无线传感器网络远程代码更新技术研究进展

况晓辉 许 飞 刘 丽
(北京系统工程研究所信息系统安全技术国家级重点实验室 北京 100101)

摘 要 高可用性是无线传感器网络的设计目标之一,远程代码更新可增加或更新运行在传感器节点上的软件,是提高无线传感器网络可用性的重要支撑。分析和总结了传感器网络远程代码更新研究领域的研究成果,阐述了主要远程代码更新机制及其待解决的问题,最后探讨了今后应研究的问题,指明了下一步研究的重点和难点。

关键词 无线传感器网络,综述,远程代码更新,网络编码,安全
中图法分类号 TP393 **文献标识码** A

Survey on Remote Code Update for Wireless Sensor Networks

KUANG Xiao-hui XU Fei LIU Li
(Science and Technology Laboratory on Information System Security, Beijing Institute of System Engineering, Beijing 100101, China)

Abstract Remote Code Update in wireless sensor networks is an important means of remote task reallocation, software update of nodes and network function reconfiguration after a wireless sensor network is deployed. It's very important to improve the availability of wireless sensor networks. This paper summarizes and concludes the existing research of remote code update in wireless sensors, some main mechanism of remote code update are introduced in this paper and their defects are pointed out. At last, the problem in future research is discussed, furthermore, the keystone and difficulty of research in wireless sensors networks are indicated.

Keywords Wireless sensor network, Survey, Remote code update, Rate-less code, Security

1 前言

随着通信技术、嵌入式技术和传感器技术的迅速发展和日趋成熟,具备通信能力、计算能力和感知能力的微型传感器节点开始在世界范围内涌现。数目众多的传感器节点协同工作,它们随机分布于监测区域周遭环境,通过自组织的无线通信方式构成传感器网络(wireless sensor network, WSN)[1]。传感器网络具有易部署、自组织、高容错、可靠性等特点,在国防军事、环境监测和预报、智能家居、建筑物状态监控、空间探索、医疗卫生、城市交通等诸多领域有着广阔的应用前景,已成为无线网络领域十分活跃的研究方向,研究人员对其开展了大量的研究工作,从不同角度对研究进展进行了综述[2-10]。

随着传感器节点技术、无线通信技术的发展,传感器网络逐渐从研究走向实用。传感器网络一旦部署,将会运行相当长的时间。然而,很多应用场景如环境监测、战场感知等无法实现对物理节点的抵近式更新。因而,迫切需要远程增加或更新运行在传感器节点上的软件,WSN远程代码更新问题逐渐成为传感器网络技术的研究热点。

由于传感器网络节点资源受限、通信带宽低等特点,WSN远程代码更新面临许多挑战。研究人员围绕WSN模型、更新需求开展了大量的研究工作,但其缺乏已有研究工作的全面分析和梳理。本文在描述WSN远程代码更新需求的基础上,对当前研究进展进行了综述,然后分别从基本远程代码更新机制、高效的更新机制以及安全的更新机制3个方面描述了典型的研究工作,并对已有工作进行了分析比较。最后指出了下一步研究的重点和难点。

2 远程代码更新的需求

远程代码更新能力是提高网络可用性和可维护性的重要手段,也是提高网络扩展性的基础。早在1978年,分布式传感器网络论坛提出的3个关键功能需求中就包含动态更新以及将新的软件版本集成到运行系统中的需求[11]。近年来,远程代码更新问题逐渐成为研究热点,远程代码更新的需求也逐渐明确[12-14]。

传感器网络远程代码更新指通过无线方式,用新的软件代码替换传感器网络节点已有代码的机制。由于低带宽、高延迟、节点资源受限等特点,WSN远程代码更新具有自身独特的需求。主要包括:

- 覆盖完备性:节点接收的软件镜像必须完整,且更新须覆盖网络中所有的节点。
- 高效性:对于节点的内存、存储资源和网络通信带宽需求尽可能小,对网络的生命周期和基本功能影响尽可能小。
- 资源敏感性:适用于资源有限的传感器网络,尽可能降低额外的处理和传输开销,支持异步链路和不稳定链路,能够扩展支持大规模、高密度的网络。
- 高安全性:不安全的远程代码更新机制将对传感器网

况晓辉(1975-),男,博士,副研究员,CCF会员,主要研究方向为无线网络、信息安全;许 飞(1981-),男,硕士,助理研究员,CCF会员,主要研究方向为无线网络,E-mail:xufei1023@126.com;刘 丽(1978-),女,硕士,助理研究员,主要研究方向为信息安全。

络的安全性带来巨大的安全风险。更新机制需满足完整性、鉴别和正确性，即确保更新来自可信源，且未被修改，只有合法的节点能够更新代码，以够抵御来自内部和外部节点的攻击。

·高鲁棒性：可诊断和处理应用、操作系统、网络的失效，容忍硬件和软件故障，监视系统状态等。

3 WSN 远程代码更新研究综述

WSN 远程代码更新问题研究可分为更新代码镜像压缩机制和远程代码分发协议两个方面。前者研究如何使得待分发的代码尽可能小，后者研究如何将远程代码高效地分发给传感器节点。已有的研究工作通常同时涉及到了两个方面，其中远程代码分发协议是研究重点。

到目前为止，WSN 远程代码更新机制的研究大致可分为两个阶段。第 1 阶段的研究重点是如何在资源有限、连接不可靠的传感器网络中实现远程代码更新，典型的研究成果包括 Deluge、Infuse 等；第 2 个研究阶段在前期工作基础上，重点研究远程代码更新的效率和安全性问题，典型的研究成果包括 Rate-less Deluge、Seluge 等。其总体脉络如图 1 所示。

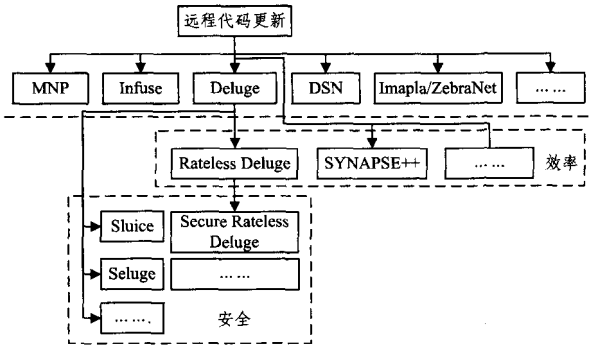


图 1 远程代码更新问题的研究脉络

3.1 基本的远程代码更新机制

围绕如何在高延迟、低带宽的 WSN 中实现远程代码更新，研究人员提出了 Deluge、DSN、Imapla、Infuse 等，其中以 Deluge 最为典型。

3.1.1 Deluge

Deluge[15] 是开源的代码分发系统，运行于 Mote 平台 TinyOS 之上。它利用增量更新方式，采用逐页分发策略，代码镜像首先按顺序被分割为固定尺寸的页，然后按序进行分发。当更新页被完整接收后，节点广播新的更新页可用，并可根据请求传送相应的报文。另一方面，接收方仅在前一页的所有报文均接收后，才请求新的页。

Deluge 使用 epidemic 协议有效地广播代码元数据。每个节点周期性地广播其代码镜像的版本以及该版本接收到的更新页的数量。为了提高节点能效，广播频率可动态调整。如果节点发现其通告频率与其他节点不同，则提高通告的频率；否则，将降低频率。因此，Deluge 能够实现快速有效的分发镜像。一旦节点通过通告报文发现邻居节点有其需要的更新页，则使用 SNACK 报文请求传输。每个 SNACK 报文包含请求页号以及所需页的位向量。根据所接收的同一个页的多个请求报文，节点计算请求报文集合，并采用轮询机制传输报文。

Deluge 采用多种消息抑制机制。为了降低冗余通告，每个节点若接收到的包含相同信息的广播报文的数量超过预先定义的门限，则抑制其通告的发送。此外，如果节点监听的请求（数据）报文与其相同，或者已经接收到，则抑制其请求报文。类似地，若节点监听到请求（数据）报文涵盖了其拟发送的报文，则节点抑制后续报文的传输。通过使用上述抑制机制，Deluge 提高了代码分发的效率。

该协议不支持异构网络，镜像将被发送给所有的节点。

3.1.2 部署支持网络（DSN）

部署支持网络（DSN）[16] 通过并行地部署一个维护性网络，来支持软件更新。在软件更新中分别访问每一个节点不可行有以下两方面原因：首先是节点数量众多，其次是节点的可访问性问题。而通过传感器网络本身进行软件更新也存在以下不足：它依赖正在运行的网络，将对传感器网络的性能造成影响，其次消耗节点的能量。而 DSN 是一个小型的、移动且临时部署的网络，可支持对传感器网络节点高效、低干扰的代码更新。

3.1.3 Imapla/ZebraNet

Impala[17] 是 ZebraNet 无线传感器网络的中间件层，该传感器网络用于跟踪野生动物。Impala 提供基于事件的中间件层，它通过应用适配器实现应用的动态更新和适配。事件由事件过滤器处理，并通过应用适配器发送给特定的应用。Zebranet 节点需广泛部署在系统管理员无法直接访问的地域，为支持应用，Zebranet 需要支持节点移动性、有限的网络带宽以及大范围的更新（从 bug 更新、软件更新到增加或删除应用等）。应用由多个、可共享的模块构成，并被分割成若干个 2k 的块。应用更新器使得应用可在更新过程中连续运行，且可同时支持多个更新，版本号用于确保更新软件与已有模块的兼容。同时，Zebranet 支持不完全更新，并在更新前提供一组简单的完整性校验机制，其软件更新包含 3 个步骤。首先，节点间交换应用模块的索引号，然后通过点播请求（使用节点 ID 寻址）更新模块，最后节点通过传输相应的模块响应其他节点的请求。如果所有节点均拥有相同的软件版本，则采用二进制回退机制避免冲突，该机制虽然可降低管理代价，但是提高了更新延迟，尤其是两个版本不同的网络互联时，更为明显。如果内存空间不足时，节点将删除老版本。当软件接收完成后，经过简单的一致性检验，将终止老版本应用，新版本模块将被加载，新应用将优先初始化。

3.1.4 Infuse

Infuse[18] 是基于 TDMA 的块数据分发协议，主要用于位置感知传感器网络。节点周期性地选择前驱和后继节点；通过有选择地监听策略降低能量开销，在其他 TDMA 时隙停止接收，并且有选择的使用前驱和后继也避免了广播风暴。基站广播的 start-download 消息中包含版本 ID 和新数据序列的 capsules 数量。然后，在后续的时隙中发送这些 capsules。接收方转发 start-download 消息，当接收到数据模块时，则将其存入 flash 中并转发。当收到最后一个 capsule 后，通告上层应用下载完成。该协议讨论了两种不同的恢复机制：基于隐含确认的 Go-Back-N 机制，以及基于显示捎带确认的 Selective-Retransmission 机制。通过前驱节点选择优化，可进一步减少能量开销。通过采用 TDMA 通信机制，infuse 比

Deluge更为有效，所有传感器节点几乎同时收到数据 capsules，避免了 CSMA 冲突。

3.1.5 MNP

MNP[19]是针对运行 TinyOS 的 MICA2 平台，使用 XNP boot loader 的传感器网络远程代码更新协议，包含以下 4 个步骤：

1)广播/请求阶段：发送源广播新的代码版本，所有感兴趣的节点发送请求。节点监听广播和请求消息，决定是否进行代码转发(该抑制策略避免网络拥塞)。

2)转发/下载阶段：源节点广播 StartDownload 消息并通告接收者，然后发送软件代码(根据报文大小进行分段)，接收方将其存入外部内存中(EEPROM)——没有确认机制，接收方在 EEPROM 中保存一个遗失段列表。

3)查询/更新阶段：源节点向所有接收方广播 Query 消息，接收方根据遗失段列表点播发送响应消息，然后源节点重新广播遗失段的报文，接着广播 Query 消息，直到没有遗失段请求消息。接收方在收到完整的镜像后，成为源节点。

4)重启阶段：在源节点不再接收到重传请求后，新的程序镜像将被传送到内存中，节点将重新完成代码更新。

节点发送 download request 给所有发送者，有助于发送者选择，同时减少了隐藏节点问题(因为潜在的发送者能够处理请求)。发送方选择算法力求在邻居节点中仅允许一个活跃的发送方。流量控制是基于速率的，由 EEPROM 写速度决定(MICA2 mote)。

3.1.6 MOAP

MOAP[20]是针对运行 TinyOS 的 MICA2 mote 平台的传感器网络的多跳、无线代码分发机制。它使用存储转发机制提供“波纹”模式的更新；遗失段由接收方通过滑动窗口标识，并使用点播消息发送请求以避免多个副本；采用 keep-alive 定时器从无应答点播重传消息中恢复，此时采用广播发送重传请求。基站广播 publish 消息，其中包括版本号。接收节点依据自己的版本号，发送更新请求 subscribe 消息。为避免不稳定链路，MOAP 采用链路统计机制。当等待周期结束，接收到所有的 subscribe 消息后，发送方开始进行数据发送。遗失段请求直接发送给发送方，且其优先级高于数据发送。一旦节点接收到所有镜像，它将成为一个发送者。如果发送方没有接收到 subscribe 消息，则将新的镜像从 EPROM 拷贝到内存中，用新代码重启。滑动窗口确认机制减少了能量消耗(减少了 EEPROM 读操作次数)，但是损失了节点乱序容忍能力。MOAP 不支持速率控制以及多发送源抑制(除了链路统计外)。

3.1.7 Trickle

Trickle[21]作为全网代码更新服务在 TinyOS/Mate 平台上运行。如果节点没有监听到一定数量的代码更新消息，则利用维护算法周期性地广播代码摘要信息。如果一个节点检测到需要更新的消息，则通过广播所需要的代码使得邻居节点及时更新。Trickle 动态调整每个节点的更新周期和流量到一个特定的频率($rx+tx$)，从而能够自动适应本地网络的密度，且在报文丢失的情况下也具有良好的扩展性。监听周期用于最小化短监听问题(未同步的节点由于定时器周期不同，可能引起冗余传输)。由于传输速率被控制在较低的水平，CSMA 隐藏节点问题不会导致过多的异常行为。通过动态修改 gossip 周期，即使在带宽有限的情况下，Trickle 也能够快速地传播代码更新。

3.2 高效的远程代码更新机制

在上述工作基础上，为进一步提高远程代码更新的效率，研究人员提出了利用无比率编码提高更新的可靠性，降低带宽开销和延迟。有代表性的研究工作包括 Rate-less Deluge、SYNAPSE++等。

3.2.1 Rate-less Deluge

在 Deluge 基础上，Rate-less Deluge[22]是一种采用 Rate-less code 编码方式提高传感器网络代码分发效率、降低重传报文数量的远程代码更新机制，分为 Rate-less Deluge 和 ACK-less Deluge 协议，二者均采用 Rate-less 编码方法，取代了 Deluge 的数据传输机制。实现和模拟仿真方法均充分表明，在高密度传感器网络中更新的效率相对于 Deluge 有明显提升。

无比率编码针对传感器网络带宽有限、可靠性不高的链路，提供了有效的输出传输手段。其基本思想是接收方不需要重传特定的报文，而是接收到一定数量的不相关报文后，即可解码获得原始消息。无比率编码的优势在于通信和能量开销小，节省了传输延迟。

随机线性编码适用于文件的分发。在模型中，长文件 X 被分割为 k 段，即 $X_1, X_2, \cdots, X_k$，然后利用下述公式编码成 m 个消息 $\{Y_1, Y_2, \cdots, Y_m\}$，$m>k$。

$$Y_i=\sum_{j=1}^{k}\beta_{i,j}X_j$$

其中，$\beta_{i,j}$是优先域 F 中随机选择的参数，以确保 $\beta_{i,j}, \cdots \beta_{i,k}$ 线性独立。从而使得任何节点接收到 k 个消息后，均可通过线性方程得到 X，如图 2 所示。

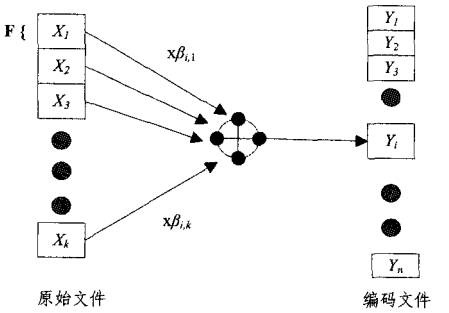


图 2 随机线性编码示意图

该技术有两个好处，解码效率很高，运算和能量开销小。其次，无比率编码具有很好的可扩展性，一旦发现由于链路状况太差，m 不足以使节点恢复数据时，可额外生成新的消息。

3.2.2 SYNAPSE++

SYNAPSE++[23]是用于无线传感器网络远程代码更新的系统。与已有工作不同，SYNAPASE++采用喷泉码(Fountain Code)进行远程代码分发，同时采用 negative acknowledgment-based ARQ 以及错误恢复机制进一步提高代码传输效率。同时，SYNAPASE 采用新的 boot loader 和内存管理模块支持任意语言开发的镜像加载。采用 Fountain Code，发送方可将 K 份资料编码后，如泉水一样不断地流向接收方，无论接收过程中漏水(丢包)程度，接收方如装水一样收满 N(略大于 K)份即可还原原始资料，其可抵御可变丢包

率的特性称为 Rate-less code。与采用 Rate-less code 的代码分发机制不同，SYNAPASE++采用基于 GF(2)的随机编码方法，仅需执行按位异或操作即可完成报文编码解码，在保持传输延迟和恢复能力的前提下，具有更高的解码效率，有效降低了传感器节点的运算开销。

3.3 安全的远程代码更新机制

随着传感器网络逐渐走向应用，安全性成为日益关注的重点问题。远程代码更新的安全问题成为研究的新热点。为适应传感器网络节点运算能力有限的特点，远程代码更新的安全机制主要采用 hash、对称加密等机制实现更新的可鉴别、完整性保证。有代表性的研究工作包括 Sluice、Seluge 以及 Secure Rate-less Deluge 等。

3.3.1 Sluice

已有的传感器远程代码更新机制主要关注效率、可靠性和多跳支持，同时假设所有节点均正确或故障终止。恶意节点能够利用代码更新协议远程安装恶意代码，破坏整个网络的运行。Sluice[24]是一种基于单向 hash 链的远程代码安全更新机制，其目标是实现安全可靠的无线传感器网络代码更新，确保恶意代码更新不能够正常传播或安装，同时正确的更新能够有效分发。该机制通过 hash 链提供鉴别和完整性保障，并有效降低了在整个更新过程中单个数字签名的代价，其基本思路如图 3 所示。

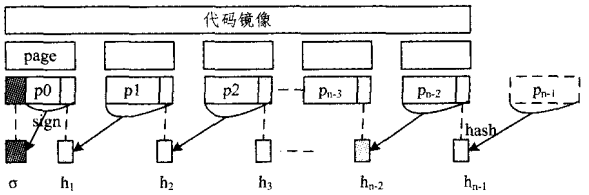


图 3 基于单向 hash 链的安全更新机制

Sluice 采用数字签名和单向 hash 链相结合的机制实现更新页的持续验证，该机制仅需要进行一次数字签名。其基本思想是通过计算每个更新页的 hash 值，并将每一页的 hash 值作为前一页的一部分，形成 hash 链，并对第一个更新页进行数字签名。

3.3.2 Seluge

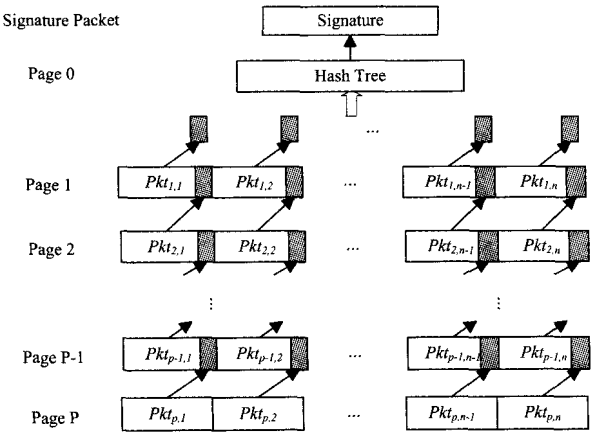


图 4 基于 hash 树的代码分发鉴别机制

在 Deluge 基础上，Seluge[25]为了提高代码更新的安全性，增加了 3 个层次的防护。1)代码分发报文的直接鉴别；2)页通告和 SNACK 的鉴别；3)抵御对签名报文的 Dos 攻击。

· 镜像的分页与鉴别

与 Sluice 采用 hash 链进行鉴别不同，Seluge 采用 hash 树进行鉴别。图 4 示出 Seluge 代码分发过程中的鉴别机制。

首先将代码镜像分割为 P 个更新页，每个更新页进一步划分为 N 个固定尺寸的报文，用 $Pkt_{i,1}$ 到 $Pkt_{i,n}$ 表示。然后将更新页 P 中每个报文的 hash 存入更新页 $P-1$ 的相应报文中，并采用 Merkle hash 树实现对更新页 1 中报文的 hash 镜像的鉴别。同时，将与 Merkle hash 树关联的报文定义为更新页 0。页 0 及其报文的构造方式如图 5 所示。

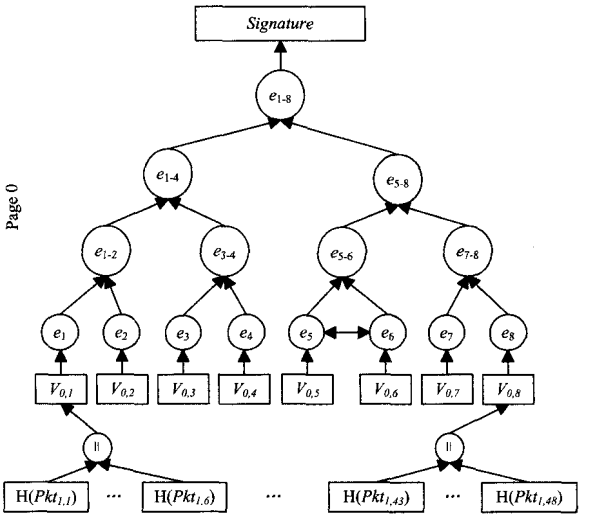


图 5 页 0 的构造流程

将更新页 1 的所有报文的 hash 值合并，$HashValues=H(Pkt_{i,1})\|\cdots\|H(Pkt_{i,n})\|$，然后将 $HashValues$ 分割为 M 份，$M=2^k$，其中 k 满足

$$\frac{N\cdot|H(\cdot)|}{2^k}+k\cdot|H(\cdot)|\leqslant \text{Maximum } payloadsize \tag{1}$$

该公式确保每个叶节点及其认证路径可通过一个报文发送。分割的部分用 $V_{0,1},V_{0,2},\cdots,V_{0,m}$ 表示，Merkle hash 树用 $V_{0,1},V_{0,2},\cdots,V_{0,m}$ 作叶节点。上图显示了 $N=48$，$M=8$ 时的 Merkle hash 树。其中 $e_i=H(V_{0,i})$，$i=1,2,\cdots,m$，并通过临近子节点构建二叉树的内部节点。每个内部节点是两个子节点的 hash 值。例如，$e_{1-2}=H(e_1\|e_2)$，$e_{1-4}=H(e_{1-2}\|e_{3-4})$。

采用 Merkle hash 树，构建 M 个报文，即为每个 $V_{0,i}$ 构建一个报文 $Pkt_{0,i}$，其中 $i=1,2,\cdots,m$。每个 $Pkt_{0,i}$ 包含 $V_{0,i}$ 及其 hash 树认证路径中的所有值(从叶节点一直到根)。例如，报文 $Pkt_{0,1}$ 包含 $V_{0,1}$，e_2，e_{3-4}，e_{5-8}。

将 hash 树的根节点、镜像的元数据(例如版本号、大小)以及上述信息的签名合并为签名报文。为描述方便，将更新页 1 至更新页 P 的报文称为数据报文，而更新页 0 的报文称为 hash 报文。除了已讨论的数据外，每个报文的包头中还需包含关于代码镜像、更新页以及报文的辅助信息。

参数 N 和 M 必须要确保 hash 报文足够小，以便能够通过传感器网络发送。根据 Deluge 的缺省配置，$N=48$。采用 64 位的 hash 函数。根据式(1)，$M=8$。因而，每个 hash 报文包含页 1 的 6 个 hash 镜像，3 个认证路径中的 hash 镜像，一共 72 字节，远小于 IEEE 802.15.4 标准中的 102 字节的最大数据载荷。

鉴别时，基站首先广播签名报文。在接收到签名报文后，每个节点验证签名，以鉴别更新页 0 的根。利用更新页 0 每个报文中的 hash 路径以及 root，节点可鉴别每个报文的完整性和来源。例如，在图 2 中，e_{1-8} 通过签名报文鉴别，当接收到包含 $V_{0,1}$，e_2，e_{3-4}，e_{5-8} 的报文 $Pkt_{0,1}$ 后，节点可通过 hash 路径鉴别，$H(H(H(H(V_{0,1}))\parallel e_2)\parallel e_{3-4})\parallel e_{5-8})=e_{1-8}$。如果相等，则接收，否则该报文为伪造报文，直接丢弃。

由于 hash 报文中包含了更新页 1 所有报文的 hash 镜像，接收后，可直接用于鉴别更新页 1 的报文。在上述例子中，$V_{0,1}=H(Pkt_{1,1})\parallel\cdots\parallel H(Pkt_{1,6})$。因此接收到更新页 1 中的报文后，可利用 hash 镜像进行鉴别。

· 通告和 SNACK 报文的鉴别

对于通告和 SNACK 报文，Seluge 采用基于 cluster keys 的局部广播鉴别机制。每一个节点生成一个 cluster key，用于鉴别其发出的所有通告和 SNACK 报文。当节点部署时，通过周期发送 hello 报文通告其邻居节点。假设传感器节点可利用已有的机制与其邻居节点建立密钥对。当从邻居节点接收 hello 报文后，每个节点在随机延迟后利用私有密钥发送 cluster key。新的节点也将其 cluster key 发送给所有邻居节点。对于每个将要发送的页通告或 SNACK 报文，发送方添加唯一的序列号（防止重放攻击），并用其 cluster key 签名。报文接收方利用发送方的 cluster key 鉴别其完整性。该方法不能唯一标识发送方，因此内部恶意节点可利用其邻居节点的 cluster key 伪造通告或 SNACK 报文。

· 抵御针对签名报文的攻击

为抵御对于签名机制的拒绝服务攻击，避免通过伪造大量签名报文迫使节点执行签名验证操作，从而导致能量耗尽，Seluge 采用基于消息挑战码（message specific puzzles）的弱鉴别机制抵御此类拒绝服务攻击。在传感器网络部署阶段，基站通过随机选择参数 K_0，利用 hash 函数 H 生成单项 hash 链 $K_0,K_1,\cdots,K_n$，其中 $K_i=H(K_{i+1})$。然后，将 K_0 在部署前预先发布给所有节点。$K_1,\cdots,K_n$ 称为 puzzle keys，K_i 为代码镜像的第 i 个版本。

Seluge 利用 message specific puzzles 作为每个代码镜像签名报文的另一层保护。对于第 i 个版本的代码镜像的签名报文 $i\parallel M_i\parallel Sig(i\parallel M_i)$，其中 i 为版本号，M_i 为签名报文的其他部分，$Sig(i\parallel M_i)$ 为基站生成的签名。签名报文 $i\parallel M_i\parallel Sig(i\parallel M_i)$ 和 K_i 组成 message specific puzzles，基站需要解决的关键问题是找到参数 P_i，使得利用 hash 函数 H 计算 $i\parallel M_i\parallel Sig(i\parallel M_i)\parallel K_i\parallel P_i$ 得到的值前 m 位均为 0，如图 6 所示。

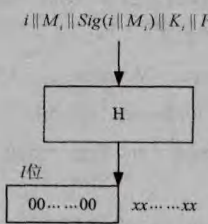


图 6　基于消息挑战码的签名保护机制

当接收到签名报文后，节点首先利用 H 和 K_i 对报文进行验证。仅验证成功（即 hash 后的值中前 m 位均为 0）才对签名进行验证，否则简单丢弃该报文。

puzzle 的验证计算代价很小，但是得到 puzzle 的值只有通过暴力破解的方法实现，而且外部攻击者无法预知当前的 puzzle key，在获得当前的 puzzle key 后、通过计算得到 puzzle 前，镜像分发过程已经结束了。因此该机制能有效抵御针对签名的 Dos 攻击。

3.3.3　Secure Rate-less Deluge

Secure Rate-less Deluge[26] 是对 Rate-less 的安全增强，采用 Merkle Hash 树和签名机制可提高分发过程的完整性、可鉴别。

在代码更新过程中，基站采用 Merkle hash 树机制计算页 hash 和根签名报文，如图 7 所示。在分发过程中，基站首先通过广播方式发送页 hash 报文和根签名报文。然后，每个更新页采用无比率编码方式分发，节点在接收到足够的报文后，恢复更新页，利用页 hash 报文中的数据进行鉴别。

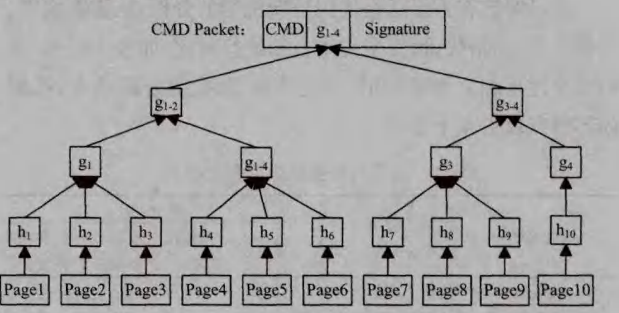
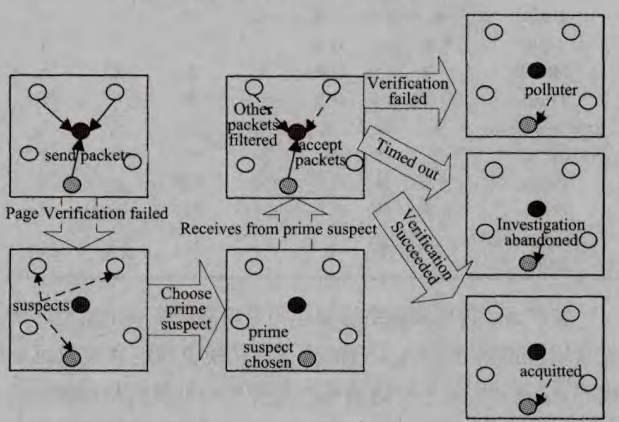


图 7　基于 Merkle hash 树的代码鉴别机制



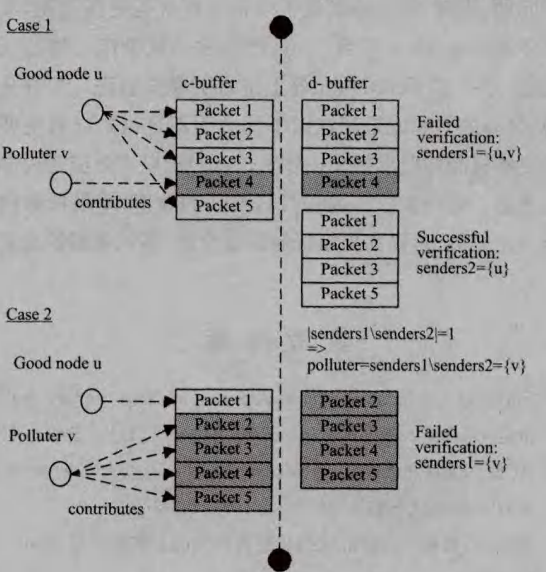


图 8　内部恶意节点识别机制

在解码过程中，每个报文均统计其来源。Secure Rateless Deluge采用正向和逆向检测方式来识别恶意内部节点，如图8所示。

在正向检测过程中，节点对接收到的报文进行鉴别。当鉴别失败时，替换某个节点发送的报文，若此时鉴别成功，则说明被替换的报文是恶意报文，其发送方为恶意节点，如图8示例1所示，发送方 u、v 的报文鉴别失败，而将节点 v 发送的报文替换为节点 u 发送的报文，则鉴别成功，表明 v 为入侵者。逆向检测与正向检测相反，仅选择某个发送方的报文进行鉴别，若鉴别失败，则说明该节点为恶意节点，如图8示例2所示，节点 v 发送的报文无法通过鉴别，表明 v 为恶意节点。但是，逆向检测需要节点发送的报文超过可还原的下限。

3.4 分析比较

通过对已有主要远程代码更新机制的分析，从更新效率、部署代价、通信代价、安全性等方面进行对比，如表1所列，其中安全性可细分为可鉴别、完整性保证、抵御外部攻击、抵御内部节点攻击4个方面。

表1 远程代码更新机制对比分析

机制名称	更新效率	部署代价	通信代价	安全性			
				可鉴别	完整性保证	抵御外部攻击	抵御内部攻击
Deluge	较高	低	较小	无	无	无	无
DSN	较高	高	较小	无	无	无	无
Imapla	中等	较低	较高	无	无	无	无
Infuse	高	较低	较低	无	无	无	无
MNP	中等	较低	中等	无	无	无	无
MOAP	中等	较低	较高	无	无	无	无
Trickle	较高	较低	较高	无	无	无	无
Rate-less Deluge	高	低	低	无	无	无	无
SYNAPSE++	高	低	低	无	无	无	无
Sluice	较高	低	较小	支持	支持	无	无
Seluge	较高	低	较小	支持	支持	支持	无
Secure Rate-less Deluge	高	低	低	支持	支持	支持	支持

结束语 传感器网络远程代码更新问题是一个新型的研究领域，逐渐得到研究人员的重视。从镜像压缩和高效远程代码更新机制、基于网络编码的远程代码更新机制到解决其安全问题，研究、提出并实现了多种远程代码更新机制，其中绝大多数均在Mote平台TinyOS构建的试验床上进行了实验验证，并在TOSSIM模拟器上进行了模拟验证。综合分析发现，已有的研究成果存在以下两方面不足：(1)已有主要的代码更新机制均要求报文按序发送；(2)难以有效应对内部节点的攻击。进一步提高远程代码更新的效率，增强抵御内部节点攻击的能力，提高更新机制的安全性，是该领域的研究重点。

参 考 文 献

[1] Estrin D, Govindan R, Heidemann J, et al. Next century challenges: scalable coordinate in sensor network [C]// Proc. of 5th ACM/IEEE Int' conf on Mobile Computing and Networking. Washington, USA: ACM Press, 1999: 263-270

[2] 任丰原，黄海宁，林闯. 无线传感器网络[J]. 软件学报，2003(7)：98-107

[3] 马祖长，孙怡宁，梅涛. 无线传感器网络综述[J]. 通信学报，2004(4)：114-124

[4] 崔莉，鞠海玲，苗勇. 无线传感器网络研究进展[J]. 计算机研究与发展，2005(1)：163-174

[5] 裴庆祺，沈玉龙，马建峰. 无线传感器网络安全技术综述[J]. 通信学报，2007(8)：113-122

[6] 李建中，高宏. 无线传感器网络的研究进展[J]. 计算机研究与发展，2008(1)：1-15

[7] 李仁发，魏叶华，付彬，等. 无线传感器网络中间件研究进展[J]. 计算机研究与发展，2008(3)：383-391

[8] 刘林峰，金杉. 无线传感器网络的拓扑控制算法综述[J]. 计算机科学，2008(3)：6-12

[9] 张国印，孙瑞华，马春光，等. 无线传感网络密钥管理及认证综述[J]. 计算机科学，2010，02：1-6

[10] Li Chang-le, Zhang Han-xiao, Hao Bin-bin, et al. A Survey on Routing Protocols for Large-Scale Wireless Sensor Networks [J]. Sensors, 2011, 11(4): 3498-3526

[11] Habermann A W. Dynamically Modifiable Distributed Systems [C]// Proc. of a Workshop on Distributed Sensor Nets, Pittsburgh, Pennsylvania. Carnegie-Mellon University. 1978: 111-114

[12] Han C-C, Kumar R, Shea R, et al. Sensor Network Software Update Mana-gement: a Survey [J]. Intl. Journal of Network Management, John Wiley & Sons, 2005(15): 283-294

[13] Koshy J, Pandy R. Remote Incremental Linking for Energy-Efficient Reprogramming of Sensor Networks [C]// Proc. of the 2nd European Workshop on Wireless Sensor Networks (EWSN'05). IEEE, 2005: 354-365

[14] Reijers N, Langendoen K. Efficient Code Distribution in Wireless Sensor Networks [C] // Proc. of the Second ACM Intl. Workshop on Wireless Sensor Networks and Applications (WSNA'03). ACM, San Diego, CA, Sept. 2003: 60-67

[15] Hui J, Culler D. The Dynamic Behavior of a Data Dissemination Protocol for Network Reprogramming at Scale[C]// Proc. of the 2nd international conference on Embedded Networked Sensor Systems. ACM, Baltimore, Maryland, USA, 2004: 81-94

[16] Beutel J, Dyer M, Meier L, et al. Next-Generation Deployment Support for Sensor Networks [R]. TIK-Report No: 207. Computer Engineering and Networks Lab, Swiss Federal Institute of Technology (ETH). , Zurich, 2004

[17] Liu T, Martonosi M. Impala: A Middleware System for Managing Autonomic, Parallel Sensor Systems [C] // Proc. of the ACM SIGPLAN Symposium on Principles and Practice of Parallel Progamming (PPoPP' 03). ACM, San Diego, California, USA, June 2003: 107-118

[18] Kulkarni S S, Arumugam M. Infuse: A TDMA Based Data Dissemination Protocol for Sensor Networks[R]. MSU-CSE-04-46. Dept. of Computer Science and Engineering, Michigan State University, MI, 2004

[19] Kulkarni S S, Wang L. MNP: Multihop Network Reprogramming Service for Sensor Networks[C]// Proc. of the 25th IEEE Intl. Conf. on Distributed Computing Systems (ICDCS' 05). IEEE, 2005

[20] Stathopoulos T, Heidemann J, Estrin D. A Remote Code Update

Mechanism for Wireless Sensor Networks [R]. CENS Tech. Report 30. Centre for Embedded Networked Sensing, UCLA, 2003

[21] Levis P, Patel N, Culler D, et al. Trickle: A Self-Regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks [C]// Proc. of the First USENIX/ACM Symposium on Network Systems Design and Implementation (NSDI 2004). San Francisco, CA, Mar. USENIX, 2004: 15-28

[22] Hagedorn A, Starobinski D, Trachtenberg A. Rate-less Deluge: over-the-air programming of wireless sensor networks using random linear codes [C]// Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN'08). IEEE Computer Society, 2008: 457-466

[23] Rossi M, Bui N, Zanca G, et al. SYNAPSE++: code dissemination in wireless sensor networks using fountain codes [J]. IEEE Transactions on Mobile Computing, 2010, 9(12): 1749-1765

[24] Lanigan P E, Gandhi R, Narasimhan P. Sluice: secure dissemination of code updates in sensor networks[C]// Proceedings of the 26th IEEE International Conference on Distributed Computing Systems(ICDCS '06). IEEE Computer Society, 2006: 53-63

[25] Hyun S, Ning P, Liu A, et al. Seluge: secure and DoS-resistant code dissemination in wireless sensor networks [C]// Proceedings of the 7th International Conference on Information Processing in Sensor Networks (IPSN'08). IEEE Computer Society, April 2008: 445-456

[26] Law Y W, Zhang Yu, Jin Jiong, et al. Secure Rate-less Deluge: Pollution-Resistant Reprogramming and Data Dissemination for Wireless Sensor Networks [J]. EURASIP Journal on Wireless Communications and Networking, 2011(1): 11-33

(上接第 235 页)

为新的服务被注册、发布，从而灵活而快速地解决各种类型用户特定的业务需求。例如船舶海上路线分析功能，即基于查找、统计、分析这一业务流程对相应功能级 Web 服务进行组装而成。

结束语 基于物联网、云计算和 SOA 架构技术建立的港口物流平台，从数据的实时采集和综合利用、状态的全面感知和监控以及资源和服务的按需提供等多个方面提升了港口物流平台的综合素质，为港口物流平台带来了全新的应用前景。

参 考 文 献

[1] Theo E, Notteboom, Winkelmans W. Structural changes in logistics: how will port authorities face the challenge? [J]. Maritime Policy & Management, 2001, 28(1): 71-89

[2] Almostairi B, Lumsden K. Port logistics platform integration in supply chain management[J]. International Journal of Shipping and Transport Logistics, 2009(1): 194-210

[3] RAdrian E, Chandra S, Etienne S, et al. Intelligent transport systems in multimodal logistics: A case of role and contribution through wireless vehicular networks in a sea port location[J]. International Journal of Production Economics, 2012, 317(1): 165-175

[4] Cheng Jun, Xu Yun-hai. The construction of port logistics information platform based on port supply chain [C]// E-Business and E-Government(ICEE), International Conference. 2011: 1-4

[5] Schuldt A, Hribernik K A, Gehrke J D, et al. Cloud Computing for Autonomous Control in Logistics[C]// 40th Annual Conference of the German Society for Computer Science (GI 2010). 2010

[6] Jedermann R, Behrens C, Laur R, et al. Intelligent Containers and Sensor Networks: Approaches to Apply Autonomous Cooperation on Systems with Limited Resources[C]// Hülsmann M, Windt K, eds. Understanding Autonomous Cooperation and Control in Logistics. The Impact on Management, Information and Communication and Material Flow, Springer, 2007: 365-392

[7] INFSO D. Networked Enterprise & RFID INFSO G. 2 Micro & Nanosystems, in co-operation with the Working Group RFID of the ETP EPOSS, Internet of Things in 2020, Roadmap for the Future[R]. Tech. Rep. Information Society and Media, 2008

[8] 罗军舟，金嘉晖，宋爱波. 云计算-体系架构与关键技术[J]. 通信学报，2011，32(7)：3-21

[9] Erl T. Service-Oriented Architecture(SOA): Concepts, Technoloty, and Design[DB/CD]. Prentice Hall PTR, 2005

[10] 肖亮. 基于物联网技术的物流园区供应链集成管理平台构建[J]. 电信科学，2011，27(4)：54-60

[11] Jun C, Wei M Y. The Research of Supply Chain Information Collaboration Based on Cloud Computing[J]. Procedia Environmental Sciences, 2011, 10: 875-880

[12] Hribernik K A, Hans C, Thoben K D. The application of the epcglobal framework architecture to autonomous control in logistics[M]// Dynamics in Logistics. Berlin Heidelberg, Springer, 2011: 365-374

[13] 杨志和，李业荣. 基于中间件和 RFID 技术的第三方物流 MIS 的应用研究[J]. 计算机应用研究，2006(s)：592-593

[14] Robison S. The next wave: Everything as a service[OL]. Executive viewpoint. http:// www. hp. com/ hPinfo/ execteam/articles/robison/08eaas. html, 2007

[15] 林云，田帅辉. 物流云服务-面向供应链的物流服务新模式[J]. 计算机应用研究，2012，29(1)：224-228

[16] SIM S K. IBM introduces ready-to-use cloud computing collaboration services get clients started with cloud computing [EB/OL]. http://www-03. ibm. com /press/us/en /pressrelease/22613. wss, 2012-09-09

[17] 张晓娟，易明巍. 基于云计算与 SOA 的企业集成架构及实现[J]. 计算机系统应用，2011，20(9)

[18] 何典，吴敏，胡春华. 物联网环境下负载均衡的低代价云存储数据副本分布[J]. 中南大学学报：自然科学版，2012，43(4)：1355-1361