Retrouver une vie privée en ligne, étude de détail de Caliopen : les indices de confidentialité et la gestion des clés.

Laurent Chemla

Retrouver une vie privée en ligne, étude de détail de Caliopen : les indices de confidentialité et la gestion des clés.

Laurent Chemla
Caliopen
24 rue des 3 fontaines
30000 Nîmes

Résumé

Caliopen est une suite logicielle libre de gestion de correspondance privée. Après une brève définition du projet, de ses objectifs en terme de protection de la vie privée et de ses ambitions, certaines spécificités susceptibles d'intéresser d'autres projets sont abordées : les « indices de confidentialité » (Privacy Index ou PI), d'une part, et le choix du DNS pour la diffusion des clés publiques, d'autre part.

Pour donner au public un meilleur contrôle de sa vie privée, Caliopen propose d'associer un indice de confidentialité à chacun des éléments de son interface. En permettant à l'utilisateur d'améliorer facilement ce niveau, cet affichage est une incitation à mieux se protéger lui-même et ses contacts. Une première ébauche des métriques utilisées pour calculer ces indices est présentée.

L'utilisation de clés de chiffrement symétriques est encouragée dans Caliopen. Nous avons fait le choix d'utiliser le DNS pour diffuser les clés publiques des utilisateurs, et nous expliquons les avantages et les inconvénients de ce choix, ainsi que brièvement les pistes alternatives.

En conclusion nous imaginons que ces deux aspects du projet Caliopen deviendront dans un futur proche des standards de fait, sinon dans l'approche particulière qui est la nôtre au moins dans l'idée qu'il est nécessaire de réintroduire la notion de confidentialité dans la correspondance privée numérique, et de faire en sorte de généraliser le chiffrement des contenus échangés.

Mots-clefs

Correspondance privée numérique, indices de confidentialité, gestion et diffusion des clés publiques.

1 Introduction

Depuis les révélations d'E. Snowden, le grand public a pris conscience du niveau de surveillance à laquelle il peut être soumis depuis qu'il utilise massivement le numérique pour sa correspondance privée. Pour autant, cette prise de conscience n'a pas donné lieu à une modification des pratiques suffisamment massive pour changer la donne : d'une part l'hyper-centralisation des services permet la surveillance des utilisateurs à moindre coût, d'autre part la très grande majorité des internautes ne se sent pas concernée par une atteinte à sa vie privée qu'elle considère comme sans importance.

L'objectif du projet Caliopen est d'inverser ces deux tendances.

2 Présentation de Caliopen

2.1 Les mauvaises réponses

À ce jour, les réponses proposées par la communauté technique sont de deux ordres : créer des

alternatives mieux sécurisées (décentralisées, ou auto-hébergées), et inventer d'autres protocoles de transport de la correspondance privée (incompatibles avec l'existant). À l'évidence, ces deux pistes - quelles que soient leurs qualités - ne concernent qu'un nombre très restreint d'utilisateurs. Seuls les plus motivés des technophiles peuvent se permettre soit de se couper de ceux de leurs contacts qui n'utiliseront jamais tel ou tel protocole ésotérique, soit d'installer et de maintenir leur propre serveur, soit même de changer d'adresse email, de faire l'effort d'en informer tous leurs proches, et de perdre en fonctionnalité sans pour autant être réellement mieux protégé.

Car rien ne sert de disposer d'une messagerie permettant le chiffrement, fut-elle localisée dans un pays à l'abri des grandes oreilles de la NSA, quand la quasi-totalité de ses correspondants continue d'utiliser Gmail et consorts : la protection de notre correspondance suppose que nos correspondants sont eux aussi protégés.

Sans même parler du fait que, de nos jours, nos correspondances passent tout autant sinon plus par les messages "privés" des réseaux sociaux (Facebook, Twitter, Linkedin...) que par le seul email, et que la seule protection de ce dernier n'empêche donc en rien la capture de métadonnées permettant la modélisation des graphes sociaux de la population (à qui parle t-on, quand, et où, sinon de quoi).

2.2 La « bonne » réponse

La direction choisie par Caliopen est différente : plutôt que de proposer une simple alternative, son ambition est d'être - au moins en phase d'adoption - un simple outil d'agrégation. Plutôt que d'avoir à se connecter à ses différents comptes en ligne, l'utilisateur de Caliopen dispose d'une interface qui lui affiche la totalité de sa correspondance privée, d'où qu'elle vienne. En dissociant le protocole sous-jacent (SMTP, XMPP, réseaux sociaux et applications diverses) du contenu (les messages privés), Caliopen espère créer de nouveaux usages et, surtout, attirer le plus grand nombre d'utilisateurs possible.

Ce n'est qu'une fois le produit adopté, pour cet usage d'agrégation, que Caliopen prend tout son sens. En assignant à chacun des éléments de son interface un « indice de confidentialité », et en l'affichant de façon systématique, Caliopen va permettre à ses utilisateurs de prendre conscience du degré d'exposition de leur vie privée en ligne. Quand un message venant de Gmail, par exemple, est affiché avec un indice quasi nul de confidentialité, on y répondra pas forcément de façon aussi libérée que s'il existe un meilleur canal. Et Caliopen utilisera par défaut ce canal là pour y répondre, en fonction des données du contact. En affichant un indice global de confidentialité du compte utilisateur, Caliopen va motiver celui-ci pour l'améliorer, en lui proposant des options pour ce faire. Et ainsi, petit à petit, chaque utilisateur de Caliopen sera poussé (par exemple en échange d'un plus grand espace de stockage, d'une plus grande notoriété...) à se créer des clés de chiffrement, à utiliser les protocoles les plus sûrs, à demander à ses contact de faire de même (et pourquoi pas à se créer un compte Caliopen).

Quand nous en serons là, nous aurons créé un outil à même d'être non seulement une alternative aux grands silos, mais aussi une alternative à l'e-mail. Tous les services basés sur Caliopen auront le choix de rejoindre un réseau privé sécurisé pour échanger entre eux les niveaux de confidentialités de leurs utilisateurs, ce réseau pourra supporter un protocole plus sûr y compris pour les échanges entre utilisateurs, tout en étant largement décentralisé.

3 Privacy Index (« indice de confidentialité »)

3.1 Philosophie du Privacy Index (PI)

L'idée d'afficher un "indice de confidentialité" associé à chaque message reçu n'est pas si neuve. La lettre de cachet était fermée, et marquée à la cire du sceau du roi. Pour garantir la confidentialité, ce sceau ne devait être brisé que par le destinataire. Un peu plus récemment, chacun était conscient qu'une lettre postée dans une enveloppe collée était plus confidentielle qu'une simple carte postale. Cette notion n'a

vraiment disparu qu'avec l'avènement de la correspondance numérique, et la proposition de la réintégrer dans nos messageries modernes n'a rien de révolutionnaire. L'IETF elle-même en envisage la syntaxe dans le RFC 7444, et d'autres, comme Enigmail par exemple, commencent à se pencher sur leur métrique (https://admin.hostpoint.ch/pipermail/enigmail-users_enigmail.net/2015-September/003341.html).

Pour Caliopen, qui envisage d'associer de tels indices à tous ses composants (interne, interface, hébergeur), ces métriques sont capitales. Ce sont elles qui redonneront à l'utilisateur la maîtrise de sa vie privée, qui le pousseront à l'améliorer, et qui permettront l'effet « boule de neige » espéré. Elles doivent donc (à l'instar du sceau ou de la colle des enveloppes) être intuitives et compréhensibles par l'utilisateur afin qu'il sache comment les améliorer. Elles ne doivent pas contenir de variable cachée et être accompagnées d'une expérience utilisateur qui l'accompagne dans sa progression.

3.2 Liste des PI

À ce jour, nous avons identifié sept éléments d'un compte Caliopen qui auront chacun leurs propres indices et leur propre calcul. Tous ne sont pas encore détaillés, mais en voici la liste ainsi que leurs éléments de métrique identifiés :

Les contacts :

- Indice du compte utilisateur Caliopen (s'il existe) : de 0 à 50 points.

- Existence d'une clé publique (type de la clé, taille de la clé, méthode de diffusion de la clé) : de 0 à 9 points.

- Type des échanges (plus ce contact utilise des protocoles sécurisés avec l'utilisateur, plus cette composante monte) : 5 points au delà de 80% d'utilisation de chiffrement, 4 points entre 60 et 80, 3 points entre 40 et 60, 2 points entre 25 et 40, 1 point au dessus de 15%.

- Niveau de complétude (plus l'utilisateur précise les informations du contact, plus cette composante monte) : de 0 à 3 points.

Il reste 33 points à attribuer.

– Les terminaux :

- Type de terminal (déclaratif : l'utilisateur est poussé à être objectif) station fixe personnelle : 10 points, station fixe familiale : 6 points, ordinateur mobile personnel : 4 points, tablette/smartphone personnelle : 3 points, ordinateur professionnel : 1 point.

- Terminal stockant la clé privée de l'utilisateur : 10 points.

- Système d'exploitation : OS libre (et à jour, stable ou mieux) : 20 points, iOS/Android/Windows (à jour), ou OS libre pas à jour : 10 points, iOS/Android/Windows (pas à jour) : 0.

- Système de fichiers (déclaratif) : 15 points si chiffré.

- Configuration SSL du navigateur : accepte TLS 1.2 : 5 points, TLS 1.1 : 4 points, TLS 1.0 : 3 points, sinon 0. Si la négociation n'accepte que RC4 ou SSLv3 c'est 0 aussi.

- Connexion au réseau local (déclaratif) : filaire: 5 points, wifi : 0.

- Pare-feu et antivirus (déclaratif) : de 0 à 5 points.

Il reste 30 points à attribuer.

– Les messages :

- Protocole utilisé : XMPP/OTR+TLS : 15 points, SMTP/ TLS et XMPP/TLS : 10 points, IRC/OTR : 8 points, SMS/SMSsecure : 6 points.

- Type de chiffrement : message chiffré avec la clé publique du ou des destinataires : 20 points.

- Niveau de confidentialité du transport : TLS certificat reconnu : 10 points, TLS auto-certifié : 5 points, OTR : 10 points.

- Niveau de confidentialité du terminal émetteur (dans une limite de 20 points)

- Niveaux de confidentialité du contact de plus bas niveau (dans une limite de 20 points)

Il reste 20 points à attribuer (durée de vie du message, par exemple ? Un message qui disparaît rapidement offrant plus de points)

– Les discussions :

- Niveau du contact de moindre confidentialité participant (dans une limite de 30 points)

- Type d'échange (XMPP/OTR + TLS seulement : 15 points, SMTP/XMPP TLS seulement : 10 points, contenant IRC/OTR : 8 points, contenant SMS/SMSSecure : 6 points, contenant autre chose : 0 point)

- Niveau moyen de confidentialité des messages (dans une limite de 30 points)

- Nombre de participants (inversement proportionnel : 10 points si 2 participants, -1 point par participant supplémentaire dans une limite de 0 point).

– Les fichiers :

- Niveau de confidentialité du contact de partage le plus bas (dans une limite de 50 points)

- Type de chiffrement (message chiffré avec la clé publique du ou des destinataires : 20 points)

– Les agendas (non définis)

– Le compte utilisateur :

À la différence des autres objets, dont le niveau ne repose que sur des choix techniques, le compte utilisateur contient une composante comportementale (50 points sur 100) permettant de perdre et de gagner des points indépendamment du seul aspect technique (le cas d'espèce est l'utilisateur qui rend publiques des conversations sécurisées et qui perd des points à ce titre : si on baisse de niveau pour des questions comportementales, il faut pouvoir remonter en corrigeant son comportement). L'utilisateur part de 0, son niveau est réévalué chaque jour (ou en temps réel lorsque c'est possible).

- Existence d'une clé publique, type et taille de la clé, méthode de création et de diffusion, nombre et origine des signatures de la clé (web of trust). De 0 à 9, à détailler.

- Niveau de confidentialité du terminal de référence (0 à 9)

- Niveau moyen de confidentialité des contacts (0 à 9)

- Utilisation de son propre MX (dans une limite de 20 points, en débat)

- Type d'identification (password : de 0 à 2, double-auth (carte de correspondances aléatoire ou OTP) : 3 à 4, certificat client : 5).

- Type de stockage des échanges (chiffrés côté serveur avec la clé publique utilisateur : 2, effacés côté serveur régulièrement : 1, sinon 0) - ceci est une variable mi-technique mi-comportementale à mieux définir à l'usage)

- Type de client utilisé pour se connecter (client natif : 2, client IMAP/TLS ou XMPP/TLS : 1, client IMAP/XMPP sans TLS : 0). On retiendra la valeur du client de plus basse valeur utilisée durant les 3 derniers jours.

- Fréquence d'utilisation de terminaux sécurisés/non sécurisés (quand l'utilisateur se connecte depuis un terminal sécurisé, cette composante augmente, elle diminue quand il se connecte depuis un terminal non sécurisé, par pas de 1 ou 2 points selon le niveau de confidentialité des différents terminaux (un terminal entre 75 et 100 vaut 2 points, un terminal entre 50 et 75 1 point, entre 25 et 50 0 point, entre 10 et 25 -1 point, et -2 points en deçà) dans une limite de 10 points).

- Fréquence de déconnexion (la déconnexion automatique de type bancaire n'est pas imaginable dans un environnement dédié à la discussion, mais on encourage l'utilisateur à se déconnecter manuellement dès qu'il cesse d'utiliser l'interface Web en lui attribuant des points lorsqu'il le fait. 1 point par déconnexion manuelle dans un délai de 2 heures après la connexion, 0 sans déconnexion entre 2 et 4 heures, -1 point sans déconnexion manuelle au delà de 4 heures, dans une limite de 5 points à affiner à l'usage).

- Fréquence du chiffrement des communications (fonction directe du pourcentage d'envoi de messages chiffrés avec la clé publique du ou des correspondants dans le total des messages émis, dans une limite de 10 points).

- Fréquence d'échanges avec des correspondants d'un niveau de confidentialité égal au supérieur au sien (plus de 50% : 2 points, entre 30 et 50% 1 point, en dessous de 30% 0 point dans une limite de 5 points).

- Une réponse non chiffrée à un message chiffré (si on dispose de la clé du correspondant) implique une pénalité de 5 points.

- L'affichage d'une discussion sécurisée sur un terminal de sécurité moindre implique une pénalité de la valeur de la différence dans une limite de 20 points.

Un dernier élément, en discussion, concernerait l'instance de Caliopen elle-même. Il s'agirait de noter, en fonction des lois locales, du modèle économique choisi, voire du niveau moyen des comptes utilisateurs hébergés, l'indice de confidentialité de tel ou tel service ayant adhéré à la future association qui gérera le projet.

4 Diffusion des clés publiques

Historiquement, la diffusion et la découverte d'une clé publique de chiffrement symétrique soulèvent de nombreux problèmes, qui sont probablement une des causes principales au faible taux d'usage des logiciels de chiffrement basés sur PGP (Pretty Good Privacy, logiciel de chiffrement symétrique).

Il n'existe pas de système de diffusion standard qui permette à la fois la découverte et la validation d'une clé publique, hors Autorités de Certification (CA) (qui posent d'autres problèmes, de part leur multiplicité et la possibilité pour une autorité de falsifier un certificat). Les serveurs de clés[1] permettent la diffusion, mais pas la validation ni l'effacement. Les clés y sont envoyées, perdues, oubliées, redondantes, et surtout ils permettent à n'importe qui d'envoyer autant de clés qu'il veut pour l'adresse d'un tiers. Les serveurs vérifiant (par échange de mail) l'émetteur d'une clé sont plus sûrs mais restent aussi aisément falsifiables qu'une CA puisque l'utilisateur ou son mandataire n'en ont pas la maîtrise.

Dès l'origine du projet, Caliopen a envisagé d'utiliser le DNS pour diffuser les clés publiques créées par ses utilisateurs : bien avant même le premier brouillon IETF sur le sujet, et suivant les préconisations du « white paper » STEED de Werner Koch et Marcus Brinkman [1]. La diffusion d'une clé PGP par le DNS

1. https://en.wikipedia.org/wiki/Key_server_%28cryptographic%29

fait particulièrement sens (même sans parler du protocole DANE (DNS-based Authentication of Named Entities). Le DNS est décentralisé et presque toujours disponible. Une adresse e-mail se décompose de façon naturelle en un identifiant utilisateur d'une part, et un nom de domaine d'autre part. Il est pratique de considérer que le gestionnaire d'une zone DNS dispose de la confiance des utilisateurs du nom de domaine utilisé.

4.1 Ta clé dans ta zone

En réalité, cela fait tellement sens que nous aurions dû nous demander pourquoi ça n'a pas été fait avant, et si ça l'avait été, pourquoi cet usage n'avait pas été généralisé. Ce doute nous aurait évité les quelques chausse-trappes dans lesquelles nous serions tombés si l'IETF n'avait pas, l'été dernier, finalisé son brouillon «Using DANE to Associate OpenPGP public keys with email addresses »[2] (désormais en attente de publication par l'IESG) [2].

La gestion des clés publiques dans le DNS pose en effet quelques problèmes :

- Il laisse des traces, quand le client email (« Mail User Agent », MUA) de l'émetteur interroge le DNS pour obtenir la clé publique du destinataire, la relation entre les deux individus transite en clair et permet à un méchant de construire facilement un graphe social (notamment si le méchant gère le DNS auquel le poste de l'émetteur fait sa demande).

- Le RFC5321 [3] impose que seul le serveur email (« Mail Transfer Agent », MTA) récepteur est autorisé à interpréter la partie locale d'une adresse e-mail. De ce fait, l'encodage de cette partie locale dans le DNS devient complexe, et la recherche difficile. Comment par exemple demander au DNS la clé publique de laurent+test@brainstorm.fr si seul le MTA gérant la zone brainstorm.fr est autorisé à considérer que '+test' ne fait pas partie de l'identifiant ?

- Enfin nous aurions certainement fait le choix d'utiliser l'existant (un « Resource Record » (RR) de type CERT, tel que défini dans le RFC 4398 « Storing Certificates in the Domain Name System (DNS) »[3]) plutôt que de créer un RR spécifique, or (et nous faisons entièrement confiance sur ce point à l'IETF) la discussion sur le brouillon «Using DANE to Associate OpenPGP public keys with email addresses » a considéré que c'était un très mauvais choix.

Par chance, donc, ce brouillon est paru à temps pour nous éviter ces erreurs. De même la recherche d'une clé publique via DANE vient à peine (9 octobre 2015) d'être intégrée à GnuPG[4]. Car malgré ces pièges, l'utilisation du DNS pour la diffusion des clés reste une très bonne idée, d'autant plus lorsqu'elle est couplée à la signature d'une zone par DNSSEC :

- La diffusion d'une clé est faite par une autorité de confiance (le gestionnaire de la zone dans laquelle l'utilisateur a choisi de créer son adresse),

- DNSSEC permet d'authentifier la clé comme étant bien celle du récepteur, presque comme une remise de clé en main propre et donc beaucoup mieux que si la clé est trouvée sur un serveur de clés. Or, et parce que Caliopen ne considère pas la confidentialité en blanc et noir mais à travers un indice variable, il est possible d'attribuer des valeurs différentes à ces méthodes et d'afficher ainsi un indice de confidentialité plus fin qui tiendra compte, comme on l'a vu, de la sécurité de la méthode de diffusion.

2. https://tools.ietf.org/html/draft-ietf-dane-openpgpkey-05

3. https://tools.ietf.org/html/rfc4398

4. Implémentation libre de GPG https://www.gnupg.org/

- La révocation d'une clé devient aisée : il suffit de la remplacer par la nouvelle clé pour que les clients voient le changement au prochain 'lookup' (idéalement fait à chaque nouvel envoi d'un courrier).

- Enfin, le brouillon IETF « SMTP and SUBMISSION Service Extensions For Address Query » [4], associé au précédent même s'il n'est qu'en phase de discussion, nous précise les bonnes manières d'éviter les autres erreurs que nous aurions pu faire. Par exemple ne jamais modifier le trousseau d'un utilisateur sans son accord, même si la clé stockée localement ne correspond pas à celle venant du DNS, ne pas interroger le DNS systématiquement (respecter le TTL de la zone), ne pas considérer qu'une clé inconnue, même diffusée dans une zone DNSSEC, est aussi sûre qu'une clé fournie de la main à la main… On peut même imaginer que le point 4.5 de ce brouillon décrit précisément la notion de PI de Caliopen en disant que le client doit indiquer le degré de confidentialité associé à une clé. Ce brouillon va devenir notre vademecum dans les mois à venir.

4.2 Les pistes alternatives

Il existe à notre connaissance au moins deux alternatives à l'étude :

- Le brouillon « SMTP and SUBMISSION Service Extensions For Address Query » (https://tools.ietf.org/html/draft-moore-email-addrquery-01) propose un mécanisme dans lequel c'est le serveur de messagerie (MX de la zone) du récipient qui répond aux requêtes de recherche d'une clé publique relative à une adresse e-mail. Cette solution résout la question du mapping de la partie locale d'une adresse puisque c'est le MTA faisant autorité qui interpréterait cette partie, en respectant donc le RFC 5321. Ce brouillon est néanmoins beaucoup moins avancé. Il n'est donc pas intégré dans GnuPG, et surtout il délègue la gestion des clés au service d'e-mail plutôt qu'au gestionnaire de la zone, ce qui pose d'autres problèmes.

- Webfinger (RFC 7033) [5], protocole de recherche d'information sur les personnes ou les organismes via HTTP, pourrait aussi être un bon candidat, s'il devenait plus largement exploité dans le futur.

4.3 En manière de conclusion

Werner Koch, bien avant nous, s'est longtemps demandé pourquoi le DNS n'avait pas été massivement utilisé pour la diffusion des clés publiques. Sa réponse étant que les grands fournisseurs d'e-mail n'y avaient pas intérêt, puisqu'une trop grande facilité de chiffrement aurait mis à mal le modèle économique basé sur la publicité ciblée, impossible si le contenu des échanges est chiffré.

Ce point reste vrai, hélas, mais il est possible d'imaginer que, si Caliopen devait être un succès en terme d'adhésion populaire, le fait de choisir ce mode de diffusion devienne un standard de fait, et pousse ainsi les très grands opérateurs à l'adopter. On a le droit de rêver !

De même, et c'est à noter, que « notre » PI pourrait bien suivre un jour le même chemin, nous sommes loin d'avoir finalisé sa conception, mais l'idée de réintégrer la notion de confidentialité dans les attributs d'une correspondance électronique fait son chemin et devrait, même si Caliopen ne devait jamais être un succès, devenir assez standard. Enigmail, outil le plus utilisé à ce jour pour le chiffrement de l'email, prend cette direction avec un indicateur de son cru [6], et l'IETF a commencé à formaliser, sinon les métriques, au moins la forme des champs techniques associés à cet indicateur (RFC 7444) [7].

L'idée n'est pas neuve (TrustedBird a été présenté aux JRES en 2009 et propose son propre label de

sécurité, et l'ensemble de protocoles XMPP [5] a déjà une norme équivalente au RFC 7444 avec XEP-0258) mais elle devient de plus en plus évidente, depuis la prise de conscience de 2013.

Bibiographie

[1] « white paper » STEED de Werner Koch et Marcus Brinkman http://g10code.com/docs/steed-usable-e2ee.pdf

[2] « Using DANE to Associate OpenPGP public keys with email addresses » https://tools.ietf.org/html/draft-ietf-dane-openpgpkey-05

[3] RFC 5321 «Simple Mail Tranfer Protocol » https://tools.ietf.org/html/rfc5321

[4] « SMTP and SUBMISSION Service Extensions For Address Query » https://tools.ietf.org/html/draft-moore-email-addrquery-01

[5] WebFinger, RFC 7033 https://tools.ietf.org/html/rfc7033

[6] Proposition de calcul d'un indice de confidentialité pour Enigmail : https://admin.hostpoint.ch/pipermail/enigmail-users_enigmail.net/attachments/20150919/0684dc41/attachment-0001.html

[7] Security Labels in Internet Email https://tools.ietf.org/html/rfc7444

[8] Présentation de TrustedBird, JRES 2009 https://2009.jres.org/planning_files/slideshow/pdf/126.pdf

5. Extensible Messaging and Presence Protocol http://xmpp.org/xmpp-protocols/rfcs/