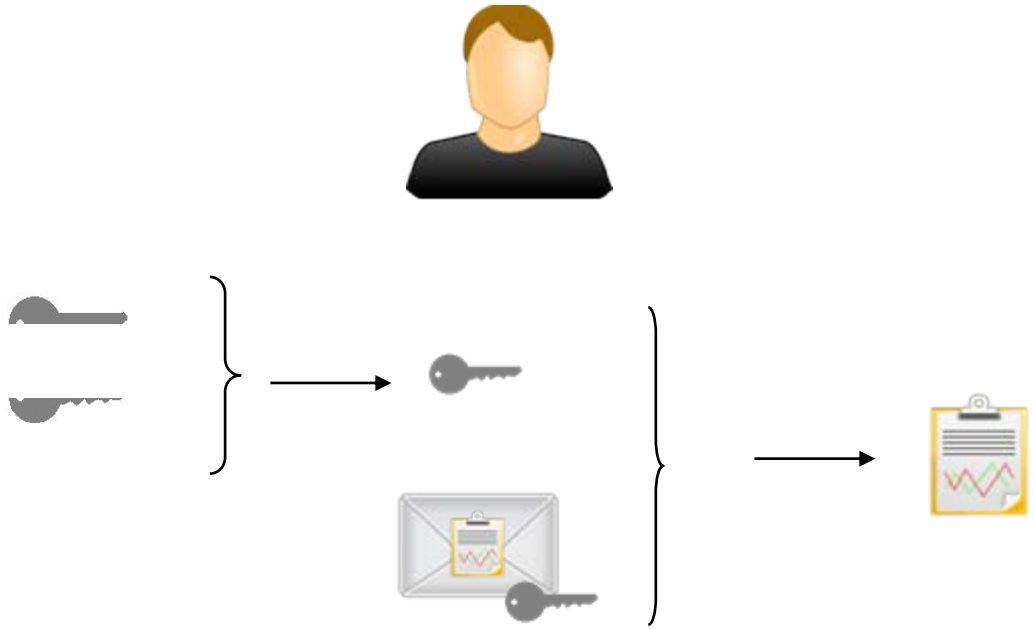
Coffre-fort : lecture (phase 1)



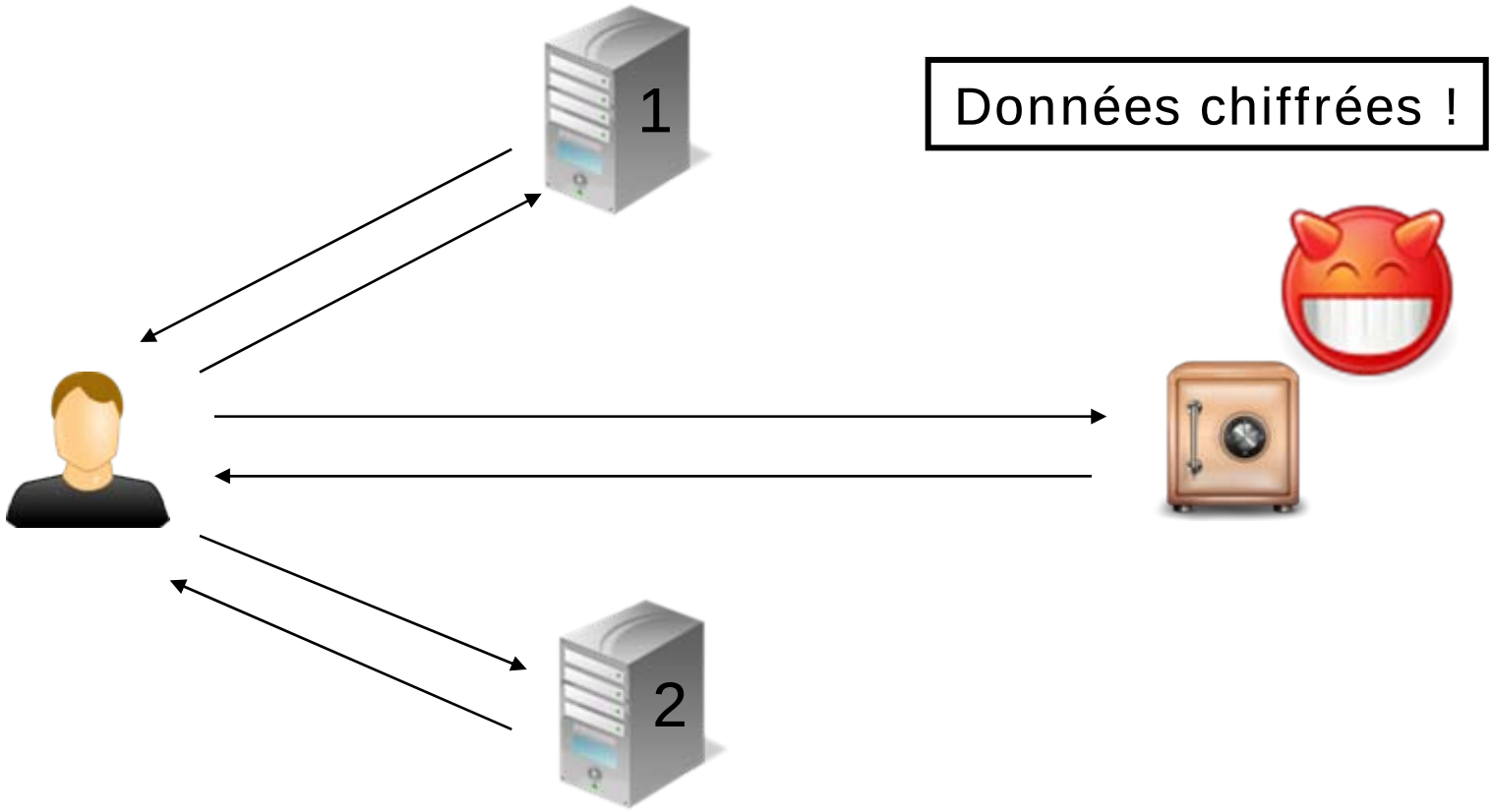
Coffre-fort : lecture (phase 2)



Coffre-fort : lecture (phase 3)

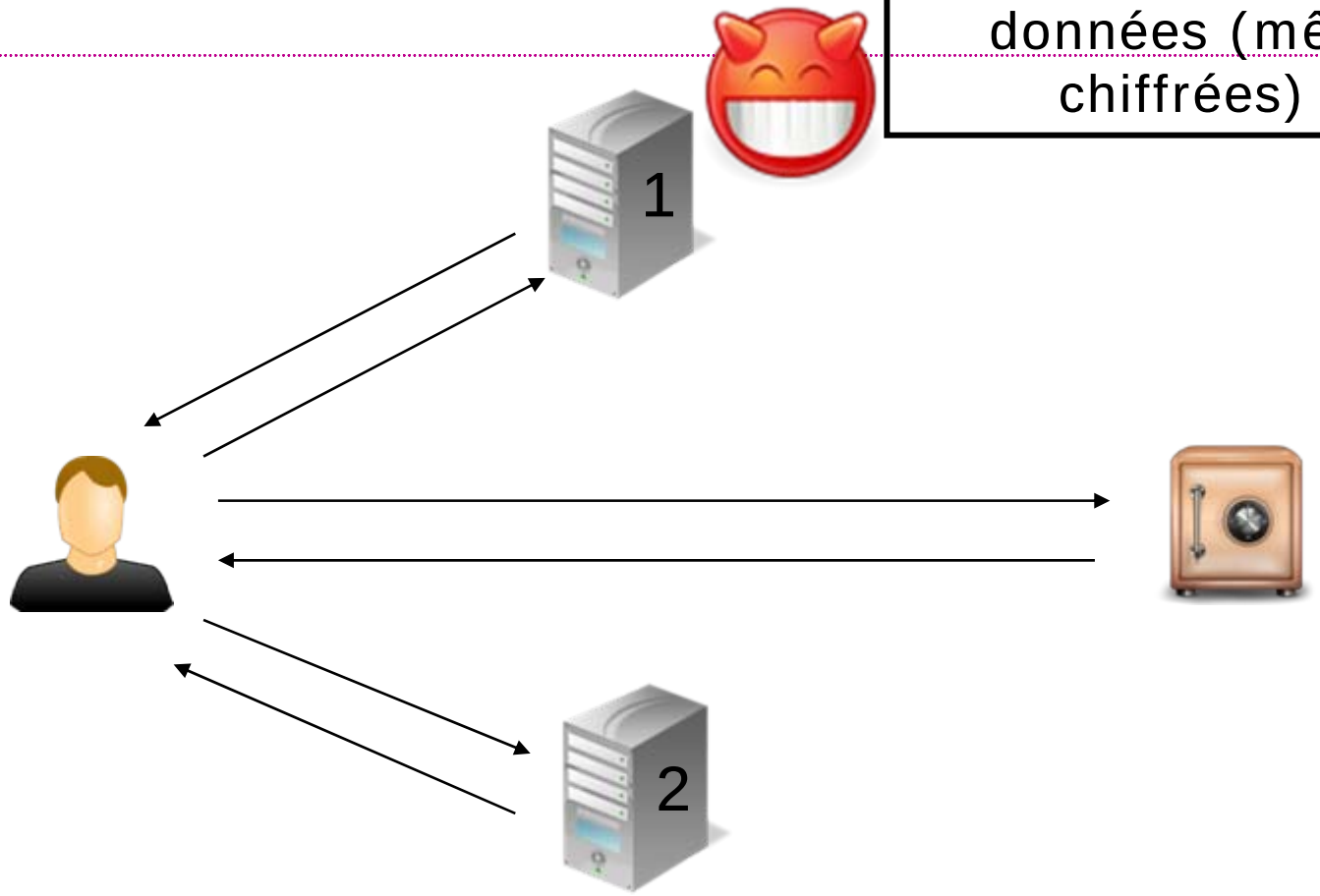


Sécurité du système (1)

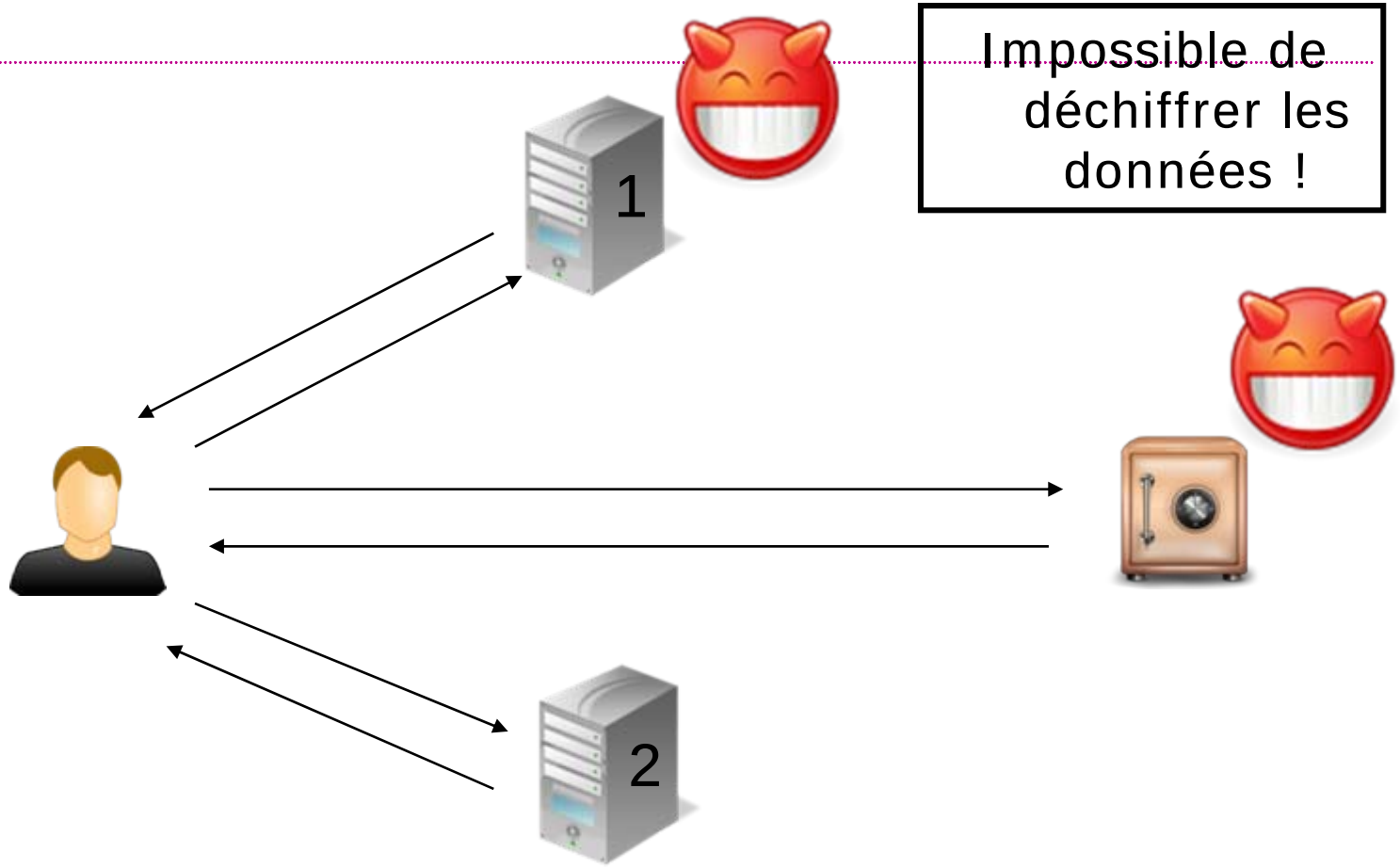


Sécurité du système (2)

Aucun accès aux données (même chiffrées) !



Sécurité du système (3)



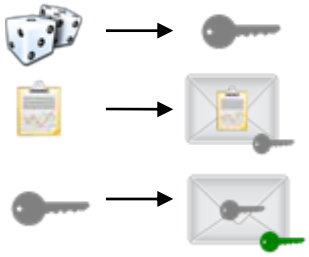
Sécurité du système (4)

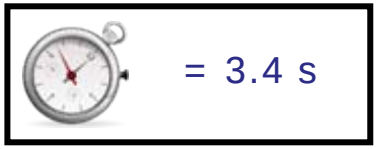
- Points d'attention
  - User and Access Management
  - Protection des clés des serveurs de déchiffrement partiel
- On peut aussi utiliser plus de deux serveurs
  - Meilleure sécurité
  - Meilleure disponibilité

Section Recherches : Étude de faisabilité

- 5.50 Go de données
  - 50000 dossiers
  - 100 champs par dossier en moyenne
- Chaque champ chiffré individuellement
- STTP : amazon web services — ZIRCON COMPUTING
- Storage Engine : notre labo
  - 4 serveurs (Core i7)
  - 1 loadbalancer

Performances : Ecriture



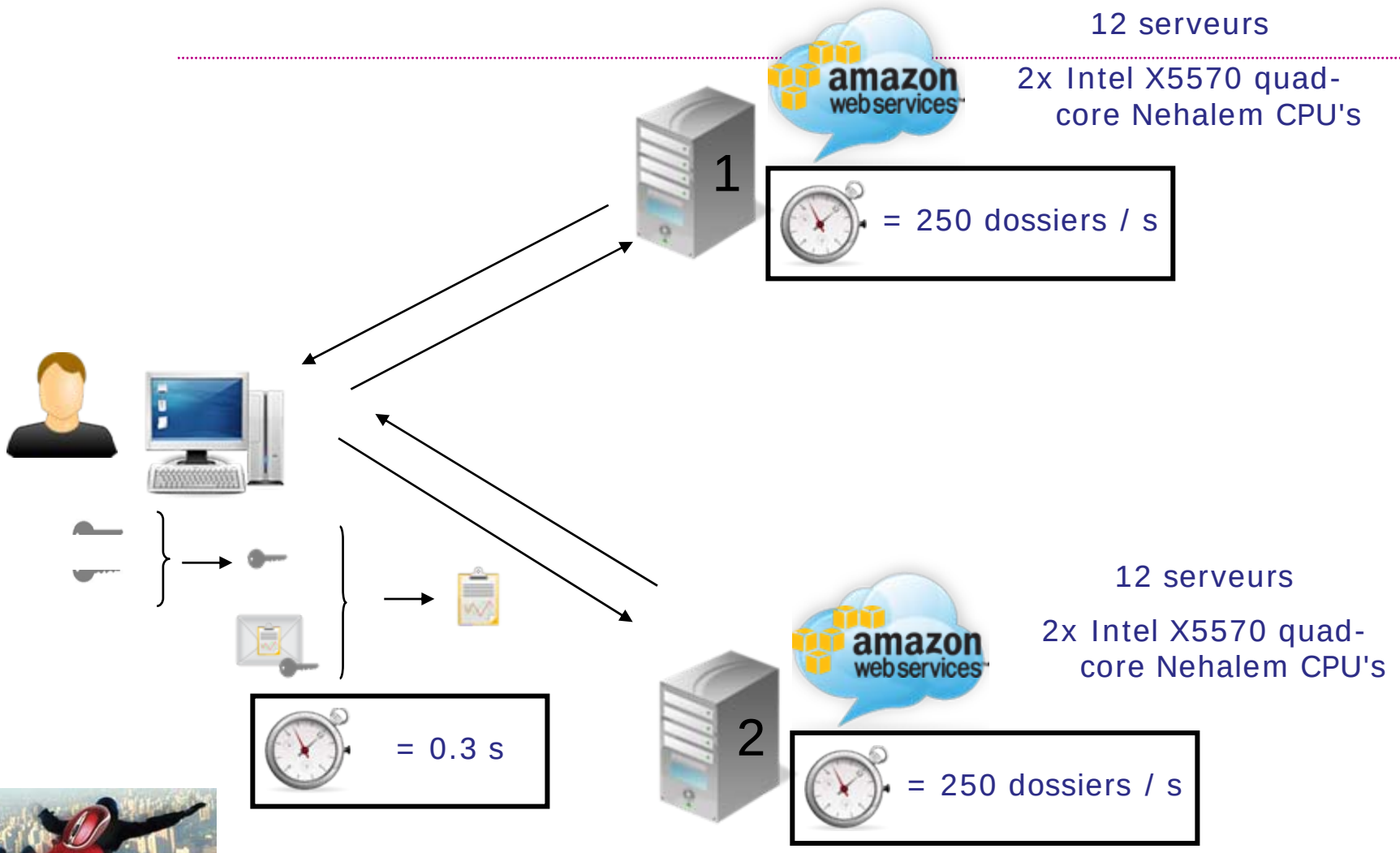


Labo : Storage Engine

4 serveurs (Core i7)

1 loadbalancer

Performances : Lecture



Conclusion

- Threshold encryption : inventé par Yvo Desmedt et Yair Frankel en 1989
- Solution proposée :
  - Utilise threshold encryption
  - Générique
  - Très différente des approches classiques pour database encryption
  - Répond aux exigences de sécurité
  - Permet le stockage dans le cloud
  - Innovante
- Mars 2011 : demande de brevet européen par Smals

Partie 2 : Alternatives au chiffrement à clé publique classique

- Motivations
- Identity-Based Encryption
- Homomorphic Encryption
- Threshold Encryption
- Conclusion

Conclusion Partie 2

- Il existe des alternatives au chiffrement classique
- Avantages :
  - Souplesse
  - Permettent de nouvelles applications
- Inconvénients :
  - Moins de standards : plus d'effort de développement
  - Moins de travail de validation : moins de garanties de sécurité

Questions ?

N'oubliez pas de remplir le formulaire d'évaluation !