# RFC 6151 : Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 7 mars 2011

Date de publication du RFC : Mars 2011

http://www.bortzmeyer.org/6151.html

---

Ce RFC marque un peu l'enterrement officiel de MD5 dans les protocoles IETF. Cet algorithme de cryptographie, très répandu et très utilisé, décrit dans le RFC 1321 [1], a connu de nombreuses attaques cryptanalytiques réussies et, pour plusieurs de ces usages, ne doit plus être considéré comme offrant une protection raisonnable. Une bonne partie de la sécurité de l'Internet reposait historiquement sur des protocoles utilisant MD5 et c'est donc un chapitre important qui se clot.

Peut-on dire aujourd'hui que MD5 est « cassé » et n'offre plus aucune protection ? La réponse doit en fait être plus nuancée et cela explique que ce très court RFC ait fait l'objet d'une aussi longue discussion à l'IETF, pour ne sortir que maintenant. En effet, si on voit souvent des affirmations hâtives de la part des ignorants, prétendant sans nuance que MD5 a été cassé, la réalité est plus complexe.

La section 1 rappelle en effet l'état de l'art. MD5 est utilisé pour deux choses :

— Pour calculer un condensat cryptographique d'un message. C'est ce que fait la commande Unix `md5sum` et cet usage est à la base de plusieurs solutions de signature.
— Pour authentifier un message en condensant le message avec une clé secrète (technique HMAC, RFC 2104).

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc1321.txt`

Dans le premier cas, MD5 n'offre en effet plus une sécurité acceptable face à un attaquant (il reste utilisable lorsque le seul but est de détecter des erreurs produites par un processus aveugle, par exemple des parasites sur la ligne). Pour le second cas, c'est moins évident.

C'est qu'il existe plusieurs types d'attaques contre les fonctions de hachage cryptographiques comme MD5 (voir le RFC 4270). Par exemple, une attaque par collision consiste en la production de deux messages ayant le même condensat. Ce sont ces attaques auxquelles MD5 est le plus vulnérable. Si un document est signé via un condensat MD5, il est aujourd'hui simple de créer deux documents ayant le même condensat MD5, ce qui annule une bonne partie de l'intérêt des signatures.

Mais MD5 résiste bien mieux à d'autres attaques comme celles par pré-image, où un attaquant cherche à fabriquer un message ayant le même condensat qu'un message donné. C'est une tâche nettement plus complexe que la collision, où l'attaquant contrôle les **deux** messages. Donc, chaque protocole ou application qui utilise MD5 doit analyser si son usage de cette fonction est acceptable ou pas, compte-tenu de l'état de l'art en cryptographie.

Pour voir la différence entre ces deux problèmes, reprenons la commande `md5sum`. Soit un fichier `toto.txt`. md5sum calcule le condensat :

```
% md5sum /tmp/toto.txt
0365023af92c568e90ea49398ffec434  /tmp/toto.txt
```

Une attaque par pré-image consisterait en la création d'un nouveau fichier ayant le **même** condensat `0365023af92c568e90ea49398ffec434` que `toto.txt`. Cela reste une tâche non-triviale. Une attaque par collision permet au contraire à l'attaquant de choisir le condensat à produire. Avec l'aide du paradoxe de l'anniversaire, c'est bien plus facile. Mais cela ne marche que si l'attaquant a le choix des données, ce qui est partiellement vrai pour les signatures mais pas pour HMAC. (Voir une bonne explication des différentes possibilités d'attaque, assez trapue mais originale, en « "*Meaningful Collisions (for SHA-1)*" <http://www.iaik.tugraz.at/content/research/krypto/sha1/MeaningfulCollisions.php> ».)

La section 2 liste les attaques **publiées** contre MD5. Attention, en cryptanalyse, toutes les attaques réussies ne sont pas publiées, certains préférant les garder pour eux. Il faut donc toujours prévoir une marge en se disant que, si une attaque publiée permet de casser un algorithme de cryptographie en un mois, les attaques non publiées peuvent sans doute le faire en quelques jours, voire quelques heures.

Donc, les attaques par collision réussies contre MD5 ont été publiées pour la première fois en 1993. Les progrès ont été importants en 2004 avec « "*Cryptanalysis of HMAC/NMAC-MD5 and MD5-MAC*" <http://eprint.iacr.org/2004/199.pdf> » de Wang, Feng, Lai et Yu. En 2006, une attaque publiée descend à une seule minute de temps d'exécution sur un portable normal. En 2007, ce genre d'attaques a pu casser des certificats X.509. Il est donc clair que MD5 ne doit plus être utilisé pour la signature numérique. (En cherchant, vous trouverez probablement un certain nombre de sites Web qui ont toujours un certificat utilisant MD5.)

Au contraire, comme le note la section 2.2, les meilleures « attaques » pré-image publiées n'ont pas encore abouti. Quelles sont les conséquences pour HMAC (section 2.3) ? S'il y a eu plusieurs tentatives de casser HMAC-MD5 (tel qu'il est utilisé, par exemple, dans TSIG, RFC 2845), aucune de celles publiées n'a réussi. Il n'y a donc pas d'urgence à supprimer MD5 lorsqu'il est utilisé ainsi, néanmoins des remplaçants sont prêts (cf. RFC 4231) et peuvent être intégrés dans les nouvelles versions des protocoles (RFC 6039). Notez que les faiblesses de MD5 ne sont évidemment pas spécifiques aux protocoles Internet, voir par exemple la documentation de GnuPG <http://www.gnupg.org/faq/weak-digest-algos.html>.