RFC 9172 : Bundle Protocol Security (BPSec)

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 1 février 2022

Date de publication du RFC : Janvier 2022

https://www.bortzmeyer.org/9172.html

Le protocole Bundle, normalisé dans le RFC 9171 [1], permet la communication entre des machines dont la connectivité est faible et/ou intermittente, par exemple dans le domaine spatial. L'absence de synchronicité interdit beaucoup de solutions de sécurité couramment utilisées sur l'Internet. Ce nouveau RFC présente donc un mécanisme spécifique pour assurer la sécurité (confidentialité et intégrité) des communications faites avec Bundle : ce mécanisme se nomme BPsec ("*Bundle Protocol security*"). Notez que le mécanisme est très général, et laisse de côté de nombreux détails.

BPsec peut assurer une sécurité de bout en bout. Ce n'est pas évident dans le contexte d'utilisation de Bundle (RFC 4838), les DTN ("*Delay Tolerant Networks*"). Bundle est un protocole qui fonctionne en « enregistre et fais suivre » ("*store and forward*"), sans liaisons directes entre émetteur et récepteur, et avec des réseaux lents, imprévisibles, à fort taux de perte de paquets. On ne peut donc pas garantir, par exemple, qu'émetteur et récepteur pourront communiquer à la demande avec un tiers de confiance, genre autorité de certification. On ne peut même pas supposer que toutes les liaisons seront bidirectionnelles, ce qui veut dire, entre autres, qu'un échange de clés Diffie-Hellman ne sera pas possible. Le protocole suppose évidemment que ledit réseau n'est pas de confiance : des méchants peuvent regarder et modifier les bits qui circulent. Cette supposition est classique en sécurité.

BPsec ne va pas essayer de garantir une authentification de chaque étape intermédiaire. D'abord, on ne sait pas si le nœud suivant est réellement adjacent dans l'espace physique (il peut y avoir des intermédiaires « invisibles ») et puis les différents nœuds par lequel le message va passer peuvent avoir des choix de sécurité incompatibles, les réseaux DTN pouvant, comme l'Internet, être composés de nœuds gérés par des organisations différentes.

Si les logiciels sont tous bien programmés, et que les clés privées utilisées n'ont pas été compromises, BPsec va garantir l'intégrité et la confidentialité des messages. BPsec est la continuation du cadre défini dans le RFC 6257, en le simplifiant et en le rendant plus réaliste.

Un peu de terminologie pour suivre ce RFC : d'abord, un rappel qu'un "*bundle*" est composé de blocs (RFC 9171, section 4.3). Ensuite :

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc9171.txt

— Bloc de sécurité ("*security block*") : un bloc d'extension dans un "*bundle*", une unité de données.
— "*Security acceptor*" : un nœud qui traite un bloc de sécurité, et le supprime ensuite. Ce n'est pas forcément le destinataire final (sauf évidemment pour les services de sécurité de bout en bout).
— Service de sécurité ("*security service*") : un service particulier, par exemple la confidentialité.
— Source de sécurité ("*security source*") : un nœud qui ajoute un bloc de sécurité. Ce n'est pas forcément l'émetteur initial.
— Vérificateur de sécurité ("*security verifier*") : un nœud qui lit et vérifie un bloc de sécurité mais qui ne le supprime pas.

La section 2 du RFC résume les points importants de la conception de BPsec :
— Sécurité au niveau du bloc, pas du "*bundle*" entier.
— Plusieurs sources ont pu mettre des blocs de sécurité dans le "*bundle*".
— Les différents nœuds ont des politiques de sécurité différentes.
— BPsec a une notion de contexte de sécurité (qui définit les techniques admissibles, par exemple les algorithmes de cryptographie), notion détaillée dans la section 9.
— Traitement déterministe des différents blocs de sécurité, notamment de l'ordre de traitement.

Il existe deux sortes de blocs de sécurité (les types de bloc figurent dans un registre IANA <https://www.iana.org/assignments/bundle/bundle.xml#block-types>, défini dans le RFC 6255), les BIB ("*Block Integrity Block*") et les BCB ("*Block Confidentiality Block*"). Les sources de sécurité ajoutent ces blocs et les "*acceptors*" les traitent. Dans le cas d'un chiffrement de bout en bout, par exemple, l'émetteur met un BCB que le destinataire déchiffrera. Dans un autre cas, on peut voir un nœud de départ ne mettre aucun bloc de sécurité (peut-être parce que ce nœud est un objet contraint, avec trop peu de ressources de calcul) mais un nœud intermédiaire ajouter un BIB pour protéger le contenu avant un voyage sur un lien qu'on sait dangereux. Ces blocs sont encodés comme spécifié dans le RFC 9171. L'ajout d'un BIB ne modifie pas le contenu du "*bundle*" mais celui d'un BCB va le faire, puisque les données seront chiffrées.

Le bloc contient notamment l'identificateur du nœud qui l'a ajouté.

Le bloc doit être mis sous forme canonique de CBOR (RFC 8949, section 4.2). Comme toujours en cryptographie, pour que les signatures soient vérifiables, il faut que les données soient sous une forme canonique.

La section 5 du RFC décrit le traitement des blocs de sécurité. Le nœud peut les passer tels quels, sans les interpréter, s'il n'est qu'un intermédiaire. Ou bien, s'il veut les valider, il va alors vérifier leurs signatures, déchiffrer, etc. En cas d'erreur, notre RFC ajoute cinq codes supplémentaires <https://www.iana.org/assignments/bundle/bundle.xml#status-reason> pour les signaler.

Pas de cryptographie sans clés et la gestion des clés est souvent le point difficile. La section 6 rappelle le problème. Comme les réseaux de type DTN seront très variés, avec des caractéristiques bien différentes, notre RFC ne décrit pas une méthode unique de gestion de clés. Disons que cette gestion est repoussée aux mises en œuvre ultérieures.

Enfin, la section 7 du RFC décrit les différentes politiques possibles, et la section 8 analyse en détail les caractéristiques de sécurité de ce système, face aux différentes menaces. Parmi les points amusants, le RFC note qu'avec les DTN, on peut s'attendre à ce que des "*bundles*" restent dans le réseau très longtemps, peut-être même des années (!) et que la cryptographie (choix des algorithmes, par exemple) doit donc être pensée pour durer.

Ce mécanisme de sécurité est mis en œuvre dans le logiciel libre ION <https://www.nasa.gov/directorates/heo/scan/engineering/technology/disruption_tolerant_networking_software_options>.