Volume XXIII 2020
ISSUE no.1
MBNA Publishing House Constanta 2020

Scientific Bulletin of Naval Academy

SBNA PAPER • OPEN ACCESS

Improved authentication method in embedded networks systems. An autonomous vehicle approach

To cite this article: Marius Rogobete and Eugen Marin, Scientific Bulletin of Naval Academy, Vol. XXIII 2020, pg.253-256.

Available online at www.anmb.ro

ISSN: 2392-8956; ISSN-L: 1454-864X

doi: 10.21279/1454-864X-20-I1-035

Improved Authentication Method in Embedded Networks Systems. An Autonomous Vehicle Approach

Marius Rogobete[1], Eugen Marin[2]

[1]marius.rogobete@gmx.de, [2]eugen.marin@gmx.de

Abstract. By adding connectivity to in-vehicle networks (including multimedia devices) and external networks (e.g. wireless and Internet) the attack surface was dramatically extended. In this context, there are several types of attacks already been demonstrated on automotive/AV control networks using compromised network connection or physical manipulation. Subsequently, the success attacks violate safety requirements, being able to disrupt system operation or even to take over operational control. In order to avoid false authentication or identity theft of devices or IoT, this paper proposes a time-based authentication method. The proposed method increases the degree of cybersecurity and allows its implementation on independent, low power mobile devices. Finally, a critical conclusion regarding the proposed authentication methods is presented.

1. Introduction

Any embedded system that is open to networks needs a specific level of security that is appropriate to its requirements.

The assurance of these complex requirements, especially for critical AV systems, is achieved through a proper authentication and authorization.

For the actual technology the authentication process is fundamental for the security of the communications, telematics commands and to prevent non-original peripherals connection as well as the digital rights management.

2. Authentication vulnerabilities

The most vulnerabilities make possible exploits of the key management especially in wireless networks. for example [Apa and Penagos, 2013] present vulnerabilities of a device that uses a graphical interface to set default values when configuring the connection, a password is generated and subsequently is used to generate the AES key. But the pseudo-random number generator uses a current time function for generator itself. Then, an attacker can calculate the password and the encryption key, thus managing the interception of network communications.

In many cases, the vulnerability exploits the protocol design, especially in terms of encryption and authentication. [Clarke, 2012] shows how an exploitation of the device can be achieved using a data package with a specific design. The attack is oriented on the certificates that are hard coded into the operating system.

The default account which support the password recovery cannot be disabled and attackers could use it by knowing the MAC address as well, thus being able to connect the external device and take over the system control.

2.1. Telematics

For telematics unit (TCU) there are ECUs able to play gateway roles, to connect different in-vehicle networks. For example, when a call is initiated, a random authentication string of three bytes is sent by the vehicle and its authentication timer for program authentication is started. In the same time, a digest of 64 bits is computed from the sent three bytes merged with a pre-shared key, and it wait for the packet answer in a lock mode. If the answer is not received in the specific authentication time or the answer is incorrect, an error message is sent, and the unit hangs up until the error is acknowledged.

A serious vulnerability occurs when multiple calls are made to a car while it is off. These lead to the same expected response and an attacker can sniffing the vehicle connection and catch the message. Therefore, it will be able to connect to the vehicle TCU by using that response and possible to take over the system control.

3. Basic concepts

Authentication schemes are very different, but for embedded systems there have common architectural elements:

- Authentication based on identity is a method that use a combination of hash and cryptographic algorithms (symmetric or not)
- Token based authentication [Emerson] when a server generated an identification token/string for device authentication process
- For non-token process, a device can use:
  - hardware based authentication that uses hardware characteristics, e.g. for True Random Number Generator/ Physical Unclonable Function
  - dedicated hardware that process the stored keys
- Procedures of authentication, could be:
  - one-way procedure when two entities communicate but only one is authenticated
  - two-way procedure when both entities authenticates each other
  - three-way uses a central authority (could be a dedicated module) that authenticates both entities

For vehicular networks, [Chan, 2014] described a two-factor authentication method, where the vehicle should be connected to a server for a successful authentication. [Tangade, 2016] describes a complex authentication protocol that uses a symmetric Hash-based Message Authentication Code (HMAC) for vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication and asymmetric cryptography (PKI) to certify the public key.

3.1. Hash-based Message Authentication Code

The main approaches of authentication schemes in embedded systems have as fundament Hash-based Message Authentication Code (HMAC).

HMAC is a digest (hashing) of a data string produced by merging a secret key (shared by sender and receiver) with a message.

The authentication scheme uses common secret key (*ScKey*) for the sender and receiver as well as a common message string (*Ms*). The secret key (*ScKey*) is used in conjunction with a key generator block to generate inner key (*Ki*) and outer key (*Ko*) (figure 1).

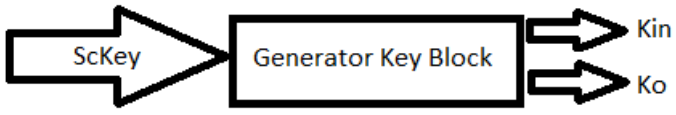

Figure 1 – Using secret key to derive inner and outer keys.

As the figure 2 shows in a very simple manner, the message (*Ms*) is combined with inner key (*Kin*), then digest the result by hashing function. This hash again combined with the outer key (*Ko*) and hashed to achieve Hash-based Message Authentication Code.

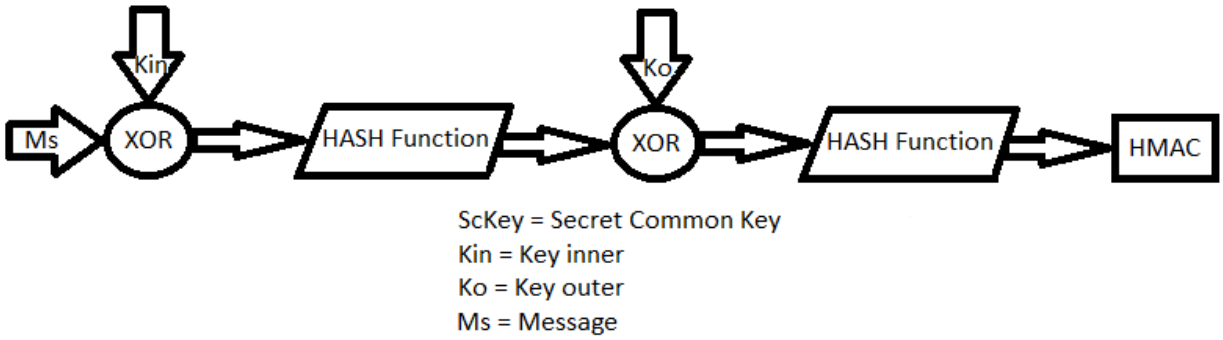

Figure 2 - Hash-based Message Authentication Code scheme

HMAC is sent to the receiver which knows the secret key, the key generator algorithm and the message. The receiver starts to produce its HMAC using the same scheme and HASH function. Finally, the HMAC produce by sender $HMAC_S$ is compared with receiver $HMAC_R$, if they match the sender is authenticated. The lengths of the key, together with the output size determine how strong the algorithm is.

4. Proposed method

Our research is using a pseudo timestamp data embedded into the authentication scheme and allows signature verification at any time.

The vehicle unit and the device should be synchronized in time but with different metering base than the universal time, for better protection only. This method permits to verify if the device attributes are revoked or even expired at the signing time.

The TCU must store a database with devices and their attributes for authentication process and even verification.

A description of the sender sequence for our method:

1. Generate the pseudo timestamp ($PTmp$) on the device (i), where *Tau* is a specific function (or just different parameterized function) for every device (i), that computes pseudo timestamp using local device time (t) with configurable granulation, synchronized with TCU time:
$$PTmp_i(t) = \tau(t, i) \quad (1)$$
2. Concatenates the pseudo timestamp with the device index:
$$Tp_i(t) = PTmp_i(t) \,||\, t \quad (2)$$
3. Compute the HMAC for sender as was previously described, and concatenates it with Tp_i
$$HmacT_i = HMAC \,||\, Tp_i \quad (3)$$
4. Encrypt HmacT with symmetric key KeyS and produce the message to be sent
$$Mess = Enc\{KeyS, HmacT_i\} \quad (4)$$

The receiver vehicle sequence (TCU):

1. Decryption the received message:
$$HmacTr = Dec\{KeS, Mess\} \quad (6)$$

2. Extract the $HMAC_S$, device index i and sender time t_s, using data size properties

$$HMAC_S = ExtrSz(HmacTr, 0,64) \quad (7)$$
$$t_s = ExtrSz\left(HmacTr, 63, T_{length}\right) \quad (8)$$
$$i = ExtrSz\left(HmacTr, 63 + T_{length} - 1, i_{length}\right) \quad (9)$$
$$t_{TCU} = \tau(t, i) \quad (10)$$
$$IF\ (t_{TCU} \neq t_s) \rightarrow AUTHENTICATION\ Error \quad (11)$$

3. Compare the received $HMAC_S$ with computed one $HMAC_R$, if they don't match, the pre-authentication is failed, if they match go to the next step

$$IF\ (HMAC_R \neq HMAC_S) \rightarrow AUTHENTICATION\ Error \quad (12)$$

4. The database could set different rights for different devices able to join to in-vehicle networks but also for TCAM functionalities. Based on these attributes set on V2X database, a device or service could access specific network resources or even telematics command.

5. Conclusions

The proposed method is able to improve in a significant manner the vehicle network connections.

The security for every device is double checked, one the HMAC and then the timestamp. Besides the fact that each device trying to connect is double checked, adding a value that varies over time at HMAC makes network fraud attempts extremely difficult, as it is an attempt to break data that is change over time.

Moreover, it offers the possibility to define specific rights into the vehicle network for every device which is recorded into the TCAM connection database.

References

[1] Apa L. and Penagos C. M., Compromising Industrial Facilities from40 Miles Away, BlackHat, 2013.

[2] Clarke J. W. , RuggedCom - Backdoor Accounts in my SCADA net-work? You don't say... seclists.org/fulldisclosure/2012/Apr/277, 2012.

[3] Emerson S., Choi Y.K., Hwang D.Y., et al. An OAuth based authentication mechanism for IoT networks. In Proceedings of the 2015 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 28–30 October 2015

[4] Chan A.C.F., Zhou J., Cyber–physical device authentication for the smart grid electric vehicle ecosystem. IEEE J. Sel. Areas Commun. 2014, 32, 1509–1517.

[5] Tangade S., Manvi S.S., Scalable and privacy-preserving authentication protocol for secure vehicular communications. In Proceedings of the 2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Bangalore, India, 6–9 November 2016.

[6] Rogobete M., Tarabuta O, Hashing and Message Authentication Code Implementation. An Embedded Approach, Scientific Bulletin of Naval Academy, Constanta 2019, ISSN: 2392-8956; ISSN-L: 1454-864X

[7] Rogobete M., Hash Function and Collision Resistance, EDUCATION AND CREATIVITY FOR A KNOWLEDGE BASED SOCIETY, Bucharest, November 2018.

[8] Gebotys C., White B., Mateos E., Preaveraging and Carry Propagate Approaches to SideChannel Analysis of HMAC-SHA256, ACM Transactions on Embedded Computing Systems (TECS),Vol.15, Feb. 2016, DOI 10.1145/2794093, ISSN:1539-9087