Sécurité des mots de passe

Stockage, génération et politiques de mots de passe

Pierre-Antoine HAIDAR-BACHMINSKA

@Hydraze / hydraze@hydraze.org

10 janvier 2017

Télécom SudParis

Lequel de ces mots de passe est le plus *solide* ?

- azerty
- dadada
- xjyt5g4axjyt5g4axjyt5g4axjyt5g4axjyt5g4axjyt5g4a
- Juge un homme par ses questions plutôt que par ses réponses

Réponse : aucun.

- Ils peuvent tous être cassés en quelques minutes
- `azerty` : top #10 des mots de passe les plus utilisés
- `dadada` : ZUCKERBERG (LinkedIn, Twitter, Pinterest, . . .)
- `xjyt5g4axjyt5g4axjyt5g4axjyt5g4axjyt5g4axjyt5g4a` : 6*“xjyt5g4a”
- `Juge un homme par ses questions plutôt que par ses réponses` : VOLTAIRE
- . . . tous dans des dictionnaires existants

Introduction

- Mots de passe : Schéma d'authentification le plus utilisé...
- ...malgré des alternatives plus robustes ?
- De plus en plus de fuites (Dropbox, LinkedIn, MySpace, Tumblr, Yahoo)
- Comment les mots de passe sont-ils attaqués ?
- Comment se défendre ?

Qui suis-je ?

- Casse des mots de passe depuis 2012 (surtout pour le fun)
- Membre de la Team HASHCAT
- Staff sur Zenk-Security

Table des matières

Table des matières

Généralités

Stockage des mots de passe

Stockage des mots de passe

- Problème : et s'il y avait une fuite de donnée ?
- Solution : stocker les mots de passe à l'aide d'une fonction à sens unique
- *Fonction de hachage cryptographique*
- `md5('INT') = 53f93baa3057821107c750323892fa92`
- `md5('Int') = 1686a6c336b71b36d77354cea19a8b52`

Généralités

Attaques en ligne

Attaques en ligne

- Problème : comment accéder à un compte en ligne ?
- Solution #1 : trouver le nom d'utilisateur et tester des mots de passe
- Très lent, inefficace et rapidement repéré
- Solution #2 : trouver une fuite incluant l'utilisateur, casser le mot de passe hors ligne, tester le mot de passe
- Souvent utilisé car les personnes réutilisent les mots de passe
- Logiciel : Hydra

Généralités

Attaques hors ligne

Attaques hors ligne

- Problème : comment retrouver un mot de passe à partir d'une empreinte ?
- Solution : on *hache* un mot de passe candidat puis on compare l'*empreinte* obtenue

Attaques hors ligne – Exemple

- On a cette empreinte : `e10adc3949ba59abbe56e057f20f883e`
- On essaie le mot de passe 1234 : `md5('1234')` → `81dc9bdb52d04dc20036dbd8313ed055`
- On essaie le mot de passe 123456 : `md5('123456')` → `e10adc3949ba59abbe56e057f20f883e`

Attaques hors ligne – Performances

- Logiciels : John The Ripper, Hashcat (yay), Ophcrack (windows, rainbow tables)
- Benchmark hashcat en conditions optimales (1 hash, bruteforce)

Device / Hash	MD5	NTLM	bcrypt(5)
i7 6850k	630 MH/s	1.3 GH/s	5 kH/s
970 GTX	10 GH/s	17.5 GH/s	5.8 kH/s

Table des matières

Méthodologie d’un attaquant

Attaque par dictionnaires

Attaque par dictionnaires

- Candidats #1 : dictionnaires, listes de mots de passe cassés, citations, *scrapping*, ...
- Top #10 personnel : `123456`, `password`, `123456789`, `12345678`, `qwerty`, `111111`, `123123`, `abc123`, `1234567`, `12345`
- Limitation : ne prend pas en compte les petites modifications (leetspeak, dates, zip codes, ...)
- Recommandation #1 : Ne pas réutiliser des mots de passe
- Recommandation #2 : Ne pas utiliser de mots ou citations connues

Méthodologie d’un attaquant

Attaque par règles de mutations

Attaque par règles de mutations

- Candidats #2 : dictionnaires + règles de mutations
- P@$$w0rd !, bONJOURdU92,
  - inversion de casse, mise en minuscules,
  - ajouts à la fin, au début, à une position donnée,
  - remplacement de caractères (leet speak)
  - 40+ règles différentes, des possibilités infinies. . .
- Très efficace (20-70% de mots de passe récupérés en quelques minutes)
- Recommandation #3 : Ne pas utiliser un mot modifié

Méthodologie d’un attaquant

Attaque par force brute

Attaque par force brute

- Candidats : aaaaa, aaaab, aaaac, aaaad, ...
- Avantages : couverture exhaustive d'un ensemble de candidats
- Inconvénients : extrêmement lent sur des mots de passe longs...
- Nombre de candidats : $(\text{taille charset})^{nombredecaractres}$

Attaque par force brute – Durée maximale de cassage

Conditions : hash seul, MD5, GPU milieu de gamme, 10 GH/s.

La durée concerne la recherche exhaustive. Si motif ou logique, beaucoup plus rapide

Charset / len	7	8	9	10	11	12	13	14
min	< 1s	21s	9m	4h	4d	4M	8y	>100y
min+dig	8s	5m	3h	4d	5M	15y	>500y	>10ky
min+maj+dig	6m	6h	16d	3y	>100y	:o	o,o	T_T

Méthodologie d’un attaquant

Attaque par masque

Attaque par masque et chaînes de Markov

- Problème : comment optimiser une attaque par force brute ?
- #1 : En utilisant des masques
- Motifs prédictibles : MAJ - mins - (chiffres / speciaux)
- Attention : politique de mots de passe restrictive = motifs
- #2 : En utilisant des probabilités !
- Modèle de chaînes de Markov par position
- Recommandations #4 : Ne pas utiliser des mots de passe courts ou comportant des motifs
- Recommandations #5 : Les mots de passe doivent faire au moins 10 caractères

Méthodologie d’un attaquant

Analyse et statistiques

Analyse et statistiques

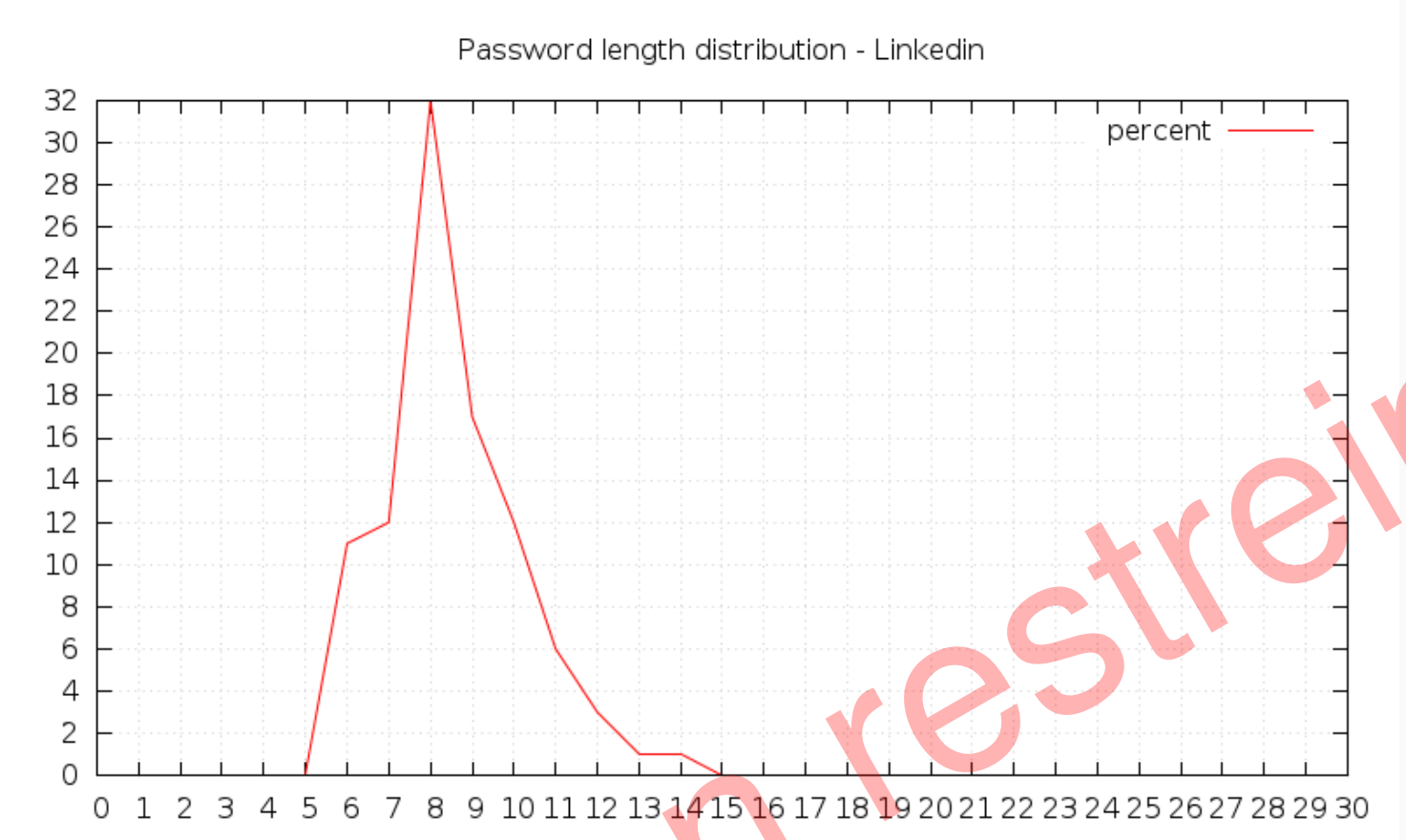
- Statistiques diverses :
  - Longueur
  - Jeux de caractères
  - Motifs
  - Mots de base
  - Règles de mutations. . .
- Utile pour créer de nouveaux masques ou de nouvelles règles
- Logiciels : Passpal, pipal, P.A.C.K.
- Recommandation #6 : *Fuck logic*.

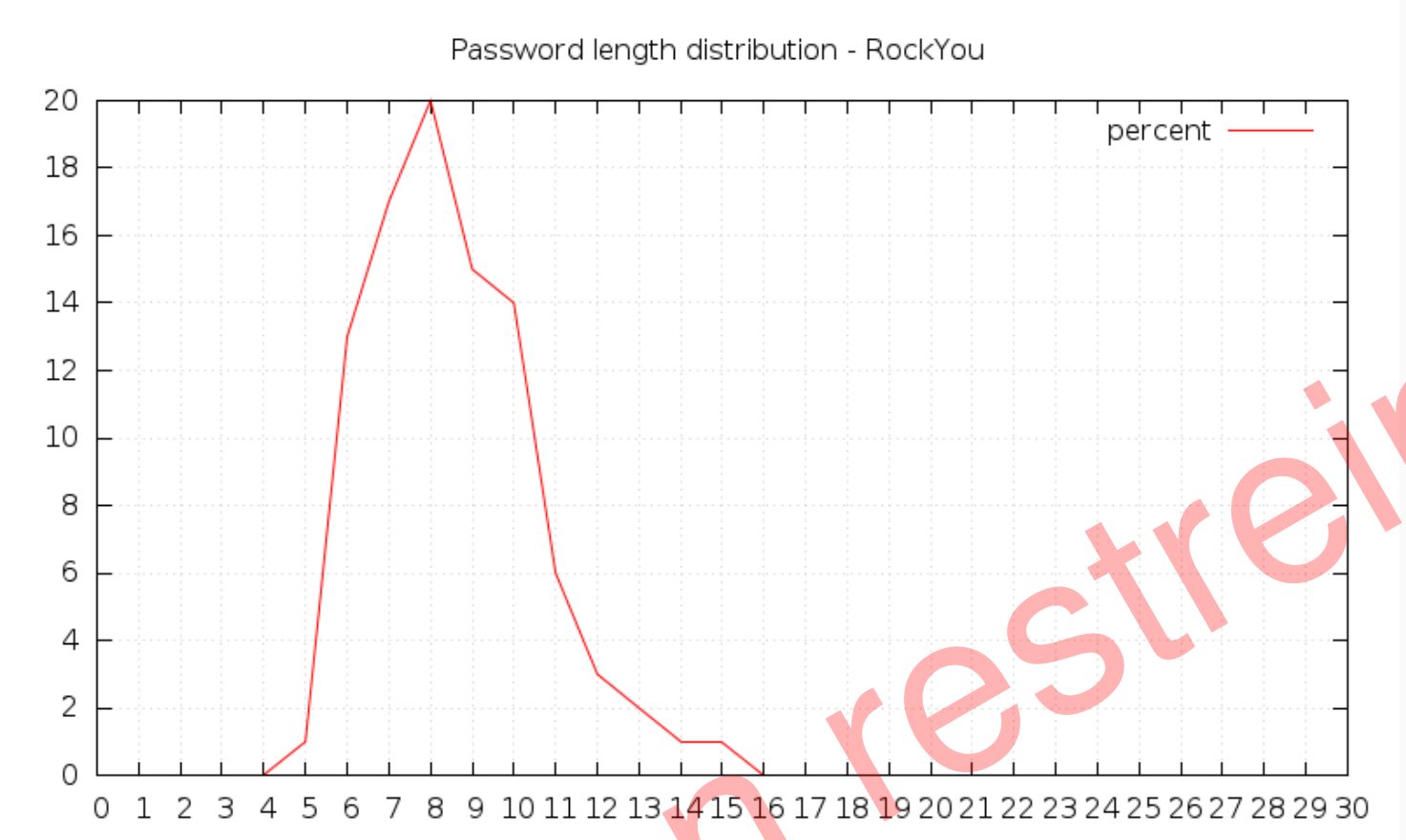
Analyse et statistiques – Fuite LinkedIn

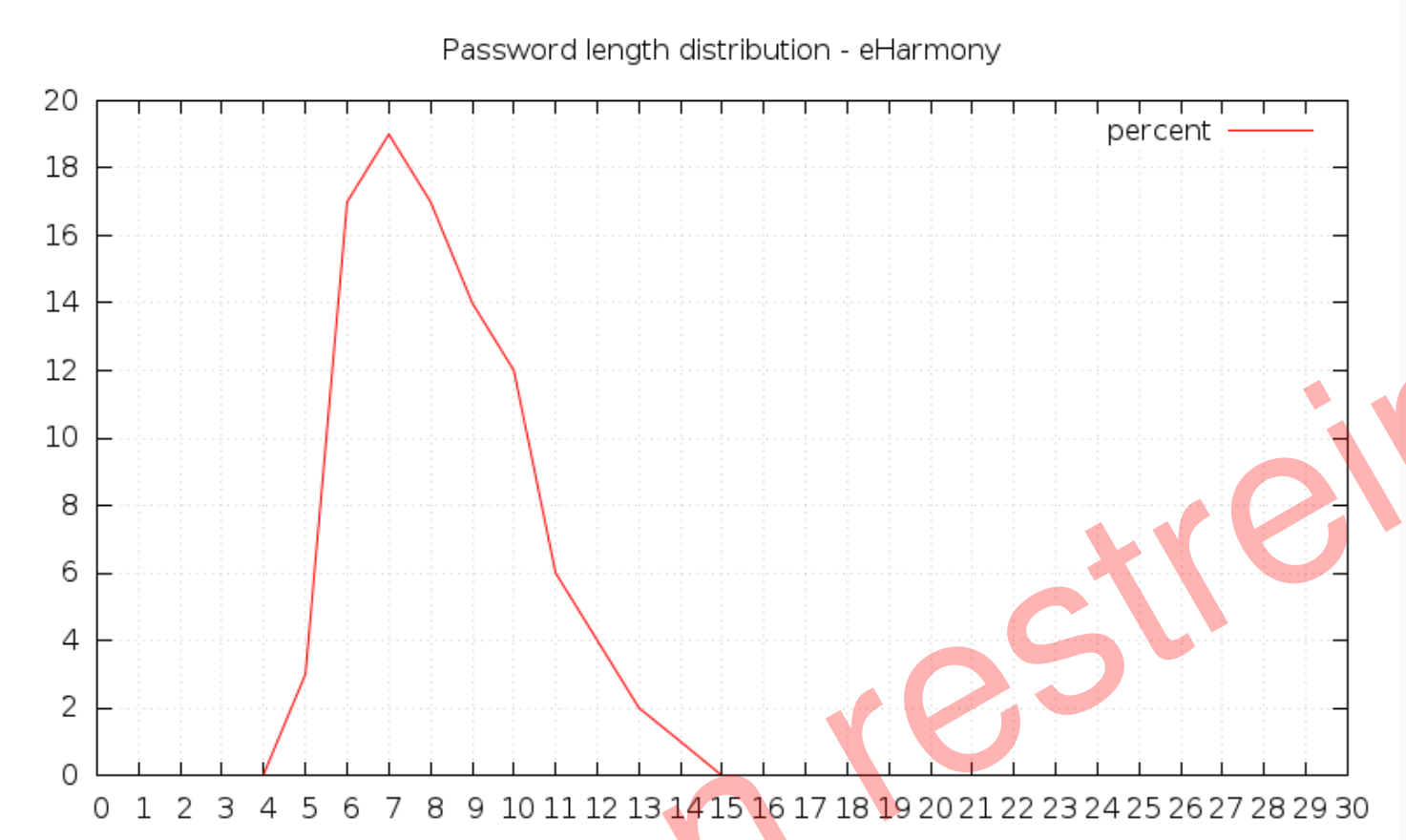
- Fuite de 2012
- 177M de mots de passe
- 62M de mots de passe uniques
- 95+% de récupération

Analyse et statistiques - Fuite LinkedIn

Count	Password
1135936	123456
207488	linkedin
188380	password
149916	123456789
95854	12345678
85515	111111
75780	1234567
51969	654321
51870	qwerty
51535	sunshine







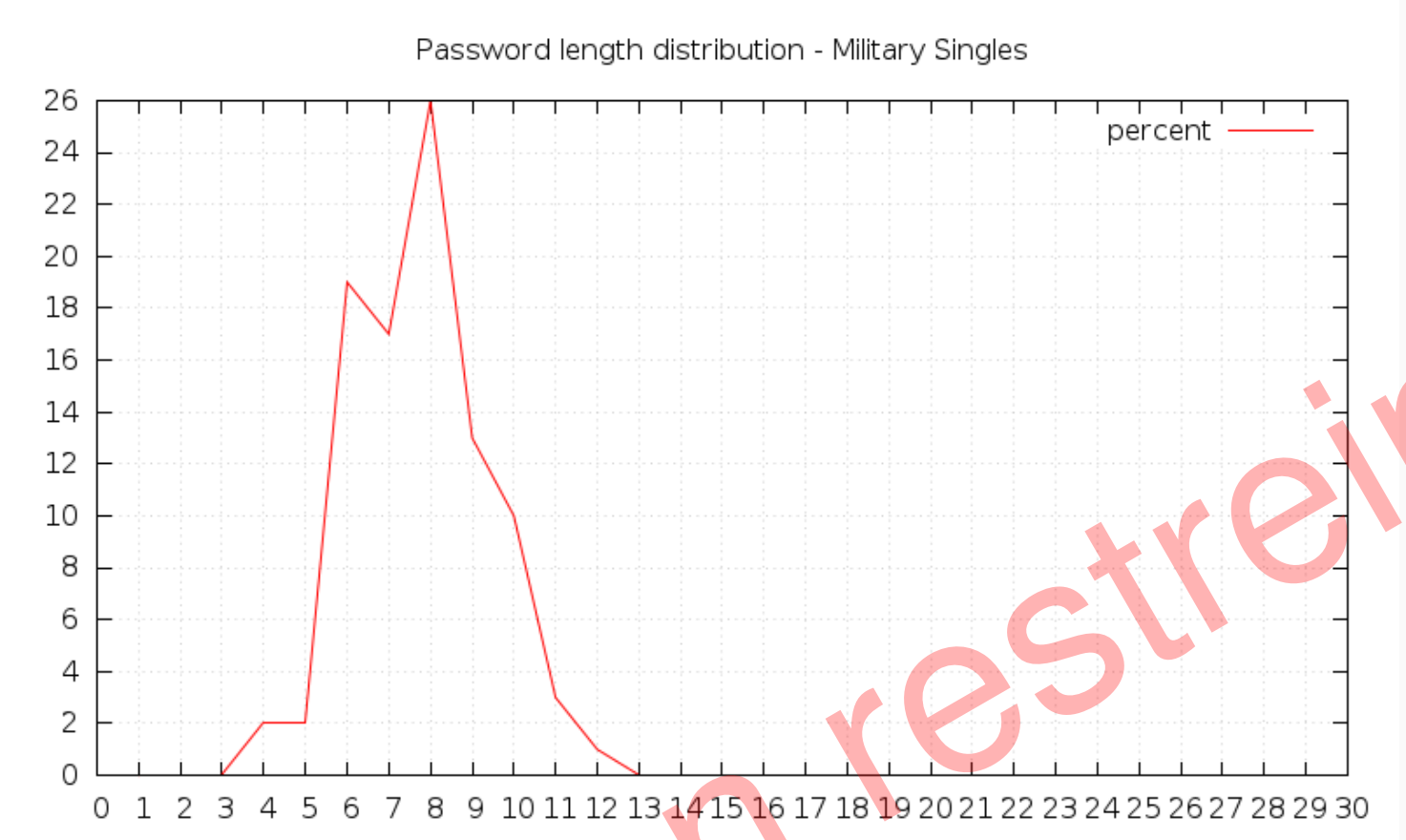


Table des matières

Quelles défenses ?

Côté utilisateur

Authentification multi-facteur

- Ce que je sais (mot de passe)
- Ce que je suis (biométrie)
- Ce que j'ai (Token, téléphone)

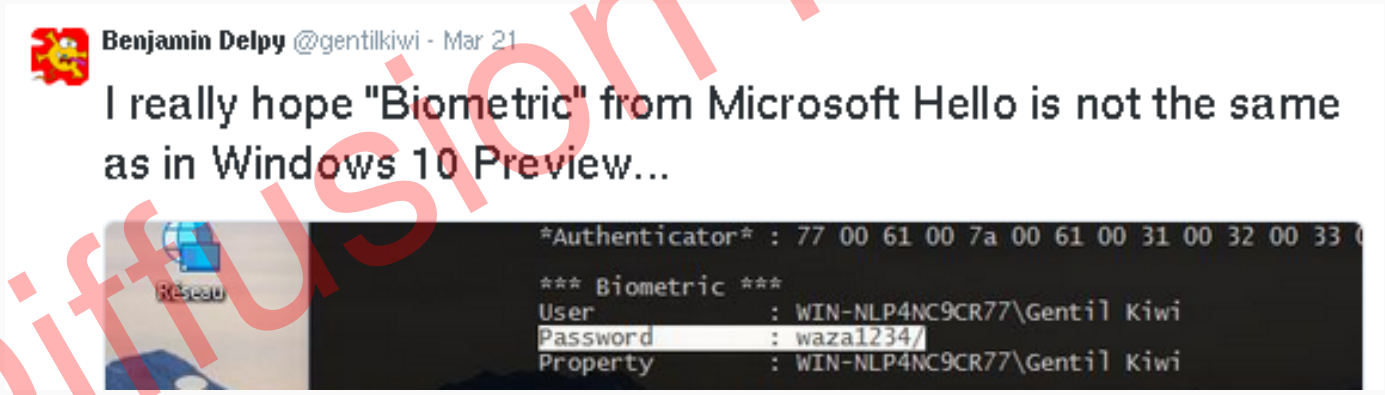
Ce que je sais

- Problème : on a besoin de mots de passe complexes et longs
- Solution : les stocker de manière sécurisé dans un logiciel
- (Non, votre navigateur ne les stocke pas de manière sécurisée. . .)

- Exemple : keepass, dashlane, lastpass, 1password, . . .
- Problème : Gestionnaire dans le *cloud* ou en local ?
- Problème : Besoin d'un mot de passe maître sécurisé ou. . . d'une phrase de passe de 4-5 mots ?

Ce que je suis

- Empreintes digitales, scan de l'iris, . . .
- Spoiler : la plupart des solutions grands publiques ne sont pas sécurisées
- Microsoft a rajouté de l'authentification biométrique dans Windows 10 ("Hello")
- Cassé avant la publication de la version finale :



[1]

1.https ://twitter.com/gentilkiwi/status/579401086479499264

Ce que je suis – Suite

- iPhone (5S- ?) authentification TouchID (empreintes) : cassé (CCC) [2]
- Problème : J'ai seulement 10 doigts, deux yeux et un visage...
- Problème : Loi très restrictive (CNIL)

2.https ://blog.lookout.com/blog/2014/09/23/iphone-6-touchid-hack/

Ce que j'ai

- *Token* matériel (Yubikey, RSA-OTP, . . .)
- Peut se souvenir d'un secret, donner un OTP, faire de la crypto, . . .
- Les SMS ne sont pas considérés comme sécurisés. . .
- . . .mais Google Authenticator et les autres logiciels générant des OTP le sont.

Quelles défenses ?

Côté Développeur

Choisir la bonne fonction de hachage

- Ne surtout pas utiliser MD5, SHA-* ou d'autres fonctions *brutes*
- Des fonctions ont été créées pour stocker des mots de passe
- **bcrypt**, scrypt, **argon2**
- Anti-GPU
- Évolutives (paramètre de coût)
- Facile à utiliser
- Ajouter un *pepper*, le *sel est obligatoire*

Évaluer la sécurité des mots de passe utilisateurs

- Interdire les mots de passe communs (top10k.txt, cf Twitter)
- Utiliser un indicateur visuel pour la sécurité des mots de passe
- Un bon indicateur.
- Utiliser zxcvbn (dropbox)
- Gère les mots de passe simples, le *keywalking*, ...
- Facile à intégrer

Quelles défenses ?

Côté RSSI

Politiques de mots de passe

- Avoir une politique à respecter c'est *chiant*
- Certaines politiques sont contreproductives
- Les utilisateurs les contourne avec des motifs...
- MAJ - min - (chiffres / spéciaux)
- Même problème avec les renouvellements périodiques forcés
- Il faut sensibiliser les utilisateurs avant tout

Conclusion

- Utiliser des mots de passe longs et aléatoires (et un gestionnaire de mots de passe)…
- …ou posez-vous les bonnes questions
- Activez l'authentification multi-facteurs dès que vous le pouvez
- Développeurs : Utilisez de meilleures fonctions de hachages !
- RSSI : Mettre en place une politique n'est pas suffisant !
- Problème de terminologie : *mot de passe* ou *phrase de passe* ou … ?