Changement d'adresses IP dans le DNS et sécurité du Web

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 8 août 2007

https://www.bortzmeyer.org/dns-rebinding-pinning.html

Un récent intérêt pour un problème de sécurité des navigateurs Web a porté sur leur relation au DNS. Qu'est-ce que le **changement** ("*rebinding*") et le DNS est-il responsable de cette faille ?

L'article "*Protecting Browsers from DNS Rebinding Attacks*" <http://crypto.stanford.edu/dns/dns-rebinding.pdf>, ainsi qu'une démonstration à grand spectacle à la conférence Black Hat ont suscité un intérêt médiatique marqué <http://www.securityfocus.com/news/11481>. Qu'y a t-il dans cet article ? Pour le comprendre, je me permets un petit détour sur le modèle de sécurité d'un navigateur Web typique.

Un navigateur comme Firefox ou Internet Explorer peut charger des ressources contenant du code à exécuter, par exemple des scripts Javascript ou bien des animations tape-à-l'œil en Flash. Ces codes « étrangers » n'ont normalement que des privilèges limités. Par exemple, un script Javascript peut faire des connexions réseau mais uniquement vers le serveur d'où il vient. Et c'est là le nœud du problème : comment savoir si la machine à laquelle parle le script est la même que celle d'où vient le script ? Que veut dire « la même machine » ? Le même nom de domaine ? La même adresse IP ? La même « identité », vérifiée par exemple avec une clé cryptographique comme le font SSH ou bien HIP ?

Outre des failles dans le modèle de sécurité lui-même (par exemple, Flash peut ouvrir des connexions réseau vers d'autres machines, s'il est autorisé par la machine d'où il vient, machine qui peut être contrôlée par l'attaquant), cette question de l'identité est au cœur du problème. La plupart des articles sur la question ont été écrits par des gens qui, n'ayant pas compris le DNS, pensait qu'il fournissait une identité, alors qu'il ne fournit qu'une **correspondance** entre un nom de domaine et une valeur (par exemple une adresse IP).

Comment se passe donc l'attaque ? L'attaquant configure un serveur, mettons `www.mechant.example.com` et publie un URL, par exemple dans les commentaires d'un blog : « vené voir plain photos top tro bien en http ://www.mechant.example.com/ ». Les utilisateurs qui cliquent sur ce lien reçoivent le code malveillant (Javascript, Java, Flash, etc) et ce code va ensuite faire des requêtes vers `www.mechant.example.com`, comme autorisé. La plupart de ces techniques d'exécution de code étranger vérifient l'identité de la machine uniquement avec le nom ! Si le TTL (la durée de vie dans le DNS) était très court, la seconde

requête, celle faite par le code malveillant, retournera aux serveurs de noms de `mechant.example.com` qui pourront alors renvoyer une autre adresse IP, par exemple située sur le réseau interne de la victime. C'est ce qu'on nomme le **changement** ("*rebinding*").

L'article contient de nombreux détails. La vulnérabilité de principe est celle indiquée ci-dessus mais elle est aggravée par d'autres problèmes comme le fait que les caches des différentes parties du navigateur ne sont pas forcément partagés et que Javascript n'utilisera peut-être donc pas la même adresse IP que le moteur principal de Firefox.

Cette attaque est normalement empêchée par un comportement fréquent dans certaines applications, l'**épinglage** ("*pinning*"). Ce nom désigne le fait que beaucoup d'applications n'appellent le résolveur (via `getaddrinfo` ou une fonction similaire) qu'une seule fois et gardent ensuite éternellement l'adresse IP résultante, quel que soit le TTL dans le DNS. L'épinglage, qui viole le protocole DNS, pose ses propres problèmes et l'article montre qu'il n'est pas sans failles, il existe en effet plusieurs trucs pour désépingler un nom.

Quelques trucs ont été proposés pour limiter les risques, comme de demander au serveur de noms récursif de refuser les réponses venues de l'extérieur si elles contiennent une adresse IP intérieure. Avec un logiciel comme BIND, cela nécessiterait une modification spécifique. Il y a aussi des logiciels comme le DNS wall <`http://code.google.com/p/google-dnswall/`> de Google qui se placent avant le serveur de noms récursif et mettent en œuvre cette fonction. Il est important de noter que DNS Wall ne permet **pas** de configurer les adresses IP considérées comme internes (il ne connait que le RFC 1918 [1]). Et qu'il comporte une faille de sécurité grave (cf. <`http://code.google.com/p/google-dnswall/issues/detail?id=1`>). Un code amélioré, basé sur DNS Wall, a finalement été intégré dans BIND (à partir de la version 9.7).

Alors, faut-il changer le DNS ? Les navigateurs ? Les deux ? D'abord, il faut bien voir que les accusations contre le DNS ou les demandes qu'on le change immédiatement sont le résultat d'une incompréhension de ce qu'est le DNS. Le DNS n'a jamais été un service d'identité, encore moins d'authentification. Le DNS est uniquement une **table** qui met en correspondance des noms de domaine et des valeurs, comme les adresses IP. Rien n'interdit au gérant de `www.mechant.example.com` de faire pointer ce nom vers une adresse IP qui ne leur « appartient » pas et c'est une fonction fondamentale du DNS (la supprimer casserait beaucoup d'applications, notamment si on n'est pas son propre hébergeur).

Mais les navigateurs n'ont pas forcément toujours tort non plus. À l'heure actuelle, il n'existe pas de mécanismes simples sur Internet pour s'assurer de l'identité d'une machine. Des applications comme SSH ont créé leur propre mécanisme. Un effort pour avoir un tel mécanisme au niveau IP est en cours, sous le nom de HIP (cf. RFC 4423) mais est encore très expérimental.

La protection de HTTP contre le changement a fait l'objet d'une discussion <`http://lists.w3.org/Archives/Public/ietf-http-wg/2010JanMar/0155.html`> lors de la mise à jour de la norme HTTP en 2010.

Un exemple d'une faille de sécurité liée au changement dans le DNS est la #1471 du Projet Zéro, concernant les jeux Blizzard <`https://bugs.chromium.org/p/project-zero/issues/detail?id=1471&desc=3`>.

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc1918.txt`