ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.12

Secure Authentication and Key Distribution Mechanism for CoAP

Akshay D. Patil
ME Student
Department of Computer Engineering
D. Y. Patil College of Engineering, Pune, India

Abstract:
With the help of Internet of things (IoT) every object no matter things or human could be connected to Internet. There are various protocols for wireless communication between devices like IEEE 802.11 Series, 802.15 Series, ZigBee, etc. However, a lot of small devices are unable to communicate efficiently with constrained resources. Internet Engineering Task Force (IETF) has developed a lightweight protocol named as Constrained Application Protocol (CoAP), which provides a request and response communications model between application endpoints. Messages in CoAP are transported over unreliable UDP communications, hence provides a lightweight reliability mechanism. Although security services depend on the authentication and key management system, they do not specify how to exchange keys securely. This paper aims to provide CoAP communication scenario and combined use of cryptographic algorithms ECDSA and ECDH to provide more security for CoAP.

Keywords: IoT, CoAP, Authentication in WSN, Key Management in WSN, Security in IoT.

I. INTRODUCTION

Internet of Things (IoT) intelligently connects all the objects no matter devices, systems or human, with self- configuring capabilities based on standard and interoperable protocols and formats. IoT has been called the Third Wave in information industry following the computer and the Internet. There are hundreds of protocols supported by IoT. Of the many protocols, wireless protocols play an important role in IoT development. IoT needs to integrate various sensors, computer and communication equipment, which are using different communication protocols [1]. Various devices exist as components in vehicles and buildings with constrained resources; it leads to a lot of variation in power computing, communication bandwidth etc. Thus lightweight protocol CoAP considered as a replacement of HTTP for being an IoT application layer protocol. CoAP is now becoming the standard protocol for IoT applications. Security is very important to protect the communication between the devices. There are three main elements when considering security, namely integrity, authentication and confidentiality [2]. The researchers in WSNs have proposed various security mechanisms which are optimized for these networks with resource constraints. Although security services also depend on the key management system, they do not specify how to exchange keys securely [2]. Many security critical applications depend on key management processes to operate and also require a high level of fault tolerance when a node is compromised. Due to various constraints on wire- less sensor networks (WSNs) traditional security mechanisms cannot be applied directly. Moreover, to be robust, the key management protocol should be lightweight and also support both pair wise and group-wise communications. However, the key management in symmetric key based solutions is costly and not scalable, especially when we consider inter domain communication in the IoT [4].

II. REVIEW OF LITERATURE

Xi Chen in his study titled Constrained Application Proto- col for Internet of Things defines CoAP, where it is used, how it works and the message format structure, also some wireless protocol are defined in this paper [1]. Jorge Granjal, Edmundo Monteiro, and Jorge S Silva state technologies enabling end-to-end Internet communications involving sensing devices: like as 6LoWPAN, RPL Routing, CoAP they also has research mechanism addressing open issues as well as research challenges and opportunity for WSNs future work as CoAP implementation in their title Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues [2]. Lavanya, Natarajan studied and compare IKE and IKE2 based on Diffie Hellman (DH) key exchange and RSA and pro- posed use of Elliptic curve analogue of key exchange (ECDH) in their title Lightweight Authentication for CoAP based IoT [3]. Seyit A. CAmtepe and B Ulent Yenerstate different security requirement, vulnerabilities and given different pair wise key distribution schemas and group wise key distribution schemas [4].Preetika Joshi, Manjuverma, Pushpendra R Verma in their title Secure Authentication Approach Using DiffieHellmanKey Exchange Algorithm for WSN explains that secure authentication approach in WSN can be achieved with the help of Diffie Hallman (DH) key exchange algorithm [5]. Ashwini Kumarin his paper titled Analysis of ECDH Key Agreement protocol through linear temporal logic shows math- ematical models for ECDH [6]. Hemant Kumar, Archana Singh in their paper titled Internet of Things: A Comprehensive Analysis and Security Implementation through Elliptic Curve Cryptography stated fundamentals of ECC [7].

III.CONSTRAINED APPLICATION PROTOCOL (COAP):

CoAP is currently uses only UDP communications protocol over 6LoWPAN, although the adoption of transport layer approaches with characteristics more close to protocols such as the Transmission Control Protocol (TCP) is still open to debate, with ongoing research addressing the adaptation of TCP for 6LoWPAN environments. Application-layer communications may enable IoT sensing applications to work with existing web applications without requiring specialized application oriented code or translation mechanisms. CoAP restricts the HTTP dialect to a subset that is well suited to the constraints of 6LoWPAN sensing devices, and may enable abstracted communications between users, applications and such devices, in the context of IoT applications [1].

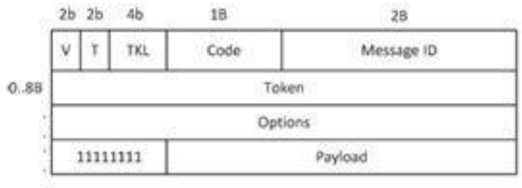

Figure.1. CoAP message format

The CoAP protocol works on the a request and response communications model at application layer between communication endpoints and enables the usage of key concepts of the web, namely the usage of URI addresses to identify the resources available on constrained sensing devices. The protocol may support end-to-end communications at the application-layer between constrained IoT sensing devices and other Internet entities, using only CoAP or in alternative by translating HTTP to CoAP at a reverse or forward gateway. Messages in the CoAP protocol are exchange asynchronously between two endpoints, and used to transport CoAP message requests and responses in communication. As we know CoAP messages are transported over UDP communications protocol which is unreliable, CoAP provides a lightweight reliability mechanism. Using this mechanism CoAP messages may be marked as reliable, for which the sender activates a simple stop-and-wait retransmission protocol mechanism with exponential back off. The receiver must acknowledge a message with a corresponding message sent by sender, if it lacks context to process the message properly, reject it with a Reset message. Acknowledge or Reset messages are related to a Confirm able message by means of a Message ID, along with the address of the corresponding endpoint. CoAP messages may also be transmitted less reliably if marked as Non-reliable, in which case the recipient does not acknowledge the request. Similarly to HTTP, CoAP defines a set of method and response codes available to applications [2].

IV. ALGORITHMS FOR WSN COMMUNCIATION SECUITY

Elliptic Curve Cryptography (ECC) is latest and best known algorithm that can be used for the secure public key distribution in WSN. It works on the concept of elliptic curve, an elliptic curve is a group of finite field and ECC uses this group for its working. The beauty of this technique is the use of elliptic curve and Elliptic Curve Discrete Logarithmic. Problem (ECDLP) is one of the major algorithms of this technique that is when an elliptic curve E and points P Q on E are given find k when Q=k[7]. As stated earlier that RSA uses large number size for its key, while ECC takes very small key size that is why here we focus on ECC.

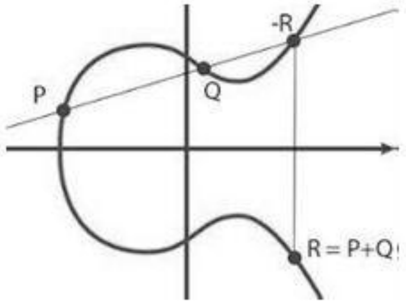

Figure.2. A Simple Elliptic Curve Example

Now if we want greatest level of security then we have to use more efficient method which contains small key size. The newly added cryptographic system will ensure the equal or higher level of security. An elliptic curve E is a curve given by an equation for a cubic or quadratic polynomials f(x).

E: y2 = f(x)

For ensuring that the curve is non-singular, f(x) has no double roots number is act as secret key for both sender and receiver of the message.

A. Elliptic Curve Diffie-Hallman (ECDH)

ECDH [5] is used for distributing public keys securely. Both parties can establish a secret value by sending only the public key of their ephemeral or static key pair to the other party. If one of parties show that key is trusted which is sent by the other party then that key pair may also be used for authentication. This is however not that common. ECDH is the elliptic curve form of the Diffie-Hellman protocol (the DH in the acronym) [8]. ECDH is used for the purposes of key agreement. Suppose two people, Alice and Bob, wish to exchange a secret key with each other [5]. Alice will generate a private key d_A and a public key $Q_A = d_AG$ (where G is the generator for the curve). Similarly Bob has his private key d_B and a public key $Q_B = d_BG$. If Bob sends his public key to Alice then she can calculate $d_AQ_B = d_Ad_BG$. Similarly if Alice sends her public key to Bob, then he can calculate $d_bQ_A = d_Ad_BG$. The shared secret is the x co-ordinate of the calculated point d_Ad_BG. Any eavesdropper would only know Q_A and Q_B, and would be unable to calculate the shared secret [6].

B. Elliptic Curve Digital Signature Algorithm (ECDSA)

ECDSA is the extension of a digital signature algorithm (DSA) by the use of Elliptical Curve Cryptography [9]. It’s the elliptic curve form of DSA. It can be used to authenticate the handshake of the TLS protocol. This authentication involves parameters defied in key agreement that can be used to derive the master secret. The authentication therefore includes the correctness of the session key. As with elliptic-curve cryptography in general, the ECDSA requires a bit size of the public key that is about twice the size of the security level, in bits. For example, at a security level of 80 bits (meaning an attacker requires a maximum of about 280operations to find the private key) the size of an ECDSA public key would be 160 bits, whereas the size of a DSA public key is at least 1024 bits. On the other hand, the size of the signature for both

DSA and ECDSA [10]: is about 4t bits, where t is the security level measured in bits, that is, about 320 bits for a security level of 80 bits.

V. PRAPOSED COMBINED APPROACH FOR COAP

For the secure authentication and key distribution mechanism in CoAP, the combination of ECDSA and ECDH can be made as follows, let's assume private $keyd_{A}integer$ and G be the elliptic curve base point, a generator of the elliptic curve with large prime order n. First, compute the public key
$Q_A = d_AG$. Clearly this is the same as with ECDH where you'd also exchange Q_A with the other party for later key derivation. Second, to sign a message, compute with k chosen uniformly at random and smaller than the group order.
R=kG Calculate e=HASH (m) where HASH is a cryptographic hash function, such as SHA-2.Let z be the Ln leftmost bits of e, where Ln is the bit length of the group order n. select a cryptographically secure random integer k from [1, n-1].
Now (r, s) would be signature pair, cleverly derive s from your private key d_A the ephemeral public key Rx, the ephemeral secret key k and the message (hash) z. Lastly, to verify a message, you first cleverly use Rx, zand the second signature part s to get u1 and u2 and to calculate
R'=u1G+u2Q
Where the first summand is a standard" public key generation" and the second is a standard ECDH exchange.

VI. CONCLUSION AND FUTURE WORK

It is known that CoAP uses UDP for communication and hence it cannot provide reliable communication over the network. ECDH is used for secure distribution of public key over non-secure channel and ECDSA is used for providing authentication between communicating entities. In this study lightweight algorithms ECDH and ECDSA are combined to provide secure authentication and key exchange mechanism for CoAP. In future, algorithm can be optimized and key size can be reduce to enhance the performance of CoAP communication.

VII. ACKNOWLEDGMENT

I would like to thank Dept. Computer of Engineering D Y Patil Collage of Engineering, Pune for support.

VIII. REFERENCES

[1]. Xi Chen, Constrained Application Protocol for Internet of Things, (at) wustl.edu

[2]. Jorge Granjal, Edmundo Monteiro, and Jorge S Silva, Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues. IEEE COMMUNICATION SURVEYS TUTORIALS, VOL. 17, NO. 3, THIRD QUARTER 2015.

[3]. Lavanya, Natarajan, Lightweight Authentication for CoAP based IoT. 6th International Conference on the Internet of Things (IoT16)

[4.] Seyit A. CAmtepe and B UlentYener, Key Distribution Mechanisms for Wireless Sensor Networks: a Survey. TR-05-07, Department of Computer Science, Rensselaer Polytechnic Institute

[5]. Preetika Joshi, Manjuverma, Pushpendra R Verma, Secure Authentication Approach Using Diffie Hellman Key Exchange Algorithm for WSN. 2015 International Conference on Control, Instrumentation, Communication and Computational Techno logies (lCCICCT)

[6]. Ashwini Kumar, Analysis of ECDH Key Agreement protocol through linear temporal logic. Recent Advances in Electrical and Computer Engineering, ISBN: 978-1-61804- 228-6

[7]. Hemant Kumar, Archana Singh, and Internet of Things: A Comprehensive Analysis and Security Implementation through Elliptic Curve Cryptography, International Journal of Current Engineering and Technology. April 2016.

[8]. https://wiki.openssl.org/EllipticCurveDiffieHellm

[9]. https://en.wikipedia.org/EllipticCurveDigitalSignature

[10].https://en.wikipedia.org/EllipticCurveDiffieHellman