基于HMAC和TEA算法的CAN总线身份认证方法研究

张之森[1,2], 李 芳[1,2], 王丽芳[1,2], 吴 艳[2]

(1. 中国科学院电工研究所, 北京 100190; 2. 中国科学院大学, 北京 100049)

摘要: 随着汽车智能化以及车联网的发展,如何保护车载网络系统的安全成为需要迫切解决的焦点问题,CAN总线作为目前应用极为广泛的车载总线,它的安全性很大程度决定了车载信息安全程度。本文提出了一种结合HMAC-SHA256和TEA加密算法的认证方式,基于挑战/应答模式的动态身份认证方法,结合CAN总线自身的特点,在认证过程中使用动态口令完成身份认证。为了验证本文所提出的认证方法的有效性,在Matlab/Simulink中进行了认证模型的搭建以及认证过程的仿真与分析,通过数据流的记录以及网络攻击模型的攻击实验表明,本文所提出的认证方法可以实现多节点验证,并且可以有效应对重放攻击,是一种安全可靠的认证方法,提高了CAN总线的安全性。

关键词: 网络安全; CAN总线; HMAC-SHA256; 重放攻击

DOI: 10.12067/ATEEE2104022 **文章编号:** 1003-3076(2021)09-0057-07 **中图分类号:** TP393

1 引言

随着车联网、智能车技术的发展,车辆信息安全问题变得越来越重要,由于攻击者可以通过连接车辆外部接口或者通过无线连接车内电控单元对车辆进行控制,过去被视为封闭车载网络可以连接到外部网络,从而增加了安全风险。目前车载网络系统中控制器局域网络(Controller Area Network, CAN)总线仍然是最常用的协议,而其缺乏足够的安全保护,CAN总线目前存在的网络安全漏洞总结[1]如下:

(1)缺乏足够的总线保护。目前CAN总线缺乏必要的安全保护,以确保信息的保密性、完整性、可用性、真实性和不可抵赖性。CAN总线上的消息可以被总线上任意节点读取。应对措施:加入消息认证码(Message Authentication Code, MAC)或数字签名保护。

(2)弱认证。对电控单元(Electronic Control Unit,ECU)进行固件修改时使用口令等弱认证技术。应对措施:加入高级别的加解密和身份认证。

(3)CAN协议滥用。由于CAN总线的广播通信与分优先级处理特性,恶意代码很容易实现拒绝服务(Denial of Sevice, DoS)攻击。比如:制造高优先级帧阻断正常帧的传输;制造错误信号帧使其他控制器从CAN总线断开;权限仲裁机制使恶意节点能够独占总线。应对措施:加入网络监测和入侵检测机制。

(4)消息泄露。通过常规诊断接口,如OBDⅡ、K线或L线即可掌握车辆运行时信息。随意使用车载诊断系统(On Board Diagnostics,OBD)软件也存在潜在的安全隐患,OBD模块通常会存储访问控制ECU的指令。应对措施:加入足够安全的加解密算法。

针对以上信息安全,研究人员从信息加密、身份认证、入侵检测、攻击检测等方面开展了深入研究。其中身份认证技术就是让ECU在通信之前先进行身份认证[2],保证互相通信的双方是可信的,它是保证车载信息安全的首要步骤,B.Groza[3]等针对CAN总线的信息安全问题,以小规模的ECU集群环境为实验平台,提出了EPSB(Efficient Protocols for Secure Broadcast)[4]、Libra-CAN(Lightweight

收稿日期: 2021-04-01

基金项目: 国家重点研发计划项目(2017YFB0102502)

作者简介: 张之森(1996-), 男, 河北籍, 硕士研究生, 研究方向为车辆智能化技术等;
李 芳(1982-), 女, 河北籍, 副研究员, 博士, 研究方向为车辆智能化技术等。

Broadcast Authentication Protocol for Controller Area Networks)[5]等多项轻量级认证协议,并对其可应用性进行了验证,然而,由于发送方和通信主机之间共享的密钥在每个会话中都不能改变,所以在窃听传输的 CAN 数据帧和 MAC 之后可能发生重放攻击。Samuel Woo 等人[6]对车辆的初始化身份认证和信息加密传输做了研究,并且对提出的方法进行了验证和评估[7]。吴尚则等人[8]在已有的身份认证方法的基础上,针对 CAN 总线设计出一种基于动态口令的身份认证方案,但是文中的方案局限于一对一认证,当面对网关对多节点或节点对多节点的认证时,认证消耗的时间会大大加长。孙瑶等人[9]在已有认证基础上,提出了一种基于信息重要性的 ECU 安全等级的身份认证方法,该机制以网关为中心,为每个 ECU 赋予唯一标识用于身份认证,最后通过协商出的密钥矩阵进行加密通信。该方法以网关为中心,没有进行 ECU 对网关的认证,存在一定风险。万爱兰等人[10]提出的基于一次性密码本车内网络的认证方法,利用网关 ECU 的安全存储模块对 ECU 进行验证,但其没考虑到恶意网关对系统的攻击。Hyo Jin Jo 等人[11]提出了一种避免攻击的总线认证协议,最终认证结果采用总线内广播节点合法性的方法,其设置一个 ECU 节点为统一认证节点,并没有使用到网关强大的运算能力。对于以上多数提出的方法,攻击者容易实现重放攻击,仅仅通过一步认证并不能完美避免重放攻击。

为解决上述问题,本文提出一种使用 HMAC-SHA256(Hash-based Message Authentication Code-SHA256)和 TEA(Tiny Encryption Algorithm)加密相结合的认证方式,实现了在多个节点或节点网关间的身份认证,即在进行消息传输前先进行节点间的身份认证,对节点间进行互相身份认证,保障下一步的消息传输安全性。采用基于 TEA 的对称加密算法进行消息传输,提高系统在认证过程中的响应速度,减少总线负载率和开销,同时保障了系统的安全性。通过 Matlab/Simulink 仿真软件搭建了包含一个网关节点,以及两个 ECU 节点的仿真模型,利用 canTool 模块对系统认证过程进行数据记录,验证了认证方法的有效性。

2 身份认证方法

2.1 设计思路

一般车载网络系统的认证分为身份认证和消息认证两种,其中身份认证即验证实体身份的真实性,而消息认证是验证消息来源的真实性,验证消息在传输过程中没有被篡改或伪造等。本文主要关注身份认证,是 CAN 网络中重要的实体节点之间的认证策略。其中重要的实体节点主要是指对安全性和可靠性要求较高的节点,比如网关节点、系统中涉及驱动、制动、转向的关键节点等,如图 1 所示为一个需要进行身份认证的网络节点实例,T-BOX、网关、车门控制器、电动助力转向、ABS 执行机构等均可以作为重要节点进行身份认证。

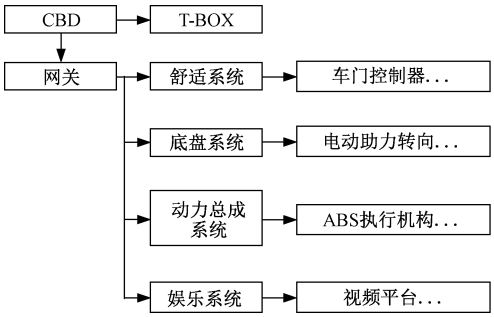

图 1 身份认证节点示意图

Fig.1 Schematic diagram of identity authentication node

本文提出的针对 CAN 总线的身份认证方法主要是基于挑战/应答模式的动态口令身份认证方法,首先初始化生成短期密钥,其意义就在于一次性,生成的口令有效期只是在这次的初始化认证阶段,将车启动到熄火停车算一个周期,每辆车的周期内有且只有一次初始化阶段。其次在网关和节点之间进行双方握手,通过随机数交互完成网关与节点之间的挑战与应答,若有节点为伪装节点,将不能返回正确的值,并被网关识别为非法节点。同时,由于网关需要通过短期密钥加密各个节点发出的随机数和节点号,若网关节点是伪装的,可以被各个节点识别。由于认证过程中的报文大多只有一次的寿命,在下一次初始化认证过程中,如果进行重放攻击,重放的还是上次所截获的报文,由于随机数已经发生改变,口令相应也发生变化,会被网络中的节点识别出来,发出危险警报。本文中的长期密钥,可以定期更新,更加地降低了安全风险。

本文在设计身份认证方法中主要用到基于 TEA 的对称加密算法以及哈希运算消息认证码 HMAC 算法。TEA 对称加密算法是由剑桥大学计算机实验室的 David Wheeler 和 Roger Needham 于 1994 年发明,它是一种分组密码算法,其明文密文块为 64 比特,密钥长度为 128 比特。TEA 算法利用

不断增加的黄金分割率值作为变化,使得每轮的加密值不同。虽然 TEA 算法简单,但有很强的抗差分分析能力,加密速度快。哈希运算消息认证码 HMAC 算法是一种使用 Hash 函数来构造消息认证码的方法。基于 Hash 函数的算法被广泛应用于对数据的完整性、合法性进行保护,是诸多数字签名设计方案。根据目前车载网络系统节点硬件的处理能力,选择 SHA256 算法作为 HMAC 中的 Hash 函数。SHA256 算法单向 Hash 函数是密码学和信息安全领域中的一个非常重要的基本算法,它是把任意长的消息转化为较短的、固定长度的消息摘要的算法。SHA256 哈希算法的实现主要包括常量初始化、信息预处理(附加填充比特和附加长度)、逻辑运算等。

2.2　设计步骤

如图 2 所示为所设计的身份认证总流程,认证方法主要分为五个步骤。步骤一:节点生成随机数并初始化生成密钥,随机数使用 TEA 加密的方法进行传输,此步骤中网关与 ECU 节点分别生成短期密钥与长期密钥,用于后续步骤的通信。步骤二:网关向各节点分发随机数,分发时使用 TEA 加密随机数,在各 ECU 节点中解密,此随机数在步骤三中用于认证。步骤三:网关对节点的认证,各 ECU 节点在步骤二得到网关发送的随机数,利用密钥通过 HMAC-SHA256 算法加密,得到密文,将密文发送至网关节点,同时网关节点也进行相同的操作,网关节点比较节点发送的密文。步骤四:认证成功分发及节点对网关的认证,网关节点向各 ECU 单元发送认证是否成功的报文,若各 ECU 为合法节点,同时各节点也可以对网关的合法性进行认证,若双方认证通过,开始信息传输。步骤五:各 ECU 单元通过报文中的节点认证是否通过标志来判断彼此之间的合法性。

步骤一:初始化生成短期密钥。由长期密钥和随机数生成短期密钥用于步骤二的通信。在认证开始前,各 ECU 和网关中存储相同的长期密钥 L_{m}。具体如下:

认证开始后,各 ECU 节点在节点内生成一个随机数 R_{x1}(x 代表 A,B,C 等节点),将随机数 R_{x1} 通过 TEA 加密的密文 T_{x1} 等发送给网关节点,同时在各个节点中,随机数 R_{x1} 与长期密钥 L_{m} 进行异或操作得到对应各节点的短期密钥 S_{x1} 等;接收方网关节点在接收到密文 T_{x1} 等后进行 TEA 解密,得到随机数 R_{x2} 等,将随机数 R_{x2} 等与长期密钥 L_{m} 进行异或操作得到对应各节点的短期密钥 S_{x2} 等;同时,各 ECU 也可以收到其他 ECU 广播的报文,比如 $\mathrm{ECU_A}$ 收到 $\mathrm{ECU_B}$,$\mathrm{ECU_C}$ 的报文后,经过解密得到其他节点的随机数 R_{b1},R_{c1} 等,在节点内各随机数与长期密钥 L_{m} 异或后得到步骤四需要的短期密钥 S_1;接收方网关节点在接收到密文 T_{x1} 等后进行 TEA 解密,得到随机数 R_{x2} 等,将随机数 R_{x2} 等与长期密钥 L_{m} 进行异或操作得到短期密钥 S_2。此短期密钥作为认证过程后续步骤四与各个节点通信的临时密钥。以 $\mathrm{ECU_A}$ 和网关为例:

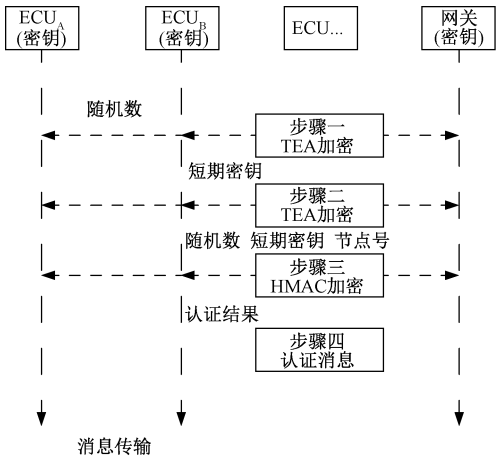

图 2　身份认证的总流程

Fig.2　Overall process of identity authentication

$$S_{x1} = L_{\mathrm{m}} \oplus R_{x1} \tag{1}$$

$$T_{x1} = T(R_{x1} \mid L_{\mathrm{m}}) \tag{2}$$

$$R_{x2} = D(T_{x1} \mid L_m) \tag{3}$$

$$S_{x2} = L_{\mathrm{m}} \oplus R_{x2} \tag{4}$$

$$S_1 = L_{\mathrm{m}} \oplus (R_{\mathrm{a1}} \oplus R_{\mathrm{b1}} \oplus R_{\mathrm{c1}} \oplus \cdots) \tag{5}$$

$$S_2 = L_{\mathrm{m}} \oplus (R_{\mathrm{a2}} \oplus R_{\mathrm{b2}} \oplus R_{\mathrm{c2}} \oplus \cdots) \tag{6}$$

式中,L_{m} 为各节点和网关长期密钥;R_{x1} 为各 ECU 节点在节点内生成的随机数;T 为 TEA 加密;D 为 TEA 解密。

步骤二:网关向各 ECU 节点分发随机数。经过步骤一,网关节点中已经存储了 $\mathrm{ECU_A}$,$\mathrm{ECU_B}$,$\mathrm{ECU_C}$ 等 ECU 单元的节点号以及对应的短期密钥,并且这个短期密钥只在本次认证过程中有效。具体如下:

网关节点产生一个随机数 R_2 以分配给各个 ECU 节点,网关节点将此随机数进行 TEA 加密得到密文 E_{x2},依次分发到每个 ECU 节点,密钥采用步骤一中的 S_{x2} 等;各 ECU 节点收到网关节点发送的密文后,进行 TEA 解密,得到随机数 R_1 等,其中解密

密钥为步骤一中 S_{x1} 等。

$$E_{x2} = T(R_2 \mid S_{x2}) \tag{7}$$

$$R_1 = D(E_{x2} \mid S_{x1}) \tag{8}$$

式中，R_2 为网关节点产生的随机数；S_{x1}，S_{x2} 为步骤一中短期密钥。

步骤三：网关对节点的认证。ECU 节点通过短期密钥、随机数、节点号等信息，发送报文给网关进行认证。具体步骤如下：

各 ECU 单元接收到步骤二的 R_1，将随机数 R_1 和本身节点号 N 进行异或，通过密钥 S_{x1} 等进行 HMAC-SHA256 算法加密，得到密文 H_1，将密文发送给网关节点进行验证。进行 HMAC-SHA256 算法加密后的密文为 256 位，为 32 个字节；网关节点根据步骤二中的随机数 R_2 和网关存储的随机数同样进行异或操作，然后根据存储的短期密钥 S_{x2} 等进行 HMAC-SHA256 算法加密，得到密文 H_2。网关节点将密文 H_1 和 H_2 进行对比，若相同则进行下一步，不同则认证失败。

$$X_{x1} = R_1 \oplus N \tag{9}$$

$$H_1 = H(X_{x1} \mid S_{x1}) \tag{10}$$

$$X_{x2} = R_2 \oplus N \tag{11}$$

$$H_2 = H(X_{x2} \mid S_{x2}) \tag{12}$$

式中，R_1 为各 ECU 单元收到的随机数；N 为各节点自身节点号；H 为 HAMC 加密。

步骤四：认证成功分发及节点对网关的认证。此步骤为步骤三认证成功后，网关节点向 ECU_A，ECU_B，ECU_C 等 ECU 单元发送认证成功的报文，网关节点将步骤二中的随机数通过步骤一中网关和各 ECU 相同的短期密钥 S_1 加密，在随机数前附步骤三中各节点是否认证成功的消息，加密前的报文 X_1 是由 0、1 组成的认证信息，后缀随机数。若 ECU_A 未认证通过，则加密前的报文 X_1 组成为：011111，后缀随机数。

网关节点将此报文 X_1 经过密钥 S_1 进行 TEA 加密后的报文 T_2 发送给各个 ECU 节点，各 ECU 节点收到加密后的报文 T_2 后，通过短期密钥 S_1 进行解密。

$$T_2 = T(X_1 \mid S_1) \tag{13}$$

$$X_1 = D(T_2 \mid S_1) \tag{14}$$

式中，X_1 为认证结果报文。

步骤五：认证结果确认。各 ECU 单元互相传输消息时，要先验证各节点是否合法，通过有无收到认证成功报文判断彼此之间的合法性。

3 实验仿真及分析

3.1 模型仿真及结果

使用 Matlab/Simulink 软件对所提出的身份认证模型进行搭建，选择 Vehicle Network Toolbox 模块下的 CAN Communication 模仿 CAN 总线传输，搭建包含网关节点与两个 ECU 单元的系统模型，并按照 2.2 节所描述的五个步骤进行认证信息的传输。认证消息选择 Channel 1 传送，Channel 2 接收，通过 CAN Pack 模块打包，CAN Unpack 对消息解包。由于信息加密需要用到 TEA 加密以及 HMAC-SHA256 加密的方法，模型中采用了 S-function 模块对随机数、节点号和密钥等进行操作。此外，为了在模型中体现出步骤的顺序，步骤二的发生由步骤一的结果触发，步骤三的发生由步骤二的结果触发，步骤四的发生由步骤三的结果触发。

仿真模型如图 3 所示，图 3 为模型整体示意图，下半部分模块为网关节点的实现框图，上半部分为 ECU_A，ECU_B 节点的实现框图。

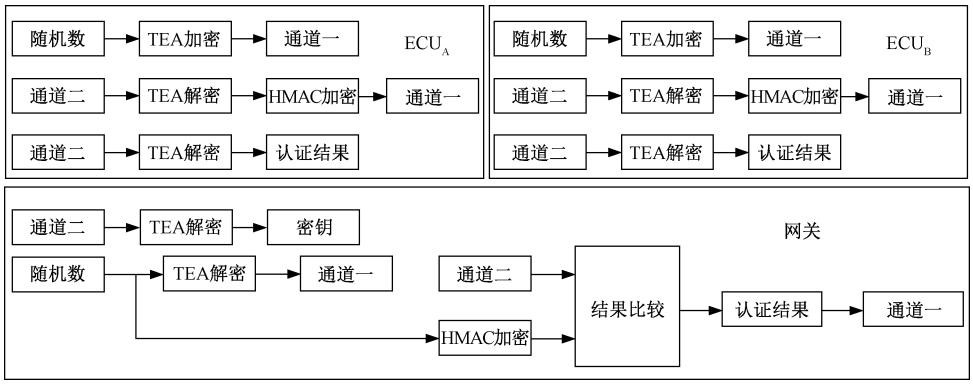

图 3 身份认证方法的整体模型

Fig.3 Overall model of identity authentication method

为了方便观察整个认证过程的传输过程以及认证结果,采用 Matlab 中的 canTool 工具对一个周期内 CAN 总线上传输的数据进行观察,canTool 可以直观地读取总线上的数据。如表 1 所示为 Channel 1 总线上一个认证周期内的 CAN 总线数据,ID 为 150、200 的消息为步骤一 CAN 总线上的数据,即为 ECU_A 和 ECU_B 节点产生的随机数加密密文。ID 为 250、300 的消息为步骤二 CAN 总线上的数据,为网关节点产生的随机数加密密文,ID 为 350~500 的消息为步骤三 CAN 总线上的数据,即为 ECU_A 节点经过 HMAC-SHA256 加密运算后发送给网关节点的加密密文,共有 4 个报文,ID 为 550~700 的消息为步骤三中 ECU_B 节点经过 HMAC-SHA256 加密运算后发送给网关节点的加密密文,共有 4 个报文。ID 为 750 的消息为步骤四的 CAN 总线上的数据,为网关节点发送给 ECU_A 和 ECU_B 节点的认证结果报文,其中包含了节点认证结果以及步骤二中的随机数。

如表 1 所示,一个周期内认证时间约为 10 ms,由于仿真环境包含网关节点和两个 ECU 节点,实际操作中随着节点数目的增多认证时间会加长。

表 1　canTool 中周期步骤信息

Tab.1　Step information in canTool in a cycle

步骤	时间戳	ID	长度	数据
步骤一	22.545 230	150	8	2D C5 F3 0E EB EA E5 19
	22.546 502	200	8	8C EC 42 5A 40 A5 D9 FD
步骤二	22.547 311	250	8	D9 9D 43 1F 07 3B 2A 0C
	22.548 205	300	8	11 03 E1 A1 F4 1F AC 83
步骤三	22.549 012	350	8	CA AB 41 B1 C2 33 AE C8
	22.549 021	400	8	FD 91 80 D4 BA 76 E6 52
	22.549 030	450	8	8E B0 B3 72 CD 48 B1 3D
	22.549 034	500	8	D5 BA 9F 45 70 CC 4B 64
	22.549 667	550	8	C3 42 92 DF CF 84 C0 C3
	22.549 673	600	8	AE 81 CB EA 78 DE 9F 78
	22.549 678	650	8	06 45 F1 54 05 E3 C5 40
	22.549 685	700	8	FD 8F 14 3D D1 D2 7A 86
步骤四	22.555 280	750	8	41 A0 CC F7 C9 16 0C 17

3.2　攻击测试与分析

进一步进行通信攻击测试,在所建立的 CAN 网络模型中加入一个攻击者节点,攻击者可以监听网络信息,发起重放报文、篡改报文等攻击。本文主要对重放攻击进行测试。设置攻击者节点为发送模式,攻击者通过报文监听,记录网络上发送的报文信息,然后在后续启动过程中向网络上重新发送该报文,试图通过身份认证。具体测试方法为:首先在一次正常认证过程中,攻击者节点记录任意时间段内 CAN 数据帧,在接下来攻击测试中,重新启动模型,将 ECU_A 和 ECU_B 设为非法节点,如表 2 所示,攻击节点向网络中重复发送上次记录的报文,由于为重放消息,在步骤一和步骤三中,ECU_A 和 ECU_B 节点发出的消息和上个周期相同。如表 3 所示,为各节点在步骤四收到的认证消息。其中随机数前包含着节点认证信息,ECU_A 和 ECU_B 认证为非法节点,后续消息发送不向该节点发送消息。通过测试,当重放消息时,在身份认证步骤中认证失败,网络检测为初始阶段认证失败。

本文提出的车辆认证协议认证过程需要的短期密钥都为认证初始阶段中随机数与节点存储的长期密钥一起生成的,下一次认证中为不同密钥,可以有效地保证消息的真实性和新鲜度。CAN 总线面对的众多攻击中,最容易受到重放攻击,在本文提出的认证方法协议中,步骤三为认证的核心工作,通过对比 HMAC-SHA256 加密密文的方法可以有效地避免重放攻击,不同时间加密的密文不同,密钥也不相同,若面对重放攻击,认证失败,结束认证流程。

本文实现的仿真方法是使用 Simulink 中的 CAN 通信模块,和实际的车辆 CAN 总线相近,仿真设计中,每个步骤采用上一步骤触发的方法设计,接近实际的 CAN 总线运行方式,但在未来实现中还需要考虑 CPU 的运算能力、计算实时性等限制。

表 2 重放攻击下 canTool 中周期步骤信息

Tab.2 Step information in canTool in a cycle under replay attack

步骤	时间戳	ID	长度	数据
步骤一	30. 603 237	150	8	2D C5 F3 0E EB EA E5 19
	30. 608 385	200	8	8C EC 42 5A 40 A5 D9 FD
步骤二	30. 609 202	250	8	B5 A7 58 98 A7 5F FB 49
	30. 609 980	300	8	A5 B2 56 F6 A8 60 B9 AE
步骤三	30. 610 866	350	8	CA AB 41 B1 C2 33 AE C8
	30. 615 779	400	8	FD 91 80 D4 BA 76 E6 52
	30. 615 793	450	8	8E B0 B3 72 CD 48 B1 3D
	30. 615 803	500	8	D5 BA 9F 45 70 CC 4B 64
	30. 615 816	550	8	C3 42 92 DF CF 84 C0 C3
	30. 620 737	600	8	AE 81 CB EA 78 DE 9F 78
	30. 620 748	650	8	06 45 F1 54 05 E3 C5 40
	30. 620 756	700	8	FD 8F 14 3D D1 D2 7A 86
步骤四	30. 625 498	750	8	01 D0 6C 1F E4 1A BA 5D

表 3 各 ECU 节点认证消息解密结果

Tab.3 Authentication results of each ECU

1 1 1	0	0	1 082 345…
ECU…	ECU_A	ECU_B	随机数

4 结论

本文提出了一种结合 HMAC-SHA256 和 TEA 加密算法的认证方式,实现在多个节点和节点网关间的身份认证,在认证过程中,由于不再是明文传输,并且通过了网关节点与 ECU 节点相互认证的方法,防止了消息被篡改和重放攻击的风险,保证了消息的真实性与完整性。通过仿真实验对认证过程的真实性和有效性进行测试,网络攻击模型的重放攻击实验表明,本文提出的认证方法是一种可靠有效的认证方法,可以提高 CAN 总线的安全级别。

参考文献 (References):

[1] Nilsson D K, Larson U E. A defense-in-depth approach to securing the wireless vehicle infrastructure [J]. Journal of Networks, 2009, 4 (7): 552-564.

[2] 张建晓 (Zhang Jianxiao). 身份认证技术及其发展趋势 (Identity authentication technology and its development trend) [J]. 信息通信 (Information and Communication), 2015, (2): 125-126.

[3] Groza B , Murvay S. Broadcast authentication in a low speed controller area network [J]. Communications in Computer & Information Science, 2012, 314: 330-344.

[4] Groza B , Murvay S. Efficient protocols for secure broadcast in controller area networks [J]. IEEE Transactions on Industrial Informatics, 2013, 9 (4): 2034-2042.

[5] Groza B , Murvay S , Herrewege A V , et al. LiBrA-CAN: A lightweight broadcast authentication protocol for controller area networks [A]. International Conference on Cryptology and Network Security [C]. Sanya, China, 2012. 12.

[6] Woo S , Jo H J , Kim I S , et al. A practical security architecture for in-vehicle CAN-FD [J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 17 (8): 2248-2261.

[7] Woo S , Jo H J , Dong H L . A practical wireless attack on the connected car and security protocol for in-vehicle CAN [J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 16 (2): 993-1006.

[8] 吴尚则 (Wu Shangze). 基于车载 CAN 总线网络的身份认证方法研究 (Research on identify authentication method based on vehicle CAN bus network) [D]. 长春: 吉林大学 (Changchun: Jilin University), 2018.

[9] 孙瑶, 王小妮, 刘鹏, 等 (Sun Yao, Wang Xiaoni, Liu Peng, et al.). 车载 CAN 总线认证与加密机制研究 (Research on authentication and encryption mechanism of vehicle CAN bus) [J]. 北京信息科技大学学报 (自然科学版) (Journal of Beijing Information Science and Technology University), 2019, 34 (3): 73-78.

[10] 万爱兰, 韩牟, 马世典, 等 (Wan Ailan, Han Mu, Ma Shidian, et al.). 基于一次性密码本的车内网身份认证协议 (In-car network identity authentication protocol based on one-time pad) [J]. 计算机工程 (Computer Engineering), 2018, 44 (6): 141-146, 161.

[11] Jo H J, Kim J H, Choi H Y, et. al. MAuth-CAN: Masquerade-attack-proof authentication for in-vehicle networks. [J] IEEE Transactions on Vehicular Technology, 2020, 69 (2): 2204-2218.

CAN bus identity authentication method based on Hash Algorithm and Tiny Encryption Algorithm

ZHANG Zhi-sen[1,2], LI Fang[1,2], WANG Li-fang[1,2], WU Yan[2]
(1. Institute of Electrical Engineering, Chinese Academy of Sciences, Beijing 100190, China;
2. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: With the development of vehicle intelligence and the Internet of Vehicles, how to protect the safety of the vehicle network system has become a focus issue that needs to be solved urgently. CAN(Controller Area Network) bus is currently a very widely used vehicle-mounted bus, and its security largely determines the degree of vehicle-mounted information security. CAN bus lacks adequate protection mechanisms and is vulnerable to external attacks such as replay attacks, modifying attacks and so on. This paper proposes an authentication method that combines HMAC(Hash-based Message Authentication Code)-SHA256 and TEA(Tiny Encryption Algorithm) algorithms. This method is based on dynamic identity authentication in challenge/response made and combined with the characteristics of the CAN bus itself and it achieves the identity authentication between the gateway and multiple ECUs. In the authentication process, dynamic passwords are used to complete identity authentication. In order to verify the validity of the authentication method proposed in this article, we built the authentication model in Matlab/Simulink and analyzed the authentication process in Matlab/canTool. Through data stream recording, and attack experiments using network attack models, it is shown that the authentication method proposed in this paper can achieve multi-node verification. Through comparison with MAC and Challenge/Response method, it can effectively deal with the replay attacks. This method is a safe and reliable authentication method, which improves the safety of the CAN bus.

Key words: network security; CAN bus; HMAC-SHA256; replay attack