

Rencontres Mondiales du Logiciel Libre 2010

Bordeaux

8 juillet 2010

Partie 2:

**Architecture pour la diffusion d'informations
personnelles**

Mikaël Ates

mates@entrouvert.com



- Enjeux
- Esquisse de l'architecture pour les échanges inter-organisations et la présentation de données personnelles
- Quelques fonctionnalités particulières
- Hébergement de données personnelles et de fonctionnalités
- Conclusion

- Enjeux
- Esquisse de l'architecture pour les échanges inter-organisations et la présentation de données personnelles
- Quelques fonctionnalités particulières
- Hébergement de données personnelles et de fonctionnalités
- Conclusion

- Diffusion d'informations personnelles omniprésente lors de la consommation de services en ligne:
 - e-commerce, réseaux sociaux, etc.
- Informations révélées par l'utilisateur :
 - "Self-asserted" : réseaux sociaux, forums, formulaires en ligne.
 - "Certifiées" : Lors de "procédures administratives" en ligne.
 - "Traces" : adresses IP, recherches sur des moteurs.

- Maîtriser : Permettre à l'utilisateur lors de procédures en ligne de diffuser de lui-même des informations qui sont aujourd'hui diffusées directement entre organisation
- Réduire : Ne diffuser que la stricte information "nécessaire"
- Contrôler :
 - Audit et journalisation
 - Contrôle d'usage



- Concevoir une architecture offrant à l'utilisateur la maîtrise de la diffusion d'informations personnelles
 - Certifiées et self-asserted
- Automatiser cette diffusion



- Obtenir des informations de confiance sur les interlocuteurs :
 - Des tiers de confiance de l'utilisateur
- Un système ergonomique
 - Authentifications multiples de l'utilisateur
 - Une interface unique

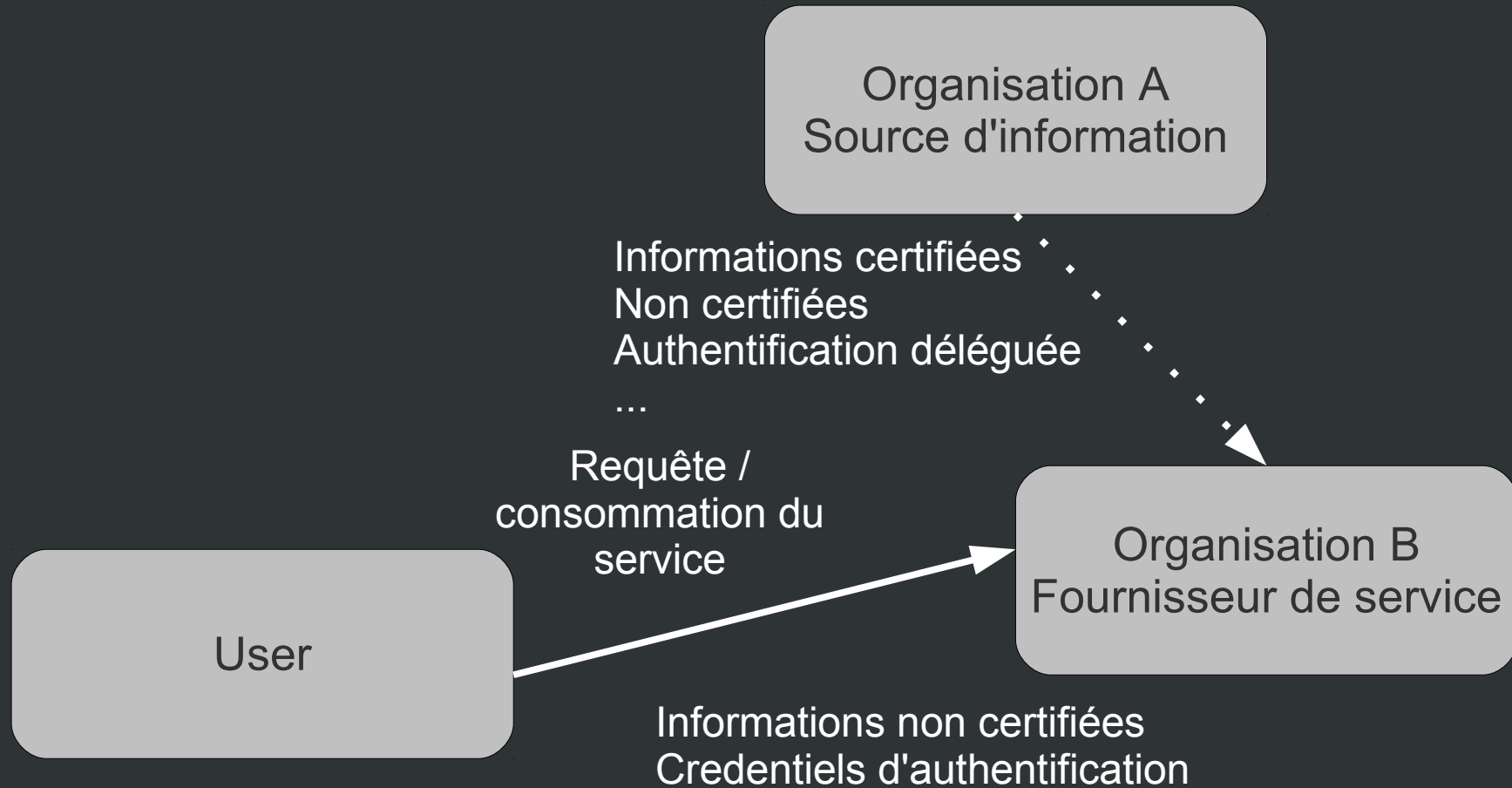


- Enjeux
- Esquisse de l'architecture pour les échanges inter-organisations et la présentation de données personnelles
- Quelques fonctionnalités particulières
- Hébergement de données personnelles et de fonctionnalités
- Conclusion

Architecture

Entr'ouvert

E-administration et identité numérique



- Architecture pour que la diffusion des données personnelles certifiées soit maîtrisée :
 - Soit l'utilisateur obtient des certificats (d'attributs) et les présente. Ex: Infocard, SAML profile Post
 - Canal d'échanges des PII certifiées au premier plan
 - Soit l'utilisateur délivre au destinataire des données des autorisations employées pour les obtenir. Ex SAML profile "Artefact"
 - Arrière-plan
 - Échanges directs entre organisations

- Un canal en arrière-plan implique le suivi d'activité par la source des données certifiées
- Cela rend impossible :
 - la mise en œuvre de la non-associativité des transactions
 - la diffusion d'informations lorsque la source est hors-ligne
- Une architecture avec des échanges au premier plan peut résoudre ces problématiques

- Architecture Web "standard": Navigateur Web agit en relai par redirection HTTP pour le transport des messages du protocole d'obtention/présentation (SAML, OpenID, etc.).
 - Pas d'action possible au cours de ces échanges : clients dits passifs
 - Les tiers gèrent l'adressage des requêtes : Suivi d'activité par le fournisseurs de certificats

- Environnement utilisateur "riche" nécessaire pour une architecture avec des échanges au premier plan permettant :
 - Le support d'un protocole d'obtention et de présentation des données et l'adressage des sources de données
 - Sélection des sources par l'utilisateur plus simple que sur une interface de sélection sur le destinataire
 - Le stockage et la "manipulation" des certificats
- Exemple: Infocard, SAML LECP profile, ID-WSF advanced client, etc.

- Un environnement utilisateur riche c'est la possibilité :
 - De faire un canal d'échanges des données certifiées au premier-plan
 - Éviter le suivi d'activité
 - Stockage: données personnelles, matériel d'authentification, certificats des tiers de confiance, etc.
 - Éviter les SSO qui se base sur la délégation de l'authentification
 - Le protocole de présentation de données certifiées peut être le même que celui de divulgation de données non certifiées
 - Maîtrise et audit de la diffusion
 - Faciliter l'unification pour l'usager de la gestion de ses données personnelles : Réaliser une interface unique
 - Privacy dashboard, e-portfolio, viewer pour réseaux sociaux, etc.
 - **Nombres d'autres fonctionnalités...**

- Enjeux
- Esquisse de l'architecture pour les échanges inter-organisations et la présentation de données personnelles
- Quelques fonctionnalités particulières
- Hébergement de données personnelles et de fonctionnalités
- Conclusion

- C'est aussi le moyen de gérer ses propres tiers de confiance, de fournir des moyens de confiance pour les usagers :
 - Requérir de la part de l'interlocuteur des certificats issus de tiers de confiance (« labélisation »)
 - Assister la récupération d'autres informations de confiance (évaluation de site de réputation)

- En mettant cet environnement utilisateur riche « en ligne », c'est la possibilité de mettre en œuvre une diffusion automatisée des données personnelles
 - Données (ou accès aux données) et fonctionnalités de contrôle d'accès aux données personnelles en ligne (PDS)
 - Contrôle d'accès gérés par l'utilisateur sur ses données personnelles
 - Permettre la diffusion d'informations certifiées ou non sur authentification et autorisation
 - Accès à des tiers « inconnus » sur certificats issus de tiers de confiance (de la même façon qu'un SP autorise des consommateurs)

- Pistes d'implémentation : les négociations de confiance automatisées
- Un agent de négociation au sein de l'environnement utilisateur riche
 - Analyse d'un règlement de contrôle d'accès
 - Algorithme de négociation pour la diffusion automatique/appliquer le règlement
 - Diffusion minimale et complétude
- Assister l'utilisateur dans la configuration de son règlement

- Ex : "Environnements pervasifs et ubiquitaires"
 - Environnement riche de l'utilisateur est publiquement adressable
 - Diffusion de dossiers médicaux *via* celui-ci sur des scènes d'accidents

- Applicables pour les données self-asserted:
 - Tous les messages postés sur des forums. Les afficheurs de contenus obtiennent les données à chaque demande d'affichage
 - Diffusion d'informations au sein de réseaux sociaux
 - **Premier contrôle de la durée de vie de ses données personnelles par l'utilisateur**

- Un environnement utilisateur riche est nécessaire :
 - Pour le stockage de certificats hors-lignes
 - Pour la présentation sélective de contenu et les preuves sur le contenu de certificats hors-lignes
 - Pour faire jouer à l'utilisateur son rôle dans les schémas cryptographiques permettant la non-associativité

- Enjeux
- Esquisse de l'architecture pour les échanges inter-organisations et la présentation de données personnelles
- Quelques fonctionnalités particulières
- Hébergement de données personnelles et de fonctionnalités
- Conclusion

- Convergence : L'environnement utilisateur c'est donc un ensemble de fonctionnalités liées à la diffusion des informations personnelles
 - De n vers 1
 - Stockage des données de n vers 1
- Enjeux
 - Mobilité
 - Automatisation de la diffusion
 - Mise à disposition des usagers de cet environnement

- La mobilité peut reposer sur :
 - Un stockage en ligne des données
 - Ou un périphérique amovible : Support du périphérique sur le terminal de l'utilisateur
- Et nécessite la disponibilité des fonctionnalités :
 - une installation sur tous les terminaux
 - ou un système accessible en ligne
- L'automatisation requière des données et des fonctionnalités en ligne
- → La mise en ligne des données personnelles semble requise

- Service hébergé pas un tiers → Confiance
- Téléphone portable / Box opérateur ?
 - (En les supposant comme serveurs en ligne sans limitation de la bande passante...)
 - Pour le grand public : Environnements non contrôlés par l'utilisateur / fermés
 - Flux entre ces terminaux et les opérateurs difficilement contrôlables par les usagers
 - **Les usagers doivent faire confiance à un tiers**
 - ~ équivalent à un serveur hébergé chez un opérateur

- Un serveur personnel « maîtrisé » par l'utilisateur :
 - Déployé par l'utilisateur
 - Où à l'opposé, déployé par un tiers (appliance) :
Confiance ?

Conclusion

- Tout semble indiquer que l'on va vers un environnement utilisateur riche : nouvelles fonctionnalités + centralisation
 - Les navigateurs s'enrichissent, ...
 - Infocard, Higgins, IDWSF Advanced Client, ...
 - Internet Of Subject, FOAF+SSL, ...
 - Projets de recherche supposent des fonctionnalités qui supposent un tel environnement (TAS3, Prime), ...

- « Qui » est légitime, de confiance, pour héberger les données personnelles et les fonctionnalités d'un environnement utilisateur riche?
 - Multiplication des offres d'hébergements de données en ligne ou sur terminaux mobiles
 - Opérateurs très actifs
- En centralisé, quelle sera la forme du serveur personnel ?
- Le libre doit se positionner rapidement pour offrir un tel environnement, que la communauté puisse contrôler à défaut du grand public, une source de confiance légitime.

- Travaux en cours sur l'implémentation d'un environnement centralisant données et fonctionnalités et qui peut être déployé (ou non) sur un hôte distinct du terminal de l'utilisateur
 - Protocole de diffusion des données : SAML2++
 - Lasso 2.2.92 (GNU GPLv2) : SAML2.0 (certifiée liberty alliance) et ID-WSF2.0
 - Certificats avancés : CL-Signature RSA
 - Librairie Cryptic 1.0 (GNU GPLv2)
 - Gestion des métadonnées de certificats
 - Contrôle d'accès sur données certifiées (XACML, outils de ATN, Décision (OrBAC))
 - Intégration de l'environnement avec les clients applicatifs (Web)

Rencontres Mondiales du Logiciel Libre 2010

Bordeaux

8 juillet 2010

Partie 1:

Certificats d'attributs avancés

API Cryptic

Mikaël Ates

mates@entrouvert.com



Plan

Entr'ouvert

E-administration et identité numérique

- Enjeu des certificats d'attributs
- Principes crypto
- Cryptic

Plan

Entr'ouvert

E-administration et identité numérique

- Enjeu des certificats d'attributs
- Principes crypto
- Cryptic

- Contrôle d'accès en environnement ouvert :
 - Autoriser un individu inconnu
 - Baser le contrôle d'accès sur des attributs de cet individu
 - Informations certifiées par des tiers de confiance du fournisseur de service
- Exemple : fournisseur de service « loueur » de voiture :
 - Individu autorisé à conduire – préfecture
 - Age > 21 ans – État civil/préfecture
 - Attestation d'assurance – Groupement des assurances FR
 - Argent - Banque

- Architecture basée sur les certificats :
 - Le user obtient des certificats digitaux (permis de conduire, pièce d'identité, attestation d'assurance, etc.)
 - Les présente durant la transaction d'obtention d'un service.

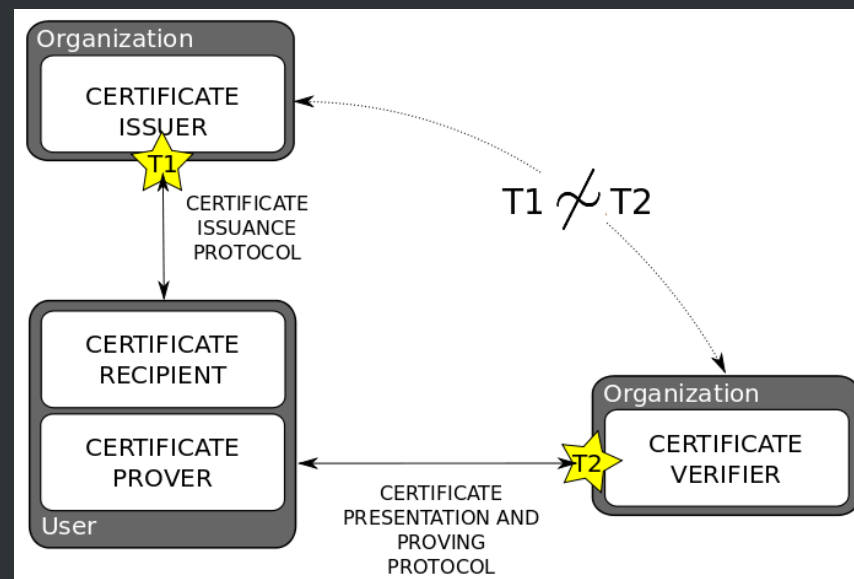
- Échanges sur les identités inter-organisations
 - Fournir à l'utilisateur les outils pour diffuser un minimum d'information
 - Adresser le problème de l'unification des enregistrements relatifs à un même usager entre organisations (par des organisations malveillantes)

- Fournir à l'utilisateur les outils pour diffuser un minimum d'information
 - Certificats « en ligne », que l'on peut obtenir à la demande, peuvent contenir un minimum d'information avec une sémantique adaptée ;
 - Lieu de naissance : Clermont-Ferrand – Date de naissance : 28/01/1982
 - → « Français majeur »
 - Certificats « hors ligne », réutilisables
 - Permettre de faire de la diffusion sélective de contenu et de conduire des preuves de propriétés sur les valeurs

- Adresser le problème de l'unification des enregistrements relatifs à un même usager entre organisations par des organisations malveillantes
 - Protéger l'usager contre des organisations malveillantes qui mutualisent leurs enregistrements de données sur un même usager pour en extraire de l'information
 - → Usager anonyme et rendre ses transactions **non associables**
 - De nombreux facteurs : temps, signature, informations révélées, etc.

- Il faut adresser chacun des facteurs connus.
- Exemples
 - temps : offline certificate
 - Informations révélées : k-anonymity
 - Né à Clermont-Ferrand le 28/01/82
 - → Français Majeur

- On se focalise sur le schéma de signature, donc sur le fait que la signature ne soit pas un facteur d'associativité :
- La « signature varie » entre l'émission du certificat et ses multiples présentations



Plan

Entr'ouvert

E-administration et identité numérique

- Enjeu des certificats d'attributs
- Principes crypto
- Cryptic

- Représentation en logarithme discret :
 - Attributs (messages, quantités) : $\{m_0, \dots, m_L\}$
 - $Y = g_0^{m_0} \cdot g_1^{m_1} \cdot \dots \cdot g_L^{m_L} \pmod{n}$
- Preuve de connaissance de x avec $y = g^x$
 - Exemple ZKPK Schnorr
 - Prouver : Avec r random, commitment : $t = g^r$
 - Verifieur : Challenge : c
 - Prouver : Réponse : $s = r - cx$
 - Verifieur : $t \stackrel{?}{=} y^c \cdot g^s$
 - $s = r - cx \rightarrow r = s + cx$
 - $\rightarrow g^r = g^s \cdot g^{cx}$
 - $\rightarrow g^r = g^s \cdot y^c$

- Prouver une représentation consiste à prouver que l'on connaît toutes les quantités représentées
 - On peut révéler une ou plusieurs valeurs de la représentation et prouver le « reste » (ne pas le révéler).
 - On révèle m_1 , on prouve $Y.g_1^{-m_1}$
- On peut prouver qu'une certaine quantité a une « propriété ». En pratique, sans révéler une valeur on peut prouver qu'elle est dans un intervalle
 - on peut ainsi prouver à partir de sa date de naissance dans une représentation, sans la révéler, que l'on est majeur
 - Prouver que $m_1 < x$ avec m_1 une date de naissance et $x = \text{date_du_jour()} - 18 \text{ ans}$

- Le fournisseur de certificats représente les attributs d'un usager et le signe = certificat d'attributs
- L'usager, présente ce certificat, et conduit des preuves de connaissances
 - Sans révéler aucun attribut = prouver que l'on possède un certificat
 - Révéler un attribut ou faire de preuves de propriétés

- La non associativité pas la CL-Signature RSA (Camenisch 01, + Lysynskaya 03) sur une représentation en logarithme discrets
 - n composite produit de 2 safe prime
 - probleme RSA $ed = 1 \pmod{\text{PHI}(n)}$
 - La clé publique du fournisseur de certificats est n .
 - Les éléments S et Z sont aussi publics, ainsi que les bases de représentation (QR_n)
 - Signature:
 - v , random connu du fournisseur et du prouver
 - $A = (Z / S^v \cdot \text{DLREP})^d \pmod{n}$
 - Prouver ce certificat revient à prouver la représentation $Z = A^e \cdot S^v \cdot \text{DLREP}$, le certificat : (A, e, v)
 - Seul le fournisseur peut calculer Z , prouver Z c'est prouver que l'on a un certificat de celui-ci.
 - Car il est le seul à pouvoir faire une inversion modulaire (RSA)

- La non associativité pas la CL-Signature sur une représentation en logarithme discrets
 - « Randomization »
 - Prendre un random r ,
 - $A' = A.S^{-r}$, e , $v' = v + er \rightarrow (A',e,v')$ est aussi un certificat valide
 - Dans la preuve de représentation, le prouver utilise A' et v'
 - Le verifieur apprend A' et pas A , A' et A sont in-nassociables.
 - Le prouver peut prouver une signature valide sur une représentation mais avec les valeur révélées le fournisseur ne « sait » pas quand il a produit cette signature

- Le vérifieur et le fournisseur peuvent être une même entité.
- Le fournisseur ne peut pas désigner publiquement un certificat → pas de révocation possible.
- Certificats avec une durée de validité limitée
- Certificats avec un nombre d'usage limité (n-time use) :
 - A l'utilisation $n+1$ le certificat devient associable.
 - Généralement accompagné d'une révocation de l'anonymat

Plan

Entr'ouvert

E-administration et identité numérique

- Enjeu des certificats d'attributs
- Principes crypto
- Cryptic

- Cryptic 1.0 GPLv2
- API en C
 - CL-Signature
 - ZKPK Shnorr
 - Preuve qu'une quantité est dans un intervalle
- Arithmétique basé sur OpenSSL
- Connecteur pour Python
- Exemples

- Bientôt Cryptic 1.1
- Connecteur pour Java
- Documentation
- Amélioration du protocole de preuve
- Tests unitaires

- Architecture de l'API
- A la racine Cryptic
 - Répertoire maths
 - Génération de groupe
 - Fonctions maths pour la preuve de quantité dans un intervalle
 - Répertoire protocol
 - CL-Signature
 - Preuve de connaissance de schnorr
 - Preuve quantité dans un intervalle
- A la racine : tests (démo) et bindings



Plateforme

Entr'ouvert

E-administration et identité numérique

- Debian / Ubuntu
- OpenSSL 0.9.8
- GLIB 2.0 - GObject 2.0

Exemples

Entr'ouvert

E-administration et identité numérique

```
import cryptic
c = cryptic.Clsig(1024,256,600,0,0,0,3)
c.generateParameters()
q1 = cryptic.charToBn("Mik")
q2 = cryptic.charToBn("Ates")
c.computeDlrepByIndex((q1,q2),(0,2),2)
c.sign()
c.verifySignatureNotRandomized()
```

Fournisseurs de certificats / Issuer

```
c.randomizeSignature()
c.verifySignatureRandomized()
dlrep = c.correctDlrepBeforeProving(c.z)
s = cryptic.ZkpkSchnorr((c.aRand,c.s,c.bases[0],c.bases[2]),4,c.modulus)
s.round1()
h = cryptic.HashForNiProofs(256)
h.addProof(s,dlrep)
h.computeHash()
s.round2WithoutOrder(h.hValue,(c.eCorrected,c.vRand,q1,q2))
```

User / Prover

```
s2 = cryptic.ZkpkSchnorr((c.aRand,c.s,c.bases[0],c.bases[2]),4,c.modulus)
s2.verifyNoninteractiveProof(dlrep,h.hValue,s.responses)
h2 = cryptic.HashForNiProofs(256)
h2.addProof(s2,dlrep)
h2.computeHash()
cryptic.cmpBn(h.hValue,h2.hValue)
```

Fournisseur de service / Verifier



- Les uses cases :
 - Non-associativité difficile à mettre en oeuvre
 - Pièce d'identité anonyme revocation synchrone
 - Dans un premier temps pas l'unlink :
 - Diffusion selective
 - Preuve de propriétés
 - Non associativité : le vote, le e-cash...
- « Only a piece of the puzzle... »