RFC 8877 : Guidelines for Defining Packet Timestamps

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 24 septembre 2020

Date de publication du RFC : Septembre 2020

https://www.bortzmeyer.org/8877.html

De nombreux protocoles Internet contiennent un champ où se trouve une **estampille temporelle** ("*timestamp*" en anglais), une indication de date et d'heure. Mais chaque protocole a défini le format de cette estampille de son côté, il n'y a pas de format standard. Ce nouveau RFC essaie de mettre un peu d'ordre là-dedans en proposant aux concepteurs de futurs protocoles :

— Trois formats standards, parmi lesquels choisir, pour avoir un format tout fait,
— Des conseils pour les cas où un protocole définirait (ce qui est sans doute une mauvaise idée) un nouveau format.

Dans quels cas des protocoles contiennent dans leurs données une estampille temporelle ? Cela peut être bien sûr parce que le protocole est voué à distribuer de l'information temporelle, comme NTP (RFC 5905[1]). Mais cela peut être aussi parce que le protocole a besoin d'indiquer le moment d'un événement, par exemple pour mesurer une latence <https://www.bortzmeyer.org/latence.html> (OWAMP, RFC 4656 ou les extensions à TCP du RFC 7323). La section 6 du RFC présente une liste non exhaustive de protocoles qui utilisent des estampilles temporelles, ce qui fournit d'autres exemples comme MPLS (RFC 6374), IPFIX (RFC 7011), TRILL (RFC 7456), etc. Il serait souhaitable que les formats de ces estampilles soient normalisés. Ainsi, utiliser le format de NTP dans un autre protocole permettrait facilement d'utiliser le temps NTP, sans perte de précision. Et la standardisation des formats permettait de les mettre en œuvre dans du matériel, comme les horloges. Bref, si vous concevez des protocoles réseau et qu'ils intègrent une information temporelle, lisez ce RFC.

Pour suivre la suite du RFC, il faut un peu de vocabulaire, rappelé en section 2. Notamment :

— Erreur sur l'estampille : la différence entre une estampille et une horloge de référence. Elle est de zéro quand tout est parfaitement à l'heure, ce qui n'arrive évidemment jamais dans le monde réel.
— Exactitude de l'estampille : la moyenne des erreurs.
— Précision de l'estampille : l'écart-type des erreurs.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc5905.txt

— Résolution temporelle de l'estampille : le temps minimal qu'un format permet de représenter. Si on a un format texte HH :MM :SS (heures :minutes :secondes), la résolution sera de une seconde.
— Période de bouclage : le temps au bout duquel on reviendra à la valeur de départ. C'est la fameuse bogue de l'an 2000 : si on stocke les années sur deux chiffres, la période de bouclage vaut cent ans. Si on stocke les estampilles sous forme d'un entier de 32 bits signé, et que la résolution est d'une seconde, cette période de bouclage est de 68 ans.

Les formats recommandés par le RFC sont décrits en section 4. Pourquoi trois formats plutôt qu'un seul ? Parce que tous les protocoles réseau n'ont pas forcément les mêmes contraintes en terme de taille du champ, de résolution temporelle, ou d'intégration avec d'autres protocoles (sur un système d'exploitation conçu pour des PC connectés à l'Internet, utiliser le format de NTP est sans doute le plus raisonnable, mais pour du matériel de métrologie complexe, celui de PTP peut être une meilleure idée).

D'abord, le grand classique, le format de NTP (RFC 5905, section 6). Il est très répandu sur l'Internet, et utilisé dans de nombreux protocoles (RFC 6374, RFC 4656, RFC 5357, et bien d'autres). Il stocke l'estampille sous forme d'un entier de 32 bits pour les secondes depuis l'"*epoch*" 1900 <https://www.bortzmeyer.org/epoch-50.html>, et d'un autre entier de 32 bits pour les fractions de seconde. Sa taille totale est de 64 bits, sa résolution de 2 puissance -32 seconde, soit 0,2 milliardièmes de seconde. Et sa période de bouclage est de 136 ans.

NTP a un autre format, sur seulement 32 bits, pour les cas où la taille compte, par exemple des objets contraints. Dans ce cas, le nombre de secondes et la fraction de seconde sont sur 16 bits chacun. Cela lui donne une période de bouclage très courte, seulement 18 heures.

Et le troisième format standard proposé est celui de PTP, dans sa version « tronquée » à 64 bits. Ce format est utilisé dans les RFC 6374, RFC 7456 ou RFC 8186. Il est surtout intéressant lorsqu'il faut interagir avec de l'instrumentation utilisant PTP. Sa résolution est d'une nanoseconde, sa période de bouclage est la même que celle de NTP mais comme son "*epoch*" est en 1970, cela remet le prochain bouclage à 2106.

(Vous noterez que notre RFC ne parle que des formats binaires pour représenter les estampilles temporelles. Il existe également des formats texte, notamment celui du RFC 3339, utilisé par exemple dans les RFC 5424, RFC 5646, RFC 6991, ou RFC 7493. Les formats de HTTP - en-têtes `Date:` et `Last-Modified:` - ainsi que l'"*Internet Message Format*" - IMF, RFC 5322 - n'utilisent hélas pas le RFC 3339.)

Deux exemples d'utilisation de ces formats binaires standards sont donnés dans notre RFC, aux sections 6.1 et 6.2, si vous êtes un ou une concepteur(trice) de protocoles qui veut inclure une estampille temporelle.

Mais, si, malgré les sages conseils du RFC, vous êtes l'auteur(e) d'un protocole et vous voulez quand même créer votre propre format binaire pour indiquer date et heure, y a-t-il des règles à suivre ? (À part la règle « ne le faites pas ».) Oui, la section 3 du RFC donne des indications (la plupart sont assez évidentes) :

— Bien documenter les raisons pour lesquelles vous tenez absolument à votre format à vous,
— indiquer évidemment la taille de chaque champ et la boutianité (ce sera du gros boutien par défaut),
— indiquer les unités utilisées, et l'"*epoch*",
— documenter le moment du bouclage, et les solutions prévues pour quand il sera atteint,
— et ne pas oublier de gérer les secondes intercalaires.

À propos de secondes intercalaires, que vous utilisiez un format à vous, ou bien la méthode recommandée d'un format existant, pensez aux aspects liés à la synchronisation des horloges entre elles (section 5 du RFC), sauf pour les cas d'horloges complètement isolées (mais les RFC sont écrits pour l'Internet, donc il est prévu de communiquer). Quel(s) protocole(s) utiliser, quelle référence de temps (UTC ? TAI ?), et comment gérer ces fameuses secondes intercalaires. Que faut-il faire si l'horloge recule, faut-il faire un changement brusque ou bien du "*smearing*" (cf. RFC 8633), etc.

Un avant-dernier détail : parfois, le format de l'estampille temporelle, et ses propriétés, sont décrits dans le protocole, ou dans un modèle formel, par exemple en YANG. C'est ce que fait OWAMP (RFC 4656) avec le champ donnant une estimation de l'erreur de mesure (section 4.1.2 du RFC 4656). Notre RFC nomme ces « méta » champs « champs de contrôle » et donne quelques règles à suivre quand on les utilise : extensibilité (on aura toujours besoin d'autre chose), taille (assez gros mais pas trop…), importance (obligatoire ou optionnel), catégorie (statique ou dynamique).

Enfin, il y a évidemment la question de la sécurité (section 9). Il est bien sûr obligatoire dans la description d'un protocole de fournir une analyse de sa sécurité (RFC 3552). Cela doit inclure les conséquences des estampilles temporelles sur la sécurité. Par exemple, de telles estampilles, si elles sont transmises en clair, peuvent renseigner un observateur sur les performances d'un réseau. Et, si elle ne sont pas protégées contre les modifications non autorisées (ce qui peut se faire, par exemple, avec un MAC), un attaquant peut fausser sérieusement le fonctionnement d'un protocole en changeant la valeur des estampilles.

Autre problème de sécurité, un attaquant actif qui retarderait délibérement les messages. Cela perturberait évidemment tout protocole qui utilise le temps. Et, contrairement à la modification des estampilles, cela ne peut pas se résoudre avec de la cryptographie. Enfin, la synchronisation des horloges est elle-même exposée à des attaques (RFC 7384). Si on déploie NTP sans précautions particulières, un attaquant peut vous injecter une heure incorrecte.