Gestion de risques appliquée aux réseaux RPL

Anthéa Mayzaud, Anuj Sehgal, Rémi Badonnel, Isabelle Chrisment

Gestion de risques appliquée aux réseaux RPL

Anthéa Mayzaud (anthea.mayzaud@inria.fr)*
Anuj Sehgal (s.anuj@jacobs-university.de)†
Rémi Badonnel (remi.badonnel@inria.fr)‡
Isabelle Chrisment (isabelle.chrisment@loria.fr)‡

Résumé : Le principe de l'Internet des Objets se traduit par le déploiement de réseaux avec pertes et à faible puissance appelés réseaux LLN [a]. Ces réseaux permettent à de nombreux équipements embarqués comme des sondes ou des capteurs de pouvoir communiquer entre eux. Un protocole de routage appelé RPL [b] a été spécialement conçu par l'IETF pour répondre aux contraintes spécifiques qu'impose ce type de réseaux. Néanmoins, ce protocole reste exposé à de nombreuses attaques de sécurité. Si des mécanismes de protection existent, leur mise en œuvre est coûteuse d'où l'intérêt d'une approche dynamique comme la gestion de risques permettant d'identifier, d'évaluer et de traiter les risques. Dans ce papier, nous proposons une approche de gestion de risques pour les réseaux RPL afin d'améliorer leur sécurité tout en minimisant la consommation de ressources induite par les contre-mesures. Nous en effectuons une évaluation à travers deux attaques spécifiques : l'attaque d'incohérence DAG et l'attaque sur le numéro de version.

a. Low power and Lossy Networks
b. Routing Protocol for LLN

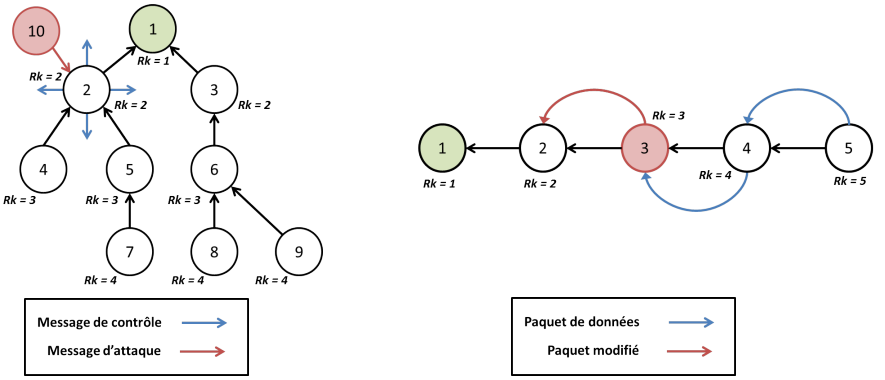
Mots Clés : Internet des Objets, Sécurité, Protocole RPL, Gestion de Risques

1 Introduction

L'émergence d'équipements à bas coût et à faibles ressources, capables de communications sans fil rend possible l'apparition de nouvelles applications allant du réseau électrique intelligent aux solutions d'e-santé. Le potentiel le plus intéressant de ces équipements réseaux contraints réside dans le fait de pouvoir être intégrés à l'infrastructure Internet existante. Ils peuvent ainsi utiliser les services déjà disponibles auxquels seront alors associés le contrôle des nœuds et leur capacité de collecte de données [AIM10].

Ces nœuds fortement contraints forment des réseaux dits de faible puissance et à pertes ou LLN tels que les réseaux de capteurs sans fil (*Wireless Sensor Network*) ou les systèmes domotiques. Ces réseaux ont généralement des contraintes fortes en matière de ressources (énergie, mémoire, puissance de calcul) et leurs liens sont caractérisés par de faibles débits et un important taux de pertes. De plus, les types de trafic ne sont pas uniquement point-à-point mais aussi point-à-multipoint et multipoint-à-point. Les protocoles de routages pour les réseaux filaires classiques (OSPF, IS-IS) et pour les réseaux ad-hoc (AODV, OSLR) ne conviennent pas aux spécifications de ce type de réseaux [LTDH09]. C'est

*. Inria Nancy Grand Est - Université de Lorraine
†. Jacobs University Bremen
‡. TELECOM Nancy - Université de Lorraine



(a) (b)

FIGURE 1: Scénarios d'attaque d'incohérence DAG. (a) L'attaquant, le nœud 10, cible le nœud 2 en envoyant des paquets avec le *flag* 'R'. (b) L'attaquant, le nœud 3, modifie les paquets reçus de ses descendants en plaçant le *flag* 'R' avant de les transmettre(Rk représente le rang du nœud).

pourquoi le groupe de travail RoLL de l'IETF [1] a conçu le protocole de routage à vecteur de distance RPL s'appuyant sur IPv6 [WTB+12]. Le protocole RPL construit des topologies appelées graphes acycliques orientés dirigés vers une destination ou DODAGs dont on peut voir des exemples sur la Figure 1. Les besoins de ces réseaux peuvent différer selon les cas : certains réseaux veulent optimiser la consommation d'énergie alors que d'autres doivent garantir la bonne réception des données. Par conséquent, RPL a été conçu de manière suffisamment flexible pour répondre aux exigences spécifiques de chaque situation en utilisant des métriques disponibles sur chaque nœud. Ces métriques sont utilisées pour optimiser la position du nœud dans le DODAG en calculant son rang par rapport à ses parents.

Le protocole RPL comporte des mécanismes de sécurité qui peuvent être utilisés pour garantir l'intégrité et la confidentialité des messages. Cependant, des fonctionnalités importantes comme la gestion des clés ne sont pas prises en compte par le standard. En outre, les algorithmes de chiffrement sont réputés pour être coûteux en mémoire et en cycles CPU ce qui pourrait considérablement affecter les performances du nœud. Cela signifie que la plupart des implantations de RPL ne mettent pas en place ces modes de sécurité. RPL reste donc exposé à de nombreuses attaques [TAD+13].

Nous proposons dans ce papier une approche de gestion de risques pour détecter et prévenir les attaques tout en préservant les ressources du réseau. La gestion de risques permet d'adapter dynamiquement la sélection de contre-mesures en fonction des menaces observées. Nous commencerons par présenter le protocole RPL et ses limites dans la Section 2. Nous détaillerons ensuite dans la Section 3 notre approche de la gestion de risques pour les réseaux RPL et évaluerons sa mise en œuvre dans la Section 4 à travers l'étude de deux attaques : l'attaque d'incohérence DAG et l'attaque sur le numéro de version. Enfin, la Section 5 présente la conclusion et identifie les travaux futurs.

1. Routing over Low power and Lossy Networks

2 Protocole RPL

Le protocole RPL est un protocole de routage à vecteur de distance utilisant IPv6, spécialement conçu par l'IETF pour répondre aux besoins des réseaux LLN. Cette section présente le fonctionnement de ce protocole et les mécanismes de protection existants.

Topologie, instance et fonction objectif. Les nœuds RPL s'interconnectent en formant une topologie spécifique appelée DODAG [2], c'est-à -dire un graphe acyclique orienté dirigé vers une destination qui est la racine du réseau. Un réseau RPL contient au moins une instance RPL qui elle-même se compose d'un ou plusieurs DODAGs. Chaque instance RPL est associée à une fonction objectif (OF) qui permet d'optimiser la topologie en fonction d'un ensemble de contraintes et/ou de métriques comme la préservation de l'énergie, le chemin le plus court ou la qualité des liens. Un nœud peut faire partie d'un seul DODAG par instance, mais peut participer à plusieurs instances simultanément.

Messages de contrôle et construction du DODAG. La construction et la maintenance des DODAGs sont réalisées grâce à des messages de contrôle ICMPv6. Plus particulièrement, trois nouveaux messages sont définis : (1) *DODAG Information Solicitation* (DIS), (2) *DODAG Information Object* (DIO) et (3) *Destination Advertisement Object* (DAO). Un nouveau nœud peut rejoindre un réseau déjà formé en diffusant un message DIS pour solliciter en réponse un message DIO qui contient des informations sur le DODAG comme le numéro de version et l'identifiant du DODAG, l'identifiant de l'instance et l'OF utilisée. Un nœud peut également attendre de recevoir un message DIO diffusé périodiquement par ses voisins. La fréquence d'envoi des messages DIO est déterminée par un temporisateur fondé sur l'algorithme Trickle [LCH+11] (appelé également temporisateur Trickle). À la moindre anomalie dans le réseau, le temporisateur Trickle est réinitialisé pour permettre à la topologie de reconverger.

Après avoir reçu un message DIO, le nœud calcule son rang en utilisant l'OF spécifiée dans ce message. Le rang d'un nœud correspond à son emplacement dans le graphe par rapport à la racine. La valeur du rang augmente toujours en descendant dans le graphe. C'est donc la racine qui a le rang le plus petit dans le graphe. Si un nœud reçoit des DIOs de voisins différents, l'émetteur avec le meilleur rang (le plus petit donc) est choisi comme le parent préféré vers lequel seront envoyés tous les messages à destination de la racine. À la fin de ce processus seulement les routes ascendantes (i.e. vers la racine) sont construites. Pour établir les routes descendantes, un nœud doit envoyer un message DAO à son parent contenant le préfixe des nœuds situés dans son sous-DODAG. Lorsque le message se propage vers la racine, les préfixes sont agrégés et les routes descendantes sont alors disponibles pour les parents.

Mécanismes de protection existants. RPL intègre différents mécanismes afin d'éviter les boucles, détecter les incohérences et réparer le graphe. Le rang joue un rôle important pour construire une topologie sans boucle. En effet, un nœud ne peut choisir qu'un parent dont le rang est inférieur au sien, autrement dit tous les nœuds se trouvant dans le sous-DODAG d'un nœud ont un rang supérieur à ce nœud. Si un nœud ne respecte pas cette propriété du rang, le graphe n'est plus acyclique. De plus, afin d'éviter les boucles, si un nœud doit changer son rang, il doit utiliser un mécanisme de *poisoning* (en annonçant un

2. Destination Oriented Directed Acyclic Graph

rang infini) ou de déconnexion (en formant un DODAG temporaire).

Dans les cas où des boucles apparaissent dans le graphe, le protocole RPL fournit une fonctionnalité appelée validation du chemin de données [3]. Des informations de contrôle sont transportées dans les paquets de données via des *flags* placés dans l'en-tête d'extension IPv6 Hop-By-Hop :

• Le *flag* 'O' indique la direction attendue du paquet, i.e., vers le haut ou le bas. Si un nœud place ce *flag* à 1 le paquet est destiné à un descendant, sinon le paquet est supposé être envoyé à un parent avec un rang inférieur, vers la racine du DODAG.

• Le *flag* 'R' indique si une erreur de rang a été détectée par un nœud transférant le paquet. Ce *flag* est mis à 1 lorsqu'un nœud observe une incohérence entre la direction supposée du paquet indiquée par le *flag* 'O' et le rang du nœud qui vient de le transférer. Le *flag* 'R' est utilisé pour réparer ce type d'anomalie appelée incohérence DODAG. Concrètement, à la première incohérence détectée, le nœud place ce *flag* à 1 et transfère le message. Lors de la réception d'un autre paquet avec le *flag* positionné à 1 et l'incohérence à nouveau détectée, le paquet est supprimé et le temporisateur Trickle est réinitialisé de sorte que les messages de contrôle sont envoyés plus rapidement, afin de refaire converger la topologie et réparer la boucle.

Deux principaux mécanismes de réparation sont utilisés dans les réseaux RPL en cas d'incohérences ou de pannes : la réparation locale et globale. La réparation locale consiste à trouver un chemin alternatif pour router les paquets. Par exemple, lorsque que la communication avec le parent préféré est rompue, un nœud peut choisir un autre parent pour transférer ses paquets. Si aucun autre parent n'est disponible, il peut aussi envoyer les paquet à un frère, i.e., un nœud de même rang [KSS12]. Si les réparations locales ne suffisent pas, la racine peut initier une réparation globale c'est-à-dire la reconstruction complète du graphe en incrémentant le numéro de version du DODAG.

Concernant la sécurité en tant que telle, RPL propose deux modes de sécurité. Le premier s'appelle mode "pré-installé" et consiste à chiffrer les messages à l'aide de clés pré-installées sur les nœuds. Le second, le mode authentifié, fonctionne comme le mode précédent. Cependant, si un nœud veut participer en tant que routeur il doit obtenir une autre clé d'une autorité authentifiée. Avec la clé pré-installée, un nœud ne peut participer qu'en tant que feuille dans le graphe. Néanmoins, le standard ne définit pas comment mettre en place concrètement ces deux modes de sécurité, dans quel contexte les sélectionner, ni comment la gestion des clés s'opère.

3 Approche de gestion de risques

La gestion de risques permet d'identifier, évaluer et traiter les risques auxquels sont confrontés les réseaux et les systèmes d'information. Elle offre de nouvelles perspectives pour activer ou désactiver dynamiquement des mécanismes de sécurité dans les réseaux RPL, et ce, dans le but de bloquer les attaques sur le réseau tout en conservant ses performances. Nous proposons d'examiner les méthodes et techniques de gestion de risques dans l'Internet des Objets afin d'établir un compromis entre la sécurité et son coût.

Le niveau de risque est traditionnellement défini comme la combinaison de la probabilité d'une attaque et de ses conséquences. On peut aussi le définir comme donné dans

3. data path validation

l'équation 1 [NIS95].

$$\mathcal{R} \quad = \quad \sum_{a \in A} P(a) \times E(a) \times C(a) \tag{1}$$

Prenons une attaque, notée a. Le risque total du réseau se définit comme la somme sur toutes les attaques possibles de chaque niveau de risque. Le niveau de risque $R(a)$ dépend de la potentialité $P(a)$ de l'attaque, de l'exposition $E(a)$ [4] du réseau RPL et des conséquences $C(a)$ de l'attaque sur le réseau si elle réussit [DBF10]. La gestion de risques consiste à surveiller, hiérarchiser et contrôler les risques [BC01]. Par exemple, si on observe une forte potentialité $P(a)$, i.e., une attaque en cours, on peut activer un mécanisme de sécurité en prenant en compte son coût afin de réduire l'exposition $E(a)$ et maintenir le niveau de risque $R(a)$ à une valeur raisonnable [GG04].

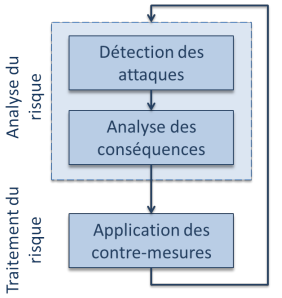


FIGURE 2: Schématisation du processus de gestion de risques

Comme le décrit la Figure 2, la gestion de risques se compose de deux principaux processus : l'analyse du risque et le traitement du risque.

L'analyse du risque se définit comme la quantification de la potentialité de l'attaque ainsi que ses conséquences. Pour cela, il est nécessaire d'évaluer la performance des techniques de détection disponibles (fondées sur la détection d'anomalies ou sur des signatures d'attaques connues) dans les environnements RPL. Il est aussi essentiel d'identifier les nœuds capables d'assurer cette activité de surveillance. Pour compléter l'analyse du risque, les conséquences des attaques en cas de succès doivent être également quantifiées. L'objectif est d'estimer l'importance relative des nœuds dans le réseau RPL et d'analyser les effets de l'attaque contre un nœud donné sur l'ensemble du réseau.

Le traitement du risque consiste, quant à lui, en la sélection et l'application des mécanismes de sécurité requis afin de réduire le niveau de risque à une valeur acceptable. Plusieurs stratégies peuvent être envisagées : (1) l'évitement c'est-à-dire que les mesures sont préventives et appliquées en anticipation de l'attaque, (2) l'optimisation où un ensemble de contre-mesures sera appliqué de manière réactive afin de réduire l'exposition des nœuds, et enfin (3) l'acceptation si l'on considère que les conséquences de l'attaque sont acceptables, par exemple, si les données transportées par le réseau ne sont pas sensibles, on peut négliger les attaques d'écoute dans certains contextes.

Si ce type d'approche dynamique a été mise en place dans différents systèmes et réseaux [GG04], [DBF10], elle n'a pas été réellement appliquée dans l'Internet des Objets.

4. Ensemble des vulnérabilités du réseau

4 Mise en œuvre

Cette section présente comment nous avons appliqué notre approche de gestion de risques dans le cas de deux attaques que nous avons identifiées et qui sont spécifiques aux réseaux RPL. Pour cela, nous avons repris le processus expliqué dans la section précédente, en identifiant les métriques utiles pour la détection des attaques et la quantification des conséquences, puis en étudiant des contre-mesures existantes ou conçues pour traiter le risque de ces attaques.

4.1 Attaques d'incohérence DAG

Description de l'attaque. En théorie, le mécanisme de validation du chemin de données, présenté dans la Section 2, a pour objectif d'améliorer la fiabilité générale du réseau. Cependant, il est possible de détourner cette fonctionnalité pour attaquer le réseau. En effet, un attaquant peut faire croire à des nœuds qu'il y a des boucles alors que la topologie est stable.

Deux approches peuvent être adoptées par l'attaquant : (1) créer des paquets malveillants directement avec le *flag* 'R' et la mauvaise direction positionnés, (2) modifier les paquets qui transitent par lui en positionnant les *flags* contenus dans l'en-tête d'extension. Dans les deux cas, la conséquence immédiate de cette attaque est l'inondation du réseau en messages de contrôle, puisque tous les nœuds victimes ayant reçus les paquets malveillants ainsi que leur voisinage réinitialiseront leur temporisateur Trickle. Ceci réduira à terme la durée de vie du réseau. Dans le second cas plus particulièrement, les paquets modifiés seront supprimés par le parent de l'attaquant (le nœud 2 dans le scénario décrit dans la Figure 1.b.) ce qui engendrera un *blackhole* ou trou noir déporté sur ce parent où les paquets de données seront perdus.

Analyse du risque. L'analyse du risque se décompose en quantification de la potentialité et des conséquences. Quantifier la potentialité de l'attaque revient à trouver une métrique traduisant l'attaque et permettant de la détecter. Dans le cas de l'incohérence DAG, une métrique efficace est le nombre de messages avec le *flag* 'R' que nous appellerons *Counter*. Si cette valeur dépasse un certain seuil, le nœud peut considérer qu'il est attaqué. Nous détaillerons comment le seuil peut être défini en fonction de la contre-mesure choisie.

Concernant la quantification des conséquences, nous avons étudié l'attaque en utilisant deux scénarios correspondant aux deux stratégies possibles pour l'attaquant, comme le décrit la Figure 1. Les métriques choisies pour évaluer le coût de l'attaque sont le surcoût en messages de contrôle reçus et envoyés ainsi que le taux de transfert exprimé comme le rapport entre le nombre de paquets reçus effectivement par la racine et le nombre de paquets de données envoyés. Cette dernière métrique est seulement utilisée pour le deuxième scénario, car le taux de transfert est inchangé lorsque l'attaquant ne modifie pas de paquets de données. Dans le premier scénario, on fait varier la fréquence de l'attaquant, c'est-à-dire le nombre de paquets malveillants envoyés par heure. On constate dans la Figure 3, que la quantité de messages de contrôle générée est multipliée par 3 pour l'attaque la moins agressive et jusqu'à 7 pour l'attaque la plus agressive. Dans le deuxième scénario, l'attaquant positionne les *flags* dans chaque paquet de données qu'il doit transférer ce qui a pour conséquence de faire chuter le taux de transfert à 0% pour les nœuds 4 et 5 situés dans son sous-DODAG, en plus des conséquences de la première attaque, à savoir le surcoût en messages.

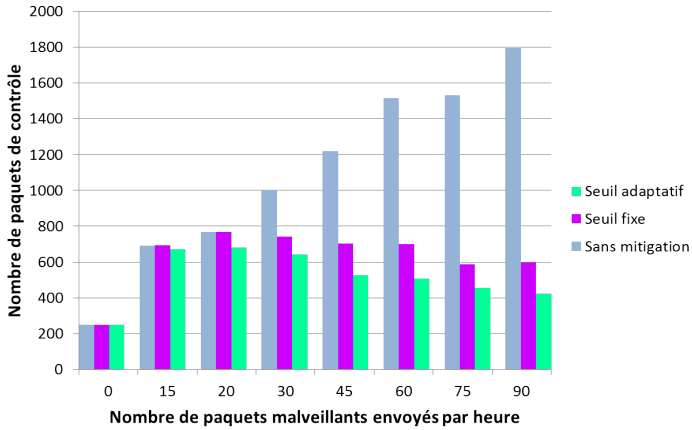


FIGURE 3: Surcharge en messages de contrôle subis par le réseau lors d'une attaque d'incohérence DAG

Traitement du risque. Afin de diminuer l'impact d'une telle attaque sur le réseau, plusieurs contre-mesures peuvent être envisagées et appliquées. La première est proposée par les auteurs de la RFC 6553 [HV12], il s'agit de limiter le nombre de réinitialisations du temporisateur Trickle dû à la détection d'incohérences DAG à 20 par heure. Autrement dit lorsque *Counter* atteint le seuil fixe 20, le nœud applique l'action "ne plus réinitialiser le temporisateur Trickle". La seconde contre-mesure que nous proposons, est de limiter aussi le nombre de réinitialisations, mais en utilisant un seuil dynamique $\lambda(r)$ comme défini par la formule 2. E_{pkts} représente le nombre de paquets reçus avec le *flag* 'R' et D_{pkt}, le nombre de paquets de données légitimes qui ont été transférés par le nœud. La variable r est calculée localement par chaque nœud ; α a été fixé de telle manière que le seuil n'atteigne jamais une valeur nulle, β a été déterminé de sorte qu'à un état sans attaque le seuil soit égal à la valeur du seuil fixe proposée par les auteurs de la RFC6553 [HV12] à savoir 20, γ quant à lui, sert à faire varier la pente de l'exponentielle décroissante afin de prendre en compte l'agressivité de l'attaquant.

$$\lambda(r) = \lfloor \alpha + \beta \cdot e^{1-\gamma \cdot r} \rfloor \text{ avec } \; r = \frac{E_{pkt}}{D_{pkt}}, \; \alpha = 5, \; \beta = \frac{15}{e}, \; \gamma = 25 \tag{2}$$

La Figure 3 présente le nombre de messages de contrôle envoyés sans mitigation, avec mitigation par seuil fixe et enfin, avec mitigation par seuil adaptatif. On constate une diminution significative du nombre de messages de contrôle générés : la surcharge en messages descend en dessous de 800 pour le seuil fixe et en dessous de 700 pour l'approche dynamique, donc un gain considérable pour l'ensemble des nœuds du réseau. Lorsqu'une contre-mesure est appliquée, la meilleure stratégie pour un attaquant est de limiter son attaque. L'avantage de l'approche dynamique par rapport au seuil fixe est qu'en cas d'attaque agressive le seuil est atteint plus vite et limite considérablement la surcharge du réseau. De plus, avec un seuil fixe, l'attaquant pourrait prévoir la meilleure stratégie à adopter et bénéficier de la réinitialisation de *Counter* toutes les heures. Le seuil dynamique ne réaugmente quant à lui que lorsque le nœud ne reçoit plus de paquets avec le *flag* 'R'. La Figure 4.a. résume le processus de gestion de risques appliqué à cette attaque.

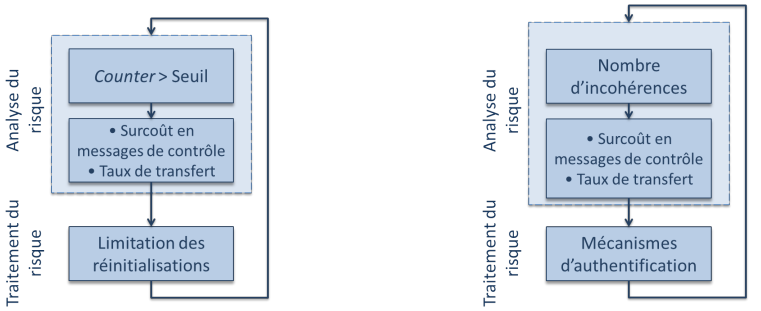
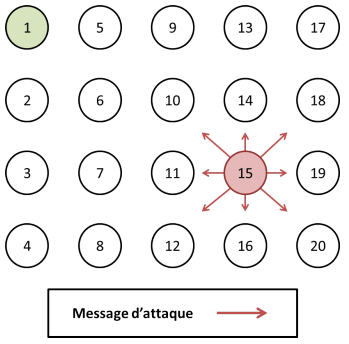


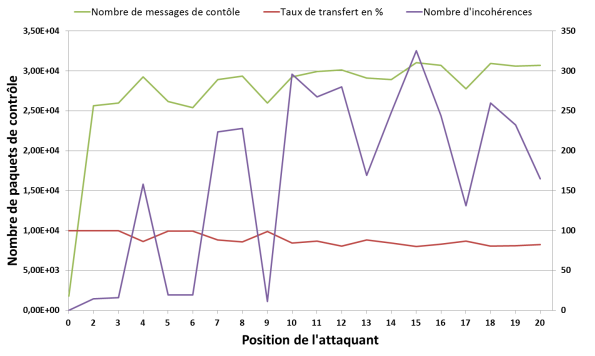
FIGURE 4: Gestion de risques appliquée à (a) l'attaque d'incohérence DAG et (b) l'attaque sur le numéro de version

4.2 Attaques sur le numéro de version

Description de l'attaque. Le numéro de version est un champ important dans les messages DIO. Il doit être propagé sans être modifié le long du DODAG. Seule la racine peut l'incrémenter afin de créer une nouvelle version du DODAG pour revalider l'intégrité du réseau et permettre une réparation globale. Si un nœud annonce une version plus ancienne, cela signifie qu'il n'a pas migré vers la nouvelle version et qu'il ne doit pas être choisi en tant que parent. Un attaquant peut changer la version du DODAG en incrémentant illégitimement le champ correspondant dans ses messages DIO avant de les transmettre à ses voisins. Ceci aura pour conséquence la génération potentielle de boucles dans le graphe et la reconstruction entière du DODAG impliquant un épuisement de la batterie des nœuds. Afin d'étudier cette attaque, nous avons utilisé la structure présentée dans la Figure 5a où nous faisons varier la position de l'attaquant qui envoie un message DIO où le numéro de version est incrémenté toutes les 60 secondes. La topologie n'est pas représentée dans ce schéma, la nature de l'attaque rendant la topologie instable durant l'expérience.



(a) Topologie en grille. L'attaquant, à la place du nœud 15 envoie des messages DIO dont le numéro de version est incrémenté.



(b) Représentation de la surcharge en messages de contrôle, du délai, du taux de transfert et du nombre d'incohérences pour chaque position de l'attaquant, 0 représente le scénario sans attaquant

FIGURE 5: Scénario utilisé et résultats obtenus pour l'attaque sur le numéro de version

Analyse du risque. Pour quantifier la potentialité de l'attaque on peut utiliser le nombre de boucles créées dans le graphe. En effet, il est possible d'inclure des compteurs notamment quand une incohérence est détectée, i.e., lorsque la direction du paquet supposée ne correspond pas à la direction réelle du paquet ; de même lorsque cette incohérence perdure (incohérence DAG) par le biais du *flag* 'R'. Cette métrique est représentée dans la Figure 5b où l'on peut observer que, plus l'attaquant a de voisins, plus il y aura d'incohérences dans le réseau comme c'est le cas en 10, 11, 12, 14 et 15. Cette attaque a fait l'objet d'une publication [MSB+14] présentant ces résultats en détail.

Afin d'évaluer les conséquences d'une telle attaque sur le réseau, on peut aussi utiliser comme métriques le nombre de paquets de contrôle générés suite à l'attaque ainsi que le taux de transfert afin de déterminer la quantité de paquets perdus. On remarque l'évidente corrélation entre le nombre d'incohérences et la quantité de paquets de contrôle générés, de même pour le taux de livraison qui est maximisé lorsque le nombre d'incohérences est minimisé.

Traitement du risque. Pour contrer cette attaque, Dvir et al. [DHB11] ont proposé un mécanisme de sécurité appelé VeRa [5] qui empêche un nœud compromis d'usurper la racine en modifiant le numéro de version. La solution utilise un système d'authentification fondé sur des opérations de hachage et de signatures avec des clés partagées. Un nœud peut ainsi vérifier que c'est la racine qui est à l'origine du changement de version. Cependant les auteurs de [LPU+13, PLU+13] ont montré que VeRa n'était pas sûr. Ils ont proposé TRAIL [6] en reprenant l'idée général de VeRa mais en incluant des challenges devant être signés par la racine. Ils ont de plus montré que leur solution était fiable pour un coût assez faible en comparaison du gain en cas d'attaque. On peut ainsi considérer cette contre-mesure comme traitement du risque bien qu'elle ne soit pas dynamiquement activable, ce qui pourrait faire l'objet d'une extension pour cette solution.

5 Conclusion

L'Internet des Objets amène au déploiement de réseaux LLNs. Ceux-ci se caractérisent par de faibles ressources en matière d'énergie, de puissance de calcul, de mémoire et reposent sur des liens contraints. Leur développement a conduit à la spécification par le groupe de travail RoLL à l'IETF, d'un protocole dédié, appelé RPL. Ces réseaux sont exposés à de nombreuses attaques. Bien que des mécanismes de sécurité soient prévus ou peuvent être adaptées, leur mise en œuvre peut dégrader les performances du réseau. Nous proposons une approche de gestion de risques dans ces réseaux afin d'obtenir un compromis entre la sécurité et son coût. L'objectif est d'adapter dynamiquement l'exposition du réseau en fonction de la potentialité d'une menace à travers l'activation ou la désactivation de contre-mesures dédiées. Nous avons évalué, grâce à l'étude de deux attaques spécifiques aux réseaux RPL, comment appliquer cette approche en identifiant des métriques pour la quantification du risque et différentes contre-mesures possibles. Comme travaux futurs, nous allons étendre notre étude à d'autres type d'attaques contre les réseaux RPL comme les attaques de *selective forwarding* et les attaques sur le rang. Afin de poursuivre nos travaux sur la gestion de risques, nous prévoyons de quantifier les performances de dif-

5. Version Number and Rank Authentication
6. Trust Anchor Interconnection Loop

férents algorithmes de sélection dynamique des contre-mesures en environnements RPL.

Remerciements

Ce travail est en partie financé par Flamingo, un projet de réseau d'excellence (ICT-318488), soutenu par la Commission Européenne.

Références

[AIM10] L. Atzori, A. Iera, and G. Morabito. The Internet of Things : A survey. *Elsevier Journal Computer Networks*, 54(15) :2787–2805, October 2010.

[BC01] T. Bedford and R. Cooke. *Probabilistic Risk Analysis : Foundations and Methods*. Cambridge University Press, 2001.

[DBF10] O. Dabbebi, R. Badonnel, and O. Festor. Managing Risks at Runtime in VoIP Networks and Services. In *IFIP AIMS'2010*, pages 89–92, June 2010.

[DHB11] A. Dvir, T. Holczer, and B. Buttyán. VeRA - Version Number and Rank Authentication in RPL. In *MASS*, pages 709–714, 2011.

[GG04] A. Gehani and K. Gershon. RheoStat : Real-time Risk Management. In *Proc. of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID'2014)*, pages 15–17, 2004.

[HV12] J. Hui and J. Vasseur. The Routing Protocol for Low-Power and Lossy Networks (RPL) Option for Carrying RPL Information in Data-Plane Datagrams. RFC 6553 (Proposed Standard), March 2012.

[KSS12] K. Korte, A. Sehgal, and J. Schönwälder. A Study of the RPL Repair Process Using ContikiRPL. In *IFIP AIMS*, pages 50–61, 2012.

[LCH+11] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko. The Trickle Algorithm. RFC 6206 (Proposed Standard), March 2011.

[LPU+13] M. Landsmann, H. Perrey, O Ugus, M. Wählisch, and T.C. Schmidt. Topology Authentication in RPL. In *Proc. of the 32nd IEEE INFOCOM. Poster*. IEEE Press, 2013.

[LTDH09] P. Levis, A. Tavakoli, and S. Dawson-Haggerty. Overview of Existing Routing Protocols for Low Power and Lossy Networks, IETF Internet Draft : draft-ietf-roll-protocols-survey-07, April 2009.

[MSB+14] A. Mayzaud, A. Sehgal, R. Badonnel, I. Chrisment, and J. Schönwälder. A Study of RPL DODAG Version Attacks (accepted). In *AIMS*, 2014.

[NIS95] NIST. An Introduction to Computer Security : The NIST Handbook, 1995.

[PLU+13] H. Perrey, M. Landsmann, O. Ugus, M. Wählisch, and T.C. Schmidt. TRAIL : Topology Authentication in RPL. 2013.

[TAD+13] T. Tsao, R. Alexander, M. Dohler, V. Daza, A Lozano, and M. Richardson. A Security Threat Analysis for Routing Protocol for Low-power and Lossy Networks (RPL), IETF Requirement Draft for Routing over Low power and Lossy Networks (ROLL), Work in progress., December 2013.

[WTB+12] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander. RPL : IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550 (Proposed Standard), IETF, 2012.