Thunderbird contre Logjam

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 22 juillet 2015

https://www.bortzmeyer.org/logjam-thunderbird.html

La sécurité informatique, c'est compliqué. Voilà qu'un utilisateur d'un serveur de messagerie que je gère me signale qu'il ne peut plus lire son courrier avec Thunderbird. Et je découvre que le problème était en fait dû au zèle que mettait Thunderbird à protéger ses utilisateurs de la faille Logjam.

Avant de revenir sur cette faille et sur la solution, voici les symptômes : Thunderbird était configuré pour récupérer le courrier avec POP. Récemment mis à jour automatiquement, il ne récupère plus de courrier. Apparemment pas de messages d'erreurs côté client, mais pas de courrier non plus. Côté serveur (un Courier), on trouve juste dans le journal :

```
pop3d-ssl: couriertls: accept: error:14094417:SSL routines:SSL3_READ_BYTES:sslv3 alert illegal parameter
```

Message sybillin, non, d'autant plus que ce serveur refuse évidemment SSLv3, comme le dit le RFC 7568 [1]. C'est ça, les messages d'erreur d'OpenSSL...

Bon, je n'ai pas eu trop à chercher, des tas de gens avaient le même problème (par exemple chez CentOS <https://bugs.centos.org/view.php?id=9050>). Deux bogues enregistrées chez Mozilla (la fondation qui développe Thunderbird) donnaient les détails : #1183650 <https://bugzilla.mozilla.org/show_bug.cgi?id=1183650> et #1184488 <https://bugzilla.mozilla.org/show_bug.cgi?id=1184488>. L'explication est simple : la mise à jour de Thunderbird (ou plus exactement celle de la bibliothèque TLS NSS dont il dépend), introduit un refus des paramètres Diffie-Hellman trop faibles.

De quoi s'agit-il encore ? La faille de sécurité Logjam, contre le protocole TLS, concerne les échanges de clés Diffie-Hellman. TLS peut fonctionner sans Diffie-Hellman (et c'est pour cela que certaines utilisateurs n'ont pas eu de problèmes) mais, si on l'utilise, l'échange de clé peut être deviné par un attaquant car les groupes Diffie-Hellman ont pu être pré-calculés, surtout s'ils sont trop petits (768 bits par exemple, taille souvent utilisée par défaut, cf. la bogue Debian #787579 <https://bugs.debian.org/787579>). Outre le site officiel de Logjam <https://weakdh.org/>, on peut consulter sur cette faille une excellente explication chez Cloudflare <https://blog.cloudflare.com/logjam-the-latest-tls-vulnerability-explained/>. Donc, Thunderbird (ou plutôt NSS) refuse désormais les paramètres Diffie-Hellman trop faibles. Testons notre serveur (le problème initial concernait aussi bien POP que IMAP, ici, je teste le serveur IMAP) :

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc7568.txt

```
% openssl s_client -connect server.bortzmeyer.org:993
CONNECTED(00000003)
...
No client certificate CA names sent
Peer signing digest: SHA512
Server Temp Key: DH, 768 bits
...
* OK [CAPABILITY IMAP4rev1 UIDPLUS CHILDREN NAMESPACE THREAD=ORDEREDSUBJECT THREAD=REFERENCES SORT QUOTA IDI
```

En effet, avec ses 768 bits (DH = Diffie-Hellman), le serveur ne suit pas les bonnes pratiques et c'est pour cela que Thunderbird coupe la communication.

Quelle est la solution ? Le site officiel de Logjam a une bonne page d'explications pour les administrateurs système <https://weakdh.org/sysadmin.html> mais, ici, j'ai simplement utilisé le commentaire 13 dans la bogue Mozilla #1184488 <https://bugzilla.mozilla.org/show_bug.cgi?id=1184488#c13>. Il faut regénérer des paramètres Diffie-Hellman. Le serveur est une machine Debian, les paramètres sont en `/etc/courier/dhparams.pem`, et le script à utiliser, `mkdhparams`, est fourni avec Courier. À noter que sa documentation est inexacte sur la méthode à utiliser pour augmenter le nombre de bits (bogue Debian #793184 <https://bugs.debian.org/793184>). J'ai choisi une taille de 3 072 bits (plutôt exagérée aujourd'hui mais c'est ce qui est recommandé par le RGS <https://www.ssi.gouv.fr/entreprise/reglementation/confiance-numerique/le-referentiel-general-d>) et j'ai donc fait :

```
server# export DH_BITS=3072

server# mkdhparams
512 semi-random bytes loaded
Generating DH parameters, 3072 bit long safe prime, generator 2
This is going to take a long time
.............................................+.......
...
server# /etc/init.d/courier-imap-ssl restart
[ ok ] Restarting courier-imap-ssl (via systemctl): courier-imap-ssl.service.
server#
```

Testons le résultat :

```
% openssl s_client -connect server.bortzmeyer.org:993
CONNECTED(00000003)
...
No client certificate CA names sent
Peer signing digest: SHA512
Server Temp Key: DH, 3072 bits
...
* OK [CAPABILITY IMAP4rev1 UIDPLUS CHILDREN NAMESPACE THREAD=ORDEREDSUBJECT THREAD=REFERENCES SORT QUOTA IDI
closed
```

Parfait, cette fois, ça marche, et Thunderbird est content, il peut récupérer le courrier de manière sécurisée.

Amusante coïncidence, le lendemain de cet article, à la réunion IETF de Prague <http://www.ietf.org/meeting/93/index.html>, Aaron Zauner faisait un excellent exposé <https://www.ietf.org/proceedings/93/slides/slides-93-saag-2.pdf> sur des tests de la sécurité des serveurs de courrier (SMTP, POP et IMAP) qu'il avait réalisés dans l'Internet. Beaucoup d'entre eux sont encore vunérables à Logjam.

---

https://www.bortzmeyer.org/logjam-thunderbird.html