

基于功能码深度检测的 Modbus/TCP 通信访问控制方法

万 明, 尚文利, 曾 鹏, 赵剑明

1. 中国科学院沈阳自动化研究所, 辽宁 沈阳 110016; 2. 中国科学院网络化控制系统重点实验室, 辽宁 沈阳 110016

基金项目: 国家自然科学基金资助项目(61501447); 中科院网络化控制系统重点实验室自主课题

通信作者: 万明, wanming@sia.cn 收稿/录用/修回: 2015-06-10/2015-11-05/2015-11-13

摘要

针对目前工业控制系统中 Modbus/TCP 通信协议面临的安全问题, 提出了一种基于功能码深度检测的 Modbus/TCP 通信访问控制方法. 该方法基于深度包解析 DPI (deep packet inspection) 技术, 将 Modbus/TCP 通信数据按功能码分类进行深度解析, 不仅实现了传统的功能码、地址范围的检测, 同时还实现了寄存器或线圈的域值检测. 研究表明, 通过采用“白名单”的访问控制模式, 该方法能够有效地隔离非法的 Modbus/TCP 通信数据流, 保障工控设备的安全性.

关键词

Modbus/TCP

访问控制

深度检测

功能码

中图分类号: TP393

文献标识码: A

Modbus/TCP Communication Control Method Based on Deep Function Code Inspection

WAN Ming, SHANG Wenli, ZENG Peng, ZHAO Jianming

1. Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China;

2. Key Laboratory of Networked Control System, Chinese Academy of Sciences, Shenyang 110016, China

Abstract

In order to resolve the security problem in Modbus/TCP protocol, we propose a Modbus/TCP communication control method based on deep function code inspection. By using the DPI (deep packet inspection) technology, the method deeply analyses the Modbus/TCP packets according to the different function codes. It can check not only the ordinary function codes and addresses, but also the values of registers or coils. In particular, our research results show that, by using the whitelist mode, the method can filter the malicious Modbus/TCP packets effectively, and can provide the security for the industry control devices.

Keywords

Modbus/TCP;
communication control;
deep inspection;
function code

1 引言

工业控制系统广泛应用于石油、化工、电力、交通、水利等领域, 是关系到国家社会和经济发展的关键基础设施. 在建设之初, 工业控制系统依据自我隔离的设计方案, 采用专用的通信协议、操作系统和软硬件设施, 禁止与其它外部网络的通信, 因此业界普遍认为工业控制系统的信息交互是安全的, 仅仅重点关注工业控制系统自身的功能安全和物理安全. 然而, 随着信息化与工业化进程的不断交叉融合, 工业控制系统开始采用传统互联网中的信息技术, 现有工业控制系统内部以及内部与外部之间开始互联互通, 同时由于工控设备的操作系统后门、漏洞及攻击隐患的逐渐暴露, 使得针对工业控制系统的网络攻击更容易实施, 因此工业控制系统面临的信息安全问题日益严峻^[1-3]. 根据美国国土安全部下属工业控制系统网

络应急响应小组(industrial control systems cyber emergency response team, ICS-CERT)的安全报告^[4], 近3年针对工业控制系统的安全事件呈明显上升趋势, 仅2013年度, 针对关键基础设施的攻击报告已达到257起.

Modbus/TCP协议是一种开放的工业控制通信协议, 它由国际Modbus组织在2002年进行延伸扩展^[5], 广泛应用于石油、化工、能源、电力等典型工业控制系统中, 并且由于协议本身的简单性和易开发性, 已经成为目前世界各国工业界领先的工业以太网通信协议, 并长期占据着主导地位. 然而, 正是由于Modbus/TCP协议的开放性、简单性和易开发性, 其安全隐患也日益暴露出来, 得到了国内外学者、研究人员的广泛关注, 在安全脆弱性、攻击检测与防护等方面开展了部分研究工作. 其中, 文[6-9]对Modbus/TCP在工业控制系统中的安全性进行了分析, 指出了Modbus/TCP协议中相关的典型安全问题与异常行

为,并提出了一些解决方法,但该研究更倾向于使用传统互联网络中的安全技术,从保障网络侧面的安全弥补协议自身的安全缺陷,并且缺少对 Modbus/TCP 应用协议的支持. 文[10-11]在传统 Modbus/TCP 协议中增加了健壮安全的认证机制,保证了 Modbus 主站与 Modbus 从站之间信息交互的真实性与完整性. 文[12]提出了一种改进的 DES (data encryption standard) 加密算法,使得 Modbus/TCP 通信支持加密传输,但是该方法缺少对加密时延的论证. 上述两种方法都需要对 Modbus/TCP 协议本身进行修改,实施性较差. 在 Modbus/TCP 攻击的入侵检测方面,文[13-15]分别提出了基于参数行为模型的入侵检测方法、基于状态的入侵检测方法和基于 PSO-SVM 的 Modbus/TCP 通信异常检测方法,虽然它们都能成功检测到针对 Modbus/TCP 协议的攻击行为,但是无法阻止这些攻击行为的发生,同时其准确性受到诸多因素制约,仍然存在着误报问题. 因此,在不改变 Modbus/TCP 协议规约的基础上,同时兼顾 Modbus 攻击的防护功能,Modbus/TCP 通信的访问控制方法得到了越来越多的重视,文[16]提出了使用 Netfilter/iptables 框架实现对 Modbus/TCP 通信数据的过滤控制,但是其不能按照功能码进行分类,同时缺少对寄存器或线圈域值的检测;文[17]设计了基于入侵检测规则与白名单相结合的 Modbus/TCP 区域与边界安全防护模型,该模型通过设定区域/边界规则实现域间的通信访问控制,但该模型无法进行规则的字段范围设置,域间通信规模大时规则数较多,影响访问控制性能;文[18]提出了基于状态的过滤系统,支持 Modbus/TCP 和 DNP3 (distributed network protocol version 3.0) 两种协议,根据通信数据包之间的状态信息过滤恶意攻击行为,但该系统对协议内容分析较粗略,并且需要同时分析多个数据包,实时性较差.

综上,Modbus/TCP 的安全性已经得到了一定的关注,在数据加密、认证、入侵检测与访问控制等方面形成了面向 Modbus/TCP 协议的安全检测与防护体系. 但是,针对 Modbus/TCP 安全性及其解决方案的研究仍然存在一些难点,主要表现在:第一,工业控制系统以可用性为先,针对 Modbus/TCP 的安全防护方法要保障控制通信的实时性,不应该影响系统的可用性;第二,Modbus/TCP 协议是一种成熟且应用广泛的工业控制通信协议,增加安全性不应该对 Modbus/TCP 协议规约进行大范围修改,同时应用 Modbus/TCP 协议的现场控制设备大多是嵌入式系统设备,其存储及计算资源有限,能否应用增加安全性之后的 Modbus/TCP 协议还有待考证;第三,Modbus/TCP 的安全解决不应局限于传统的网络层和传输层防护机制,而是需要对 Modbus/TCP 协议进行深度解析. 为此,综合考虑工业控制系统的可用性需求,在不改变 Modbus/TCP 协议规约的基础上,本文提出了一种基于功能码深度检测的 Modbus/TCP 通信访问控制方法. 不同于文[16],该方法基于深度包解析 DPI (deep packet inspection) 技术^[19-20],将 Modbus/TCP 通信数据按功能码分类进行深度解析,不仅实现了传统的功能码、地址范围的检测,同时还实现了寄存器或线圈的域值检测,同时采用“白名单”模式^[21],更加深层次地保

障了 Modbus 从站不被非法访问,实现了“区域隔离”的设计思想,提高了工业控制系统的安全性.

2 Modbus/TCP 协议及安全性分析

Modbus/TCP 是 OSI (open system interconnection) 模型第 7 层的应用层报文传输协议,采用典型的主从式通信模式,Modbus 主站向 Modbus 从站发送请求报文,Modbus 从站依照协议规约给出应答响应. Modbus/TCP 协议的报文格式如图 1 所示.

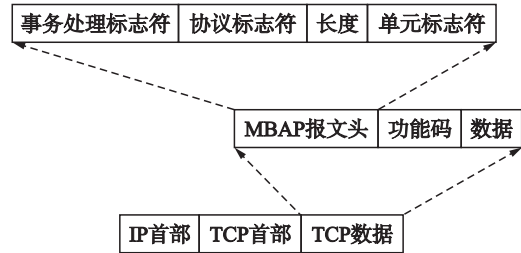


图1 Modbus/TCP 数据报文格式

Fig.1 Modbus/TCP message format

Modbus/TCP 报文主要包括 3 个部分: MBAP (modbus application protocol) 报文头、功能码和数据. 其中, MBAP 报文头是一种专用报文头,通过它能够在 TCP/IP 上识别 Modbus 应用数据单元,该部分主要包含事务处理标志符、协议标志符、长度、单元标志符四个域. 功能码是执行操作的标志字段,当 Modbus 主站发送请求报文时,功能码域通知 Modbus 从站执行哪种操作. 数据部分可以看作是功能码的参数,依赖于具体的功能码,指明了执行操作的具体数据.

Modbus/TCP 协议是基于标准以太网 TCP/IP 协议基础之上的工控应用协议,这就导致传统 TCP/IP 协议的安全缺陷被继承下来,有可能遭受到例如 DoS 攻击、IP 欺骗攻击等常见的网络攻击. 同时,由于 Modbus/TCP 协议在设计之初并没有考虑到信息安全问题,这就使得 Modbus/TCP 协议自身也存在着一些安全漏洞,容易被黑客利用,主要包括以下几点^[7-9]:

首先,Modbus/TCP 缺乏认证,任何一个接入到工业控制系统的计算机都可以使用一个不合法的 IP 地址和 Modbus 功能码即可建立一个 Modbus 会话,这就使得黑客可以扮演任意一个 Modbus 主站去恶意访问 Modbus 从站.

其次,Modbus/TCP 缺乏授权,没有基于角色权限的访问控制机制,这就使得任何一个用户都可以执行任意的功能,例如,黑客控制某一 Modbus 主站后,可以利用主站向 Modbus 从站发送执行某一功能的命令,而此功能原本是不允许该 Modbus 主站访问的.

再者,Modbus/TCP 缺乏加密,功能码、地址、寄存器或线圈域值都是明文传输,这就使得黑客很容易捕获传输数据并解析,导致工艺的机密信息泄露.

最后,Modbus/TCP 缺乏完整性检测,通信数据有可能被篡改而无法校验. 例如,黑客可以将合法的 Modbus/TCP 数据报文篡改成超长的畸形 Modbus/TCP 数据报文,

造成 Modbus 从站缓存区溢出,从而导致 Modbus 从站宕机,影响工业控制系统正常运行.

3 功能码深度检测的访问控制方法

3.1 基本访问控制模型

访问控制是工业控制系统安全防护中重要手段之一,其限制了工业控制系统中主体向客体的访问,防止对某资源的未经授权使用,保障数据资源在合法范围内的访问和管理.图2给出了基于功能码深度检测的 Modbus/TCP 通信基本访问控制模型,该模型实施于 Modbus 主站与 Modbus 从站之间,防止 Modbus 主站对 Modbus 从站的恶意攻击或误操作等行为.本模型符合工业控制系统主流的“纵深防御”体系思想^[2, 22],保障工业控制系统中不同安全区域的安全通信,通过设置 Modbus/TCP 的访问控制规则策略,管理和控制出入不同安全区域的信息流.如图2所示,协议完整性检测判定数据报文是否符合 Modbus/TCP 协议规约,功能码分类深度解析完成了对数据报文逐字段的深度提取与分析,访问控制功能根据访问控制规则策略检查深度解析出来的关键字段信息,判断是否允许数据报文通过,从而实现对 Modbus/TCP 通信的访问控制.

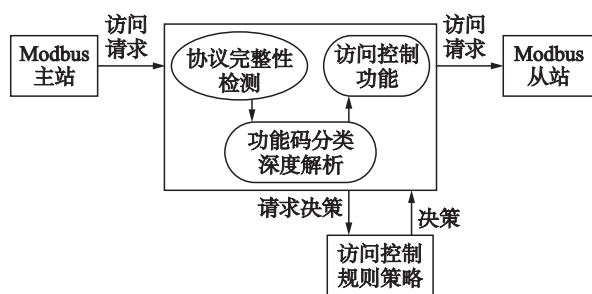


图2 基本访问控制模型

Fig.2 Basic communication control model

3.2 协议完整性检测

当一个恶意攻击者想要破坏 Modbus 从站时,一种非常简单的方法就是构造一个不符合 Modbus/TCP 协议规约的畸形数据报文,虽然畸形数据报文的端口也是 502 (指明是一个标准的 Modbus/TCP 数据报文),但是超长数据载荷可能造成 Modbus 从站缓存区溢出.因此,当捕获到 Modbus 主站和 Modbus 从站之间的通信数据报文后,本方法首先对 Modbus/TCP 数据报文进行协议完整性检测,检验每个数据包是否符合 Modbus/TCP 协议的规定,过滤掉异常的畸形数据报文,保障 Modbus 主站与 Modbus 从站之间通信数据的正确性.协议完整性检测主要检查 Modbus/TCP 数据报文的 3 个部分:

首先,检测 Modbus/TCP 数据报文的源端口号或者目的端口号是否为 502,这是最基本的检查,判断此数据报文是否是一个标准的 Modbus/TCP 数据报文.

其次,检测 Modbus 协议标志符,判断 Modbus/TCP 数据报文中 MBAP 报文头的协议标志符是否为 0x0000,因为 Modbus/TCP 协议规定 Modbus 主站和 Modbus 从站通过协议标识符是否为 0x0000 来识别 Modbus/TCP 协议.

最后,检测 Modbus/TCP 数据报文的应用数据长度,通过判断数据报文中 TCP 载荷的长度是否等于 MBAP 报文头中的长度域数值加上 6 个字节,其中 6 个字节表示 MBAP 报文头中事物处理标识符、协议标识符和长度三个域所占字节数之和.判别公式如下:

$$L_{\text{length-MBAP}}? = L_{\text{TCP}} - \sum_{i=1}^3 LB_i \quad (1)$$

其中, $L_{\text{length-MBAP}}$ 为 MBAP 报文头中的长度域数值, L_{TCP} 为 TCP 载荷的长度, $LB_i (i=1, 2, 3)$, 分别表示 MBAP 报文头中事物处理标识符、协议标识符和长度三个域所占字节数, ? 表明判断公式两端是否相等.

3.3 功能码分类深度解析

完成 Modbus/TCP 数据报文的协议完整性检测之后,本方法基于深度包解析技术,按照 Modbus/TCP 的协议规定,将数据报文中的应用层信息按功能码分类进行深度解析,提取出数据报文中每一个关键字段信息,不仅解析出功能码、起始地址、地址数等功能字段信息,同时还解析出被访问的具体寄存器或线圈的域值信息,为访问控制功能提供检测对象.

Modbus/TCP 协议对每一个功能码都进行了详细地功能描述,每一个功能码的作用不一样,使得 Modbus/TCP 数据报文格式中数据部分的组成形式也不一样,例如,对于功能码“01”读线圈,其数据部分主要为起始地址(2 个字节) + 线圈数量(2 个字节),而对于功能码“05”写单个线圈,其数据部分主要为输出地址(2 个字节) + 输出值(2 个字节).因此,应该按照功能码进行分类,针对不同的功能码对 Modbus/TCP 数据报文中的数据部分进行深度解析,获取代表不同意义的关键字段信息.图3给出了功能码分类深度解析的主要执行过程,如图所示,本文将 Modbus 功能码主要分为 3 类:读操作类、写操作类和其它操作类,其中,读操作类包括对“01”读线圈、“02”读离散输入、“03”读保持寄存器、“04”读输入寄存器、“20”读文件记录和“43”读设备识别码等功能码进行深度解析,写操作类包括“05”写单个线圈、“06”写单个寄存器、“15”写多个线圈、“16”写多个寄存器和“21”写文件记录等功能码进行深度解析,其它操作类包括“22”屏蔽写寄存器和“23”读/写多个寄存器两个功能码进行深度解析.功能码分类深度解析可以获得 Modbus/TCP 数据报文中更加细粒度的组成信息,越详细的关键字段信息使得针对 Modbus/TCP 通信的访问控制越精确,同时也更加便于对访问控制规则的设置,访问控制规则也更具针对性,访问控制的实施也更有效.

3.4 访问控制规则策略

根据功能码分类深度解析出的关键字段信息,访问控制功能检查这些信息是否符合初始设置的规则策略,判断 Modbus/TCP 数据报文是否合法.访问控制的规则策略指在系统安全策略级上表示授权策略,是对访问如何控制及如何做出访问决定的高层指南,在 Modbus/TCP 通信的访问控制中,访问控制规则策略能够深入到 Modbus/TCP 通信数据的应用层内容,即支持针对 Modbus/TCP 协议的深度设置.值得一提的是,本文所提出方法的访问控制规则

策略采用“白名单”模式, 不仅支持对传统的 IP 地址、协议、端口号的设置, 还支持针对 Modbus/TCP 协议的功能码、地址范围以及寄存器或线圈的域值设置. 相比现有的一些工控系统访问控制方法, 本文所提出方法的规则策略能够支持对具体写入寄存器或者线圈的域值进行限定, 具有更细腻的规则粒度, 图 4 给出了本文的规则策略粒度与

一般的规则策略粒度的对比, 其中, 实例的源 IP 地址为 192.168.1.2, 目的 IP 地址为 192.168.1.3, 协议为 tcp, 源端口号为 1024, 目的端口号为 502, Modbus 功能码为 06 写单个寄存器, 寄存器的地址范围在 100~200 之间, 寄存器中写入值的范围在 0~50 之间, 符合以上规则的 Modbus 通信数据包执行允许操作, 即 Pass.

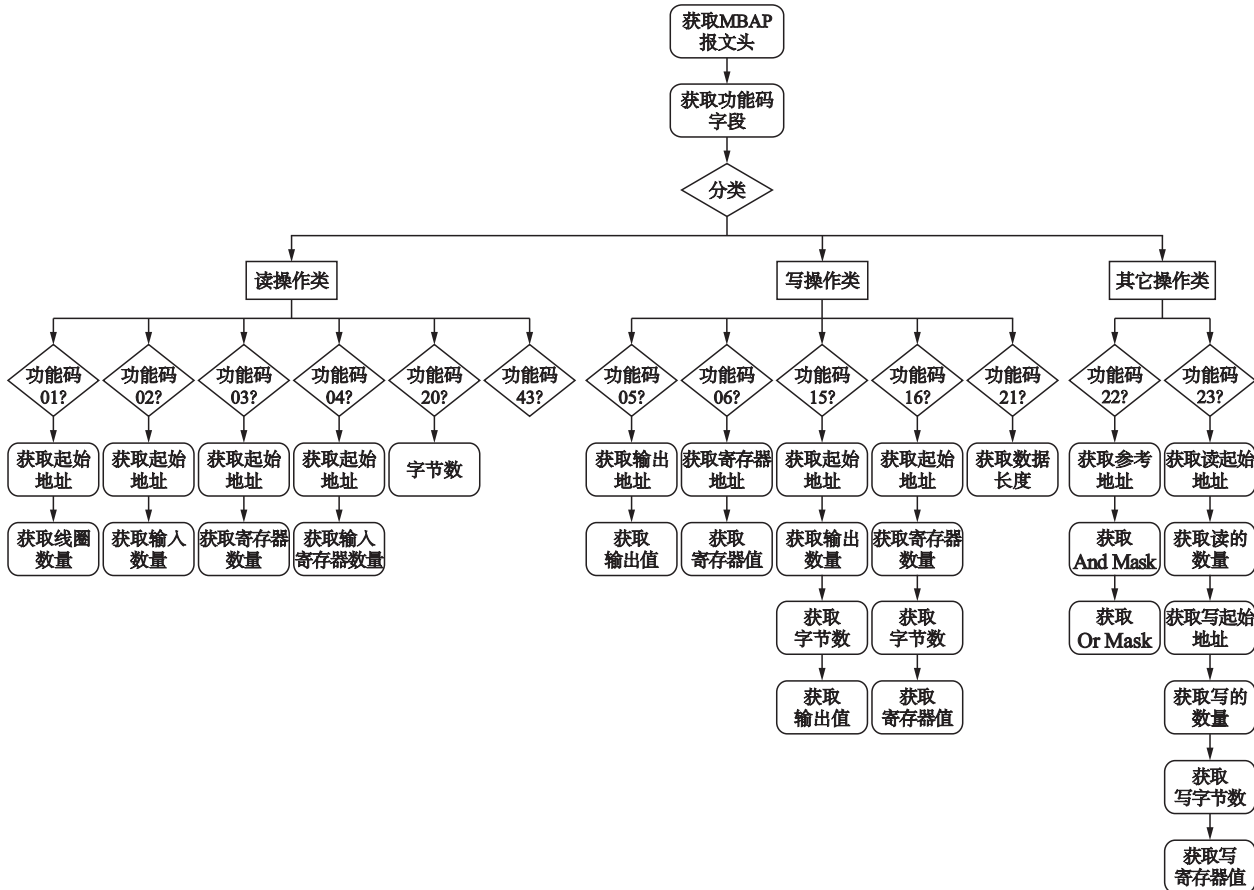


图 3 功能码分类深度解析的执行过程

Fig.3 Procedure of deep inspection based on different function codes

操作	IP地址	协议	端口号	功能码	地址范围	阈值
实例	Pass	源: 192.168.1.2 目的: 192.168.1.3	tcp	源: 1024 目的: 502	06	100~200
Modbus应用字段						0~50
一般的规则粒度						
本方法的规则粒度						

图 4 规则粒度对比

Fig.4 Comparison of rule granulation

3.5 规则执行算法

图 5 给出了访问控制规则策略执行算法的伪代码, 针对规则策略 Rule 1 到 Rule n , 分别执行以下匹配操作: 检查数据包的源、目的 IP 地址是否与规则设定的一致, 若一致, 则检查数据包的传输层协议是否为 tcp、目的端口是否为 502, 若是, 则进行功能码的分类检查: 若功能码属于写

操作类, 则继续检查数据包的地址与域值信息是否符合规则设定, 若符合, 则对此数据包执行 Pass 操作, 否则与下一条规则进行匹配; 若功能码属于读操作类, 则继续检查数据包的地址信息是否符合规则设定, 若符合, 则对此数据包执行 Pass 操作, 否则与下一条规则进行匹配; 若功能码属于其它操作类, 如果功能码等于 22, 则检查数据包的

And_Mask 和 Or_Mask 信息是否符合规则设定, 若符合, 则对此数据包执行 Pass 操作, 否则与下一条规则进行匹配, 如果功能码等于 23, 则分别检查数据包的读地址、写地址和域值信息是否符合规则设定, 若符合, 则对此数据包执行 Pass 操作, 否则与下一条规则进行匹配. 倘若所有规则都没有匹配成功, 则对此数据包执行 Drop 操作, 即丢弃此数据包.

```

for Rule 1 to Rule n
do check SourceIP and DestinationIP
if matched
then do check tcp? and DestinationPort?=502
if matched
then do check FunctionCode
if FunctionCode ∈ [Writing Operation Set]
then do check DataAddress
if matched
then do check DataValue
if matched
then do pass the packet
else do jump to next rule
else if FunctionCode ∈ [Reading Operation Set]
then do check DataAddress
if matched
then do pass the packet
else do jump to next rule
else if FunctionCode ∈ [Other Operation Set]
if FunctionCode=22
then do check And_Mask and Or_Mask
if matched
then do pass the packet
else do jump to next rule
else if FunctionCode=23
then do check ReadingDataAddress
if matched
then do check WritingDataAddress
if matched
then do check DataValue
if matched
then do pass the packet
else do jump to next rule
else do jump to next rule
else do jump to next rule
else do jump to next rule
else do jump to next rule
else do jump to next rule
else do jump to next rule
else do jump to next rule
else do jump to next rule
if n rules are unmatched
then do drop the packet

```

图 5 规则执行算法伪代码

Fig.5 Pseudo-code of rule implementation algorithm

4 安全性分析

通过在 Modbus 主站和 Modbus 从站之间实施本文所提出的方法, 合理地设置访问控制规则策略, 能够有效限制对 Modbus 从站的恶意访问, 从而保障 Modbus 从站的安全性.

首先, 针对 Modbus/TCP 缺乏认证, 本方法可以通过限制访问者 IP 的方式防止不合法用户对 Modbus 从站的访问, 因为本方法采用“白名单”模式, 即不符合规则策略中 IP 地址的 Modbus/TCP 数据报文会被过滤, 从而使得只有合法 IP 地址的 Modbus/TCP 通信才能顺利通过.

其次, 针对 Modbus/TCP 缺乏授权, 本方法可以通过设置功能码、地址范围和域值等规则信息来限制 Modbus 主站的访问权限, 即使黑客控制某一 Modbus 主站, 也无法向 Modbus 从站发送该主站权限之外的控制命令, 在一定程度上保护了 Modbus 从站的正常运行.

最后, 针对 Modbus/TCP 缺乏完整性检测, 本方法提

供了协议完整性检测机制, 分别检测了 Modbus/TCP 数据报文的端口号、MBAP 报文头的协议标志符和长度, 端口号的检测表明此数据报文为 Modbus/TCP 协议的数据报文, 协议标志符的检测表明此数据报文能够被 Modbus 从站识别, 长度的检测能够在一定程度上防止利用超长数据载荷的缓存区溢出攻击.

5 实验验证与分析

5.1 功能验证

本文将本方法实现于 Linux 透明网桥中(以下称之为访问控制网桥), 并串接在 Modbus 主站与 Modbus 从站之间, Modbus 主站采用 modpoll 实现对从站的控制, 通过设置访问控制规则策略, 限制 Modbus 主站对 Modbus 从站的访问行为, 从而验证本方法的有效性.

本文以功能码“16”写多个寄存器为例, Modbus 主站的 IP 地址为 192.168.1.101, Modbus 从站的 IP 地址为 192.168.1.4, 服务端口为 502, 本次实验的访问控制规则如下:

白名单规则:

[操作: Pass][源 IP: 192.168.1.101][目的 IP: 192.168.1.4]
[协议: tcp][源端口: any][目的端口: 502]
[功能码: 16][地址范围: 8 ~ 12][域值范围: 90 ~ 110]

默认规则:

[操作: Drop][源 IP: any][目的 IP: any][协议: tcp][源端口: any][目的端口: any]

正常情况下, 本文利用 modpoll 向 Modbus 从站执行写单个寄存器操作, 功能码是 16, 写寄存器的地址为 10, 写入值为 100, 这完全符合白名单规则, 图 6 给出了 modpoll 的执行结果, 从图中可以看出, 此次操作执行成功, modpoll 给出了“Writing 1 reference. ”, Modbus 从站的第 10 个寄存器被写入值 100, 也就是说此次操作的数据流顺利通过网桥到达 Modbus 从站. 为了验证能够拦截不合理的域值, 通过向第 10 个寄存器写入值 200, 图 7 给出了 modpoll 的执行结果, 从图中可以看出, modpoll 出现了“Reply time-out!”错误, 即回答超时, 表明此次操作的数据包被成功拦截, 因此 modpoll 没有收到 Modbus 从站的应答数据包, 此次操作没有执行成功. 同理, 为了验证能够拦截不合理的地址值, 通过向第 14 个寄存器写入值 100, 图 8 给出了 modpoll 的执行结果, 从图中可以看出, 同样出现了“Reply time-out!”错误, 此次操作同样没有执行成功. 另外, 为了验证对不合理功能码的拦截作用, 通过执行功能码 15 的写多个线圈操作, 对第 10 个线圈写入值为 1, 图 9 给出了 modpoll 的执行结果, 从图中可以看出, 出现了“Reply time-out!”错误, 说明此次操作没有执行成功. 从上述几个简单实验可以看出, 本文所提出的访问控制方法能够控制 Modbus 主站对 Modbus 从站的访问操作, 隔离非法的 Modbus 通信数据流, 防止对 Modbus 从站的恶意操作, 从而保障 Modbus 从站的正常运行.

为了与文[16]的方法进行对比, 本文在 Iptables/Netfilter 加载了文[16]所提及的 Modbus/TCP 过滤模块, 同样应用

在 Modbus 主站与 Modbus 从站之间,限制 Modbus 主站对 Modbus 从站的访问行为. 本次实验的主要白名单规则设置如下:

```
iptables -A FORWARD -p tcp -s 192.168.1.101 -d 192.168.1.4
-m modbus --funcode 16 --refnum 10 --allowtcp 1 -j ACCEPT
```

这里, funcode 表明允许的功能码, refnum 表明允许写入的地址,同时设置默认的规则操作为丢弃. 同样假设第 10 个寄存器的合法写入域值应该在 90 至 110 之间.

```
C:\Users\Administrator>D:\modpoll\win32\modpoll.exe -n tcp -t 4 -r 10 192.168.1.4 100
modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 10, count = 1
Communication.....: 192.168.1.4, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

Written 1 reference.

C:\Users\Administrator>
```

图 6 正常访问下 modpoll 执行结果

Fig.6 The result of modpoll under normal condition

```
C:\Users\Administrator>D:\modpoll\win32\modpoll.exe -n tcp -t 4 -r 10 192.168.1.4 200
modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 10, count = 1
Communication.....: 192.168.1.4, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

Reply time-out!

C:\Users\Administrator>
```

图 7 写入值为 200 时 modpoll 执行结果

Fig.7 The result of modpoll when the value is 200

```
C:\Users\Administrator>D:\modpoll\win32\modpoll.exe -n tcp -t 4 -r 14 192.168.1.4 100
modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 14, count = 1
Communication.....: 192.168.1.4, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

Reply time-out!

C:\Users\Administrator>
```

图 8 写入寄存器地址为 14 时 modpoll 执行结果

Fig.8 The result of modpoll when the register address is 14

```
C:\Users\Administrator>D:\modpoll\win32\modpoll.exe -n tcp -t 4 -r 14 192.168.1.4 100
modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 14, count = 1
Communication.....: 192.168.1.4, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

Reply time-out!

C:\Users\Administrator>
```

图 9 对功能码 05 操作时 modpoll 执行结果

Fig.9 The result of modpoll when the function code is 05

当 modpoll 向 Modbus 从站执行正常写寄存器操作时,即完全符合白名单规则,如图 10 所示,操作的功能码为 16,寄存器地址为 10,写入值为 95,可以看出此次操作执行成功,modpoll 给出了“Writing 1 reference.”,第 10 个寄

存器被写入值 95,说明此次操作的数据流被成功放行至 Modbus 从站. 当 modpoll 采用同样的功能码和寄存器地址,而写入值为 1 000,向 Modbus 从站执行异常的写寄存器操作时,如图 11 所示,此次操作同样执行成功,modpoll 给出了“Writing 1 reference.”,第 10 个寄存器被写入值 1 000,说明该方法并没有成功过滤掉此次异常操作的数据包. 另外,当 modpoll 分别执行图 8 和图 9 两次实验的操作时,其相应的结果与图 8 和图 9 描述的一致. 从上述实验可以看出,文[16]的方法虽然在一定程度上保护 Modbus 从站不受非法访问,但也存在着一些不足之处:首先,该方法无法限制寄存器值的写入范围,Modbus 从站有可能接受过大或过小的域值,对控制系统造成破坏;其次,该方法的白名单规则无法设置一个连续的地址区间,每条规则只能设置一个地址值,过多的规则会影响该方法的执行效率. 综上所述,本文所提出的方法要优于文[16]的方法.

```
C:\Users\Administrator>D:\modpoll\win32\modpoll.exe -n tcp -t 4 -r 10 192.168.1.4 95
modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 10, count = 1
Communication.....: 192.168.1.4, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

Written 1 reference.

C:\Users\Administrator>
```

图 10 正常访问下 modpoll 写入值为 95 时执行结果

Fig.10 The result of modpoll when the value is 95 under normal condition

```
C:\Users\Administrator>D:\modpoll\win32\modpoll.exe -n tcp -t 4 -r 10 192.168.1.4 1000
modpoll 3.4 - FieldTalk(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2013 proconX Pty Ltd
Visit http://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 10, count = 1
Communication.....: 192.168.1.4, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

Written 1 reference.

C:\Users\Administrator>
```

图 11 异常访问下 modpoll 写入值为 1 000 时执行结果

Fig.11 The result modpoll result when the value is 1 000 under abnormal condition

5.2 模拟实验分析

为了分析本文的访问控制方法在真实工业控制系统环境中的安全防御效果,本文搭建了基于 Modbus/TCP 通信的工业 SCADA(supervisory control and data acquisition)系统模拟实验环境,如图 12 所示,该环境分为 3 层:数据监控层包含两个 Modbus 主站(工程师站和恶意攻击者),其中工程师站不仅具有对现场过程监视功能,而且具有修改现场工艺参数的能力;控制单元层由支持 Modbus/TCP 通信的控制 PLC(programmable logic controller)组成,通过访问控制网桥与数据监控层连接,主要实现接受工程师站的控制命令,根据自身的控制逻辑驱动现场执行机构进行响应;虚拟现场设备层由 3 个开关量和液位传感器等组成,3 个开关量分别代表容器的两个进料口阀门开关和一个出口口阀门,物料 A 和物料 B 依次通过阀门开关 A 和 B 进

入容器合成产品 C, 当达到液位上限时, 关闭阀门开关 A 和 B, 打开阀门开关 C, 排出产品 C, 达到液位下限时, 重复上述操作过程。

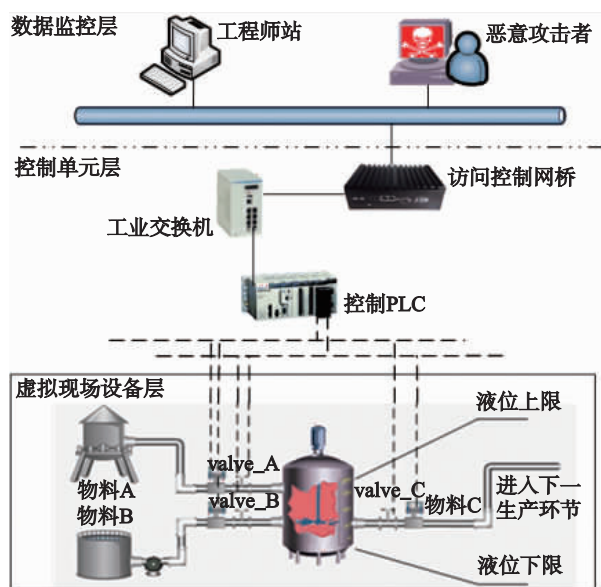


图 12 实验环境拓扑图

Fig.12 Experimental environment topology

本次实验中设定虚拟现场设备每 5 min 完成一次上述工艺操作流程, 采用写线圈的方式控制开关量, 每一个开关拥有一个线圈地址。假设恶意攻击者不知道具体的控制参数, 只是盲目地向控制 PLC 发送功能码为 15 的写线圈数据包, 试图改变虚拟现场设备的工作流程, 攻击者的攻击速率为每分钟 80 ~ 100 个数据包 (每个数据包中只有一个随机的线圈地址, 相应的域值随机的写入 1)。访问控制规则策略设置为只允许 01 和 15 功能码对 3 个开关的线圈地址进行读和写操作的数据包才能通过, 其它的数据包默认为丢弃。图 13 给出了攻击时经过访问控制过滤的前后对比, 由图可知, 几乎所有的攻击流量都被访问控制网桥过滤, 过滤后的流量呈现周期性的变化, 即每 5 min 完成一次工艺流程, 值得注意的是, 在过滤后的流量中出现了一个异常点, 这是因为攻击者这种盲目随机的方式恰巧产生了一个符合访问控制规则策略的数据包, 从而造成了设备工作异常, 然而这是一种极端的攻击情况, 在实际的工艺操作过程中并不会出现, 并且出现这种情况的概率非常小, 仅仅为 $(1/2^{16}) \times 3$, 随着工艺的复杂程度越来越高, 这种概率将变得越来越小。

5.3 性能分析

为了分析本文所提出访问控制方法的性能指标, 本文采用 IXIA Optixia XM2 网络性能测试仪对访问控制网桥分别在 1 条规则和 200 条规则下进行数据包转发性能测试, 访问控制网桥的基本配置如下: Linux 内核版本 2.6.24、内存 4 GB、CPU 为 Atom D525 (1.8 GHz)、网卡为 10/100/1 000 Mbps 自适应以太网卡, 测试方法采用串联回路的方式, 如图 14 所示, IXIA Optixia XM2 设备一个端口负责发

送不同速率的测试流量, 经过访问控制网桥转发后, 另一个端口接受转发后的测试流量, 从而分析出吞吐量、时延、丢包率等指标。

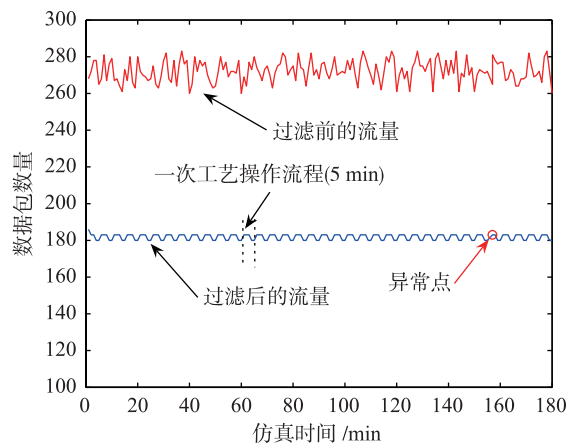


图 13 过滤前后的流量对比

Fig.13 Comparison of packet numbers before and after filtering



图 14 测试拓扑图

Fig.14 Performance test topology

本文首先进行了吞吐量的测试, 采用 64 字节短包在丢包率为 0 的情况下测试转发速率, 结果发现吞吐量达到了 140 Mbit/s, 完全满足工业以太网百兆级网络通信的需求。在此基础上, 本文在分别设置 1 条规则和 200 条规则情况下, 针对 64 字节短包、512 字节中长包和 1 518 字节长包进行了丢包率和延迟测试, 实际测试流量为 200 Mbit/s (双向, 每一方向测试流量为 100 Mbit/s), 512 字节中长包, 结果如表 1 所示, 本文发现在 1 条规则和 200 条规则情况下, 丢包率都为 0, 并且转发延迟也都在百微秒级, 而实际上大多数工业控制系统的工业以太网转发延迟达到毫秒级即可满足可用性需求。由以上分析可知, 采用本方法的访问控制网桥不仅能够为工业控制系统提供安全保护, 而且还能够满足工业应用的网络通信需求。

6 原型平台搭建

为了说明工业病毒如何对关键工控设备的攻击及本文所提出访问控制方法的防护作用, 本文采用灯箱的方式搭建了石化液位控制系统攻击与防护演示平台, 如图 15 所示。该平台模拟了炼油、化工生产工艺中液体净化的液位

控制过程, 通过上位机组态软件中堆缓冲区溢出漏洞(该漏洞主要原因是上位机组态软件未经正确的验证用户输入, 使得写缓冲区超出其长度内容, 造成缓冲区的溢出), 工业病毒破坏上位机组态软件的堆栈, 使得上位机组态软件的程序执行指针指向工业病毒中的恶意代码, 从而运行恶意代码, 恶意代码主要进行两部分工作: 一是, 保持上位机监控画面仍然按照正常工艺流程显示, 令攻击无法被工程师或操作员通过监控画面察觉; 二是, 构造 Modbus/TCP 写寄存器的指令, 使得 3 个子净化罐同时向主净化罐注入液体, 造成主净化罐液体超出警戒线而发生爆炸. 整个攻击的操作过程如下: 首先, 将上位机插入带有工业病毒的 U 盘, 利用 Windows XP 自带的 AutoRun 功能, 工业病毒会主动将其恶意代码存至上位机中一段指定的内存段中, 然后, 工业病毒利用上位机组态软件的堆缓冲区溢出漏洞, 执行恶意代码, 进行破坏操作, 不仅使得上位机显示一切正常, 而且发送恶意的 Modbus 控制指令攻击 PLC, 使得净化罐中液体超过警戒线而无法及时排出, 因此在工作人员无法察觉的情况下导致重大事故. 当将访问控制网桥串接于上位机与 PLC 之间时, 再次插入带有病毒的 U 盘, 可以观察到, 上位机界面正常显示, 同时控制系统未发生任何故障. 根据演示平台, 作者还搭建了石化液位控制系统攻击与防护的真实原型系统, 如图 16 所示, 从而证明了本文所提出访问控制方法的可行性.

表 1 不同规则数下的性能测试

Tab.1 Performance test under different rule numbers

测试条件		1 条规则		200 条规则	
包长	测试流量	丢包率	延迟	丢包率	延迟
64 Byte	200 Mbit/s	0	342.60 μ s	0	192.76 μ s
512 Byte	200 Mbit/s	0	699.22 μ s	0	901.28 μ s
1 518 Byte	200 Mbit/s	0	891.82 μ s	0	884.03 μ s

7 总结

本文首先对 Modbus/TCP 协议进行了安全性分析, 然后提出了一种基于功能码深度检测的 Modbus/TCP 通信访问控制方法, 分别介绍了该方法的基本模型、协议完整性检测、功能码分类深度检测以及访问控制规则策略, 不仅



图 15 石化液位控制系统攻击与防护演示平台

Fig.15 Attack and protection demonstration platform for petrochemical level control system

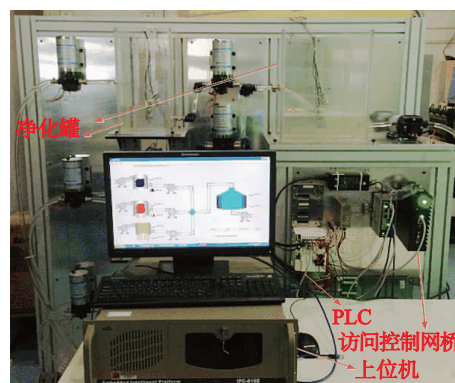


图 16 石化液位控制系统攻击与防护真实原型系统

Fig.16 Attack and protection prototype system for petrochemical level control system

实现了传统的功能码、地址范围的检测, 同时还实现了寄存器或线圈的域值检测. 最后通过安全性分析和模拟实验, 本文验证了该方法能够有效地拦截不合理的 Modbus/TCP 通信数据流, 从而保障 Modbus 从站不受非法的访问. 在此基础上, 本文还搭建了石化液位控制系统攻击与防护演示平台及相应真实原型系统, 证明了该方法的实际可行性.

参考文献

- [1] 彭勇, 江长清, 谢丰, 等. 工业控制系统信息安全研究进展[J]. 清华大学学报: 自然科学版, 2012, 52(4): 1396-1408.
Peng Y, Jiang C Q, Xie F, et al. Industrial control system cybersecurity research[J]. Journal of Tsinghua University: Science and Technology, 2012, 52(4): 1396-1408.
- [2] Kang D H, Kim B K, Na J C. Cyber threats and defence approaches in SCADA systems[C]//2014 16th International Conference on Advanced Communication Technology (ICACT). Piscataway, NJ, USA: IEEE, 2014: 324-327.
- [3] Knijff R M. Control system/SCADA forensics, what's the difference?[J]. Digital Investigation, 2014, 11(3): 160-174.
- [4] Industrial Control Systems Cyber Emergency Response Team. ICS-CERT year in review[EB/OL]. (2014-02-24)[2015-03-01]. https://ics-cert.us-cert.gov/sites/default/files/documents/Year_In_Review_FY2013_Final.pdf.
- [5] Modbus-IDA. Modbus messaging on TCP/IP implementation guide v1.0a[EB/OL]. (2004-06-04)[2015-03-01]. http://www.modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0a.pdf.
- [6] Goldenberg N, Wool A. Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems[J]. International Journal of Critical In-

- frastructure Protection, 2013, 6(2): 63–75.
- [7] Kobayashi T H, Batista A B, Brito A M, et al. Using a packet manipulation tool for security analysis of industrial network protocols[C]//IEEE Conference on Emerging Technologies and Factory Automation. Piscataway, NJ, USA: IEEE, 2007: 744–747.
- [8] NSFOCUS. Industry control system security report[EB/OL]. (2013–06–24)[2015–03–02]. http://www.nsfocus.com.cn/report/NSFOCUS_ICS_Security_Report_20130624.pdf.
- [9] Huitsing P, Chandia R, Papa M, et al. Attack taxonomies for the Modbus protocols[J]. International Journal of Critical Infrastructure Protection, 2008(1): 37–44.
- [10] Hayes G, Ei-khatib K. Securing modbus transactions using hash-based message authentication codes and stream transmission control protocol[C]//2013 Third International Conference on Communications and Information Technology. Piscataway, NJ, USA: IEEE, 2013: 179–184.
- [11] Phan R C W. Authenticated modbus protocol for critical infrastructure protection[J]. IEEE Transactions on Power Delivery, 2012, 27(3): 1687–1689.
- [12] 王婷婷. SCADA 系统中数据传输安全性研究[D]. 上海: 华东理工大学, 2013.
- Wang T T. Security research on SCADA system data transmission[D]. Shanghai: East China University of Science and Technology, 2013.
- [13] Erez N, Wool A. Control variable classification, modeling and anomaly detection in Modbus/TCP SCADA systems[J]. International Journal of Critical Infrastructure Protection, 2015, 10(1): 59–70.
- [14] Fovino I N, Carcano A, Murel T D L, et al. Modbus/DNP3 state-based intrusion detection system[C]//2010 24th IEEE International Conference on Advanced Information Networking and Applications. Piscataway, NJ, USA: IEEE, 2010: 729–736.
- [15] 尚文利, 张盛山, 万明, 等. 基于 PSO-SVM 的 Modbus TCP 通讯的异常检测方法[J]. 电子学报, 2014, 42(11): 2314–2320.
- Shang W L, Zhang S S, Wan M, et al. Modbus/TCP communication anomaly detection algorithm based on PSO-SVM[J]. Acta Electronica Sinica, 2014, 42(11): 2314–2320.
- [16] Pothamsetty V, Franz M. Transparent Modbus/TCP filtering with Linux[EB/OL]. (2004–05–11)[2015–03–02]. <http://modbusfw.sourceforge.net/>.
- [17] 张盛山, 尚文利, 万明, 等. 基于区域/边界规则的 Modbus/TCP 通讯安全防御模型[J]. 计算机工程与设计, 2014, 35(11): 3701–3707.
- Zhang S S, Shang W L, Wan M, et al. Security defense module of Modbus TCP communication based on region/enclave rules[J]. Computer Engineering and Design, 2014, 35(11): 3701–3707.
- [18] Igor N F, Alessio C, Andrea C, et al. Critical state-based filtering system for securing SCADA network protocols[J]. IEEE Transactions on Industrial Electronics, 2012, 59(10): 3943–3950.
- [19] Liu C, Pan Y, Chen A, et al. A DFA with extended character-set for fast deep packet inspection[J]. IEEE Transactions on Computers, 2013, 63(8): 925–937.
- [20] Yang Y E, Le H, Prasanna V K. High performance dictionary-based string matching for deep packet inspection[C]//2010 IEEE INFOCOM. Piscataway, NJ, USA: IEEE, 2010: 1–5.
- [21] Myungkeum Y. Using whitelisting to mitigate DDoS attacks on critical Internet sites[J]. IEEE Communications Magazine, 2010, 48(7): 110–115.
- [22] Stouffer K, Falco J, Scarfone K. NIST SP 800-82, Guide to industrial control systems (ICS) Security Revision 1[EB/OL]. (2013–05)[2015–03–02]. <http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r1.pdf>.

作者简介

万 明(1984–), 男, 博士, 副研究员. 研究领域为工业控制系统信息安全, 互联网网络与信息安全.

尚文利(1974–), 男, 博士, 副研究员, 硕士生导师. 研究领域为工业控制系统信息安全, 机器学习.

曾 鹏(1976–), 男, 博士, 研究员, 博士生导师. 研究领域为无线传感器网络, 工业无线通信.

(上接第 247 页)

- [20] Rosenstein M T, Collins J J, Deluca C J. A practical method for calculating largest Lyapunov exponents from small data sets[J]. Physica D, 1993, 65(1): 117–134.
- [21] Wang J, Wang W, Wu S. A new support vector machine optimized by simulated annealing for global optimization[J]. International Journal of Precision Engineering and Manufacturing, 2012, 2(1): 8–14.

作者简介

李 琨(1983–), 男, 博士, 副教授. 研究领域为复杂工业系统建模、控制与诊断, 软测量方法及应用.

韩 莹(1982–), 女, 硕士, 讲师. 研究领域为复杂系统建模及预测.

黄海礁(1976–), 男, 本科, 工程师, 区长. 研究领域为石油生产自动化理论与应用.