Delft University of Technology, Bachelor Seminar of Computer Science and Engineering

A Comparative Study on Lightweight Authentication Protocols for IoT

Thomas Eckhardt, Daily Supervisor: Miray Ayşen, Responsible Professor: Zekeriya Erkin

Cyber Security Group
Department of Intelligent Systems
Delft University of Technology

Abstract

Wireless Sensor Networks(WSN's) are networks of sensor that wirelessly communicate to each other. The communication of these sensors needs to be secured to prevent leaking of potentially sensitive information of the data sent between the user, gateway and sensors. For WSN's lightweight authentication, protocols have been developed in order to provide lightweight authentication for resource constrained devices. This paper performs a comparative study of authentication protocols for WSN's. This is done by comparing the performance and examining the attack types to which a protocol is vulnerable. From this comparison, a possible improvement for newer authentication protocols is proposed.

1 Introduction

The Internet of Things (IoT) is a life-improving field in which objects get connected to the internet (Yang, Zhang, Liu, & Zhang, 2020). Use cases come in different types of environments. Lee, Yoo, & Kim, 2016 illustrate an energy management framework that can be used to monitor and control the energy consumption of a factory. Something similar is the Smart Health Sensing System (SHSS), but instead of monitoring a factory, this monitors and controls a patients' health (Kumar, Tiwari, & Zymbler, 2019). In both examples, authentication protocols play a vital role in securing the system. When there is no proper authentication, an intruder would be able to take control of the factory. When monitoring and controlling the health of a person, improper authentication could lead to life-threatening situations. These reasons display the importance of lightweight authentication protocols.

In this paper, a comparative study on the existing authentication protocols for Wireless Sensor Networks (WSN) is made. This is done by performing a performance and security analysis. In the performance analysis, the computational cost of a protocol is examined. This is done by identifying where the most computational load of the protocol is located. Having a high load on the user side would be less of a problem than a high load on the sensor node. This is because the user side is not resource constraint computationally. In the security analysis, the strengths and weaknesses of the protocols are examined. By checking to which attack types a protocol is resistant, this analysis is made. This comparison will help in identifying where the current authentication protocols are lacking and what could be improved upon.

The 5 protocols that will be compared are the ones from:

- **Wong, Yuan, Jiannong, & Shengwei, 2006**
- **Vaidya, Makrakis, & Mouftah, 2010**
- **Liu & Chung, 2017**
- **Gope & Hwang, 2016**
- **Jiang et al., 2017**

Some of these protocols are more recent, such as the ones from Liu & Chung, 2017, Gope & Hwang, 2016 and Jiang et al., 2017. The ones from Wong et al., 2006 and Vaidya et al., 2010 were one of the first protocols to be proposed. Hence, they had an influence on the development of other protocols.

In section 2 It is explained what WSN are. In section 3 the methodology of this research will be explained and in section 4 related studies will be shown. The chosen protocols will be closer examined in 5 and analyzed for performance and security in section 6 and 7. The results of these analyses will be discussed in section 8. Conclusions and possible future improvements will be discussed in section 10

2 Background

A subset of the IoT is a Wireless Sensor Network (WSN). A WSN is a network of sensors that wirelessly communicate to each other. These networks are comprised of sensor nodes that have multiple functions. They can collect, process and communicate collected data back to the user (Sánchez-Álvarez, Linaje, & Rodríguez-Pérez, 2018). WSN's are becoming more common in fields such as vehicular pollution level (Ullo & Sinha, 2020), wildlife (Vaidya et al., 2010) and healthcare monitoring (Gope & Hwang, 2016). The market for WSN's is set to grow from 3,282.2 million in 2018 to 8,669.8 million in 2025 (Grandviewresearch, 2018). Wireless Sensor Networks are made up of 3 parts:

- **User:** The user of the system. This is the person that can access the data of the sensor nodes.
- **Gateway Node (GW Node):** Can register new users and sensor nodes to the WSN. In some papers, this would be referred to as the Registration Center (RC).

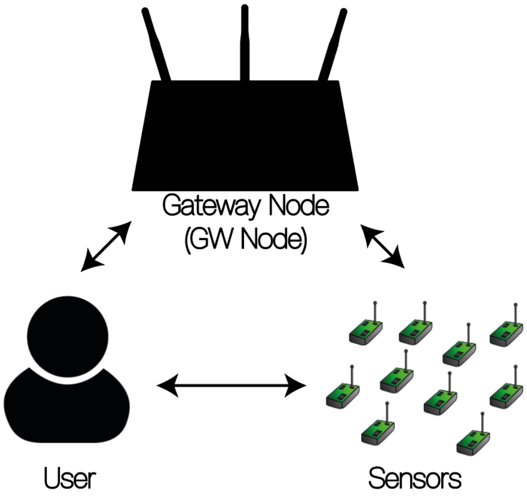

Figure 1: Communication of a wireless sensor network

- **Sensor Nodes:** These are the nodes that send the actual information to the user.

A graphical illustration of the communication between the nodes can be seen in figure 1. The communication takes place via low power wireless media such as LoRaWAN (Augustin, Yi, Clausen, & Townsley, 2016), ZigBee and Bluetooth to the user. Which protocol to choose can be dependent on number of nodes, data transmission rates and intended range of the application (Tsantilas et al., 2020).

The sensor nodes in WSN's can have several constraints. Seo et al., 2016 proposed a sensor node the size of a grain of dust. This sensor is to be used as an implant and can be powered, and communicated with, via ultrasound. The size puts a limit on its computational capabilities. Although this device is wirelessly powered, most devices in WSN's are powered via batteries (Turkanovic & Holbl, 2013). Those devices would benefit in consuming less energy since that would benefit the battery life of the device. According to OWASP the top 3 vulnerabilities of IoT devices are hardcoded passwords, lack of integrity/authenticity and a lack of authentication/authorization (OWASP, 2018). Hence, it is important that the data send over these networks is properly secured and that lightweight authentication protocols exist.

There are three phases in an authentication protocol:

- **Registration phase:** A user must apply for login credentials at the Gateway Node(GW Node). This can be in the form of a password and/or password.
- **Login phase:** The user shows their login credentials for authentication.
- **Authentication phase (Verification phase):** The credentials are checked to establish a secure connection.

Showing and checking the credentials happens in the same phase in most protocols. This would effectively combine the login and authentication phase, which would then be referred to as the authentication phase. Some protocols choose to divide this authentication in a login and verification phase (Vaidya et al., 2010). Other protocols might add phases to their protocols (e.g. password renewal or node addition phases).

3 Methodology

The goal of this research is to compare the existing authentication protocols for WSN's. The list of chosen protocols are not necessarily the most state of the art authentication protocols, as well as protocols that greatly influenced the other protocols. The protocols that will be compared are the ones from Wong et al., 2006, Vaidya et al., 2010, Liu & Chung, 2017, Gope & Hwang, 2016 and Jiang et al., 2017.

This is done through an extensive literature survey. In search for authentication protocols that needed comparing articles were searched by using Google Scholar and DBLP. Connected papers supported in finding articles that might be related to that article, such as a cryptanalysis of a protocol. Mendeley was used in order to manage and create the references.

4 Related work

Yang et al. made a survey on authentication protocols for Machine to Machine communications (M2M), Internet of Vehicles (IoV), Internet of Energy (IoE) and Internet of Sensors (IoS). These four are all subsets of the Internet of Things (IoT). The IoS is what in this paper would be referred to as WSN's. This paper names the attacks to which Liu and Chung, Gope and Hwang and Jiang et al. are vulnerable. It however excludes the older protocols by Wong et al. and Vaidya et al.. It also does not perform a performance analysis on any of these protocols.

Rajeswari and Seenivasagam provide a comparative study on authentication protocols for WSN's. This study mentions the papers by Wong et al. but in turn, does not include the other protocols that are compared in this paper. A performance analysis is only included in text format and it does not provide a detailed comparison of the protocols. The different attack types are explained, but the vulnerabilities of each protocol are not examined.

The papers of Gope & Hwang, 2016, Vaidya et al., 2010 and Liu & Chung, 2017 provide a performance comparison compared to other protocols. These comparisons however only includes the time of hash operation and excludes the time of concatenation, xor and message operations. The paper of Jiang et al. only compares itself to other protocols utilising ECC.

5 Protocols

In the following sections, each individual authentication protocol will be closer examined. The 5 chosen protocols can be seen in table 1 together with their type of security.

A smart card is a physical card that gets handed to the user upon registration. It is able to perform the computations needed for authentication of the user. In the examination a number of notations are used, these notions are explained in table 2.

Table 1: The security types of the different protocols

Name	Security type
Wong et al., 2006	Password
Vaidya et al., 2010	Smart-card and password
Liu & Chung, 2017	Smart-card and password
Gope & Hwang, 2016	Smart-card and password
Jiang et al., 2017	Smart-card and Elliptic Curve Cryptography (ECC)

Table 2: Notion used in the examination of the protocols

Name	Explanation
U	A user
GWN	Gateway node
DID	Dynamic identity
h	A one-way hash function
SID	The identity of the sensor
$SHID$	The shadow identity of a user
ID	The identity of the user
ID_G	The identity of the GWN
PW	The password of the user
K	Master key
SK	Session key
T or TS	Timestamp
TC	Temporal credential
Ts_{ug}	Temporal credential
r	Random number
$\|\|$	Bitwise concatenation
$\oplus$	Bitwise xor operation

5.1 Wong et al., 2006

This paper by proposes an authentication protocol intended for WSN's. It chose a private key instead of public key authentication protocols. This is because of the high computational load and the dependence on a Trusted Third Party (TTP). A TTP is a device that facilitates communication between sensor nodes and the user. An example of a public key authentication protocol is the one by Benenson & Gedicke, 2005. They dispute that the use of ECC in WSN's is a feasible solution. Wong et al. reckon that the sensor nodes could form the bottleneck of the system when there is a lot of traffic. This paper is the oldest authentication protocol that is being compared. This can also be seen in its features, as it only uses a hash and password-based protocol, and thus no smart cards are used for this protocol. The protocol has the following phases:

1. Registration phase
2. Login phase
3. Authentication phase

In the registration phase, a user should submit its ID and chosen password. The gateway then in turn computes the hashes: $A = h(userID||key)$ and $B = h(A||h(PW))$. When these hashes are computed, the GW node sends a reply to the user that the registration was successful. The GW passes the data $(userID, PW, A, B, TS)$ to be stored on the GW-PC's database in plain text. The data $(userID, A, TS)$ is sent to the sensor login nodes, which allows the user to log in. The timestamp is used to check if a record has not expired yet.

In the login phase, a user can request data generated by the sensors. When the user successfully send its login credentials (userID and password) to one of the sensor login nodes, that node can now verify if the right credentials are sent. The node computes $B^* = h(A||h(PW^*))$, $C2 = (B^* \oplus A)$ and $C1 = h(T \oplus B^*)$ and then sends $(userID^*, C2, C1, T)$ to the GWN to be used for authentication.

The GWN now also computes $C1^* = h(B \oplus T)$ and $C2^* = (B \oplus A)$. These are then checked against the data sent by the login node. If they are equal, the GW will send a message to the user that the login was successful. The handshake for the protocol has now finished.

5.2 Vaidya et al., 2010

This paper by proposes a two-factor authentication protocol. Unlike Wong et al., this protocol utilizes a password and smart-card based authentication. This smart-card is issued upon registration. Together with the password, the user can now log in into the sensor and GW node. The protocol has the following phases:

1. Registration phase
2. Authentication phase
   (a) Login phase
   (b) Verification phase

In the registration phase, the user sends its identity (ID_i) and password (PW_i) to the GWN. The GWN now computes the hash $Ni = h(IDi||PWi) \oplus h(K)$ in which K is only known to the GWN. The GWN in turn makes a smart card with parameters $h(\cdot), ID_i, N_i, h(PW_i)$ and x_a. This smart card now gets sent to the user to be used for authentication.

When the user wants to access data from the sensor nodes, it should authenticate itself. This can be done by inserting the smart-card and entering the login credentials ID_i and PW_i. If the login credentials are correct, the following hashes are computed: $DID_i = h(ID_i||PW_i) \oplus h(x_a||T)$ and $C_i = h(Ni||x_a||T)$. DID stands the dynamic login identity of the user U. The smart card then sends $< DID_i, C_i, T >$ to the GWN to be used for verification.

The GWN computes the hashes: $h(ID_i||PW_i) = DID_i \oplus h(x_a||T)$ and $C_i^* i = h((h(ID_i||PW_i) \oplus h(K))||x_a||T)$. Whenever C_i^* equals C_i the login request is accepted. The GWN now sends $< DID_i, A_i, T' >$ in which $A_i = h(DID_i||S_n||x_a||T')$, with secret parameter x_a, to one of the dedicated sensor login-nodes (S_n). The sensor node now computes A_i in the same way and checks whether they are equal. If this is the case the sensor node will now respond to the request of the user

5.3 Liu & Chung, 2017

This paper by proposes an identity-based authentication protocol intended for wireless healthcare sensor networks, which utilises the id of the user to compute a public key. It was developed to monitor medical information about a patient in a secure way. The protocol makes use of the Diffie-Hellman Problem for bilinear pairing. A smart card and password-based security ensures that the data is only accessible to the medical personnel. The protocol is divided into a setup, registration, login and verification phase.

1. Setup phase
2. Registration phase
3. Login phase
4. Verification phase

In the setup phase the GWN does 3 things. It sets a bilinear map e: $G_1 \times G_1 \rightarrow G_2$ and $P_0 \in G_1$. Generates two one way hash function $H_1 : \{0, 1\} \rightarrow G_2$ and $H_2 : G_2 \rightarrow \{0, 1\}$. The GW also selects a random number $S_0 \in Z_q^*$ and computes public parameter $P_{pub} = S_0 * P_0$.

In the registration phase, the user needs to register with ID_i and PW_i to the GWN. The GWN now calculates $U_{priv} = S_0 * U_{pub}$ and uses it along with h, ID_i, PW_i, a to create a personalized smart-card. The parameter a is a private parameter, not known to the user, that is stored on the smart card. The smart card now gets sent to the user to be used for authentication.

In the login phase the user uses its smart-card, ID and PW to make a login request. The smart-card computes $Sig = r * U_{priv}$ with $r = h(ID||PW||a)$ and sends Sig, r, T_L, I to the GWN. The GWN is now able to verify the user by first checking if correct login credentials are send and $\hat{e}(P_0, Sig)$ should equal $\hat{e}(P_{pub}, r * U_{pub})$. The GWN now sends T_u, b, ID to all sensor nodes in order to notify them that the user is legal.

5.4 Gope & Hwang, 2016

This paper by proposes a smart-card based authentication protocol. The protocol is made of 4 phases:

1. Registration phase
2. Anonymous authentication and key exchange phase
3. Password renewal phase
4. Dynamic node addition phase

In the registration phase the user needs to submit their identity ID_u to the GWN. Note that in this protocol the password is not sent to the GWN. The GWN will now compute $K_ug = h(ID_u||n_g) \oplus ID_G$, create shadow-IDs $SHID = \{shid_1, shid_2, ...\}$ with corresponding emergency keys $K_{em} = \{k_{em1}, k_{em2}, ...\}$. The GWN now continues to calculate $sid_j = h(ID_U||r_j||K_{ug})$ and $k_{em_j} = h(ID_U||sid_j||r_j)$. As a security feature against replay attacks, a transaction number Ts_{ug} is stored for communication between the user and GWN. Whenever a message is sent this number gets incremented. The GWN is now able to personalize the smart card with$\{K_{ug}, (SHID, K_{em}, Ts_{ug}, h(\cdot))\}$ and sends it to the user. The user may now choose a password PSW_U and replace $K_{ug}, SHID, K_{em}$ with $K_{ug}^* = K_{ug} \oplus h(h(ID_U) \oplus h(PSW_U))$, $SHID^* = SHID \oplus h(h(ID_U) \oplus h(PSW_U))$ and $K_{em}^* \oplus h(h(ID_U) \oplus h(PSW_U))$. The user adds the computation $f_U^* = h(h(K_{ug}) \oplus h(PSW_U) \oplus h(ID_U))$ to change the smart card to contain $\{K_{ug}^*, f_U^*, (SHID^*, K_{em}^*), Ts_{ug}, h(\cdot)\}$.

The authentication phase for this protocol is pictured in figure 2 which is located in Appendix A. A graphical explanation of this phase is more appropriate because of the amount of computation that needs to be done. Note that in the explanation above SID is replaced with $SHID$, this would otherwise be confusing compared to the other protocols that use SID as the identity of the sensor. A difference with other protocols is that the sequence numbers now need to be verified.

5.5 Jiang et al., 2017

This paper proposes a two-factor authentication protocol based on smart-cards, passwords and Elliptic Curve Cryptography (ECC). The protocol has the following phases:

1. Registration phase
2. Login phase
3. Authentication phase
4. Password change phase

Before the registration phase, the GWN uses ECC in order to compute the private key (x) and public ($y = xP$) key pair. A user may now register with a identity ID_i, password PW_i and random number r_i. The user then calculates the hash $HPW_i = h(PW_i||ID_i||r_i)$ which can then be sent along with ID_i to the GWN. The GWN will now calculate $TC_i = h(K_{GWN-U}||ID_i||TE_i)$ and $PTC_i = TC_i \oplus HPW_i$. TC_i is now stored along with ID_i in the verification table. The GWN will now use $\{h(\cdot), y, TE_i, PTC_i\}$ to personalize a smart-card to send back to the user. In turn the user will compute $HPW_I'{=}h(h(ID_i||PW_i||r_i) \bmod m)$ with m $2^8 \leq m \leq 2^{16}$. The user now store r_i and HPW_i onto the smart-card.

In the login phase, the user should enter the smart-card into a terminal and enters their password. The smart-card will now compute $HPW_i^* = h(h(ID_i||PW_i||r_i) \bmod m)$ and compare it to HPW_i' that is on the smart-card. When these values are equal, the smart-card will compute $TC_i = PTC_i \oplus h(PW_i||ID_i||r_i)$ to be used for authentication.

In the authentication phase, the user computes $A_i = aP, D_i = ay = axP, DID_i = ID_i \oplus h(A_i||D_i)$ and $C_i = h(ID_i||TS_1||D_i||A_i||TC_i)$ which can then be sent to the GWN. The GWN performs the computations $D_i = xA = xaP, ID_i = DID_i \oplus h(A_i||D_i)$, and $TC_i = h(K_{GWN-U}||ID_i||TE_i)$. It also calculates C_i^* and checks whether it is equal to C_i. When this is the case, the GWN will continue performing computations $TC_j = h(K_{GWN-S}||SID_j), DID_{GWN} = ID_i \oplus h(DID_i||TC_j||TS_2)$, and $C_{GWN} = H(ID_i||TC_j||A_i||TS_2)$. The data $\{TS_2, DID_i, DID_{GWN}, C_{GWN}, A_i\}$ is now sent to the sensor node. The sensor now will now check for equality on $h(ID_i||TC_j||A_i||TS_2)$

Table 3: Performance of the registration phase of various authentication protocols

Name	**Registration**		
	User	**GW node**	**Sensor node**
Wong et al., 2006	T_{mes}	$3T_h + 2T_{\|\|} + T_{mes}$	-
Vaidya et al., 2010	T_{mes}	$3T_h + T_{\oplus} + T_{\|\|}$	-
Liu & Chung, 2017	T_{mes}	$T_{pu} + T_{pr}$	-
Gope & Hwang, 2016	$6T_h + 3T_{\oplus} + T_{mes}$	$5T_h + 3T_{\oplus} + 8T_{\|\|} + T_{mes}$	-
Jiang et al., 2017	$2T_h + 2T_{\|\|} + T_{mod} + T_{mes}$	$T_h + T_{\oplus} + 2T_{\|\|} + T_{mes}$	-

Table 4: Performance of the authentication phase of various authentication protocols

Name	**Authentication**		
	User	**GW node**	**Sensor node**
Wong et al., 2006	T_{mes}	$T_h + 2T_{\oplus} + T_{mes}$	$3T_h + 2T_{\oplus} + T_{\|\|} + 2T_{mes}$
Vaidya et al., 2010	$3T_h + T_{\oplus} + 4T_{\oplus} + T_{mes}$	$4T_h + T_{\oplus} + 8T_{\|\|} + T_{mes}$	$T_h + 3T_{\|\|} + T_{mes}$
Liu & Chung, 2017	$3T_{\|\|} + T_h + T_{\oplus} + T_{mes}$	$T_h + T_{\oplus} + T_{mes}$	$T_{\|\|} + 2T_h + 3T_{\oplus} + T_{mes}$
Gope & Hwang, 2016	$10T_h + 8T_{\oplus} + 15T_{\|\|} + T_{mes}$	$7T_h + 5T_{\oplus} + 11T_{\|\|} + 2T_{mes}$	$3T_h + T_{\oplus} + 4T_{\|\|} + T_{mes}$
Jiang et al., 2017	$5T_h + 2T_{\oplus} + 9T_{\|\|} + T_{mes}$	$8T_h + 2T_{\oplus} + 21T_{\|\|} + 2T_{mes}$	$6T_h + 4T_{\oplus} + 9T_{\|\|} + T_{mes}$

with $ID_i = DID_{GWN} \oplus h(DID_i||TC_j||TS_2)$. If they match, the sensor node can now generate a random key $b \in Z^*_{p-1}$ and perform the computations $B_j = bP, SK_{ij} = H(bA_i) = H(abP), andC_j = h(TC_j||ID_i||SID_j||B_j||TS_3)$. The data $\{SID_j, TS_3, C_j, B_j\}$ is now sent back to the GWN, which can be used to check whether $h(TC_j||ID_i||SID_j||B_j||TS_3)$ is equal to Cj. If this is the case, the GWN computes $E_{GWN} = h(ID_i||TC_i||D_i||B_j||TS_4)$, and sends $SID_j, TS_4, B_j, E_{GWN}$ to the user. The user now computes $H(ID_i||TC_i||D_i||B_j||TS_4)$ and matches it against E_{GWN}. The session is established if this the case The user may now compute the shared session key $SK_{ij} = h(aB_j) = h(abP)$.

6 Performance analysis

A key aspect of WSN's are the computational constraints of the sensor nodes. The goal of a protocol should be to distribute the load away from the sensor nodes, since those are the most resource constrained , and guide it more towards the GWN and user (Sánchez-Álvarez et al., 2018).

The performance of the chosen protocols can be seen in table 3 and 4. Table 3 is for the performance of the registration phase and 4 is for the authentication phase. This performance is set forth per user, GW and sensor nodes. For making a proper comparison, the authentication and login phase are combined. This is the case for the protocols of: Wong et al. and Liu and Chung. The performance is expressed as a combination of time notions:

- T_h: Execution time for a one-way hash operation
- $T_{\oplus}$: Execution time for a xor operation
- $T_{||}$: Execution time for a concatenation operation
- T_{mes}: Execution time for sending a message

The results in the tables originate from the descriptions of the authentication protocols in the papers. This data is generated by analyzing the protocols proposed in the papers. The amount of all the different time notions have been individually counted per node of the protocol.

7 Security Analysis

Performance is not the only aspect of a protocol that is of importance. An authentication protocol should also be resistant to various attack types. An intruder could use these vulnerabilities in order to attack a WSN. In table 5 the results of this analysis are displayed. An explanation of the various attack types:

- **Replay Attack:** An attack in which an intruder replays a message send by the user or sensor node.
- **Impersonation Attack:** An attack in which an intruder is able to impersonate himself as a legitimate user.
- **Stolen-Verifier Attack:** An attack in which the verification credentials get stolen by an intruder

- **Stolen Smart Card Attack:** An attack in which the smart-card, created by the GWN, is intercepted by an intruder. This is related to the Stolen-Verifier attack.
- **Guessing Attack:** An attack in which an intruder tries to guess the password of a user.
- **Denial of Service Attack:** An attack in which requests are continuously sent to the Gateway and Sensor nodes, with the intention to disrupt the service of an application.
- **Node Compromise Attack:** An attack in which an intruder gains physical access to a sensor node. An intruder could now be able to extract or change data sent by the sensor node. In WSN's this is a common issue since sensor nodes can be deployed over a wide area and will probably be unattended.
- **Eavesdropping Attack:** An attack in which unsecured data, that is being sent over a wireless medium is intercepted by an intruder.
- **Forgery Attack:** An attack in which the intruder would make the user unwillingly perform actions. It could be the case that the hacker changes the user's password.
- **SID Modification Attack:** An attack in which an intruder tries to change the ID of a sensor node.

Table 5: Vulnerabilities of the authentication protocols

		Authentication protocols				
		Wong et al., 2006	Vaidya et al., 2010	Liu & Chung, 2017	Gope & Hwang, 2016	Jiang et al., 2017
Attack types	Replay Attack	✘	✓	✓		
	Impersonation Attack		✓	✓	✓	✓
	Stolen-Verifier Attack	✘	✓	✓		
	Guessing Attack		✓	✓		
	Denial of Service Attack		✓	✓		✘
	Node Compromise Attack		✓		✓	
	Eavesdropping Attack					
	Stolen Smart Card Attack				✓	✓
	Tracking attack					✓
	Forgery Attack	✘				
	SID Modification Attack					✘

The papers of Vaidya et al., 2010 and Gope & Hwang, 2016 found that the protocol proposed by Wong et al., 2006 is not resistant against Replay, Stolen-Verifier and Forgery attacks. The Stolen-Verifier attack is caused by the lookup table of credentials. These credentials could be stolen and then be used to log in. Another flaw is the ability to log in multiple times with the same user id.

The paper of Choi, 2018 has shown that the protocol by Jiang et al. has some serious security vulnerabilities. The first is the lack of mutual authentication. The sensors can verify the user with the help of the gateway, but the users are not able to verify a sensor node. The second vulnerability is the risk for a SID modification attack. An attacker sends a different sensor node identity to the user. Since a user is not able to authenticate the sensor nodes, the user will believe it is a sensor. The third vulnerability is that the identity of the sensor node is not anonymous, and in turn, an attacker can find out to which sensor nodes a user is communicating. The sensor nodes are also susceptible to DoS attacks. This could result in draining the battery of the sensor nodes. The last vulnerability is the fact that based on the data in the sensor node, a hacker is able to identify the user id ID_i.

8 Discussion

From the performance analysis, several differences between the authentication come to light. The first being the division of the computational load over the different nodes. In actual use cases of these protocols the User and GW nodes do not lack the necessary resources (Jiang et al., 2017). Leveraging computations over from sensor nodes to gateway or user could benefit the scalability of the system, and perhaps help shrink the size of the sensor nodes. These resources come in the form of maximum CPU performance and battery life. This means that a higher load on these nodes will not result in any system bottlenecks. The sensor nodes, on the other hand, could suffer under high load situations. Having less stress on the sensor node could in turn lead to a longer-lasting battery (Sánchez-Álvarez et al., 2018).

From the security analysis can be seen that not all authentication protocols can ensure a perfectly secure connection. The paper by (Wong et al., 2006) is shown to be vulnerable to multiple attack types such as replay, stolen-verifier and forgery attacks. But in turn, what should be said for this protocol was that it was one of the first protocols to be proposed. A lot of authentication protocols based their protocols on the flaws of this protocol. The paper by (Jiang et al., 2016) is also not fully secure, being susceptible to a DoS attack could have a serious impact on the performance of the protocol.

A fully functional authentication protocol would be both low on computational needs and high on security features. Whenever a protocol is

The biggest improvement to the existing authentication protocols would be a protocol that puts less strain on the sensor nodes. This could aid the rapid development of smaller nodes. The protocols by Vaidya et al., 2010 and Liu & Chung, 2017 show that they have quite low computational demands on the sensor node, while still preserving resistant to most attack types. A way for this to be done is by combining the efficiency of Liu & Chung, 2017 with the extra security measures of Gope & Hwang, 2016. The use of sequence number for resistance against replay attacks would benefit the resistance of the authentication protocol.

9 Responsible Research

This research is done by an extensive literature survey. Anyone that has access to the literature that is in the reference section can replicate this research. The data that is in the results section is produced by analyzing the individual authentication protocols. By counting all the time notions one can come to the same results. Since I have not been a part of the development in any of the schemes there is not any bias in the results.

10 Conclusions and Future Work

The main difference between the authentication protocols is the division of computational load between the User, GW and Sensor nodes. A lower computational load on the sensor nodes could has a positive influence on the performance of the network. The schemes by (Vaidya et al., 2010) and (Liu & Chung, 2017) have one of the lowest computational requirements compared to the other protocols.

An improvement for authentication protocols could be made in combining certain aspects of different protocols. Taking the lightweight smart-card and ECC based protocol of Liu & Chung, 2017 and the extra security measures of (Gope & Hwang, 2016) in the form of sequence number, would make for a lightweight and secure authentication protocol.

Further research recommendations should be to develop even more lightweight in terms of load on the sensor nodes. For the comparative study a translation from time notions to actual number could aid the understandability of the results. This would make it easier to identify which protocol is more efficient. This could aid in having a more meaningful comparison and make it easier to compare a larger number of protocols.

References

Augustin, A., Yi, J., Clausen, T., & Townsley, W. M. (2016). A study of Lora: Long range low power networks for the internet of things. *Sensors (Switzerland)*, *16*(9). doi: 10.3390/s16091466

Benenson, Z., & Gedicke, N. (2005). Realizing Robust User Authentication in Sensor Networks Chair of Wireless Networks. *System*, 1–5.

Choi, Y. (2018). Cryptanalysis on privacy-aware two-factor authentication protocol for wireless sensor networks. *International Journal of Electrical and Computer Engineering*, *8*(1), 605–610. doi: 10.11591/ijece.v8i1.pp605-610

Gope, P., & Hwang, T. (2016). A Realistic Lightweight Anonymous Authentication Protocol for Securing Real-Time Application Data Access in Wireless Sensor Networks. *IEEE Transactions on Industrial Electronics*, *63*(11), 7124–7132. doi: 10.1109/TIE.2016.2585081

Grandviewresearch. (2018). *Industrial wireless sensor network market size, share trends analysis report by component (hardware, software, service), by type, by technology, by application, by end use, and segment forecasts, 2019 - 2025.* Retrieved from https://www.grandviewresearch.com/industry-analysis/industrial-wireless-sensor-networks-iwsn-market

Jiang, Q., Kumar, N., Ma, J., Shen, J., He, D., & Chilamkurti, N. (2017). A privacy-aware two-factor authentication protocol based on elliptic curve cryptography for wireless sensor networks. *International Journal of Network Management*, *27*(3), 1–17. doi: 10.1002/nem.1937

Jiang, Q., Ma, J., Wei, F., Tian, Y., Shen, J., & Yang, Y. (2016). An untraceable temporal-credential-based two-factor authentication scheme using ECC for wireless sensor networks. *Journal of Network and Computer Applications*, *76*, 37–48. Retrieved from http://dx.doi.org/10.1016/j.jnca.2016.10.001 doi: 10.1016/j.jnca.2016.10.001

Kumar, S., Tiwari, P., & Zymbler, M. (2019). Internet of Things is a revolutionary approach for future technology enhancement: a review. *Journal of Big Data*, *6*(1). Retrieved from https://doi.org/10.1186/s40537-019-0268-2 doi: 10.1186/s40537-019-0268-2

Lee, H., Yoo, S., & Kim, Y. W. (2016). An energy management framework for smart factory based on context-awareness. *International Conference on Advanced Communication Technology, ICACT*, *2016-March*, 685–688. doi: 10.1109/ICACT.2016.7423520

Liu, C. H., & Chung, Y. F. (2017). Secure user authentication scheme for wireless healthcare sensor networks. *Computers and Electrical Engineering*, *59*, 250–261. doi: 10.1016/j.compeleceng.2016.01.002

OWASP. (2018). *Owasp internet of things top 10 2018.* Retrieved 2018, from https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf

Rajeswari, S. R., & Seenivasagam, V. (2016). Comparative Study on Various Authentication Protocols in Wireless Sensor Networks. *Scientific World Journal*, *2016*(iii). doi: 10.1155/2016/6854303

Sánchez-Álvarez, D., Linaje, M., & Rodríguez-Pérez, F. J. (2018). A framework to design the computational load distribution of wireless sensor networks in power consumption constrained environments. *Sensors (Switzerland)*, *18*(4). doi: 10.3390/s18040954

Seo, D., Neely, R. M., Shen, K., Singhal, U., Alon, E., Rabaey, J. M., … Maharbiz, M. M. (2016). Wireless Recording in the Peripheral Nervous System with Ultrasonic Neural Dust. *Neuron*, *91*(3), 529–539. Retrieved from http://dx.doi.org/10.1016/j.neuron.2016.06.034 doi: 10.1016/j.neuron.2016.06.034

Tsantilas, S., Spandonidis, C., Giannopoulos, F., Galiatsatos, N., Karageorgiou, D., & Giordamlis, C. (2020). A comparative study of wireless communication protocols in a computer vision system for improving the autonomy of the visually impaired. *Journal of Engineering Science and Technology Review*, *13*(1), 72–76. doi: 10.25103/jestr.131.10

Turkanovic, M., & Holbl, M. (2013). An improved dynamic password-based user authentication scheme for hierarchical wireless sensor networks. *Elektronika ir Elektrotechnika*, *19*(6), 109–116. doi: 10.5755/j01.eee.19.6.2038

Ullo, S. L., & Sinha, G. R. (2020). Advances in smart environment monitoring systems using iot and sensors. *Sensors (Switzerland)*, *20*(11). doi: 10.3390/s20113113

Vaidya, B., Makrakis, D., & Mouftah, H. T. (2010). Improved two-factor user authentication in wireless sensor networks. *2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob'2010*, *8*(3), 600–606. doi: 10.1109/WIMOB.2010.5645004

Wong, K. H., Yuan, Z., Jiannong, C., & Shengwei, W. (2006). A dynamic user authentication scheme for wireless sensor networks. *Proceedings - IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, *2006 II*, 244–251. doi: 10.1109/SUTC.2006.1636182

Yang, T., Zhang, G. H., Liu, L., & Zhang, Y. Q. (2020). A survey on authentication protocols for internet of things. *Journal of Cryptologic Research*, *7*(1), 87–101. doi: 10.13868/j.cnki.jcr.000352

A Figures

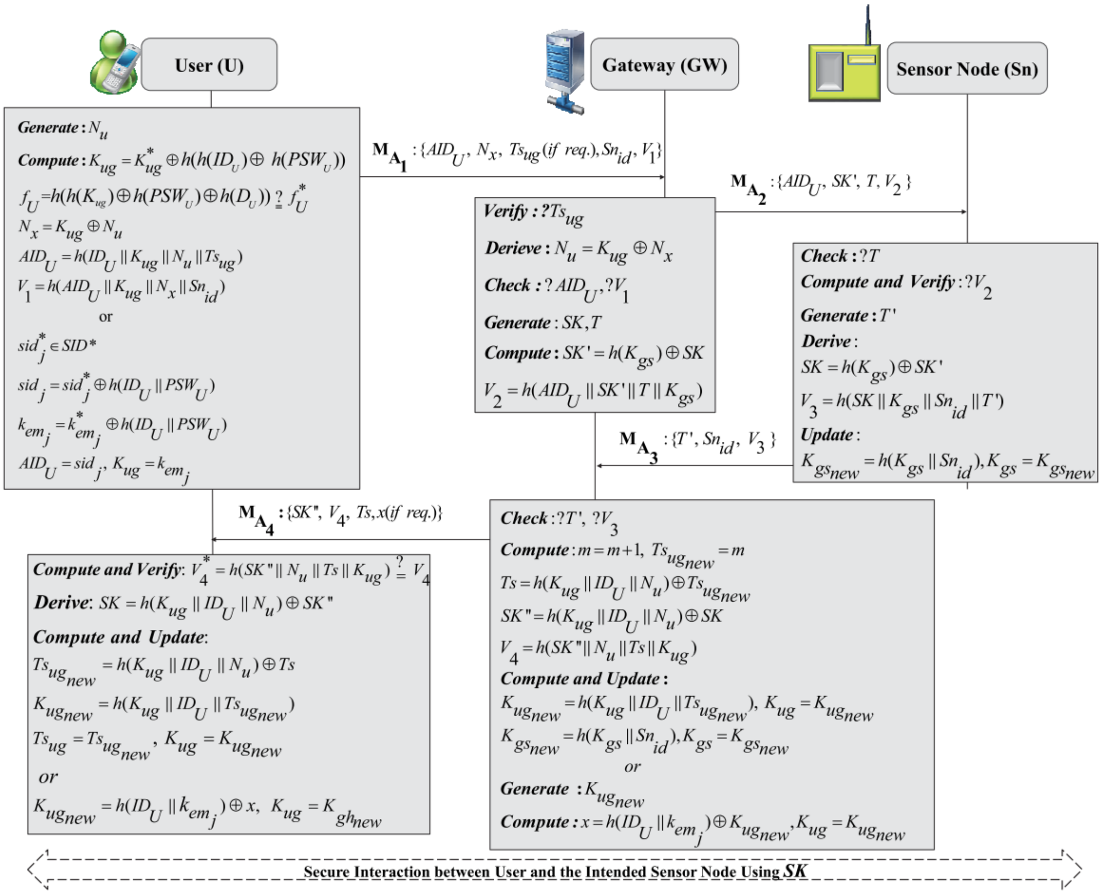
User (U)
Gateway (GW)
Sensor Node (Sn)
Generate : N_u
Compute : $K_{ug} = K_{ug}^* \oplus h(h(ID_U) \oplus h(PSW_U))$
$f_U = h(h(K_{ug}) \oplus h(PSW_U) \oplus h(D_U)) \overset{?}{=} f_U^*$
$N_x = K_{ug} \oplus N_u$
$AID_U = h(ID_U \| K_{ug} \| N_u \| Ts_{ug})$
$V_1 = h(AID_U \| K_{ug} \| N_x \| Sn_{id})$
or
$sid_j^* \in SID^*$
$sid_j = sid_j^* \oplus h(ID_U \| PSW_U)$
$k_{em_j} = k_{em_j}^* \oplus h(ID_U \| PSW_U)$
$AID_U = sid_j, K_{ug} = k_{em_j}$
$\mathbf{M}_{\mathbf{A_1}}$: $\{AID_U, N_x, Ts_{ug}(if\ req.), Sn_{id}, V_1\}$
Verify : ? Ts_{ug}
Derieve : $N_u = K_{ug} \oplus N_x$
Check : $?\, AID_U, ?V_1$
Generate : SK, T
Compute : $SK' = h(K_{gs}) \oplus SK$
$V_2 = h(AID_U \| SK' \| T \| K_{gs})$
$\mathbf{M}_{\mathbf{A_2}}$: $\{AID_U, SK', T, V_2\}$
Check : $?T$
Compute and Verify : $?V_2$
Generate : T'
Derive :
$SK = h(K_{gs}) \oplus SK'$
$V_3 = h(SK \| K_{gs} \| Sn_{id} \| T')$
Update :
$K_{gs_{new}} = h(K_{gs} \| Sn_{id}), K_{gs} = K_{gs_{new}}$
$\mathbf{M}_{\mathbf{A_3}}$: $\{T', Sn_{id}, V_3\}$
Check : $?T', ?V_3$
Compute : $m = m+1,\ Ts_{ug_{new}} = m$
$Ts = h(K_{ug} \| ID_U \| N_u) \oplus Ts_{ug_{new}}$
$SK'' = h(K_{ug} \| ID_U \| N_u) \oplus SK$
$V_4 = h(SK'' \| N_u \| Ts \| K_{ug})$
Compute and Update :
$K_{ug_{new}} = h(K_{ug} \| ID_U \| Ts_{ug_{new}}),\ K_{ug} = K_{ug_{new}}$
$K_{gs_{new}} = h(K_{gs} \| Sn_{id}), K_{gs} = K_{gs_{new}}$
or
Generate : $K_{ug_{new}}$
Compute : $x = h(ID_U \| k_{em_j}) \oplus K_{ug_{new}}, K_{ug} = K_{ug_{new}}$
$\mathbf{M}_{\mathbf{A_4}}$ **:** $\{SK'', V_4, Ts, x(if\ req.)\}$
Compute and Verify: $V_4^* = h(SK'' \| N_u \| Ts \| K_{ug}) \overset{?}{=} V_4$
Derive: $SK = h(K_{ug} \| ID_U \| N_u) \oplus SK''$
Compute and Update:
$Ts_{ug_{new}} = h(K_{ug} \| ID_U \| N_u) \oplus Ts$
$K_{ug_{new}} = h(K_{ug} \| ID_U \| Ts_{ug_{new}})$
$Ts_{ug} = Ts_{ug_{new}},\ K_{ug} = K_{ug_{new}}$
or
$K_{ug_{new}} = h(ID_U \| k_{em_j}) \oplus x,\ K_{ug} = K_{gh_{new}}$
Secure Interaction between User and the Intended Sensor Node Using SK

Figure 2: Authentication and key exchange phase (Gope & Hwang, 2016)