Itinérance dans les réseaux LoRaWAN †

Mohamed Hamnache, Rahim Kacimi, André-Luc Beylot

Itinérance dans les réseaux LoRaWAN[†]

Mohamed Hamnache[1], Rahim Kacimi[2] et André-Luc Beylot[1]

[1] *Université de Toulouse, IRIT-ENSEEIHT, Toulouse, France*
[2] *Université de Toulouse, IRIT-UT3, Toulouse, France*

LoRaWAN est une technologie prometteuse pour les objets connectés. Elle est déjà utilisée dans de nombreux contextes en raison de ses qualités intrinsèques. Cependant, comme pour toutes les technologies sans fil, la mobilité reste une préoccupation majeure amenant les terminaux hors de portée de leur opérateur. Dans cet article, nous étudions la capacité d'itinérance inter-opérateurs dans des scénarios de mobilité. Nous proposons un nouveau mécanisme d'itinérance fondé sur la résolution DNS et la migration du contexte des terminaux entre réseaux. Nous étendons l'architecture LoRaWAN en maintenant l'intégrité des mécanismes existants et avec un minimum d'exigences de configuration préalable. Afin de valider la solution, nous avons mis en œuvre une plateforme de test en l'intégrant dans la suite logicielle Chirpstack.

Mots-clefs : LoRaWAN, Itinérance, Mobilité

1 Introduction

L'Internet des objets (IoT) constitue des enjeux cruciaux pour les sociétés modernes et intelligentes, regroupant une grande variété de technologies de communication, principalement par le biais de capteurs sans fil alimentés par batterie. L'une des technologies les plus en vogue pour l'industrie et le monde académique est celle des réseaux étendus à faible consommation (LPWAN) et en particulier LoRaWAN [AVT+17].

Ils complètent les réseaux cellulaires et sans fil en répondant aux attentes de l'IoT. LoRaWAN offre des caractéristiques exceptionnelles et bien adaptées : longue portée, faible consommation d'énergie, faible débit, capacité réseau élevée, terminaux peu coûteux, souplesse de déploiement [HKB20]. Cela convient parfaitement à de nombreux contextes IoT tels que la surveillance de l'environnement, les villes intelligentes, l'agriculture de précision... L'utilisation des bandes sans licence permet aux individus et aux organisations de gérer leurs propres réseaux privés sans impliquer de tiers. Cela donne naissance à un large écosystème de réseaux LoRaWAN fournis par de multiples opérateurs, permettant d'atteindre une large couverture.

Pour certaines applications telles que le suivi, la localisation et la logistique intelligente, les terminaux sont capables de quitter les cellules de leur opérateur d'origine. Malheureusement, dans ces scénarios de mobilité [ASN+19], les opérateurs ne peuvent pas fournir une couverture plus large qui ne serait pas viable économiquement pour un seul opérateur ni techniquement pratique. Cependant, LoRaWAN est une technologie open source qui permet aux opérateurs et aux particuliers de créer des réseaux privés ou publics. Par conséquent, l'interconnexion de tous ces réseaux via des accords d'itinérance permet d'atteindre une sorte de couverture mondiale. L'unification de ces réseaux par l'activation de l'itinérance [JSN+21] LoRaWAN nécessite la capacité (i) de découvrir le réseau domestique du terminal mobile, (ii) de vérifier que l'opérateur autorise l'itinérance pour ce terminal, et (iii) d'en récupérer le contexte.

Dans cet article, nous proposons une nouvelle solution d'itinérance dans les réseaux LoRaWAN. Plus précisément, nous étendons l'architecture LoRaWAN sans compromettre les mécanismes existants pour interconnecter différents opérateurs. Notre solution est conçue pour être intégrée dans les infrastructures déployées avec un minimum de configuration préalable. Nous introduisons une souscription du réseau au service d'itinérance et le provisionnement des terminaux avec des champs JoinEUI formatés. Nous proposons ensuite des mécanismes d'itinérance : résolution DNS et migration du contexte entre réseau d'origine et réseau visité. Un banc d'essai de notre proposition a été mis en œuvre pour en démontrer la faisabilité.

[†] **Ce travail a été accepté et publié à IEEE LCN 2021 (ref. [HKB21]) et CoRes 2022.**

2 Solution d'Itinérance dans les Réseaux LoRaWAN

Décrivons maintenant l'extension de l'architecture LoRaWAN permettant la continuité de service aux terminaux en dehors des zones de couverture de leur opérateur [HKB21]. La topologie du réseau LoRaWAN est décrite comme une étoile d'étoiles avec quatre composants : le serveur réseau (NS), le serveur d'application (AS), les passerelles et les terminaux. Dans les scénarios classiques, les terminaux communiquent avec leur NS domestique (hNS) via les passerelles. Cependant, ils peuvent se trouver hors de la couverture de leur opérateur et visiter des zones couvertes par d'autres réseaux. Une solution évidente consiste à exploiter ces infrastructures d'autres opérateurs par le biais d'accords d'itinérance. Ainsi, afin d'établir une interconnexion entre les opérateurs participant à l'itinérance, nous avons étendu l'architecture LoRaWAN avec des composants logiciels locaux (HLA et VLA) et centraux (MA) comme le montre la figure 1.

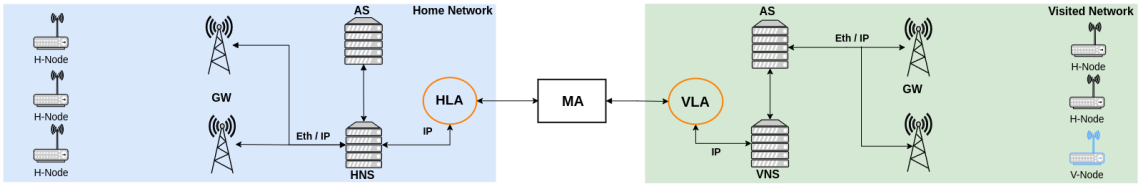

FIGURE 1 : Architecture LoRaWAN étendue

2.1 Master Agent (MA)

Ce composant central crée une interconnexion entre réseaux LoRaWAN. La mise en place d'un mécanisme d'itinérance devient essentielle pour permettre les communications entre opérateurs. Le MA commence par fournir une procédure de souscription au réseau afin d'attribuer un Net-ID unique de 32 bits à chaque réseau participant à l'itinérance. Il est extrait des trames reçues pour effectuer une résolution DNS afin de chercher le réseau domestique du terminal. Cette résolution diffère de la résolution DNS classique : elle permet de trouver le réseau domestique d'un nœud à partir d'un JoinEUI unique.

2.2 Local Agent (LA)

Ce composant local est connecté au NS. Il joue le rôle d'agent local visité (VLH) lorsqu'il traite les trames provenant de nœuds itinérants à destination du serveur de réseau visité (vNS) ou d'agent local domestique (HLA) pour celles provenant d'un réseau partenaire dans le cadre d'accords d'itinérance. Un VLA (ou HLA) est divisé en 4 sous-composants : 1) Le client MA représente une interface de connexion avec le MA. Il s'abonne aux services fournis par le MA pour une nouvelle souscription au partenariat d'itinérance ou pour une résolution DNS. 2) Le client NS interagit avec le serveur de réseau (NS). Il traite les trames entrantes, les filtre en fonction du type de message et gère les terminaux. 3) Le Device Context Manager récupère le contexte du nœud sur le serveur du réseau domestique, puis l'envoie au VLA correspondant. Il sera stocké dans le vNS. 4) Le Peer Manager gère l'échange des transmissions entre le vNS et le hNS après que le JoinRequest a abouti et que le contexte du terminal est arrivé à son extrémité.

2.3 Procédure d'Activation

Pour s'associer à un réseau LoRaWAN, un terminal doit le rejoindre. Il envoie une demande d'adhésion ou ré-adhésion (JoinRequest, RejoinRequest) et attend un accusé de réception (JoinAccept). La procédure d'adhésion dans l'architecture d'itinérance LoRaWAN proposée nécessite deux phases principales.

2.3.1 Approvisionnement des Terminaux

Le JoinEUI est un identifiant codé sur 64 bits. Il est utilisé pour cibler le serveur Join qui gère la procédure d'activation des terminaux. Avant d'autoriser un terminal à utiliser un réseau partenaire, il doit recevoir un JoinEUI spécifique attribué par le HLA associé à son opérateur. Le JoinEUI est engendré de manière unique par le HLA afin de permettre la résolution DNS précédemment évoquée. Il est divisé en deux champs de quatre octets. Le premier représente un Net-ID unique attribué par le MA après une souscription au réseau.

A la réception de JoinRequest, ce champ est extrait et utilisé pour trouver le serveur de réseau domestique d'un terminal donné. Les quatre derniers octets représentent un identifiant unique universel attribué par le HLA. L'étape d'approvisionnement JoinEUI est très importante pour identifier de manière unique les terminaux afin d'acheminer le trafic vers le hNS correspondant.

2.3.2 Protocole d'Activation d'un Terminal Itinérant

La procédure d'activation pour la première visite d'un terminal d'un opérateur partenaire est résumée dans la Figure 2. Pour des raisons de lisibilité le vNS et le hNS ne sont pas représentés. Ils s'intercalent comme présenté dans la Figure 1.

Lorsque l'itinérance est activée, le vNS gère les couches physique et MAC. Le terminal et le vNS doivent effectuer un handshake d'activation : un JoinRequest est envoyé par le terminal et un JoinAccept par le vNS. Le JoinRequest est envoyé en clair et contient le JoinEUI obtenu dans la phase précédente, le DevEUI et un Nonce. Comme il est envoyé en clair, tout réseau LoRaWAN l'interceptant peut en lire le contenu, y compris le vNS. Lorsque le vNS reçoit des trames de sources inconnues, il les transmet au VLA qui effectue une vérification de base pour valider la structure de la trame et détecter le type de message.

Si la vérification est réussie, le VLA construit une requête composée du JoinRequest reçu sur le vNS et d'une VLAReq qui contient l'adresse IP du vNS et son nom de domaine. Cette demande est ensuite envoyée au MA qui en vérifie la source. Un JoinRequest valide envoyé par un terminal itinérant doit contenir un Net-ID connu par le MA. En effet, le MA extrait le Net-ID du JoinEUI reçu dans le JoinRequest. Si le MA connaît ce Net-ID, il procède à une résolution DNS et transmet la requête au HLA correspondant. Ensuite, il répond au VLA avec une DNSResp comprenant l'adresse IP du hNS et son nom de domaine.

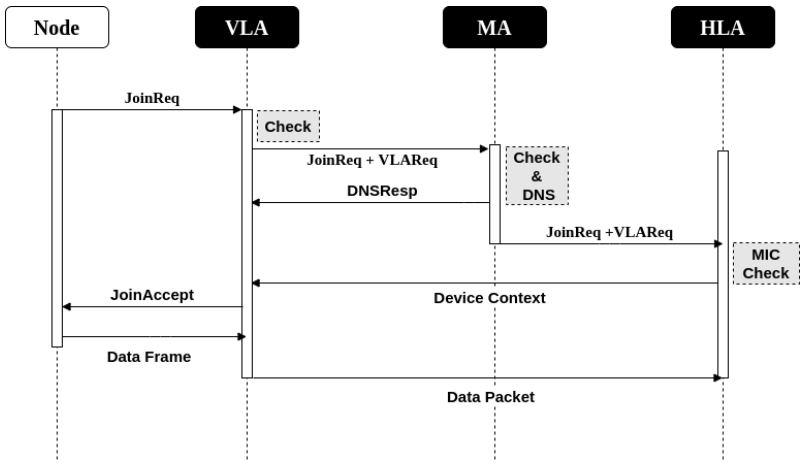

FIGURE 2 : Protocole d'itinérance LoRaWAN

Dès que la demande du MA est reçue sur le HLA, il effectue un contrôle d'intégrité. En effet, seules les entités possédant le NwkKey peuvent affirmer que le message n'a pas été altéré grâce au code d'intégrité du message(MIC). En retour, le HLA répond au VLA correspondant en utilisant les informations incluses dans la VLAReq en envoyant le contexte du terminal. Une fois ce contexte reçu sur le VLA, il est stocké sur le vNS. À ce stade, le vNS est en mesure de répondre nœud par un JoinAccept. Une fois que le terminal est activé, il peut commencer à envoyer des trames. À la réception d'une trame sur le vNS, celui-ci la transmet au hNS correspondant sur la base du DNSResp déjà stocké lors des phases précédentes. Si un changement est intervenu sur l'adresse IP et/ou le nom de domaine du hNS, le MA envoie une mise à jour au VLA.

Toutes ces communications pour la résolution DNS et la récupération du contexte du terminal ne sont effectuées qu'à la première visite du réseau ou pour la mise à jour du contexte du nœud. Puisque le contexte du terminal est stocké sur le vNS, la couche MAC est entièrement gérée par celui-ci, qui est capable de gérer l'activation des nœuds itinérants. Ainsi, le vNS peut traiter les prochaines JoinRequest ou ReJoinRequest provenant de nœuds mobiles dont le contexte a déjà migré et est stocké.

3 Implantation et Résultats

Pour valider la solution conçue pour l'itinérance LoRaWAN présentée dans la section 2, nous avons implanté et déployé un banc d'essai fondé sur la suite logicielle ChirpStack. L'utilisation de ChirpStack est justifiée par son architecture modulaire qui permet son intégration dans des infrastructures existantes et qui est ouverte à d'éventuelles extensions. Chirpstack, au moment de la rédaction de cet article, met en œuvre une résolution DNS spécifique à des fins d'itinérance passive. Nous avons étendu l'architecture LoRaWAN avec trois composants principaux pour mettre en œuvre un nouveau mécanisme de résolution DNS qui s'appuie sur le JoinEUI pour délocaliser le contrôle de la couche MAC du terminal itinérant, qui sera maintenu par le vNS. L'implantation des composants est détaillée comme suit : i) Le MA est un composant central écrit en python qui fournit une API offrant deux services principaux : la génération de Net-ID et la résolution DNS fondée sur le JoinEUI pour trouver le réseau domestique d'un terminal. ii) Le VLA/HLA est un composant python divisé en plusieurs sous-composants : des clients pour interagir avec chirpstack et le MA, un composant pour gérer la migration et la sauvegarde du contexte des terminaux et enfin un gestionnaire de communication entre les opérateurs partenaires.

Pour évaluer la solution d'itinérance proposée, nous avons considéré un déploiement de deux réseaux LoRaWAN sur deux sites différents, séparés de 6 km dans une zone urbaine. Plus précisément, dans notre déploiement, nous avons utilisé deux passerelles Mikrotik wAP LR8b sur les deux sites. Nos expérimentations sont menées avec des terminaux Pycom Lopy4 et Microchip RN2483 intégrés à des Raspberry Pi3. Une première version de notre solution est disponible sur github et peut être consultée sur [Ham22].

L'étude des délais engendrés par le protocole d'itinérance montre que la résolution DNS et la migration du contexte des terminaux prend en moyenne 360 ms. Ainsi, 95% des terminaux sont enregistrés et activés sur le vNS en un maximum de 380 ms. Cela montre que nos mécanismes n'augmentent pas de manière significative le temps d'activation. Ainsi, ils répondent parfaitement à la contrainte des deux fenêtres d'écoute de longueur de 2s des terminaux de la Classe A.

4 Conclusion

Dans cet article, nous avons étudié la capacité d'itinérance inter-opérateurs fondée sur la résolution DNS et la migration du contexte du terminal. Nos contributions portent sur 3 axes : la recherche des réseaux domestiques des terminaux fondée sur la résolution DNS en utilisant le JoinEUI, la conception et la mise en œuvre d'une extension à l'architecture LoRaWAN pour gérer l'itinérance et offrir une gestion inter-opérateurs et enfin le déploiement d'un banc de test en environnement réel. Les nœuds LoRaWAN mobiles traversant une cellule LoRaWAN partenaire peuvent donc communiquer avec les réseaux domestiques correspondants sans surcharge de mobilité.

Références

[ASN+19] W. Ayoub, A. E. Samhat, F. Nouvel, M. Mroue, and J. Prévotet. Internet of mobile things : Overview of lorawan, dash7, and nb-iot in lpwans standards and supported mobility. *IEEE Communications Surveys Tutorials*, 21(2) :1561–1581, 2019.

[AVT+17] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne. Understanding the limits of lorawan. *IEEE Communications Magazine*, 55(9) :34–40, 2017.

[Ham22] Mohamed Hamnache. mohamedhamnache/lorawan-roaming-doc : Lorawan handover roaming, 2022.

[HKB20] M. Hamnache, R. Kacimi, and A. L. Beylot. L3sfa : Load shifting strategy for spreading factor allocation in lorawan systems. In *45th IEEE Conf. on Local Computer Networks (LCN)*, pages 216–224, 2020.

[HKB21] M. Hamnache, R. Kacimi, and A.-L. Beylot. Unifying lorawan networks by enabling the roaming capability. In *46th IEEE Conf. on Local Computer Networks (LCN)*, pages 371–374, 2021.

[JSN+21] H. Jradi, A. E. Samhat, F. Nouvel, M. Mroue, and J.-C. Prévotet. Overview of the mobility related security challenges in lpwans. *Computer Networks*, 186 :107761, 2021.