Demain, les développeurs… ?

En quelques années à peine s'est élevée dans une grande partie de la population la conscience diffuse des menaces que font peser la surveillance et le pistage sur la vie privée.

Mais une fois identifiée avec toujours plus de précision la nature de ces menaces, nous sommes bien en peine le plus souvent pour y échapper. Nous avons tendance surtout à chercher qui accuser… Certes les coupables sont clairement identifiables : les GAFAM et leur hégémonie bien sûr, mais aussi les gouvernements qui abdiquent leur pouvoir politique et se gardent bien de réguler ce qui satisfait leur pulsion sécuritaire. Trop souvent aussi, nous avons tendance à culpabiliser les Dupuis-Morizeau en les accusant d'imprudence et de manque d'hygiène numérique. C'est sur les utilisateurs finaux que l'on fait porter la responsabilité : « problème entre la chaise et le clavier », « si au moins ils utilisaient des mots de passe compliqués ! », « ils ont qu'à chiffrer leur mails », etc. et d'enchaîner sur les 12 mesures qu'ils doivent prendre pour assurer leur sécurité, etc.

L'originalité du billet qui suit consiste à impliquer une autre cible : les développeurs. Par leurs compétences et leur position privilégiée dans le grand bain numérique, ils sont à même selon l'auteur de changer le cours de choses et doivent y œuvrer.
Les pistes qu'expose Mo Bitar, lui-même développeur (il travaille sur StandardNotes, une application open source *de notes qui met l'accent sur la longévité et la vie privée) paraîtront peut-être un peu vagues et idéalistes. Il n'en pointe pas moins une question intéressante : la communauté des codeurs est-elle consciente de ses responsabilités ?*

Qu'en pensent les spécialistes de la cybersécurité, les adminsys, la communauté du développement ? – les commentaires sont ouverts, comme d'habitude.

Article original : [The Privacy Revolution that never came]
Traduction Framalang : tripou, david, goofy, audionuma, MO, lyn., Luc et un anonyme.

La révolution de la vie privée n'a jamais eu lieu

Voici pourquoi les développeurs de logiciels détiennent la clef d'un nouveau monde

par Mo Bitar

Actuellement, c'est la guerre sur les réseaux, et ça tire de tous les côtés. Vous remportez une bataille, ils en gagnent d'autres. Qui l'emporte ? Ceux qui se donnent le plus de mal, forcément. Dans cette campagne guerrière qui oppose des méga-structures surdimensionnées et des technophiles, nous sommes nettement moins armés.

Des informations. C'est ce que tout le monde a toujours voulu. Pour un gouvernement, c'est un fluide vital. Autrefois, les informations étaient relativement faciles à contrôler et à vérifier. Aujourd'hui, les informations sont totalement incontrôlables.

Les informations circulent à la vitesse de la lumière, la vitesse la plus rapide de l'univers. Comment pourrait-on arrêter une chose pareille ? Impossible. Nos problèmes commencent quand une structure trop avide pense qu'elle peut le faire.

Telle est la partie d'échecs pour la confidentialité que nous jouons tous aujourd'hui. Depuis le contrôle de l'accès à nos profils jusqu'au chiffrement de nos données en passant par un VPN (réseau privé virtuel) pour les rediriger, nous ne sommes que des joueurs de deuxième zone sur le grand échiquier des

informations. Quel est l'enjeu ? Notre avenir. Le contrôle de la vie privée c'est le pouvoir, et les actions que nous menons aujourd'hui déterminent l'équilibre des pouvoirs pour les générations et sociétés à venir. Quand ce pouvoir est entre les mains de ceux qui ont le monopole de la police et des forces armées, les massacres de masse en sont le résultat inévitable.

Alors, où se trouve la révolution sur la confidentialité de nos informations que nous attendons tous ? Ce jour d'apothéose où nous déciderons tous de vraiment prendre au sérieux la question de la confidentialité ? Nous disons : « Je garde un œil dessus, mais pour le moment je ne vais pas non plus me déranger outre mesure pour la confidentialité. Quand il le faudra vraiment, je m'y mettrai ». Ce jour, soit n'arrivera jamais, soit sous une forme qui emportera notre pays avec lui. Je parle des États-Unis, mais ceci est valable pour tout pays qui a été construit sur des principes solides et de bonnes intentions. Bâtir un nouveau pays n'est pas facile : des vies sont perdues et du sang est inutilement versé dans le processus. Gardons plutôt notre pays et agissons pour l'améliorer.

Les gouvernements peuvent être envahissants, mais ni eux ni les gens ne sont mauvais par nature : c'est l'échelle qui est problématique. Plus une chose grandit, moins on distingue les actions et les individus qui la composent, jusqu'à ce qu'elle devienne d'elle-même une entité autonome, capable de définir sa propre direction par la seule force de son envergure.

Alors, où est notre révolution ?
— Du côté des développeurs de logiciels.

Les développeurs de logiciels et ceux qui sont profondément immergés dans la technologie numérique sont les seuls actuellement aptes à déjouer les manœuvres des sur-puissants, des sans-limites. Il est devenu trop difficile, ou n'a jamais vraiment été assez facile pour le consommateur moyen de suivre

l'évolution des meilleurs moyens de garder le contrôle sur ses informations et sa vie privée. La partie a été facile pour le Joueur 1 à tel point que le recueil des données s'est effectué à l'échelle de milliards d'enregistrements par jour. Ensuite sont arrivés les technophiles, des adversaires à la hauteur, qui sont entrés dans la danse et sont devenus de véritables entraves pour le Joueur 1. Des technologies telles que Tor, les VPN, le protocole Torrent et les crypto-monnaies rendent la tâche extrêmement difficile pour les sur-puissants, les sans-limites. Mais comme dans tous les bons jeux, chaque joueur riposte plus violemment à chaque tour. Et notre équipe perd douloureusement.

Même moi qui suis développeur de logiciels, je dois admettre qu'il n'est pas facile de suivre la cadence des dernières technologies sur la confidentialité. Et si ce n'est pas facile pour nous, ce ne sera jamais facile pour l'utilisateur lambda des technologies informatiques. Alors, quand la révolution des données aura-t-elle lieu ? Jamais, à ce rythme.

Tandis que nous jouissons du luxe procuré par la société moderne, sans cesse lubrifiée par des technologies qui nous libèrent de toutes les corvées et satisfont tous les besoins, nous ne devons pas oublier d'où nous venons. Les révolutions de l'histoire n'ont pas eu lieu en 140 caractères ; elles se sont passées dans le sang, de la sueur et des larmes, et un désir cannibale pour un nouveau monde. Notre guerre est moins tangible, n'existant que dans les impulsions électriques qui voyagent par câble. « Où se trouve l'urgence si je ne peux pas la voir ? » s'exclame aujourd'hui l'être humain imprudent, qui fonctionne avec un système d'exploitation biologique dépassé, incapable de pleinement comprendre le monde numérique.

Mais pour beaucoup d'entre nous, nos vies numériques sont plus réelles que nos vies biologiques. Dans ce cas, quel est l'enjeu ? La manière dont nous parcourons le monde dans nos vies numériques. Imaginez que vous viviez dans un monde où, dès que vous sortez de chez vous pour aller faire des courses,

des hommes en costume noir, avec des lunettes de soleil et une oreillette, surveillent votre comportement, notent chacun de vos mouvements et autres détails, la couleur de vos chaussures ce jour-là, votre humeur, le temps que vous passez dans le magasin, ce que vous avez acheté, à quelle vitesse vous êtes rentré·e chez vous, avec qui vous vous déplaciez ou parliez au téléphone – toutes ces métadonnées. Comment vous sentiriez-vous si ces informations étaient recueillies sur votre vie, dans la vraie vie ? Menacé·e, certainement. Biologiquement menacé·e.

Nos vies sont numériques. Bienvenue à l'évolution. Parcourons un peu notre nouveau monde. Il n'est pas encore familier, et ne le sera probablement jamais. Comment devrions-nous entamer nos nouvelles vies dans notre nouveau pays, notre nouveau monde ? Dans un monde où règnent contrôle secret et surveillance de nos mouvements comme de nos métadonnées ? Ou comme dans une nouvelle *vieille Amérique*, un lieu où être libre, un lieu où on peut voyager sur des milliers de kilomètres : la terre promise.

Construisons notre nouveau monde sur de bonnes bases. Il existe actuellement des applications iPad qui apprennent aux enfants à coder – pensez-vous que cela restera sans conséquences ? Ce qui est aujourd'hui à la pointe de la technologie, compréhensible seulement par quelques rares initiés, sera connu et assimilé demain par des enfants avant leurs dix ans. Nous prétendons que la confidentialité ne sera jamais généralisée parce qu'elle est trop difficile à cerner. C'est vrai. Mais où commence-t-elle ?

Elle commence lorsque ceux qui ont le pouvoir de changer les choses se lèvent et remplissent leur rôle. Heureusement pour nous, cela n'implique pas de se lancer dans une bataille sanglante. Mais cela implique de sortir de notre zone de confort pour faire ce qui est juste, afin de protéger le monde pour nous-mêmes et les générations futures. Nous devons accomplir aujourd'hui ce qui est difficile pour le rendre

facile aux autres demain.

Jeune nerd à qui on vient de demander de sauver le monde, dessin de Simon « Dr Gee » Giraudot, Licence Creative Commons BY SA

Développeur ou développeuse, technophile… vous êtes le personnage principal de ce jeu et tout dépend de vos décisions et actions présentes. Il est trop fastidieux de gérer un petit serveur personnel ? Les générations futures ne seront jamais propriétaires de leurs données. Il est trop gênant d'utiliser une application de messagerie instantanée chiffrée, parce qu'elle est légèrement moins belle ? Les générations futures ne connaîtront jamais la confidentialité de leurs données. Vous trouvez qu'il est trop pénible d'installer une application *open source* sur votre propre serveur ? Alors les générations à venir ne profiteront jamais de la maîtrise libre de leurs données.

C'est à nous de nous lever et de faire ce qui est difficile pour le bien commun. Ce ne sera pas toujours aussi dur. C'est dur parce que c'est nouveau. Mais lorsque vous et vos ami·e·s, vos collègues et des dizaines de millions de développeurs et développeuses auront tous ensemble fait ce qui est difficile,

cela restera difficile pendant combien de temps, à votre avis ? Pas bien longtemps. Car comme c'est le cas avec les économies de marché, ces dizaines de millions de développeurs et développeuses deviendront un marché, aux besoins desquels il faudra répondre et à qui on vendra des produits. Ainsi pourra s'étendre et s'intensifier dans les consciences le combat pour la confidentialité.

Pas besoin d'attendre 10 ans pour que ça se produise. Pas besoin d'avoir dix millions de développeurs. C'est de vous qu'on a besoin.

- *Vous pouvez faire un premier pas en utilisant et soutenant les services qui assurent la confidentialité et la propriété des données par défaut. Vous pouvez aussi en faire profiter tout le monde : rendez-vous sur Framalibre, et ajoutez les outils libres et respectueux que vous connaissez, avec une brève notice informative.*

Plus rien ne marche, qu'est-ce qu'on fait ?

Désormais conscients et informés que nos actions et nos données en ligne sont faciles à espionner et l'enjeu de monétisation en coulisses, il nous restait l'espoir que quelques pans des technologies de sécurité pouvaient encore faire échec à la surveillance de masse et au profilage commercial. Pas facile pour les utilisateurs moyens d'adopter des outils et des pratiques de chiffrement, par exemple, cependant de toutes parts émergent des projets qui proposent de nous aider à y accéder sans peine.

Mais quand les experts en sécurité, quittant un moment leur regard hautain sur le commun des mortels à peine capables de choisir un mot de passe autre que 123AZERTY, avouent qu'ils savent depuis longtemps que tout est corrompu directement ou indirectement, jusqu'aux services soi-disant sécurisés et chiffrés, le constat est un peu accablant parce qu'il nous reste tout à reconstruire…

Plus rien ne fonctionne

*article original : Everything is broken par **Quinn Norton***

Traduction Framalang : Diab, rafiot, Omegax, Scailyna, Amine Brikci-N, EDGE, r0u, fwix, dwarfpower, sinma, Wan, Manu, Asta, goofy, Solarus, Lumi, mrtino, skhaen

Un beau jour un de mes amis a pris par hasard le contrôle de plusieurs milliers d'ordinateurs. Il avait trouvé une faille dans un bout de code et s'était mis à jouer avec. Ce faisant, il a trouvé comment obtenir les droits d'administration sur un réseau. Il a écrit un script, et l'a fait tourner pour voir ce que ça donnerait. Il est allé se coucher et il a dormi environ quatre heures. Le matin suivant, en allant au boulot, il a jeté un coup d'œil et s'est aperçu qu'il contrôlait désormais près de 50 000 ordinateurs. Après en avoir pratiquement vomi de trouille, il a tout arrêté et supprimé tous les fichiers associés. Il m'a dit que finalement il avait jeté le disque dur au feu. Je ne peux pas vous révéler de qui il s'agit, parce qu'il ne veut pas finir dans une prison fédérale ; et c'est ce qui pourrait lui arriver s'il décrivait à qui que ce soit la faille qu'il a découverte. Cette faille a-t-elle été corrigée ? Sans doute… mais pas par lui. Cette histoire n'est en rien exceptionnelle. Passez quelque temps dans le monde des hackers et de la sécurité informatique, et vous entendrez pas mal d'histoires dans ce genre et même pires que celle-là.

Il est difficile d'expliquer au grand public à quel point la technologie est chancelante, à quel point l'infrastructure de

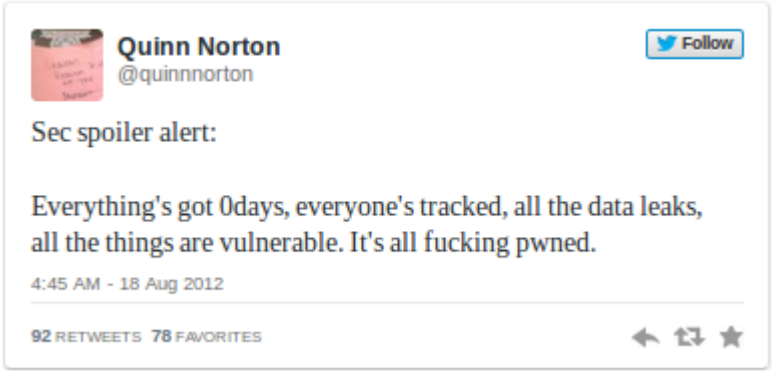
nos vies ne tient qu'avec l'équivalent informatique de bouts de ficelle. Les ordinateurs et l'informatique en général sont détraqués.

Quand c'est codé avec les pieds, bonjour les vautours

Pour un bon nombre d'entre nous, en particulier ceux qui ont suivi l'actualité en matière de sécurité et les questions d'écoutes sauvages, rien de surprenant dans toutes les dernières révélations. Si nous ne connaissions pas les détails, nous savions tous, dans le monde de la sécurité, que la technologie est vacillante et malade. Depuis des années nous voyons tourner les vautours qui veulent profiter de cet état de fait. La NSA n'est pas et n'a jamais été le grand prédateur unique fondant sur Internet. C'est simplement le plus gros de ces charognards. S'ils arrivent à aller aussi loin, ce n'est pas parce que leurs employés sont des dieux des maths.

Si la NSA s'en sort si bien, c'est parce que les logiciels en général sont merdiques.

Huit mois avant que Snowden ne fasse ses révélations, j'ai twitté ça :



« alerte de sécu : tout a une faille 0 day, tout le monde est suivi à la trace, toutes les données fuitent, tout est vulnérable, tout est compromis jusqu'à l'os. »

J’en étais arrivée à cette conclusion un peu désespérée : chercher des logiciels de qualité est un combat perdu d’avance. Comme ils sont écrits par des gens n’ayant ni le temps ni l’argent nécessaires, la plupart des logiciels sont publiés dès qu’ils fonctionnent assez bien pour laisser leurs auteurs rentrer chez eux et retrouver leur famille. Pour nous le résultat est épouvantable.

Si les logiciels sont aussi mauvais, c’est parce qu’ils sont très complexes, et qu’il cherchent à parler à d’autres logiciels, soit sur le même ordinateur, soit au travers du réseau. Même votre ordinateur ne peut plus être considéré comme unique : c’est une poupée russe, et chaque niveau est fait de quantité d’éléments qui essaient de se synchroniser et de parler les uns avec les autres. L’informatique est devenue incroyablement complexe, alors que dans le même temps les gens sont restés les mêmes, pétris de la même boue grise originelle pleine d’une prétention à l’étincelle divine.

Le merdier qu’est votre ordinateur sous Windows est tellement complexe que personne sur Terre ne sait tout ce qu’il fait vraiment, ni comment.

Maintenant imaginez des milliards de petites boites opaques qui essaient en permanence de discuter les unes avec les autres, de se synchroniser, de travailler ensemble, partageant des bouts de données, se passant des commandes… des tous petits bouts de programmes aux plus gros logiciels, comme les navigateurs – c’est ça, Internet. Et tout ça doit se passer quasi-simultanément et sans accrocs. Sinon vous montez sur vos grand chevaux parce que le panier de la boutique en ligne a oublié vos tickets de cinéma.

On n’arrête pas de vous rappeler que le téléphone avec lequel vous jouez à des jeux stupides et que vous laissez tomber dans les toilettes au troquet du coin est plus puissant que les ordinateurs utilisés pour la conquête de l’espace il y a de cela quelques décennies à peine. La NASA dispose d’une armée

de génies pour comprendre et maintenir ses logiciels. Votre téléphone n'a que vous. Ajoutez à cela un mécanisme de mises à jour automatiques que vous désactivez pour qu'il ne vous interrompe pas au beau milieu d'une séance de Candy Crush…

À cause de tout ça, la sécurité est dans un état effrayant. En plus d'être truffés de bugs ennuyeux et de boîtes de dialogue improbables, les programmes ont souvent un type de faille piratable appelée *0 day* (« zéro jour ») dans le monde de la sécurité informatique. Personne ne peut se protéger des *0 days*. C'est justement ce qui les caractérise : 0 représente le nombre de jours dont vous disposez pour réagir à ce type d'attaque. Il y a des *0 days* qui sont anodins et vraiment pas gênants, il y a des *0 days* très dangereux, et il y a des *0 days* catastrophiques, qui tendent les clés de la maison à toute personne qui se promène dans le coin. Je vous assure qu'en ce moment même, vous lisez ceci sur une machine qui a les trois types de *0days*. Je vous entends d'ici me dire : « Mais, Quinn, si personne ne les connaît comment peux-tu savoir que je les ai ? » C'est parce que même un logiciel potable doit avoir affaire avec du code affreux. Le nombre de gens dont le travail est de rendre le logiciel sûr peut pratiquement tenir dans un grand bar, et je les ai regardé boire. Ce n'est pas rassurant. La question n'est pas : « est-ce que vous allez être attaqué ? » mais : « quand serez-vous attaqué ? »

Considérez les choses ainsi : à chaque fois que vous recevez une mise à jour de sécurité (apparemment tous les jours avec mon ordi sous Linux), tout ce qui est mis à jour a été cassé, rendu vulnérable depuis on ne sait combien de temps. Parfois des jours, parfois des années. Personne n'annonce vraiment cet aspect des mises à jour. On vous dit « Vous devriez installer cela, c'est un patch critique ! » et on passe sous silence le côté « …parce que les développeurs ont tellement merdé que l'identité de vos enfants est probablement vendue en ce moment même à la mafia estonienne par des script kiddies accrocs à

l’héro ».

Les bogues vraiment dangereux (et qui peut savoir si on a affaire à eux lorsqu’on clique sur le bouton « Redémarrer ultérieurement » ?) peuvent être utilisés par des hackers, gouvernements, et d’autres horreurs du net qui fouillent à la recherche de versions de logiciels qu’ils savent exploiter. N’importe quel ordinateur qui apparaît lors de la recherche en disant « Hé ! Moi ! Je suis vulnérable ! » peut faire partie d’un botnet, en même temps que des milliers, ou des centaines de milliers d’autres ordinateurs. Souvent les ordinateurs zombies sont possédés à nouveau pour faire partie d’un autre botnet encore. Certains botnets patchent les ordinateurs afin qu’ils se débarrassent des autres botnets, pour qu’ils n’aient pas à vous partager avec d’autres hackers. Comment s’en rendre compte si ça arrive ? Vous ne pouvez pas ! Amusez-vous à vous demander si votre vie en ligne va être vendue dans l’heure qui suit ! La prochaine fois que vous penserez que votre grand-mère n’est pas cool, pensez au temps qu’elle a passé à aider de dangereux criminels russes à extorquer de l’argent à des casinos offshore avec des attaques DDoS.

Récemment un hacker anonyme a écrit un script qui prenait le contrôle d’appareils embarqués Linux. Ces ordinateurs possédés scannaient tout le reste d’Internet et ont créé un rapport qui nous en a appris beaucoup plus que ce que nous savions sur l’architecture d’Internet. Ces petites boîtes hackées ont rapporté toutes leurs données (un disque entier de 10 To) et ont silencieusement désactivé le hack. C’était un exemple délicieux et utile d’un individu qui a hacké la planète entière. Si ce malware avait été véritablement malveillant, nous aurions été dans la merde.

Et ceci parce que les ordinateurs sont tous aussi inévitablement défectueux : ceux des hôpitaux et des gouvernements et des banques, ceux de votre téléphone, ceux qui contrôlent les feux de signalisation et les capteurs et les systèmes de contrôle du trafic aérien. Chez les

industriels, les ordinateurs destinés à maintenir l'infrastructure et la chaîne de fabrication sont encore pires. Je ne connais pas tous les détails, mais ceux qui sont les plus au courant sont les personnes les plus alcooliques et nihilistes de toute la sécurité informatique. Un autre de mes amis a accidentellement éteint une usine avec un '"ping"' malformé au début d'un test d'intrusion. Pour ceux qui ne savent pas, un '"ping"' est seulement la plus petite requête que vous pouvez envoyer à un autre ordinateur sur le réseau. Il leur a fallu une journée entière tout faire revenir à la normale.

Les experts en informatique aiment prétendre qu'ils utilisent des logiciels d'un genre complètement différent, encore plus géniaux, qu'eux seuls comprennent, des logiciels faits de perfection mathématique et dont les interfaces semblent sortir du cul d'un âne colérique. C'est un mensonge. La forme principale de sécurité qu'ils offrent est celle que donne l'obscurité – il y a si peu de gens qui peuvent utiliser ces logiciels que personne n'a le moindre intérêt à concevoir des outils pour les attaquer. Sauf si, comme la NSA, vous voulez prendre le contrôle sur les administrateurs systèmes.

Une messagerie chiffrée et bien codée, il ne peut rien nous arriver, hein ?

Prenons un exemple que les experts aiment mettre sous le nez des gens normaux qui ne l'utilisent pas : OTR. OTR, ou *Off The Record messaging*, ajoute une couche de chiffrement aux échanges via messagerie instantanée. C'est comme si vous utilisiez AIM ou Jabber et que vous parliez en code sauf que c'est votre ordinateur qui fait le code pour vous. OTR est bien conçu et robuste, il a été audité avec attention et nous sommes bien sûrs qu'il ne contient aucune de ces saloperies de vulnérabilités zéro jour.

Sauf que OTR n'est pas vraiment un programme que vous utilisez

tel quel.

Il existe un standard pour le logiciel OTR, et une bibliothèque, mais elle ne fait rien par elle-même. OTR est implémentée dans des logiciels pour des neuneus par d'autres neuneus. À ce stade, vous savez que ça va se terminer dans les pleurs et les grincements de dents.

La partie principale qu'utilise OTR est un autre programme qui utilise une bibliothèque appelée ‘“libpurple”’. Si vous voulez voir des snobs de la sécurité aussi consternés que les ânes qui ont pondu leur interface, apportez-leur ‘“libpurple”’. ‘“Libpurple”’ a été écrit dans un langage de programmation appelé C.

Le C est efficace dans deux domaines : l'élégance, et la création de vulnérabilités jour zéro critiques en rapport avec la gestion de la mémoire.

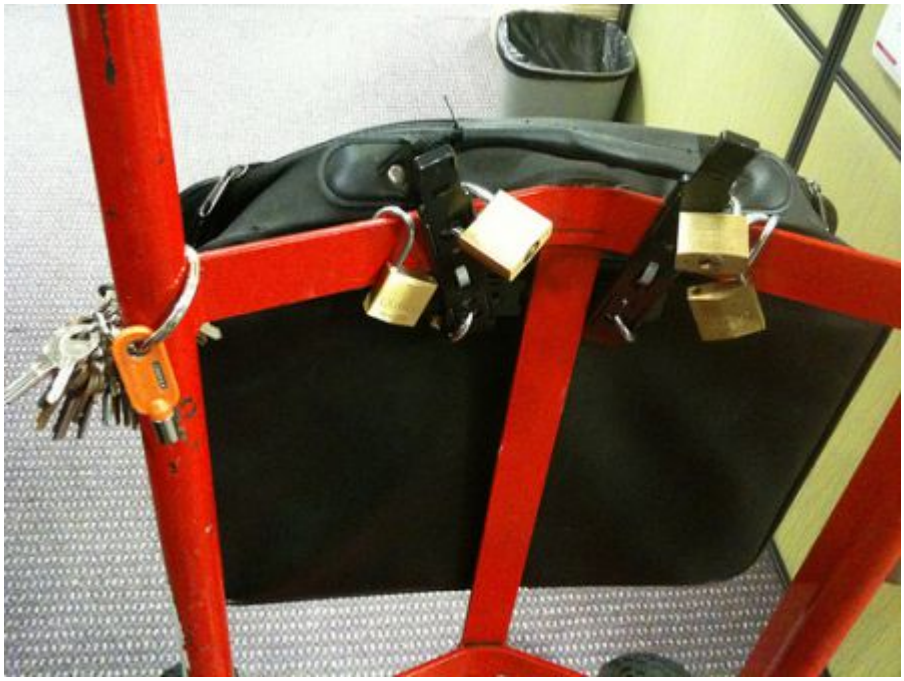
Heartbleed, le bogue qui a affecté le monde entier, permettant la fuite de mots de passe et de clés de chiffrement et qui sait quoi encore ? — Du classique et superbe C.

La ‘“libpurple”’ a été écrite par des gens qui voulaient que leur client de discussion *open source* parle à tous les systèmes de messagerie instantanée du monde, et se foutaient complètement de la sécurité ou du chiffrement. Des gens du milieu de la sécurité qui en ont examiné le code ont conclu qu'il y avait tellement de façons d'exploiter la ‘“libpurple”’ que ça n'était probablement pas la peine de la patcher. Elle doit être jetée et réécrite de zéro. Ce ne sont pas des bugs qui permettent à quelqu'un de lire vos messages chiffrés, ce sont des bugs qui permettent à n'importe qui de prendre le contrôle total de votre ordinateur, regarder tout ce que vous tapez ou lisez et même probablement vous regarder vous mettre les doigts dans le nez devant la webcam.

Ce magnifique outil qu'est OTR repose sur la ‘“libpurple”’ dans la plupart des systèmes où il est utilisé. Je dois

éclaircir un point, car même certains geeks n'en ont pas conscience : peu importe la force de votre chiffrement si celui qui vous attaque peut lire vos données par-dessus votre épaule, et je vous promets que c'est possible. Qu'il sache le faire ou pas encore, cela reste néanmoins possible. Il y a des centaines de bibliothèques comme '"libpurple"' sur votre ordinateur : des petits bouts de logiciels conçus avec des budgets serrés aux délais irréalistes, par des personnes ne sachant pas ou ne se souciant pas de préserver la sécurité de votre système.

Chacun de ces petits bugs fera l'affaire quand il s'agit de prendre le contrôle de tout le reste de votre ordinateur. Alors on met à jour, on remet à jour, et peut-être que ça mettra les intrus dehors, ou peut-être pas. On n'en sait rien ! Quand on vous dit d'appliquer les mises à jour, on ne vous dit pas de réparer votre navire. On vous dit de continuer à écoper avant que l'eau n'atteigne votre cou.



(Crédit image : sridgway, licence CC BY 2.0)

Pour prendre un peu de recul par rapport à cette scène d'horreur et de désolation, je dois vous dire que la situation est tout de même meilleure que par le passé. Nous disposons aujourd'hui d'outils qui n'existaient pas dans les années 90,

comme le ‘“sandboxing”’, qui permet de confiner des programmes écrits stupidement là où ils ne peuvent pas faire beaucoup de dégâts. (Le « sandboxing » consiste à isoler un programme dans une petite partie virtuelle de l’ordinateur, le coupant ainsi de tous les autres petits programmes, ou nettoyant tout ce que ce programme essaie de faire avant que d’autres puissent y accéder).

Des catégories entières de bugs horribles ont été éradiqués comme la variole. La sécurité est prise plus au sérieux que jamais, et il y a tout un réseau de personnes pour contrer les logiciels malveillants 24h sur 24. Mais ils ne peuvent pas vraiment garder la main. L’écosystème de ces problèmes est tellement plus vaste qu’il ne l’était ne serait-ce qu’il y a dix ans, qu’on ne peut pas vraiment dire que l’on fait des progrès.

Les gens, eux aussi, sont cassés

« Je vous fais confiance… » est ce que j’aime le moins entendre de la part des mes sources Anonymous. C’est invariablement suivi de bribes d’informations qu’ils n’auraient jamais dû me confier. Il est naturel de partager quelque chose de personnel avec quelqu’un en qui on a confiance. Mais c’est avec exaspération que je dois rappeler aux Anons qu’avant d’être connectés à un autre être humain ils sont d’abord connectés à un ordinateur, relayé à travers un nombre indéterminé de serveurs, switches, routeurs, câbles, liaisons sans fil, et en bout de chaîne, mon ordinateur parfaitement ciblé par les attaques. Tout ceci se déroule le temps d’une longue inspiration. Cela semble une évidence, mais il est bon de le rappeler : les humains ne sont pas conçus pour penser de cette manière.

Personne n’arrive à utiliser les logiciels correctement. Absolument tout le monde se plante. OTR ne chiffre pas avant le premier message, un fait que des éminents professionnels de la sécurité et des hackers qui subissent une chasse à l’homme

dans une vingtaine de pays oublient en permanence. Gérer toutes les clés de chiffrement et de déchiffrement dont vous avez besoin pour garder vos données en sûreté sur plusieurs appareils, sites, et comptes est théoriquement possible, de la même façon que réaliser une appendicectomie sur soi-même est théoriquement possible. *Il y a un gars qui a réussi à le faire en Antarctique, pourquoi pas moi, hein ?*

Tous les experts en programmes malveillants que je connais ont un jour oublié ce que faisait là un certain fichier, ont cliqué dessus pour le voir et ensuite compris qu'ils avaient exécuté un quelconque logiciel malveillant qu'ils étaient censés examiner. Je sais cela parce que ça m'est arrivé une fois avec un PDF dans lequel je savais qu'il y avait quelque chose de mauvais. Mes amis se sont moqués de moi, puis m'ont tous confessé discrètement qu'ils avaient déjà fait la même chose. Si quelques-uns des meilleurs spécialiste de rétro-ingénierie de logiciels malveillants ne peuvent surveiller leurs fichiers malveillants, qu'espérer de vos parents avec cette carte postale électronique qui est prétendument de vous ?

Les pièces jointes exécutables (ce qui inclut les documents Word, Excel, et les PDF) des emails que vous recevez chaque jour peuvent provenir de n'importe qui (on peut écrire à peu près ce que l'on veut dans le champ « De : » d'un email) et n'importe laquelle de ces pièces jointes pourrait prendre le contrôle de votre ordinateur aussi facilement qu'une vulnérabilité jour zéro. C'est certainement de cette façon que votre grand-mère s'est retrouvée à travailler pour des criminels russes, ou que vos concurrents anticipent tous vos plans produits. Mais dans le monde d'aujourd'hui, vous ne pourrez sûrement pas conserver un emploi de bureau si vous refusez d'ouvrir des pièces jointes. Voilà le choix qui s'offre à vous : prendre en permanence le risque de cliquer sur un dangereux programme malveillant, ou vivre sous un pont, laissant sur la pelouse de votre ancienne maison des messages

pour dire à vos enfants combien vous les aimez et combien ils vous manquent.

Les experts de la sécurité et de la vie privée sermonnent le public à propos des métadonnées et des réseaux d'échange de données, mais prendre en compte ces choses est aussi naturel que de se faire une batterie de tests sanguins tous les matins, et à peu près aussi facile. Les risques sur le plan sociétal de renoncer à notre vie privée sont énormes. Et pourtant, les conséquences pour chacun de ne pas y renoncer sont immédiatement handicapantes. Il s'agit au final d'un combat d'usure entre ce que l'on veut pour nous-mêmes et nos familles, et ce que l'on doit faire pour vivre dans notre communauté en tant qu'humains – un champ de mines monétisé par les entreprises et monitoré par les gouvernements.

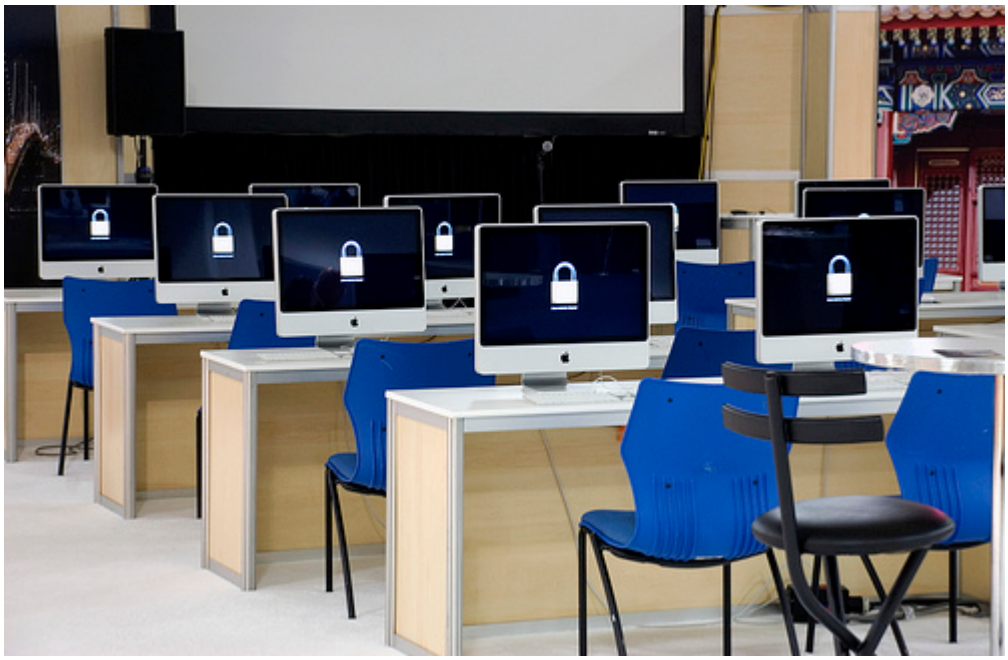
Je travaille en plein là-dedans, et je ne m'en sors pas mieux. J'ai dû une fois suivre un processus pour vérifier mon identité auprès d'un informateur méfiant. J'ai dû prendre une série de photos montrant où je me trouvais ainsi que la date. Je les ai mises en ligne, et on m'a permis de procéder à l'interview. Au final, il se trouve qu'aucune de ces vérifications n'avait été envoyées, parce que j'avais oublié d'attendre la fin du chargement avant d'éteindre nerveusement mon ordinateur. « Pourquoi m'avez-vous quand même permis de vous voir ? » demandais-je à ma source. « Parce qu'il n'y a que vous qui pourrait faire une chose aussi stupide », m'a-t-il répondu.

Touché.

Mais si cela m'arrive à moi, une adulte relativement bien entraînée qui fait attention à ce genre de sujets systématiquement, quelle chance ont les gens avec de vrais boulots et de vraies vies ?

Finalement, c’est la culture qui est cassée.

Il y a quelques années, j’ai rencontré plusieurs personnes respectées qui travaillent dans la confidentialité et la sécurité logicielle et je leur ai posé une question. Mais d’abord j’ai dû expliquer quelque chose : « La plupart des gens n’ont pas de droits d’administration sur les ordinateurs qu’ils utilisent. »



(Crédit image : amelungc, licence CC BY 2.0)

C’est-à-dire que la plupart des gens qui utilisent un ordinateur dans le monde n’en sont pas propriétaires… Que ce soit dans un café, à l’école, au travail, installer une application bureautique n’est pas directement à la portée d’une grande partie du monde. Toute les semaines ou toutes les deux semaines, j’étais contacté par des gens prêts à tout pour améliorer la sécurité et les options de confidentialité, et j’ai essayé de leur apporter mon aide. Je commençais par « Téléchargez le… » et on s’arrêtait là. Les gens me signalaient ensuite qu’ils ne pouvaient pas installer le logiciel sur leur ordinateur. En général parce que le département informatique limitait leurs droits dans le cadre de la gestion du réseau. Ces gens avaient besoin d’outils qui

marchaient sur ce à quoi ils avaient accès, principalement un navigateur.

Donc la question que j'ai posée aux hackers, cryptographes, experts en sécurité, programmeurs, etc. fut la suivante : quelle est la meilleure solution pour les gens qui ne peuvent pas télécharger de nouveau logiciel sur leurs machines ? La réponse a été unanime : aucune. Il n'y a pas d'alternative. On me disait qu'ils feraient mieux de discuter en texte brut, « comme ça ils n'ont pas un faux sentiment de sécurité ». À partir du moment où ils n'ont pas accès à de meilleurs logiciels, ils ne devraient pas faire quoi que ce soit qui puisse déranger les gens qui les surveillent. Mais, expliquais-je, il s'agit d'activistes, d'organisateurs, de journalistes du monde entier qui ont affaire à des gouvernements et des sociétés et des criminels qui peuvent vraiment leur faire du mal, ces gens sont vraiment en danger. On me répondait alors que dans ce cas, ils devraient s'acheter leurs propres ordinateurs.

Et voilà, c'était ça la réponse : être assez riche pour acheter son propre ordinateur, ou bien littéralement tout laisser tomber. J'ai expliqué à tout le monde que ce n'était pas suffisant, j'ai été dénigrée lors de quelques joutes verbales sans conséquences sur Twitter, et je suis passée à autre chose. Peu de temps après, j'ai compris d'où venait l'incompréhension. Je suis retourné voir les mêmes experts et j'ai expliqué : dans la nature, dans des situations vraiment dangereuses – même quand les gens sont traqués par des hommes avec des armes – quand le chiffrement et la sécurité échouent, personne n'arrête de parler. Ils espèrent seulement ne pas se faire prendre.

La même impulsion humaine qui nous pousse vers le hasard et les loteries depuis des milliers d'années soutient ceux qui luttent même quand les chances sont contre eux. « Peut-être bien que je m'en sortirai, autant essayer ! » Pour ce qui est de l'auto-censure des conversations dans une infrastructure

hostile, les activistes non techniques s'en sortent de la même manière que les Anons, ou que les gens à qui l'on dit de se méfier des métadonnées, ou des réseaux d'échanges de données, ou de ce premier message avant que l'encodage OTR ne s'active. Ils foirent.

Cette conversation a été un signal d'alerte pour quelques personnes de la sécurité qui n'avaient pas compris que les personnes qui devenaient activistes et journalistes faisaient systématiquement des choses risquées. Certains ont rallié mon camp, celui où on perd son temps à des combats futiles sur Twitter et ils ont pris conscience que quelque chose, même quelque chose d'imparfait, pouvait être mieux que rien. Mais beaucoup dans le domaine de la sécurité sont toujours dans l'attente d'un monde parfait dans lequel déployer leur code parfait.

Alors apparaît l'*Intelligence Community* (Communauté du renseignement), ils s'appellent entre eux le IC. Nous pourrions trouver ça sympathique s'ils arrêtaient d'espionner tout le monde en permanence, et eux aimeraient bien que l'on cesse de s'en plaindre. Après avoir passé un peu de temps avec eux, je pense savoir pourquoi ils ne se préoccupent pas de ceux qui se plaignent. Les IC font partie des humains les plus surveillés de l'histoire. Ils savent que tout ce qu'ils font est passé au peigne fin par leurs pairs, leurs patrons, leurs avocats, d'autres agences, le président, et parfois le Congrès. Ils vivent surveillés, et ne s'en plaignent pas.

Dans tous les appels pour augmenter la surveillance, les fondamentaux de la nature humaine sont négligés. Vous n'allez pas apprendre aux espions que ce n'est pas bien en faisant encore plus qu'eux. Il y aura toujours des failles, et tant qu'elles existeront ou pourront être utilisées ou interprétées, la surveillance sera aussi répandue que possible. Les humains sont des créatures généralement égocentriques. Les espions, qui sont humains, ne comprendront jamais pourquoi vivre sans vie privée est mal aussi longtemps

qu’ils le feront.

Et pourtant ce n’est pas cela le pire. La catastrophe culturelle qu’ils provoquent rend plus facile leur boulot d’épier le monde. Les aspects les plus dérangeants des révélations, ce sont le marché des failles *0 day*, l’accumulation des moyens de les exploiter, l’affaiblissement des standards. La question est de savoir qui a le droit de faire partie de ce « nous » qui est censé être préservé de ces attaques, écoutes et décryptages et profilages. Quand ils ont attaqué Natanz avec Stuxnet et laissé tous les autres centres nucléaires vulnérables, nous avons été tranquillement avertis que le « nous » en question commençait et finissait avec l’IC lui-même. Voilà le plus grand danger.

Quand le IC ou le DOD ou le pouvoir exécutif sont les seuls vrais Américains, et que le reste d’entre nous ne sommes que des Américains de deuxième classe, ou pire les non-personnes qui ne sont pas associées aux États-Unis, alors nous ne pouvons que perdre toujours plus d’importance avec le temps. À mesure que nos désirs entrent en conflit avec le IC, nous devenons de moins en moins dignes de droits et de considération aux yeux du IC. Quand la NSA accumule des moyens d’exploiter les failles, et que cela interfère avec la protection cryptographique de notre infrastructure, cela veut dire qu’exploiter des failles contre des gens qui ne sont pas de la NSA ne compte pas tellement. Nous sécuriser passe après se sécuriser eux-mêmes.

En théorie, la raison pour laquelle nous sommes si gentils avec les soldats, que nous avons pour habitude d’honorer et de remercier, c’est qu’ils sont supposés se sacrifier pour le bien des gens. Dans le cas de la NSA, l’inverse s’est produit. Notre bien-être est sacrifié afin de rendre plus aisé leur boulot de surveillance du monde. Lorsque cela fait partie de la culture du pouvoir, on est en bonne voie pour que cela débouche sur n’importe quel abus.

Mais le plus gros de tous les problèmes culturels repose toujours sur les épaules du seul groupe que je n'aie pas encore pris à partie – les gens normaux, qui vivent leurs vies dans cette situation démentielle. Le problème des gens normaux avec la technologie est le même qu'avec la politique, ou la société en général. Les gens pensent être isolés et sans pouvoir, mais la seule chose qui maintient les gens seuls et sans pouvoir est cette même croyance. Ceux qui travaillent ensemble ont un énorme et terrible pouvoir. Il existe certainement une limite à ce que peut faire un mouvement organisé de personnes qui partagent un rêve commun, mais nous ne l'avons pas encore trouvée.

Facebook et Google semblent très puissants, mais ils vivent à peu près à une semaine de la ruine en permanence. Ils savent que le coût de départ des réseaux sociaux pris individuellement est élevé, mais sur la masse, c'est une quantité négligeable. Windows pourrait être remplacé par quelque chose de mieux écrit. Le gouvernement des États-Unis tomberait en quelques jours devant une révolte générale. Il n'y aurait pas besoin d'une désertion totale ou d'une révolte générale pour tout changer, car les sociétés et le gouvernement préfèreraient se plier aux exigences plutôt que de mourir. Ces entités font tout ce qu'elles peuvent pour s'en sortir en toute impunité – mais nous avons oublié que nous sommes ceux qui les laissons s'en sortir avec ces choses.

Si les ordinateurs ne satisfont pas nos besoins de confidentialité et de communication, ce n'est pas en raison d'une quelconque impossibilité mathématique. Il existe un grand nombre de systèmes qui pourraient chiffrer nos données de façon sécurisée et fédérée, nous disposons de nombreuses façons de retrouver la confidentialité et d'améliorer le fonctionnement par défaut des ordinateurs. Si ce n'est pas ainsi que les choses se passent en ce moment c'est parce que nous n'avons pas exigé qu'il en soit ainsi, et non pas parce que personne n'est assez malin pour que ça arrive.

C’est vrai, les geeks et les PDG et les agents et les militaires ont bousillé le monde. Mais en fin de compte, c’est l’affaire de tous, en travaillant ensemble, de réparer le monde.

Sauvegardes et garde-fous (Libres conseils 9/42)

Chaque jeudi à 21h, rendez-vous sur le framapad de traduction, le travail collaboratif sera ensuite publié ici même.

Traduction Framalang : Sky, LIAR, lerouge, yann, Goofy, peupleLa, KoS, Nys, Julius22, okram, 4nti7rust, zn01wr, lamessen

Des sauvegardes pour votre santé mentale

Austin Appel

Austin Appel, alias « scorche », est un professionnel de la sécurité informatique qui passe son temps à casser (il est dûment autorisé, évidemment) des choses précédemment réputées sécurisées. On le croise souvent enseignant le crochetage de serrure durant des conférences de sécurité et de hacking. Dans le monde de l’open source, il fait une foule de choses pour le projet Rockbox et a œuvré bénévolement pour le projet One Laptop Per Child (un ordinateur portable par enfant).

Les sauvegardes c’est bien. Les sauvegardes c’est super. Un administrateur compétent fait toujours des sauvegardes régulières. On apprend ça dans n’importe quel manuel traitant

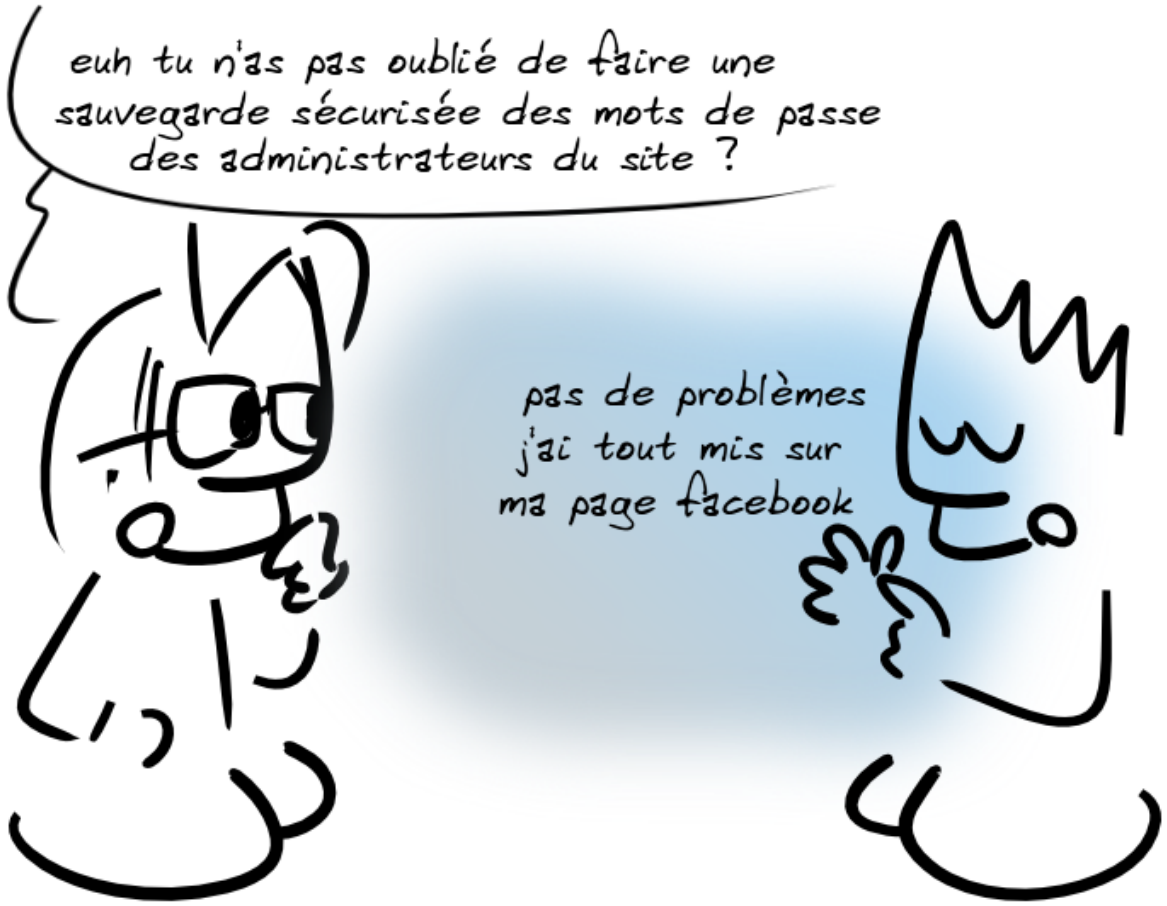
de l'administration des serveurs. Le problème c'est que les sauvegardes ne sont vraiment utiles qu'en cas d'absolue nécessité. Lorsque quelque chose de grave arrive au serveur ou à ses données et qu'on est forcé de se replier sur autre chose, les sauvegardes viendront à point nommé. Cependant, cela ne devrait jamais arriver, n'est-ce pas ? À n'importe quel autre moment, à quoi cela sert-il pour vous et votre environnement serveur d'avoir des sauvegardes ?

Avant d'aller plus loin, il est important de noter que ce conseil vaut pour les administrateurs serveurs des plus petits projets *open source* – la majorité silencieuse. Si vous maintenez des services qui vont engendrer une grande frustration, et même peut-être faire du tort s'ils sont indisponibles, vous devriez considérer ceci avec la plus grande circonspection.

Pour le reste d'entre nous qui travaillons sur d'innombrables petits projets ayant des ressources limitées, nous avons rarement deux serveurs séparés pour la production et les tests. En vérité, avec tous les services qu'un projet open source doit maintenir (système de gestion de version, services web, listes de diffusion, forums, ferme de compilation, bases de données, traceurs de bogues ou de fonctionnalités, etc.), des environnements de test séparés sont souvent de l'ordre du rêve. Malheureusement, l'approche courante de l'administration systèmes est d'avancer avec précaution et mettre les systèmes à jour uniquement en cas de nécessité absolue, afin d'éviter tout problème de dépendance, de code cassé, ou n'importe laquelle des millions de choses qui pourraient mal se dérouler. La raison pour laquelle vous êtes nerveux n'est pas que vous pourriez manquer d'expérience. Il est important de savoir que vous n'êtes pas seul dans ce cas. Que nous l'admettions ou non, beaucoup d'entre nous ont été (et sont probablement encore) dans cette situation. Il est triste que cette inaction – découlant de la peur de détruire un système fonctionnel – conduise souvent à des services en

fonctionnement qui ont souvent plusieurs versions de retard, ce qui implique de nombreuses failles de sécurité potentiellement sérieuses. Cependant, soyez assuré que ce n'est pas la seule manière de jouer le jeu.

Les gens ont tendance à jouer un jeu différent selon qu'ils aient une infinité de vies ou qu'ils doivent recommencer depuis le début dès lors qu'une seule erreur a été commise. Pourquoi devrait-il en être autrement pour de l'administration systèmes ? Aborder le concept de sauvegardes avec un état d'esprit offensif peut complétement changer votre conception de l'administration systèmes. Au lieu de vivre dans la peur d'une *dist-upgrade* complète (ou de son équivalent pour yum, pacman, etc.), celui qui est armé de sauvegardes est libre de mettre à jour les paquets d'un serveur, confiant dans le fait que ces changements pourront être annulés si les choses tournent au vinaigre. La clé du succès réside tout entière dans l'état d'esprit. Il n'y a aucune raison d'avoir peur tant que vous avez vos données sauvegardées sous la main comme filet de sécurité lorsque vous sautez le pas. Après tout, l'administration système est une expérience d'apprentissage permanente.



Bien sûr, si vous ne validez pas vos sauvegardes, vous reposer sur elles devient un jeu très dangereux. Heureusement, les administrateurs systèmes expérimentés savent que le commandement « Garde des sauvegardes à jour » est toujours suivi par « Valide tes sauvegardes ». À nouveau, c'est un mantra que les gens aiment réciter. Ce qui, en revanche, ne tient pas de façon élégante dans un mantra entraînant est la manière de valider rapidement et simplement ses sauvegardes. La meilleure manière de dire qu'une sauvegarde est fonctionnelle est, bien sûr, de la restaurer (de préférence sur un système identique qui n'est pas en cours d'utilisation). Mais, une fois encore, en l'absence d'un tel luxe, on doit faire preuve d'un peu plus de créativité. C'est là (tout du moins pour les fichiers) que les sommes de contrôle peuvent vous aider à vérifier l'intégrité de vos fichiers sauvegardés. Dans *rsync*, par exemple, la méthode utilisée par défaut pour déterminer quels fichiers ont été

modifiés consiste à regarder la date et l'heure de la dernière modification, ainsi que la taille du fichier. Cependant, en utilisant l'option '-c', rsync utilisera une somme de contrôle MD4 de 128 bits pour déterminer si les fichiers ont changé ou non. Bien que ce ne soit pas toujours la meilleure idée à mettre en œuvre à chaque fois en toute occasion – à cause d'un temps d'exécution beaucoup plus long qu'un rsync normal et d'une utilisation accrue des accès disques – cette méthode permet de s'assurer que les fichiers sont intègres.

Le rôle d'un administrateur systèmes peut être éprouvant par moments. Il n'est cependant pas nécessaire de le rendre plus stressant que nécessaire. Avec le bon état d'esprit, certaines commandes de précaution apparemment à but unique et limité peuvent être utilisées comme des outils précieux qui vous permettent de progresser de façon agile, tout en gardant votre santé mentale intacte et la vitesse tant appréciée dans les projets open source.

Être administrateur systèmes : ne pas s'enfermer dans une spécialité ? (Libres conseils 8/42)

Chaque jeudi à 21h, rendez-vous sur le framapad de traduction, le travail collaboratif sera ensuite publié ici même.

Traduction Framalang :lerouge, lamessen, CoudCoud, Kev, peupleLa (relectures),Goofy, Jej, Julius22, kalupa, 4nti7rust, ga3lig, Tsigorf, maat

Aimer l'inconnu

Jeff Mitchell

Jeff Mitchell passe ses journées de travail à s'activer sur tout ce qui touche aux ordinateurs et aux réseaux et son temps libre à barboter dans toutes sortes de projets de logiciels libres et open source. Ce qu'il préfère c'est la convergence des deux. Après avoir travaillé en tant qu'administrateur systèmes professionnel de 1999 à 2005, il maintient son niveau de compétences en les mettant bénévolement au service de projets libres en divers lieux. Ces temps-ci, son activité pour le Libre est dédiée à l'administration systèmes pour KDE[1] et c'est l'un des développeurs principaux du lecteur Tomahawk[2]. Jeff vit actuellement à Boston, aux États-Unis.

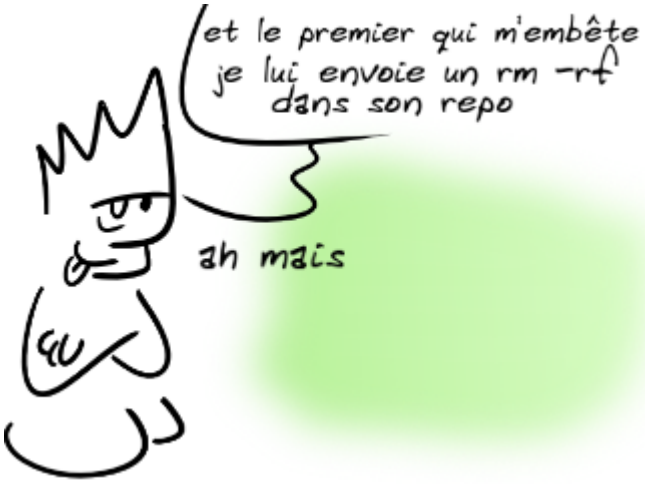
Récemment, à mon travail, j'ai fait partie d'une équipe qui faisait passer les entretiens d'embauche pour un poste d'administrateur systèmes. Après avoir parcouru quelques dizaines de curriculum vitae nous avons finalement convoqué notre premier candidat. Celui-ci – appelons-le John – avait aussi bien l'expérience de petites structures, style laboratoire informatique, que de plus vastes opérations dans des centres de données. À première vue, les choses se présentaient bien, si ce n'est qu'il avait eu cette réponse bizarre à quelques-unes de nos questions : « je suis administrateur systèmes ». Le sens de cette phrase n'a pas été immédiatement clair pour nous, jusqu'à ce que l'échange suivant ait lieu :

> *Moi : Donc, vous avez dit que vous n'avez pas d'expérience avec Cisco IOS, mais qu'en est-il des réseaux en général ?*
>
> *John : Eh bien, je suis administrateur systèmes.*
>
> *Moi : Oui, mais que diriez-vous sur les concepts de réseau ? Les protocoles de routage comme BGP ou OSPF, les VLANs, les*

ponts réseaux…

John, exaspéré : Je suis administrateur systèmes.

C’est à ce moment-là que nous avons compris ce qu’il voulait dire. John ne nous disait pas qu’il connaissait toutes ces choses que nous lui demandions puisqu’il était administrateur systèmes ; il nous expliquait que *parce qu*‘il était administrateur systèmes, il n’en savait rien. John était administrateur systèmes et cela signifiait pour lui que ces tâches étaient celles d’un administrateur réseau. Sans surprise, John n’a pas obtenu le poste.



Dans bien des projets *open source*, la spécialisation est une malédiction et non une bénédiction. Qu’un projet relève d’une catégorie ou l’autre dépend souvent de la taille de l’équipe de développement ; la spécialisation à l’extrême peut entraîner de graves perturbations dans un projet en cas de départ d’un développeur, que ce soit en bons ou mauvais termes, qu’on le regrette ou non. Il en va de même pour les administrateurs systèmes de projets *open source*, bien que la pénurie générale de ces derniers semble autoriser aux projets une marge de tolérance parfois dangereuse.

L’exemple le plus flagrant qui me vienne à l’esprit impliquait un projet spécifique dont le site de documentation (y compris toute celle de l’installation et de la configuration) était indisponible depuis plus d’un mois. La raison : le serveur

était en panne et la seule personne qui en avait l'accès naviguait sur un « bateau pirate » avec les membres du parti pirate suédois. C'est une histoire vraie.

Cependant, tous les points de défaillance ne sont pas dus à l'absence des administrateurs systèmes ; certains sont artificiels. Sur un gros projet, les décisions des droits d'accès à l'administration systèmes étaient assumées par un seul administrateur. Il ne s'était pas seulement réservé certains droits d'accès uniquement pour lui-même (vous l'avez deviné : oui, il a disparu pendant un certain temps, et oui, cela a causé des problèmes) ; il avait aussi décidé de la façon dont les droits d'accès devaient être accordés, en fonction de la confiance qu'il portait personnellement au candidat. La « confiance », dans ce cas, se fondait sur une seule chose : non pas le nombre de membres de la communauté qui se portait garant pour cette personne, ni depuis combien de temps cette personne était un contributeur actif et de confiance pour le projet, ni même depuis combien de temps il connaissait lui-même cette personne dans le cadre de ce projet. Au lieu de cela, elle reposait sur la façon dont il connaissait personnellement quelqu'un, ce par quoi il entendait la façon dont il connaissait cet individu en personne. Vous imaginez bien à quel point cela est adapté à une équipe d'administrateurs systèmes disséminée sur toute la planète…

Bien sûr, cet exemple ne fait qu'illustrer la grande difficulté pour un administrateur systèmes *open source* de trouver le juste milieu entre sécurité et capacité. Les grandes entreprises peuvent se permettre d'avoir du personnel redondant, et ce, même si le travail se répartit selon différentes responsabilités ou domaines de sécurité. La redondance est importante. Mais qu'en est-il si la seule possibilité d'avoir une redondance pour l'administrateur systèmes est de prendre la première personne se présentant au hasard sur votre canal IRC ou une personne quelconque

proposant son aide ? Comment pouvez-vous raisonnablement avoir confiance en cette personne, ses capacités et sa motivation ? Malheureusement, seuls les contributeurs principaux du projet ou une petite partie d'entre eux peuvent savoir quand la bonne personne se présente en utilisant le même modèle de toile de confiance[3] qui sous-tend une grande partie du reste du monde *open source*. L'univers des projets *open source*, leurs besoins et les personnes qui veulent contribuer à un projet particulier forment une extraordinaire diversité ; par conséquent, la dynamique humaine, la confiance, l'intuition et la manière d'appliquer ces concepts à un projet open source sont de vastes sujets, bien au-delà de la thématique de ce court article.

Une chose importante a cependant facilité la découverte de cette ligne d'équilibre sécurité/capacité : l'essor des systèmes de gestion de versions distribués, ou DVCS (NdT : Distributed Version Control System, système de gestion de version distribué). Auparavant, les contrôles d'accès étaient primordiaux car le cœur de tout projet *open source* – son code source – était centralisé. Je me rends bien compte que beaucoup doivent penser : « Jeff, tu devrais pourtant le savoir, le cœur d'un projet, c'est sa communauté, pas son code ! ». Ma réponse est simple : les membres de la communauté vont et viennent, mais, si quelqu'un fait accidentellement un « rm -rf » sur tout l'arbre du système de gestion de versions de votre projet et que vous manquez de sauvegardes, combien de ces membres de la communauté vont continuer à s'investir dans le projet et aider à tout recommencer à zéro ? (Mes propos se basent sur une histoire vraie dans laquelle un membre de la communauté saoul qui s'énervait à déboguer un bout de code, lança un « rm -rf » sur toute sa contribution, avec l'intention de supprimer tout le code du projet. Par chance, il n'était pas administrateur systèmes et n'avait donc pas accès au dépôt central, et il était trop saoul pour se rappeler qu'il travaillait seulement sur une copie du projet.)

Le code du projet est son cœur ; les membres de sa communauté en sont l'énergie vitale. Privé de l'un ou de l'autre, vous aurez du mal à garder un projet vivant. Avec un logiciel de gestion de version (NdT : VCS pour version control system) centralisé, si vous n'avez pas eu la présence d'esprit de mettre en place un système de sauvegarde régulier, vous pourriez, avec de la chance, ré-assembler l'arborescence complète du code source à partir des différents éléments contribués qu'auront gardés les autres personnes. Mais pour la majorité des projets, l'historique du code est aussi important que le code lui-même et cela, vous l'aurez tout de même entièrement perdu.

Ce n'est plus le cas désormais. Quand tous les clones locaux ont tout l'historique du projet et que des sauvegardes de secours peuvent être effectuées chaque nuit, en lançant une tâche planifiée aussi simple que « git pull », les dépôts centralisés ne sont plus que des outils de coordination. Cela en diminue l'importance de quelques degrés. Le projet doit toujours être protégé contre les menaces aussi bien internes qu'externes : les systèmes non corrigés sont toujours vulnérables à des exploits bien connus. Un administrateur systèmes malveillant peut tout mettre sans dessus dessous, un système d'authentification déficient peut permettre l'entrée de codes malveillants dans la base, et un « rm -rf » accidentel sur le dépôt central peut toujours coûter cher en temps de développement. Mais ces défis peuvent être surmontés, et à l'ère des serveurs privés virtuels (VPS) abordables et des centres d'hébergement de données, les absences des administrateurs systèmes peuvent également être compensées. (Il vaut mieux cependant s'assurer d'avoir un accès redondant au DNS ! Oh et mettez aussi vos sites internet sur un dépôt vérifié et certifié [DVCS] , et faites des branches pour les modifications locales. Vous me remercierez plus tard.)

Les DVCS permettent la redondance du cœur de votre projet pour trois fois rien, ce qui est une bonne façon d'aider les

administrateurs systèmes à dormir la nuit et nous donne l'impression d'être un peu des maîtres du temps. Cela veut aussi dire que si vous n'êtes pas sur un DVCS, arrêtez de lire immédiatement et passez sur l'un d'eux. Ce n'est pas qu'une question d'espace de travail et d'outils. Si vous vous souciez de la sécurité de votre code et de votre projet, vous migrerez.

La redondance du code source est une nécessité, et en général plus vous avez de redondances, plus vos systèmes sont robustes. Il semble aussi évident que vous voulez une redondance de vos administrateurs systèmes ; ce qui vous semblera peut-être moins évident, c'est que l'importance de la redondance ne se joue pas tant en termes de personnes qu'en termes de niveau des compétences. John, l'administrateur systèmes, a travaillé dans des centre de stockage des données et au sein d'entreprises qui avaient des systèmes d'administration redondants mais des niveaux de compétences rigides, définis. Cela fonctionne dans de grandes entreprises qui peuvent payer pour embaucher de nouveaux administrateurs systèmes avec des compétences à la carte. Mais la plupart des projets *open source* n'ont pas ce luxe. Vous devez faire avec ce que vous avez. Cela veut dire bien sûr que, pour que l'administration systèmes soit redondante, une solution – et c'est parfois la seule – consiste à répartir la charge : d'autres membres du projet prenne chacun une ou deux compétences jusqu'à ce qu'il y ait redondance.

Il n'y a guère de différence entre le côté développement et le côté créatif d'un projet ; si la moitié de votre programme est écrite en C++, l'autre moitié en Python, et qu'un seul développeur sait programmer en Python, son départ du projet provoquera de gros problèmes à court terme et pourrait aussi causer de sérieux problèmes à plus long terme. Encourager les développeurs à se diversifier et à se familiariser avec d'autres langages, paradigmes, bibliothèques, etc. entraîne que chacun de vos développeurs gagne en valeur ; cela ne

devrait pas choquer : l'acquisition de nouvelles compétences est le résultat d'un apprentissage qui se poursuit tout au long de la vie, et un personnel mieux formé a aussi plus de valeur. (Cela rend aussi le curriculum vitae de chacun plus attractif, ce qui devrait être une bonne motivation.)

La plupart des développeurs *open source* que je connais considèrent comme un défi et un plaisir de s'aventurer sur de nouveaux territoires : c'est justement ce genre d'état d'esprit qui les a menés à développer de l'*open source* au départ. De même, les administrateurs de systèmes *open source* sont une denrée rare, et ne peuvent se permettre de s'enliser dans une routine. De nouvelles technologies intéressantes pour les administrateurs systèmes apparaissent constamment et il existe souvent de nouvelles façons d'utiliser des technologies actuelles ou anciennes afin de renforcer l'infrastructure ou d'améliorer leur efficacité.

John n'était pas un bon candidat parce qu'il apportait peu de valeur ajoutée ; et il apportait peu de valeur car il n'était jamais allé au-delà des limites du rôle qui lui était attribué. Les administrateurs systèmes *open source* qui tombent dans ce piège ne nuisent pas seulement au projet dans lequel ils sont impliqués sur le moment, ils réduisent leur valeur pour d'autres projets utilisant des technologies d'infrastructure différentes et qui auraient vraiment besoin d'un coup de main ; cela diminue les capacités globales de la communauté *open source*. Pour un administrateur de logiciel libre efficace, il n'existe pas de zone de confort.

1. http://www.kde.org/ ↩
2. http://www.tomahawk-player.org/ ↩
3. http://fr.wikipedia.org/wiki/Toile_de_confiance ↩