A Blockchain-based Authentication Framework For IoT Devices

Andrew Wall
CYBR 535
College of Electrical Engineering and Computer Sciences
Huntington, WV
Wall48@marshall.edu

Carly Rushbrook
CYBR 535
College of Electrical Engineering and Computer Sciences
Huntington, WV
Rushbrook@marshall.edu

Evan Henry
CYBR 535
College of Electrical Engineering and Computer Sciences
Huntington, WV
Henry136@marshall.edu

Abstract*—Internet of Things (IoT) devices are the unconventional computing devices which wirelessly gather and transfer data over a network. Examples of these devices include wearable health monitors, smart home security systems, virtual assistant technology like Amazon Alexa, and many more. With the ever-increasing reliance on the internet, IoT devices have become a requisite in everyday life, but with any network connected gadget there is always a need for security. Traditional security schemes used in standard computing machines are most often ineffective for IoT devices due to the resource intensive processes necessary for these schemes to function. This often leads to IoT devices collecting and transmitting security-sensitive data unprotected, which can have catastrophic effects on the users of these devices if their data falls into the wrong hands. In this project, we propose a blockchain-based authentication framework for IoT devices. Our proposed solution will secure communications with these devices without taxing their limited resources.

***Keywords*—IoT devices, blockchain, authentication**

I. INTRODUCTION

Internet of Things (IoT) devices are the unconventional computing devices which wirelessly gather and transfer data over a network. Examples of these devices include wearable health monitors, smart home security systems, and virtual assistant technology like Amazon Alexa. With the world's ever-increasing reliance on the internet, IoT devices have become a requisite in everyday life, but with any network connected gadget there is always a need for security. Traditional security schemes used in standard computing machines are most often ineffective for IoT devices due to the resource intensive processes necessary for these schemes to function. This often leads to IoT devices collecting and transmitting security-sensitive data unprotected, which can have catastrophic effects on the users of these devices if their data falls into the wrong hands. We aim to create a secure framework for IoT devices using blockchain that will ensure safe communications without taxing the IoT devices' limited resources. We will do this in three steps: 1. Creating and implementing an authentication scheme. 2. Performing tests on: runtime, CPU time, and energy consumption 3. Confirming that our scheme is secure by using the security verification tool AVISPA. .

II. BACKGROUND AND MOTIVATION

A. Background

Internet of things (IoT) devices are the non standard computing devices that collect and transfer data over a network. These devices have limited resources compared to conventional computing devices, which can leave something like security that is not necessary for operation, left unmanaged.

Blockchain is a shared immutable record. The individual blocks of a blockchain contain data that is strung together by hash identifiers that link back to the previous block. When a block of data is altered the hash identifier will also change. When a hash changes, the blocks that follow no longer chain together, making malicious activity easily detectable.

Public key encryption, also known as asymmetric encryption, uses a pair of keys referred to as a public key and a private key that are used for encryption and decryption. Anything encrypted by the public key can only be decrypted by the private key and vice versa.

Exclusive OR (XOR) is a common component in cryptographic ciphers that uses modulus addition and a key to apply an encryption on a string of text.

B. Motivation

Blockchain technology is very popular in today's fast moving world. When deciding on what we should do for our project we wanted to do something we found interesting. We have all been interested in blockchain technology and wanted to dive deeper. So we built our project around using blockchain technology.

III. OUR PROPOSED SCHEME

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

A. Notation Table

Pub()	Function to encrypt using a public key
TID	A device's true identity
TID_x	The true identity XOR'd with the nonce
XOR	Exclusive OR operation
Priv_x	A device's private key XOR'd with the nonce
RM	Response message
DS	Digital signature

B. Initialization Phase

The Initialization Phase is divided into four steps and is used to initialize the blockchain and devices on the network and establish the public and private keys for each device. Below is a brief summary of the steps taken in the Initialization Phase and the communication between the Server and a device on the network.

1) Server:
   Generates Server's Public and Private Keys
   Creates blockchain
   Publishes Sever's Public Key to the blockchain
2) Device:
   Generate nonce
   Generate timestamp
   Send initialization message to server, IM = Pub(nonce, timestamp, hash(nonce,timestamp))
3) Server:
   Receive IM message from Device
   Decrypt with Private Key
   Verify timestamp
   Verify hash(nonce, timestamp)
   Generate device's true id (TID)
   Generate device's Public and Private Key
   Perform Exclusive OR operation on TID using nonce, TID_x = XOR(TID, nonce)
   Perform Exclusive OR operation on device's Private Key using nonce, Priv_x = XOR(Private Key, nonce)
   create response message, RM = (TID_x, Priv_x, timestamp, hash(TID_x, Priv_x, timestamp))
   Generate Digital Signature (DS) using Server's Private Key, DS(TID_x, Priv_x, timestamp, hash(TID_x, Priv_x, timestamp))
   Send RM and DS to device
4) Device:
   Receive RM and DS from Server
   Verify DS using Server's Public Key
   Verify Timestamp
   Verify hash(TID_x, Priv_x, timestamp)
   Retrieve TID from TID_x using XOR operation
   Retrieve Private Key from Priv_x

C. Authentication Phase

The process of authentication is broken up into twelve steps. The first step starts off the process by having device one obtain device two's public key from the blockchain. Next, device one encrypts its own blockchain ID with device two's public key. After that has been completed, device one sends its encrypted blockchain ID to device two. Once the blockchain ID's have been exchanged, device two then kicks off the next major part of the process. Starting with generating a nonce (a randomly generated number), a timestamp, and a hash value of the data. Device two also encrypts the data it generated with device one's public key that was retrieved earlier in the process. After this step, device 2 sends this encrypted data over to device one. Device one then decrypts the information sent by device two with its own private key. After this has been completed, device one digitally signs the information sent over by device two with its own private key. This is to ensure that anyone who decrypts with the public key knows the message came from who they say it came from. After this step, device one

creates an encrypted message containing the digital signature that was generated earlier, its own blockchain ID, and the data that was sent over. This data contains the nonce, timestamp, and a hash. This data is sent to device two where the message is decrypted using its private key. The final step in the process ends with device two checking the digital signature. If the digital signature matches, the authentication process has been completed.

IV. EXPERIMENTATION AND SECURITY VERIFICATION

To test our scheme we will perform experimentation and perform security verification to prove it's efficiency and security. For the experiments, we will record the performance metrics for our scheme. Specifically, we will record execution time, CPU time, and energy consumption. We will perform security verification using AVISPA. AVISPA stands for Automated Validation of Internet Security Protocols and Applications. AVISPA is an automated, push-button, security verification tool.

References

[1] R. Nagpal, "Blockchain-based Authentication of devices and people," *Medium*, 17-Dec-2018. [Online]. Available: https://medium.com/blockchain-blog/blockchain-based-authentication-of-devices-and-people-c7efcfcf0b32. [Accessed: 13-Feb-2022].

[2] "What is asymmetric encryption?," Cloudflare. [Online]. Available: https://www.cloudflare.com/learning/ssl/what-is-asymmetric-encryption/. [Accessed: 13-Feb-2022].

[3] Wikipedia Contributors, "XOR cipher," Wikipedia, 12-Dec-2021.[Online]. Available: https://en.wikipedia.org/wiki/XOR_cipher. [Accessed: 13-Feb-2022]