Est-ce la même chose d'accéder à une donnée individuelle, et d'avoir un accès en masse ?

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 14 décembre 2011

https://www.bortzmeyer.org/bulk-or-not.html

Dans les discussions sur la protection de la vie privée, une confusion est souvent faite entre « donnée accessible publiquement » et « la totalité des données est récupérable » (ce qu'on nomme en anglais le "*bulk access*").

Par exemple, cette confusion est souvent faite dans le cas de l'accès aux données stockées dans le DNS. N'importe qui peut interroger les serveurs DNS de `.fr` pour savoir si le nom `anemelectroreculpedalicoupever` existe ou pas (on peut aussi le faire via le protocole whois). En revanche, le fichier comportant tous les noms existants dans `.fr` n'est pas disponible. (Certaines zones permettent cet accès <https://www.bortzmeyer.org/recuperer-zone-dns.html>.) N'y a-t-il pas une incohérence ? Si les données sont publiques, quel mal y aurait-il à donner un accès à l'ensemble de ces données, un « "*bulk access*" » (accès en masse) ?

Techniquement, la différence peut en effet sembler mince : si on peut faire une requête DNS (ou whois) pour un nom, il est trivial de faire une boucle pour essayer plein de noms. Cela se nomme une attaque par dictionnaire et les serveurs de `.fr` en voient régulièrement. Mais ce n'est pas très discret.

Et surtout, penser que l'accès individuel (éventuellement répété dans une boucle) équivaut à l'accès en masse, c'est oublier l'explosion combinatoire, qui limite sérieusement les possibilités d'une attaque par dictionnaire. Imaginons qu'on soit intéressé par les variantes de `mabanqueserieuse.example`. Imaginons également qu'on se limite aux variations d'un seul caractère. `mabanqueserieuse` a 17 caractères. En exploration systématique par des requêtes DNS, il faudrait 36 essais (les lettres d'ASCII, plus les chiffres et le tiret, moins le caractère existant) par caractère soit 612 essais. Et cela ne teste que les substitutions, pas les ajouts ou suppressions (dont on verra plus loin qu'ils existent). Bref, de tels tests seraient assez bavards et feraient râler l'administrateur des serveurs DNS (et c'est encore plus net avec whois). Dans la plupart des cas, énumérer toutes les variantes « intéressantes » (pour faire ensuite des requêtes DNS) n'est pas faisable.

Si on a l'accès en masse, tout est plus simple, car de superbes algorithmes existent pour rechercher de manière plus efficace. Voyons un exemple avec le programme agrep (tre-agrep en fait), `-E 1` signifiant qu'on cherche les noms qui ne diffèrent que d'un seul caractère :

```
% grep mabanqueserieuse example.txt
mabanqueserieuse.example

% tre-agrep -E 1 mabanqueserieuse example.txt
mabanqueserieuse.example
mabanqueserieusr.example
mabanqueserieusse.example
mbanqueserieuse.example
manbanqueserieuse.example
...
```

et on trouve ainsi de nombreuses autres variantes (testé avec une banque réelle, où presque toutes les variantes avaient été enregistrées par un bureau d'enregistrement situé aux Bahamas). Le fait d'avoir accès à la totalité de la base permet également des recherches sur une partie du nom et de trouver ainsi les "*doppelgangers*" comme `wwwmabanqueserieuse.fr`.

Dans ce cas précis, vous me direz peut-être que détecter les cybersquatteurs opérant depuis un paradis fiscal ne serait pas une mauvaise chose. Mais mon but était de montrer que l'accès aux données en masse permettait des recherches bien plus poussées, et que cela peut se faire au détriment d'innocents (par exemple des particuliers harcelés par les détenteurs de titre de propriété intellectuelle, comme dans l'affaire Milka).

C'est pour cela que le mécanisme NSEC3 du RFC 5155[1] était important. Sans lui, il était possible d'énumérer tous les noms d'une zone DNS signée avec DNSSEC. Certaines personnes avaient relativisé ce risque en disant « les données DNS sont publiques, de toute façon », ce qui est une sérieuse erreur, comme indiqué plus haut.

Si vous préférez aborder le problème sous l'angle juridique, il faut lire les articles L. 342-1 à 3 du Code de la Propriété Intellectuelle <`http://www.legifrance.gouv.fr/affichCode.do?idSectionTA=LEGISCTA000006161661&cidTexte=LEGITEXT000006069414&dateTexte=20111214`> (merci à Thomas Duboucher pour les indications).

Les curieux noteront que les algorithmes de recherche approximative de texte sont proches de ceux utilisés en génomique comme Smith et Waterman ou Needleman et Wunsch. En effet, la recherche d'une séquence de bases dans un génome ne peut pas se faire littéralement, comme avec grep. En raison des mutations et des erreurs dans le séquençage, la correspondance n'est jamais parfaite, et il faut donc accepter, comme dans l'exemple avec tre-agrep, un certain nombre de différences.

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc5155.txt`