Nouvelle approche de communication sécurisée des objets connectés basée sur le protocole MQTT

Abdessamad Mektoubi, Hicham Lalaoui Hassani, Abdelouahed Zakari, Olaf Malassé

NOUVELLE APPROCHE DE COMMUNICATION SÉCURISÉE DES OBJETS CONNECTÉS BASÉE SUR LE PROTOCOLE MQTT

Abdessamad MEKTOUBI[*1], Hicham LALAOUI HASSANI[2]*, Abdelouahed ZAKARI[3#], Olaf MALASSÉ[4]
*Laboratoire RITM, CED Science de l'ingénieur, Ecole Supérieure de Technologie, Université Hassan II Casablanca, Maroc
[#3]LITA Université de Lorraine Metz, France
[4]Arts et Métiers ParisTech Metz, France

Résumé : L'internet des Objets « Internet of Things : IoT» est l'interconnexion des objets du monde physique via le réseau internet en se basant sur les techniques et protocoles de communication existants. L'utilisation accélérée et massive de ce type du réseau dans une multitude de domaine, allons de grands applications industrielles jusqu'aux petites utilisations quotidiennes, soulève des nouvelles problématiques sérieuses relatives au sujet de la sécurité. En effet, des rapports récents sur la cybernétique ont mis en évidence la vulnérabilité des réseaux d'objets [1], les risques ne cessent d'accroitre d'une façon exponentielle surtout avec le déploiement des réseaux intelligents. Ces objets distribués en réseau internet vont créer des millions de nouvelles tentatives d'attaques, ainsi des dangers considérables sur l'intégrité des données. Pour éviter les conséquences désastreuses, la sécurité doit être intégrée afin de vérifier un ensemble de critères, à savoir : la résistance aux attaques, l'authenticité des données, le contrôle d'accès et d'assurer la vie privée des d'utilisateurs [2]. Vu les capacités limitées des objets internet en CPU, mémoire, bonde passante et énergie, une nouvelle vague des protocoles et architectures ont été créés et/ou reconçues pour combler les besoins spécifiques en qualité de service pour ce type de réseau, parmi ces protocoles nous pouvons cité Message Queue Telemetry Transport (MQTT). C'est un protocole open source crée par IBM qui utilise le pattern Publish/Subscribe et se servie d'une petite bande passante. Dans ce travail, nous proposons une nouvelle approche de communication sécurisée basée sur le protocole MQTT. En effet dans un réseau non sécurisé (internet), nous utilisons les techniques et les algorithmes standard de la cryptographie notamment la signature numérique, les fonctions de hachage et l'algorithme RSA, afin de sécurisé la communication et l'échange des données dans un réseau de capteur.

Mots clefs: Sécurité d'information, internet of things, protocol MQTT.

I- Introduction

Depuis sa création, internet s'est développé d'un petit réseau de quelque nœud dans le laboratoire (nom de laboratoire) qui est devenu un réseau à l'échelle mondiale connectant des milliards de nœuds. Aujourd'hui, internet constitue le moyen de communication le plus utilisé pour échanger des informations ainsi que des services. L'histoire de l'internet a connu deux grandes évolution, le web 1.0 qui est l'âge des premiers navigateurs et des pages web statiques, le web 2.0 qui est une révolution des clients riches et récemment le début d'une troisième génération celle de "Internet des objets" ou tout simplement le web 3.0. Cette dernière présente la connexion des objets physique à internet, permettant la lecture des données ou le contrôle des objets à distance qui sont connectés au monde virtuelle à travers des composants électroniques intitulés les "Objets Intelligents".

L'évolution scientifique et industrielle dans le domaine d'électronique a permis la fabrication en masse des objets intelligents à faible coût. Le résultat, est une croissance du nombre d'objets connectés à internet, ainsi la démocratisation de leur utilisation dans une multitude de domaines, sans oublier l'effet sur l'amélioration des services existant (application en temps réel, domotique…). Tout cela a permet de créer des changements socio-économiques, en déclenchant des grands challenges techniques pour faire face aux risques éventuels.

Le nombre des objets connectés passera de 25 milliard en 2015 à 50 milliard en 2020 [3]. Dans cette optique, c'est légitime de poser des questions sur l'infrastructure, les architectures, les technologies utilisées, l'analyse et le stockage des informations massives générées par ces objets, sans oublier la sécurité de ces systèmes notamment la résistance aux attaques, l'authenticité des données, le contrôle d'accès ainsi la sécurisation de la vie privée des d'utilisateurs [2]. La sécurité pour IoT (Internet of Things) est plus critique, vu que les objets intelligents ont des capacités de calcule, de mémoire et d'énergie très limitées. Il représente le maillon le plus

faible dans ces infrastructures, surtout l'accès aux sources d'informations critiques et d'ordre privé [1].

Dans ce travail nous proposons une nouvelle approche d'échange d'information, sécurisée pour des objets internet, en se basant sur le protocole MQTT, les algorithmes de cryptographie RSA et AES. Ce document est organisé de la manière suivante : Apres une introduction générale, nous présentons l'état de l'art afin de tirer les limites des méthodes existantes de sécurité. Nous avons pu situer notre problématique par rapport à l'état de l'art. Nous introduisons également une modélisation de l'approche proposée. Des conclusions et perspectives viennent à la fin de ce document.

II-Etat de l'art

La cryptographie est largement utilisé pour sécuriser les systèmes d'information, en revanche l'utilisation de ces algorithmes dans le contexte d'internet des objets pénalise largement les performances, sachant que les objets intelligents en générale possède des ressources limitées. Pour remédier à ce problème la tendance va vers l'utilisation des algorithmes de cryptographie légère, en effet l'organisation internationale de normalisations a publié l'ISO/IEC 29192-x:2012 qui sont un ensemble de standards pour la cryptographie légère.

Pour le chiffrement par bloc, le standard ISO/IEC 29192-2:2012 propose deux algorithmes alternatifs au AES (Advanced Encryption Standard)[4] , le premier est PRESENT[5] un algorithme de chiffrement par bloc de 64 bits , avec une clé de 80 ou 128 bits, le deuxième est CLEFIA[6] un algorithme de chiffrement par bloc de 128 bits , avec une clé de 80, 128 ou 192 bits.

Pour les algorithmes de chiffrement par flux, le projet eSteame est un projet de l'European Network of Excellence in Cryptology II de 2004 jusqu'à 2008 a pour but la conception des nouveaux algorithmes de chiffrement par flux. Il propose 7 algorithmes dont 3 sont dédiés au système à des ressources limitées, Grain v1 [7], MICKEY 2.0 [8] et Trivium [9].

Pour les fonctions de hachage le SHA-3 issu de la NIST hash function competition qui a élu l'algorithme Keccak [10] en octobre 2012, représente l'alternative de SHA-2 pour les systèmes avec des ressources limitées.

Pour les algorithmes asymétriques, le RSA [11] reste l'algorithme le plus utilisé malgré la taille de ces clés relativement grand, les algorithmes de cryptographie se basent sur les courbes elliptiques représente l'alternative au RSA avec des clés beaucoup moins petite mais qui souffre d'un nombre de brevets sur des aspects de l'algorithme.

MQTT est idéal pour les communications Machine to Machine et les environnements internet of things, il se sert d'une très faible bande passante, le protocole offre trois niveaux de qualité de service, un mécanise de notification en cas de déconnections de client. En Novembre 2014 MQTT v3.1.1 est devenu un standard l'OASIS (Advancement of Structured Information Standards) [12].Une nouvelle variation de protocole qui intitule MQTT - SN (MQTT for Sensor Networks) destiné aux environnements qui n'utilisent pas les protocoles TCP/IP comme Zigbee.

III Problématique

Dans un environnement MQTT, la sécurité de communication entre le broker MQTT et les utilisateurs est assuré par le protocole SSL (Secure Socket Layer), cependant ce protocole n'est pas suffisant pour une sécurité optimale pour la communication selon le protocole MQTT. En effet, nous avons constaté en premier lieu que cette sécurité n'intervient pas au niveau du broker. Un utilisateur quelconque ayant accès au broker pourra accéder à toutes les informations, ainsi que les utilisateurs connectés au broker qui sont en écoute sur un Topic reçoivent toutes les informations sur les autres Topics. L'intervention des algorithmes symétriques comme DES [13] ou AES et les algorithmes asymétriques comme RSA répondent aux problématiques cités ci-dessus moyennant le cryptage/décryptage des messages. Ce mode de sécurité aborde le partage des clés confidentielles entre les utilisateurs. Dans la pratique, un nombre important d'utilisateurs peut nuire à cette sécurité, la diffusion de ces clés pour un nombre aussi important d'utilisateur pourra perdre sa confidentialité.

Dans ce travail, nous proposons une nouvelle approche de communication et de gestion des clés de sécurité en se basant sur le protocole MQTT et l'infrastructure des clés publiques « PKI ». Dans un premier temps, nous avons proposé cette solution à base d'algorithme RSA.

IV- Approche et modélisation proposée :

Le principal objectif de la contribution consiste à sécuriser la distribution d'un flux de messages entre un ensemble d'utilisateurs du protocole MQTT [12].

Pour cette raison, nous avons opté pour la solution d'Autorité de Certification « AC » afin de générer deux types de certificat, la première pour les clients et la deuxième pour les Topics (sujets).

Dans le cas d'un ensemble de clients certifiés par le même AC et qui souhaitent échanger des messages à travers un Topic donné, l'AC d'abord génère une clé privée et un certificat pour cette Topic (figure 1). Ces deux derniers seront publiés pour les clients certifiés, ce qui permettra l'échange des messages en toute sécurité. Lorsqu'un nouveau client souhaite participer au flux d'un Topic donné de manière sécurisé, il doit envoyer une demande de certificat pour pouvoir décrypter les messages publiés.

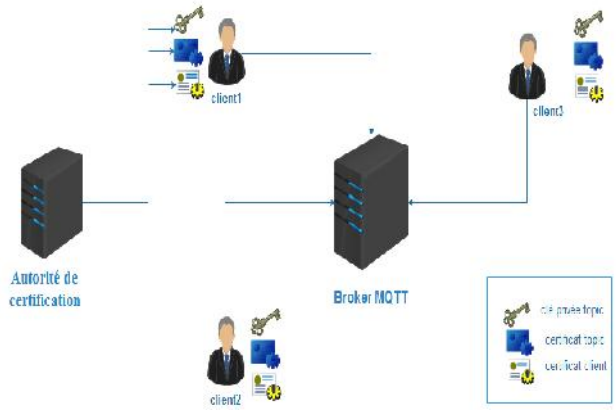

Figure 1 : Architecture de communication MQTT

La demande de certification est publiée par le client demandeur sur un Topic de demandes (Topic réservé aux demandes). Cette demande permettra d'obtenir le certificat du Topic qui contient deux informations, le Topic cible et le Topic de réponse. Le Topic de réponse est une chaine de caractères généré aléatoirement par le client qui est en écoute.

Le traitement des demandes est géré par les clients écoutants sur le Topic demandes. Après la réception de la demande, l'un des clients écoutants sur le Topic vérifie la disposition du certificat demandée. Dans le cas favorable ce client construit une réponse qui contient le Topic cible et son certificat. Par la suite il publie ce message sur le Topic réponse à cette demande (figure 2).

Après la publication et le traitement de la demande, le client demandeur reçoit le certificat du Topic par le biais du Topic réponse. Le client demandeur vérifie également si le certificat est signé par l'AC et pourra par la suite diffuser des messages crypté en utilisant la clé publique du Topic.

La phase de décryptage des messages consiste à ce que l'utilisateur possède la clé privée du Topic. Pour ce faire, il publie une demande de la clé privée sur le Topic des demandes. Cette demande se compose de deux parties, la première est celle du Topic cible et la seconde et celle du Topic réponse ainsi que le certificat du client qui sont crypté par la clé publique du Topic.

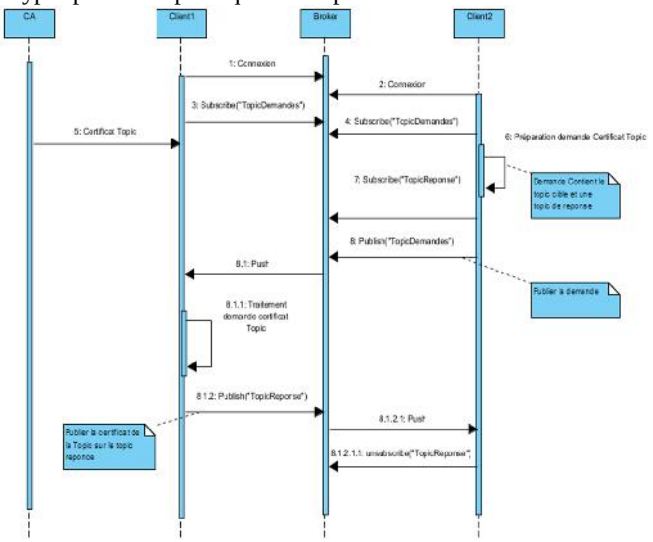

Figure 2 : Diagramme de séquence de demande de certificat de Topic

Les demandes de clés privée sont traitées par les clients écoutant sur le Topic des demandes. Après la réception de la demande de clé privée, l'un des clients écoutant sur le Topic vérifie la disponibilité de la clé privée. Dans le cas favorable ce client peut décrypter la demande par la clé privée de Topic. Une fois cette demande est décrypté, il passe à la vérification du certificat du client demandeur et par la suite il constitue une réponse qui contient le Topic cible et sa clé privée. Ces derniers sont cryptés par la clé publique accompagnée au certificat client demandeur. Finalement, la réponse à la demande de clé privée est publiée sur le Topic réponse jointe à la demande.

Le client demandeur doit décrypter la réponse à sa demande avec sa clé privée. Toute cette démarche permettra à n'importe quel client certifié par l'AC d'envoyer et de recevoir des messages de manière sécurisée sur un Topic donné en utilisant la clé privée et public de Topic.

Pour modéliser cette approche, nous avons unifié la forme des messages échanger qui sont représentés par la classe 'Message'. Cette classe possède deux attribues, le premier attribut représente le type de message et le second attribut représente le détail du message. Ce dernier de type 'Détail Message' est une interface implémentée par les classes modélisant les types des messages échangés, à la fois les messages simples et

cryptés, les demandes et les réponses de certificat ainsi que les demandes et les réponses des clés privées.

L'approche proposée a été réalisée et testée dans un environnement Android. Dans cette optique, nous avons implémenté une solution qui permet d'illustrer, d'une part l'échange de messages sous format texte (cryptés ou non cryptés) via un broker MQTT et d'autre part la diffusion des certificats et des clés privées propres à chaque Topic. La plate-forme développée est constitué principalement de trois parties (voir figure 3). La première composante représente le service MQTT qui est un service Android asynchrone exécuté en arrière-plan. Elle s'occupe de la gestion des communications avec le Broker (Connect, Publish, Subscrib et Disconnect).

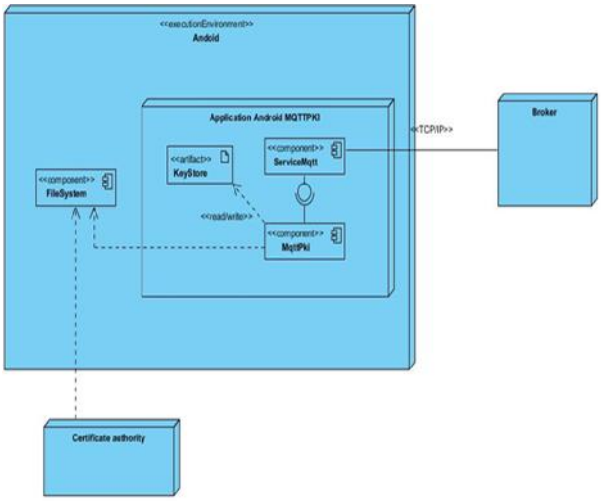

Figure 3 : Architecture de communication MQTT

La seconde partie est un Keystore crypté par un mot de passe, sa fonction principale est le stockage des certificats et des clés privés en assurant une sécurité optimale. La dernière partie est la pièce maitresse de notre application, elle assure plusieurs tâches à savoir, la gestion des interfaces graphiques, les fonctions de cryptage, la lecture/écriture des certificats et clé privée depuis le keystore, la communication avec le service MQTT et finalement la partie de gestion des messages et des demandes.

VI- Conclusions et perspectives :

Dans ce travail, nous avons contribué à l'amélioration de la sécurité de communication et d'échange d'information dans le monde des objets internet (IoT). En se basant sur un protocole non sécurisé. Nous avons proposé une plate-forme d'échange de message entre utilisateurs, pour un Topic donnée, avec un niveau de sécurité acceptable. Malgré l'effort fourni, il reste des points que nous souhaitons traiter dans le futur travail, à savoir la saturation de réseau en cas d'un nombre important d'utilisateurs. Ainsi par rapport aux algorithmes de cryptage utilisés, nous souhaitons faire une comparaison de ces résultats avec des résultats obtenus en utilisant les courbes elliptiques et d'autres algorithmes symétriques légers. La gestion complète de cycle de vie des clefs est un aspect à aborder. Et finalement pour tester la robustesse de notre plate-forme il est nécessaire de simuler quelques attaques et voir la réaction de l'architecture vis-à-vis des intrusions.

Bibliographies:

[1] Carlo Maria Medaglia and Alexandru Serbanati "An Overview of Privacy and Security Issues in the Internet of Things" (1 p 402)

[2] Rolf H. Weber "Internet of Things – New security and privacy challenges" (2010) sciencedirect

[3] Dave Evans "The Internet of Things How the Next Evolution of the Internet Is Changing Everything" http://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

[4] NIST (2001). "Advanced encryption standard" FIPS 197, US Department of Commerce, Washington, DC, November 2001.

[5] A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, and C. Vikkelsoe ,"PRESENT: An Ultra-Lightweight Block Cipher" Springer 2007

[6] The 128-bit Blockcipher CLEFIA: Algorithm Specification, Revision 1.0 (2007), Sony Corporation, http://www.sony.net/Products/cryptography/clefia/download/data/clefia-spec-1.0.pdf

[7] Hell, M., Johansson, T., Meier, W.: Grain: a stream cipher for constrained environments. Int. J. Wirel. Mob. Comput. (IJWMC) 2(1), 86–93 (2007)

[8] Babbage, S.H., Dodd, M.W.: The stream cipher MICKEY 2.0, revised ECRYPT stream cipher submission,http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickeyp3.pdf

[9] Canniere, C.D., Preneel, B.: Trivium A Stream Cipher Construction Inspired by Block Cipher Design Principles (2005), http://www.ecrypt.eu.org/stream/papersdir/2006/021.pdf

[10] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: The Keccak reference

[11] R.L. Rivest, A. Shamir, and L. Adleman "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems" 1977

[12] OASIS Message Queuing Telemetry Transport (MQTT) Version 3.1.1 http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.pdf

[13] Data Encryption Standard: Federal Information Processing Standard (FIPS), Publication 46, National Bureau of Standards, U.S. Department of Commerce, Washington, DC (January 1977)