A Lightweight Authentication and Key Exchange Protocol for IoT

摘要-- 物联网环境中的安全问题至关重要，因为在很多情况下，物联网设备提供的感官信息需要安全地共享。然而，为物联网设备提供认证和安全的通信可能是一个挑战。物联网设备有许多限制，包括计算、功率、内存和能源方面的限制。此外，它们往往要通过网关/水槽才能连接到网络。为了与网络的其他部分进行安全通信，物联网设备需要信任网关/水槽，这就需要设备对网关进行认证，反之亦然。我们还寻求支持安全通信，即使物联网设备和网关与网络的其他部分断开连接。在本文中，我们为这种物联网环境提供了一个轻量级的认证和密钥交换协议，其中物联网设备和网关通过无线通道进行通信。我们的协议取决于每对设备有两个独特的密钥，一个主密钥和一个初始会话密钥，在配置时提供。会话密钥是不断变化的，它被用作会话期间安全地交换帧的密钥。该协议是轻量级的，只使用对称密钥加密法和基于哈希消息验证码（HMAC）的密钥推导函数（HKDF）来提供验证、密钥交换、保密性和消息完整性。该协议不依赖于任何可信的第三方（TTP），很适合断开连接的物联网环境。密钥从不通过网络交换，提供完美的前向保密性。该协议在所需的计算量、内存和能源使用方面是高效的。

物联网设备的使用越来越多，传感器在我们的生活中越来越无处不在。有许多领域的关键功能，如监测病人的健康状况和环境，取决于从物联网设备收集的数据。一些功能可能经常发送大量的数据，如用于视频监控。这些数据必须以安全和认证的方式共享。
物联网设备是有资源限制的，在处理、内存和电源的可用性方面都是有限的，特别是因为很多设备都是由电池供电。它们也大多使用无线作为通信的物理媒介。物联网设备使用网关/水槽作为主要实体
2018年去中心化物联网安全与标准研讨会（DISS)
2018年2月18日，美国加州圣地亚哥
编号：1-891562-51-7
https://dx.doi.org/10.14722/diss.2018.23004 www.ndss-symposium.org
通过它与环境的其他部分进行通信。出于这个原因，我们寻求在物联网设备和与之通信的网关/水槽之间进行安全和认证的通信。然而，传统的认证和密钥建立协议并不完全适用于物联网环境，因为它有特定的限制。由于对可信第三方（TTP）的依赖，以及其高计算要求，使用公钥加密技术进行认证和密钥建立是不可行的。公钥加密法对计算能力、内存和电池消耗提出了严格的要求，而这些要求在典型的物联网设备中是无法满足的。物联网环境可能还需要在断开连接的模式下运行，无法访问TTP。
在没有可信第三方的情况下，两个节点之间的认证和密钥交换需要在节点之间建立某种形式的先验共享秘密。此外，重要的是要消除系统中的单点 "故障"（秘密暴露）。出于这个原因，我们寻求拥有一个以上的秘钥。每个秘钥都有不同的用途。此外，我们认为协议提供完美前向保密性（PFS）是很重要的。PFS意味着即使长期秘钥在某一时刻以某种方式被知道，所有过去的会话通信仍然必须是安全的。此外，每个会话的不同会话密钥允许用一个会话密钥加密的信息数量较少，使攻击者更难用密码分析法找到会话密钥。此外，即使攻击者能够以某种方式找到会话密钥，该会话密钥（针对该会话）在解密未来会话的数据时也没有用。这促使我们在本文提出的用于物联网环境的轻量级认证和密钥交换协议中使用有限寿命的会话密钥。

现在物联网安全的一个常见做法是，在物联网设备上设置
物联网设备已经配备了数字证书，由制造商签署，用于认证和密钥交换。然而，这种常见的方法要求物联网设备具有高计算能力和高功耗，因为数字证书依赖于公钥密码学。相比之下，我们建议使用对称密钥加密技术，这要简单得多。此外，常见的方法需要更多的内存，因为与对称密钥密码学和数字证书相比，公钥密码学使用更长的密钥，而数字证书是必须存储的几个KB。我们提出的协议对内存的要求要小得多。依靠数字证书而不与TTP联系，会产生单一的 "失败 "点，因为唯一用于认证的系统秘密是私钥。如果唯一的系统秘密以某种方式被破坏，用这种常见的方法，认证和密钥交换可以由攻击者完成；相比之下，拟议的协议并不完全依赖一个系统秘密进行认证和密钥交换，所以即使系统秘密之一以某种方式被发现，认证和密钥交换过程也不能被攻击者欺骗。
我们提出的协议在物联网设备和网关之间有两个共享的密钥：一个共享的长期对称主密钥，用Km表示，和一个共享的短期对称会话密钥，用Ks表示。初始会话密钥，用Kiks表示。主密钥和初始会话密钥都是手动插入到物联网设备和网关/水槽中一次。每个网关验证物联网设备，反之亦然，使用两个设备之间交换的信息，这些信息使用共享Km进行加密，并使用共享Kiks进行散列。此外，该协议确保两端继续拥有一个共享的Ks，用于加密和散列会话消息。更重要的是，秘密密钥从未在网络上明确交换。会话密钥Ks的重新认证和更新是定期进行的。更新之间交换的信息被称为 "会话"。每个会话都有一个阈值，它是该会话中交换的最大帧数。会话密钥Ks的更新是基于前一个会话中交换的帧的随机集合，以及前一个Ks和主密钥Km。因此，即使攻击者在某个时候以某种方式找到了Ks，他也无法计算出下一个会话密钥并解密信息，除非他同时拥有Km和当前的Ks。此外，由于很难有一个完美的通道，所有的帧都被攻击者捕获[12]，我们的协议迫使攻击者面对三个不同的障碍。因此，克服物联网系统的安全性要困难得多。
此外，我们的协议考虑到了物联网环境的各种限制和要求。它使用轻量级的机制：对称密钥加密法和基于哈希消息验证码（HMAC）的密钥衍生函数（HKDF）[2], [6]，用于保密性、消息完整性、验证和密钥交换。它力求最大限度地减少认证和密钥交换的信息数量（和交换的总字节数）[3], [7], [9]。它也不以任何方式依赖中央的、受信任的第三方。这提供了一个安全、快速以及能源和空间高效的认证和密钥交换协议。
本文的其余部分组织如下。下一节介绍了相关工作。第三节描述了系统模型和假设，包括考虑的网络攻击模型。第四节详细讨论了提议的协议，第五节讨论了提议协议的安全性分析。我们在第六节得出结论。

有许多认证和密钥建立协议被提议用于物联网环境中。有些使用公钥加密法进行认证和密钥建立[5]，[10]，考虑到物联网设备的资源限制，这种方法很昂贵。另一些则要求使用中央可信的第三方[8]，如证书颁发机构（CA）进行认证和密钥建立。然而，这很难在断开连接的环境中使用。另一种方法是依赖硬件能力，使用物理不可克隆函数（PUF）为认证和密钥建立引入随机性[1]。虽然PUF方法是有帮助的，但它仍然没有在设备中广泛部署，这促使我们去寻找替代方案。
一种方法是单独研究密钥建立[12]，利用无线信道的随机性来更新会话密钥。这种方法主要依赖于一个合理的假设，即无线信道并不完美（无损失）。该设计为每个会话使用不同的会话密钥，这也是我们利用的一种设计理念。一个会话是由 "一次性帧"（OTFs）的阈值数量定义的，这个数量是事先商定的。OTFs是精确地传输和接收一次的帧（也就是说，没有重传这些帧）。它假定两个实体都以一个公开知名的固定值作为初始会话密钥开始。这个密钥最初用于加密会话的帧。在会话期间，OTF存储在两端，在OTF的阈值被传达后，会话密钥被更新。为了更新会话密钥，当前的会话密钥和OTF一起被用来递增地生成新的

会话密钥。该设计取决于攻击者不具有与接收者相同的信道条件，因此将看到不同的OTF子集。然而，一个拥有完美信道的攻击者（例如，非常接近发射器）使得这个方案容易受到被动（例如，窃听）和主动攻击（例如，劫持）的影响。因此，我们的协议消除了这种对OTF和攻击者拥有不完美无线信道的依赖，并建立在[12]中提出的技术之上。

III. 物联网环境的限制和要求

物联网环境与一些制约因素有关。最重要的是，物联网设备的能量有限，因为它们主要依赖电池（例如，一次性纽扣电池，在有限的使用下最多可能持续1-2年）。其次，物联网设备的处理能力有限，内存也有限，只有几KB的RAM和几十KB的EEPROM。物联网设备很少有快速可重写的非易失性数据存储。

此外，物联网设备和网关可能必须在一个断开的环境中运行，限制了对中央实体或受信任的第三方的访问。最后，管理这些设备的方法需要是可扩展的，因为可能会有非常多的物联网设备，而且尽可能少的人为干预是很重要的。

A. 网络模型和假设

我们考虑的通用网络拓扑结构如图1所示。有多个物联网设备和一个网关，每个物联网设备只通过网关与其他节点进行通信。在本文中，我们假设无线链路上的MAC层协议可靠地、按顺序地传递数据包。

一个新的物联网设备添加到这个物联网网络中，首先要与网关进行认证，反之亦然，以建立与其他环境的通信。我们假设

物联网设备和网关都有数量有限的非易失性存储，以EEPROM的形式，可以存储共享密钥。我们进一步假设有限的手动配置是可行的，因此管理员可以在成对的基础上为网关和物联网设备设置共享密钥。我们假设攻击者没有对设备的物理访问权，因此无法访问存储在设备上的配置的密钥。

B. 攻击场景

在这一节中，我们考虑了进入链路的攻击者可能寻求利用我们设计的协议中的漏洞的一系列可能方式。然后，我们描述（简要地）这些攻击是如何被我们在本文中描述的协议所阻止的。

1）攻击者可能试图嗅探或修改会话信息。

2. 攻击者可能试图通过窥探认证和密钥交换流量来发现秘钥。
3. 攻击者可能试图修改或欺骗认证和密钥交换信息，以造成干扰。
4. 攻击者可能试图发起重放攻击以造成破坏。
5. 攻击者可能会发起中间人攻击，在未经授权的情况下访问机密信息，并可能改变它们。

   我们相信，我们的协议旨在挫败所有这些攻击，我们已经对该协议进行了安全分析（在第五节简要描述），以验证这一点。

IV. 轻量级认证和密钥交换协议

如前所述，对于每个网关-物联网设备对，我们维护两个共享秘钥：一个共享的长期对称主密钥，称为Km；以及第二个共享的短期对称会话密钥，称为Ks。最初，Km和Ks被手动插入两个设备的非易失性存储中，如EEPROM。手动添加的Ks的具体初始值用Kiks表示。每个物联网设备和网关对都有一对独特的Km和Kiks，专门用于它们的使用，Km和Kiks在我们的协议中的认证中发挥着重要作用。物联网设备和网关使用用共享Km加密的信息和用共享Kiks散列的信息进行认证，而两个节点都使用共享Ks在会话中加密和散列信息。该协议利用认证加密，即Counter with CBC-MAC（CCM）区块

密码模式[4]，[11]，以确保所有阶段的保密性和消息认证。在CCM模式下，CBC-MAC用于使用对称秘钥（Km或Ks，取决于阶段）计算整个帧（头、非ce和有效载荷）的消息认证码（MAC），而Counter模式用于使用非ce和对称秘钥（Kiks或Ks，取决于阶段）对有效载荷以及MAC进行加密，而头需要是明文，以便另一端可以了解帧的一些信息，以便处理它，如MAC地址。当上一个会话期间交换的会话帧的数量达到预设的阈值（由双方事先商定）时，一个新的会话开始。

Ks为每个会话定期更新，并且从不在网络上明确交换。此外，当更新Ks时，使用以前的Ks值，所以Ks的值是累积的，这意味着新的会话密钥Ks取决于以前的会话的Ks。

我们提出的协议利用了常规框架中的随机性，并利用它来产生更多的随机会话密钥。两个实体通过将一组随机选择的未来会话帧纳入新生成的Ks来整合随机性。每当Ks更新时，这组未来的会话帧被同意并使用另一个秘密密钥Km进行安全通信。由于两个实体都知道帧的阈值，可以计算出直到下一次Ks更新的帧序列号的范围。使用一个随机数发生器，网关在这个范围内选择一组随机的未来帧序列号，以便两个实体在缓冲区内只保留这些帧。对于每个会话，选择的帧序列号的数量越大，新的Ks就越随机。这种随机性为Ks的交换提供了更好的安全性。然而，我们注意到，物联网设备的内存有限，这可能会限制它们可以缓冲的会话帧的数量。

因此，在每一个新的会话中，攻击者就更难跟上Ks的更新。为了违反保密性，攻击者需要同时拥有Km和Ks。虽然很难有一个完美的信道来捕获所有的帧[12]，但我们不假设攻击者是否有一个完美的信道来捕获物联网设备或网关发送的所有帧的能力。当任何一端的Ks因任何原因出现同步失败时，无论是恶意的还是非恶意的，任何一个设备都可以随时启动Ks的重置，将其重置为由Kiks衍生的新的随机密钥，同时还有nonces，因此他们可以一直拥有一个稳定的通信渠道。

A. 新物联网设备的初始设置阶段

当物联网设备首次加入网络时，物联网设备和网关所需的初始设置是相互认证和协商Ks。这是用共享的Km和已经手动安装的Kiks完成的。

首字母缩写词定义

Kiks Ks

ID RandFrmSeqsi (i-1) i

RandFrmSeqsi+1 i (i+1)

RandFrmi 基于RandFrmSeqsi的上一个会话（会话（i-1））的随机帧。

KeyLength 派生密钥的期望长度

表一: 术语

进入网关和新的物联网设备。图2显示了协议的互动。

物联网设备网关

IDI，"初始化"，$E\#\$$（Nonce1）。

$E\#\$(MAC\#()$(消息*1))*

信息*1*

IDG, $E\#\$(Nonce2,$

RandFrmSeqsi+1, Nonce1）。

$E\#\$(MAC\#()$(信息2))

信息2

物联网设备IDI, $E\#\$$(h(Nonce2 ||验证RandFrmSeqsi+1))。

网关。 $E\#\$(MAC\#()$(信息*3))*

信息*3*
闸道
IDG, E#$('Ack') , 验证*IoT*
E#$(MAC#()(Message 4)) 设备。
信息4
K= *HKDFK()* (Nonce1 *K*= *HKDFK()* (Nonce1 ||
|| Nonce2, Nonce2, KeyLength）。
KeyLength）。
会话信息会话信息交换开始交换开始
图2：初始设置阶段
a) 消息1：物联网设备将其IDI、"初始化"、Nonce1和MACKiks（消息1），用Km加密后发送给网关。为了证明其身份，认证者，也就是此时的物联网设备，提供了一个新的挑战，Nonce1，这将被申请者Gateway使用。此外，使用Kiks来计算整个消息的MAC，包括Nonce，可以验证数据的完整性和消息的真实性，从而有助于防止以某种方式确定Km的攻击者可能造成的破坏。
b) 消息2：在收到物联网设备的第一个消息后，网关首先选择一组未来帧的随机序列号RandFrmSeqsi+1，以便在下次更新Ks时考虑。这与Nonce2（网关挑选的新挑战）和Nonce1一起被传达给物联网设备，作为消息2的一部分。Nonce2、RandFrmSeqsi+1和Nonce1是用主密钥加密的，以便与物联网设备进行保密交换。因此，当这个会话开始时，物联网设备将保留一份具有这些序列号的帧的副本，以便在下一次Ks更新时使用。为了让物联网设备认证网关，同时确认它收到了正确的RandFrmSeqsi+1集，网关还计算了整个消息的MAC，包括使用Kiks作为密钥的Nonce1和RandFrmSeqsi+1，并将其发送给物联网设备。因此，当网关使用Kiks计算相应的MAC时，物联网设备可以验证网关有正确的Kiks并正确接收了Nonce1。网关发送的所有参数，IDG，加密的Nonce2，连同RandFrmSeqsi+1，以及Nonce1都是用Km加密，用Kiks散列。

c) 消息3：当物联网设备收到消息2时，它验证刚刚收到的MAC的哈希值与它根据使用Km的解密消息计算的值相匹配。物联网设备将网关标记为已认证，并确认物联网设备已收到正确的RandFrmSeqsi+1集。此外，物联网设备验证Gateway有正确的Kiks，并且已经从消息1中正确地接收了Nonce1，从而验证了Gateway。
物联网设备计算Nonce2的哈希值和
h(Nonce2||RandFrmSeqsi+1)和MACKiks(Message 3)被传送给Gateway，以便它认证物联网设备，并确认物联网设备正确接收了RandFrmSeqsi+1。物联网设备发送IDI，以及使用Km对h(Nonce2||RandFrmSeqsi+1)和MACKiks(消息3)进行加密。
d) 消息4：当网关收到消息3时，它验证了MAC。如果有效，网关将物联网设备标记为已认证，并且它知道物联网设备已收到正确的RandFrmSeqsi+1集。这也让网关知道，物联网设备有正确的Kiks，并且已经从消息2中正确收到Nonce2。网关使用Km向物联网设备发送IDG、加密的ack和MACKiks（消息4），以确认网关正确接收了消息3。网关现在将Ks设置为从Nonce1和Nonce2衍生的随机密钥，用作盐的输入（随机值），使用Kiks作为HKDF密钥。
当物联网设备收到消息4时，它验证MAC。如果有效，物联网设备知道消息3被正确接收。物联网设备将Ks设置为由Nonce1和Nonc+e2得出的随机密钥，使用Kiks作为HKDF密钥。
B. 网关和物联网设备之间的正常通信
在物联网设备和网关之间的正常通信期间，两个设备已经商定了一个Ks，并且它们使用Ks

安全地交换消息。在这个阶段，两个设备都会保留对应于当前RandFrmSeqsi+1的帧，用于下一次Ks更新，直到达到交换帧的阈值。不仅对手不知道RandFrmSeqsi+1，也有可能不是对手收到的所有会话帧。物联网链接层协议，如IEEE 802.15.4，也能够保护数据的保密性和完整性。因此，我们的协议为链接层提供了用于此目的的Ks。图3中显示了一个近似的会话互动。正常的会话帧用Ks进行加密和散列，然后发送到另一端。一个用Ks加密和散列的ack，由接收方发送，以确认正确收到一个帧。当交换的帧达到阈值时，我们进入下一个阶段，更新K