Software Engineering and Applications 软件工程与应用, 2020, 9(6), 497-506
Published Online December 2020 in Hans. http://www.hanspub.org/journal/sea
https://doi.org/10.12677/sea.2020.96057

Hans汉斯

基于网络流量异常监测的工业控制系统安全技术研究

徐友洪，李剑萍，吴宏良

衢州职业技术学院，浙江 衢州
Email: 6247158@qq.com

收稿日期：2020年11月16日；录用日期：2020年12月17日；发布日期：2020年12月24日

摘 要

随着科技的发展，在智能自动化制造中工业控制系统逐渐被接入互联网，而当前互联网上存在着大量的攻击，直接影响着工业控制系统的安全，工控系统面临的安全形势也越来越严重。因此，工业控制系统与关键基础设施的网络安全受到高度的关注，为有效抵御恶意软件对工业控制系统的攻击，网络入侵检测系统是一个常用的措施，其分为两大主要的策略，一种策略是采用统计分析与机器学习的异常监测，另一种策略是采用攻击特征或规则进行比对的特征监测。本文提出一种监测工业控制系统网络出现的异常封包的技术，该技术的核心技术在于寻找TCP和UDP协议数据部分的规律性，并构造一个正常行为模型。通过工业控制系统网络内布置的蜜罐技术，系统模型还可以额外产出特征，协助过滤已知的攻击。该方法适用于建立在TCP与UDP之上的工业控制系统协议，并将检测模型嵌入到工业防火墙中，实现对Modbus/TCP与BACnet/IP异常报文检测。

关键词

网络安全，异常监测，工控系统，协议通讯网络

Research on Security Technology of Industrial Control System Based on Network Traffic Abnormal Monitoring

Youhong Xu, Jianping Li, Hongliang Wu

Quzhou College of Technology, Quzhou Zhejiang
Email: 6247158@qq.com

Received: Nov. 16th, 2020; accepted: Dec. 17th, 2020; published: Dec. 24th, 2020

文章引用: 徐友洪, 李剑萍, 吴宏良. 基于网络流量异常监测的工业控制系统安全技术研究[J]. 软件工程与应用, 2020, 9(6): 497-506. DOI: 10.12677/sea.2020.96057

Abstract

With the development of science and technology, industrial control systems are gradually connected to the Internet in intelligent automated manufacturing, and there are a large number of attacks on the Internet, which directly affect the safety of industrial control systems, and the security situation facing industrial control systems is becoming more and more serious. The network security of industrial control systems and critical infrastructure has been highly valued in recent years. In order to resist malicious software attacks against industrial control systems, network intrusion detection systems are a commonly used method, which is divided into two main strategies. One kind of anomaly detection uses statistical analysis and machine learning, and the other is misuse detection that uses attack characteristics or rules to compare. A technology for detecting abnormal packets in the industrial control system network is proposed in this paper. The core concept of the technology is to find the regularity of the TCP and UDP protocol payloads, and construct a normal behavior model. Through the honeypot is arranged in the industrial control system network, the system model can also generate additional features to help filter known attacks. Our method is suitable for industrial control system protocols built on TCP and UDP, and the detection model is embedded in the industrial firewall to realize the detection of Modbus/TCP and BACnet/IP abnormal messages.

Keywords

Network Security, Abnormal Monitoring, Industrial Control System, Protocol Communication Network

1. 引言

随着全球经济一体化进程的加速，工业信息化及物联网技术高速发展，在智能自动化制造中，工业控制系统被认为是一个核心模块，因为它被广泛应用在多种领域的过程控制上，部分工业控制系统也能够以某些方式连接到互联网等公共网络，病毒、木马、入侵攻击、拒绝服务等安全威胁也正在向工业控制系统扩散。工业控制系统(Industrial Control System, ICS)中的网络安全问题势必延缓工业 4.0 的采用。许多企业领导者发现 ICS 网络安全挑战非常难以理解，因为众多因素导致其非常复杂。此外，开发工业控制系统解决方案的工程师可能尚未看到在设备层面的重大网络安全要求。保障工业控制系统安全的传统方法依赖于限制对网络和设备的访问，并通过信息技术(IT)解决方案监控网络流量。在工厂中使用设备的产品负责人会发现如果将网络安全问题视为 IT 问题，就很容易解决。然而，随着工业 4.0 的出现，传统方法将不再足以保障工业控制系统的安全。如果公司没有应对终端设备安全问题的策略，ICS 网络安全面临的挑战最终将延缓工业 4.0 的采用[1] [2] [3]。一般常见工控协议中包含了大量的命令字，如读取、写入数据等，然而其中一部分高级或协议约定的自定义功能往往会给用户安全带来更多的威胁，如 Modbus 协议的从机诊断命令将会造成从机设备切换到侦听模式，CIP 协议某些命令字还能导致设备直接重启，SiemensS7 协议的 STOP CPU 功能将会导致 PLC 程序运行停止，在大多数的情况下用户在上位机进行组态时仅会使用协议的某些读取数据功能、固定范围和固定地址的写数据功

能，而协议栈上更多的功能则不会应用于系统集成中。如果对协议字段的掌握和对协议命令字的掌握，便可以很灵活地将可能给用户带来风险和威胁的一些隐藏功能，单独使用应用层防火墙对报文进行深度过滤，或者使用 IDS 进行报警提醒，Modbus 和 Siemens S7 协议中常见的威胁样例以及报警规则的应用方式如表 1 及表 2 所示。

Table 1. Threat examples and alarm rules of Modbus protocol
表 1. Modbus 协议的威胁样例以及报警规则

程度等级	威胁行为描述	潜在危害
高	主站下发 08 号功能码	可能导致设备进入 Standby 状态
高	主站下发 90(5A)号功能码-stop (Schneider)	导致 PLC CPU 进入 STOP 停机状态
高	主站下发 90(5A)号功能码-download (Schneider)	PLC 的内部程序可能正在被替换程
中	主站下发 90(5A)号功能码-upload (Schneider)	设备将工程上传至主站可能造成信息泄露
中	主站一次下发 modbus 报文超过 260 个字节	超出 modbus tcp 协议标准组包长度，可能导致设备拒绝服务
低	主站下发 43(2B)号功能码	导致设备及其固件版本信息泄露

Table 2. Threat examples and alarm rules of Siemens S7 protocol
表 2. 西门子 S7 协议的威胁样例以及报警规则

程度等级	威胁行为描述	潜在危害
高	主站下发 STOP/RUN 命令	导致设备进入停机状态/导致设备被启机初始化
高	主站下发 download block 命令	PLC 的内部程序可能正在被替换程
高	主站下发 delet block 命令	PLC 的内部程序块可能正在被删除
中	主站下发错误的密码请求	正在未授权访问
低	主站下发 read szl 请求	正在尝试获取设备模块信息、固件信息

工业控制系统如 SCADA 系统、DCS 系统和 PLC 等目前已广泛应用于工业、能源、交通、水利以及市政等国家关键基础设施领域，是工业自动化的核心中枢神经。如何能够保护工业控制系统免于遭受恶意软件的攻击成为一项重要的议题[4] [5] [6] [7]，网络入侵检测系统是一种发展许久的防御措施，监测的方法可分为两大策略，一是基于特征的监测，另一是基于异常的监测。基于异常的网络入侵检测，透过观察正常网络流量的内容或是通讯量的变化，定义规则或是训练分类器，借此判断监视中的网络流量有无异常，这类方法在工业控制系统已有相关的研究，例如视觉传感网络、建筑自动化系统等，有些研究是针对特定的装置，或是特定的通讯协议，有些研究则提出较为通用性的方法。基于特征的网络入侵检测，需要事先准备攻击特征，透过检查网络通讯中是否含有攻击特征的方式达到入侵检测。这个方法虽然直接且明确，但需要先有攻击样本，再由资安专家的观察分析后才能归纳出攻击特征，因此无法用来防御零日攻击。Snort 是一款开源的网络入侵检测软件，其采用规则来判断是否有网络攻击行为，在此研究中我们将采用其规则(Snort rule)的形式来记录数据(payload)的特征。除了使用网络入侵检测系统来保护工业控制系统网络，蜜罐技术(Honeypot)也是一种重要的防御手段。蜜罐技术是一种透过引诱攻击来搜集信息的诱饵系统，通过模仿真实存在的装置或是服务来吸引攻击者，在攻击手段日新月异的状况下，蜜罐有机会预先搜集到未知的攻击手法。Conpot 是有名的工业控制系统蜜罐，可以使用的协议包括 Modbus/TCP、S7、HTTP、SNMP、BACnet 等。

2. 工业网络异常监测方法

2.1. 网络异常监测

异常监测的研究很多，面对工业控制系统时，根据网络流量的稳定性来做异常监测的判断依据，搭配统计分析、机器学习等方法为正常的网络状态建立分类器，而在仅有正常网络流量作为训练数据的情况，一元分类成为一项异常监测的重要方法。Matti Mantere 等人[8]提出一个针对工业控制系统网络的异常监测模块，并且实作在 Bro NSM (network security monitor)。Zhiyuan Zheng 等人[4]提出一个针对 BACnet 通讯协议的异常监测方法。Ye T.等人[9]提出一个双重行为特性的异常监测方法，这里的双重行为指的分别是功能控制行为和处理数据行为，使用 behavior extraction 算法进行特征获取，再利用这些特征进行一元分类的训练，建立正常行为的分类器。由于工业控制系统相对于传统 IT 系统有较稳定的通讯模式，也就是周期性和时间顺序性，双重行为的分析方法便是利用这样的特性产生的。

2.2. 工业控制系统蜜罐技术

蜜罐技术(Honeypot)是一种对攻击方进行欺骗的技术[10] [11]，通过布置一些作为诱饵的主机、网络服务或者信息，诱使攻击方对它们实施攻击，从而可以对攻击行为进行捕获和分析，了解攻击方所使用的工具与方法，推测攻击意图和动机，能够让防御方清晰地了解他们所面对的安全威胁，并通过技术和管理手段来增强实际系统的安全防护能力。开源工控蜜罐中，主要针对 modbus、s7、IEC-104、DNP3 等工控协议进行模拟，其中 conpot 和 snap7 是相对成熟的蜜罐代表，conpot 实现了对 s7comm、modbus、bacnet、HTTP 等协议的模拟，属于低交互蜜罐，conpot 部署简单，协议内容扩展方便，并且设备信息是以 xml 形式进行配置，便于修改和维护。Snap7 是专门针对西门子 PLC 的蜜罐，基本实现了 s7comm 协议栈。它可以模拟实际设备的信息与状态，而且实现常用 PLC 操作的交互。但这些这些主流的虚拟蜜罐只能模拟单一工控协议，因此只能捕获单一工控协议的攻击数据。为提高蜜罐的部署能力，降低蜜罐部署成本，陆续有研究者提出采用低交互蜜罐与高交互蜜罐混合部署的架构，在合适的时候调度合适的蜜罐，在学术领域陈之为混合蜜罐，示例如下：

1) Snort 和 honeybrid 联合方案

Snort 主要进行低高交互流量的鉴别，并通知 Honeybrid 网关，便于后续步骤的进行。Honeybrid 网关包括决策引擎和重定向引擎，负责协调前端和后端之间的过滤和重定向，决策引擎用于选择感兴趣的流量，重定向引擎用于透明地重定向流量，联合方案如图 1 所示。

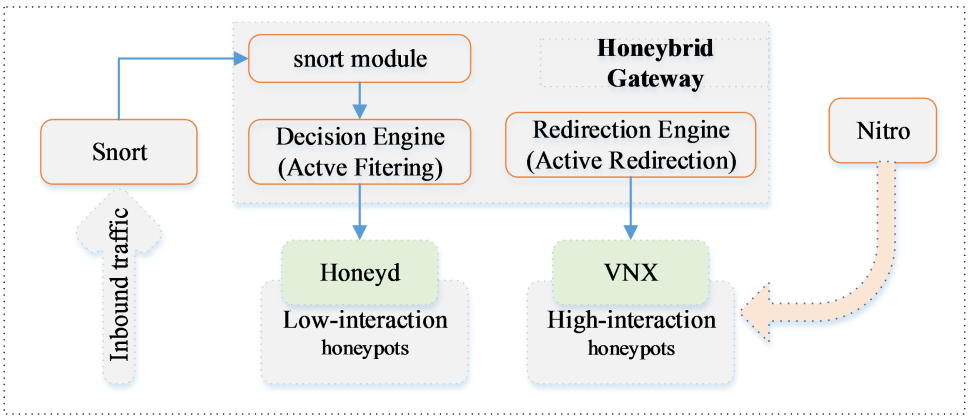

Figure 1. Scheme of snort and honeybrid
图 1. Snort 和 honeybrid 联合方案

其中 honeybrid 是一种典型的混合蜜罐框架，主要有如下四部分模块(决策引擎、重定向引擎、控制引擎和日志引擎)，示意图如图 2 所示。

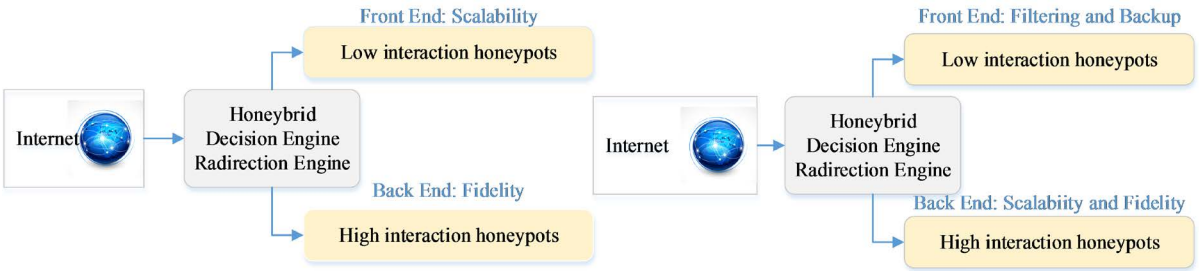

Figure 2. Mixed honeypot framework
图 2. 混合蜜罐框架

这四个组件围绕目标概念进行阐述，目标概念包含基于蜜罐的实验的规范。因此，每当我们想要运行基于蜜罐的新实验时，我们必须考虑我们想要收集的流量类型以及我们想要收集它的方式，即具有多少粒度和控制程度。目标由四个声明组成：一个过滤规则，它使用 tcpdump 语法定义此目标应处理的确切流量类型；一个前端规则，定义哪个蜜罐应该首先与传入的攻击流量进行交互，以及接受此传入流量的标准是什么；一个可选的后端规则，用于定义流量被重定向到哪个蜜罐以进行更详细的分析，以及决定重定向流量的标准是什么；一个可选的控制规则，定义如何限制蜜罐启动的传出流量。

2) Snort + SDN 方案

SDN 是软件定义网络，它以下发 flowtable 的形式完成对流量的控制，所提出的混合蜜罐架构如图 3 所示。它主要由一个基于 OpenFlow 的交换机来管理控制平面，它负责重定向攻击者和不同蜜罐之间的连接。在控制平面中，开源 IDSSnort 用于分析流量以生成警报，并通过 UNIX 套接字将警报消息发送到控制器应用程序。根据警报消息，决策引擎(DE)将决定转发或重定向连接并发信号通知重定向引擎(RE)以执行相应的动作。在上述方案中都是用了 Snort 工具，Snort 是一种入侵检测工具，可以针对数据包进行单包解析，在监听到数据包后首先会对来源数据包进行解析，然后提取特征，匹配规则，从而发出告警信息，示意图如图 4 所示。

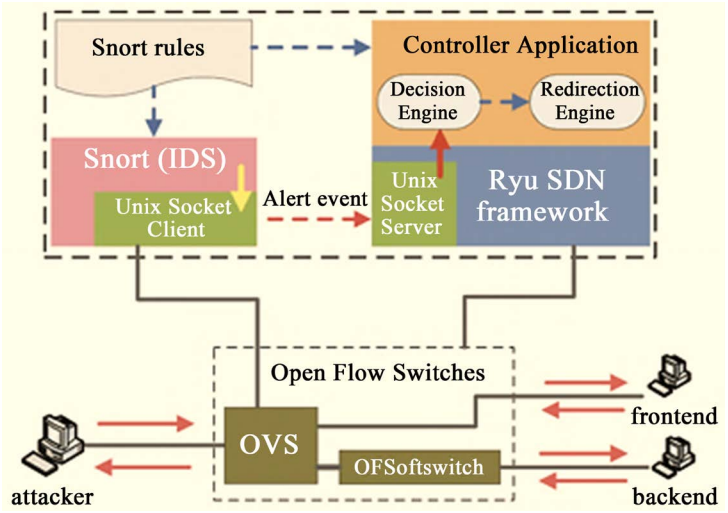

Figure 3. Mix frame work of Snort and SDN
图 3. Snort + SDN 方案

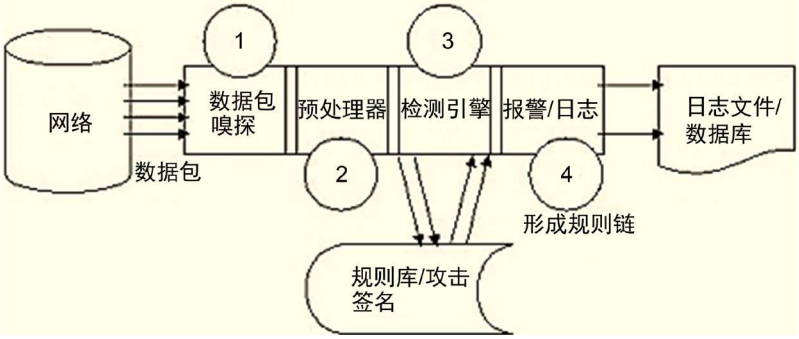

Figure 4. Intrusion detection
图 4. 入侵检测

利用在蜜罐框架中，是利用了它的数据解析功能与告警功能；它在匹配到对应的信息后，可以发出信号，从而使得下一步的处理程序可以进行处理。

3. 理论分析

本文提出的系统架构如图 5 所示，控制系统网络通常分为三个层级，由上而下分别是监控层、控制单元层和现场设备层，在系统架构内泛指其中任何一层，假定该网络内设有数个蜜罐与数个装置，而这些装置拥有各自的正常行为模型之后就成为被保护的目标。

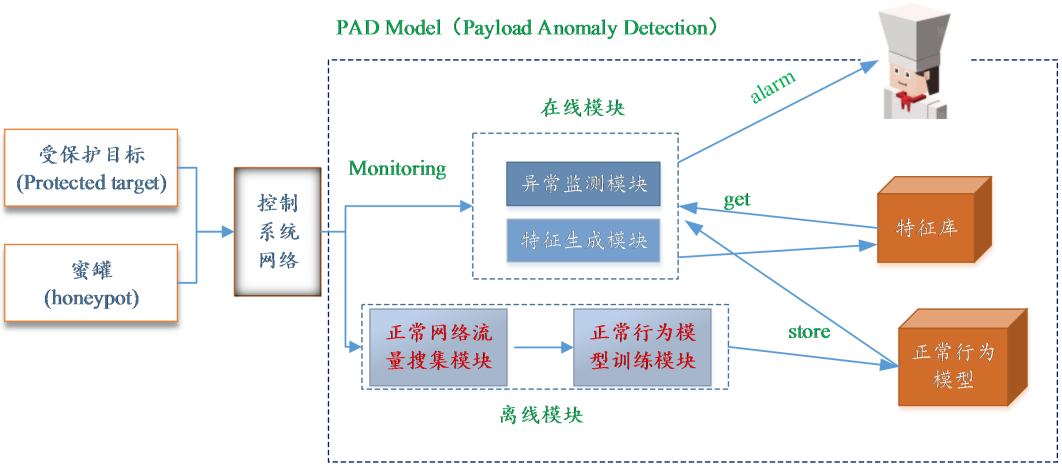

Figure 5. System architecture
图 5. 系统架构

3.1. 系统架构

数据异常监测模型分为离线与在线两大模块，离线模块包含正常网络流量搜集模块与正常行为模型训练模块，是系统运作的预处理部分，其产出的正常行为模型会被储存，在线模块使用。在线模块包含异常监测模块与特征生成模块，控制系统网络内所有的封包都将送至异常监测模块检查，一旦发现异常便发送警报给管理者。数据异常监测模型包含四个模组，分别是正常流量模组、正常行为训练模组、异常监测模组和特征生成模组。正常封包模组会根擦不同目的地 IP-Port 将封包分组，而这些目的地 IP 对应的装置称为保护目标(target)。正常行为训练模组会分析正常网络流量中 TCP 和 UDP 协定的数据，替保证目标建立正常行为模型；异常监测模组会监控工业控制系统网络流量，根据封包不同的流向选择对

应的正常行为模型来分类网络封包；若有封包流向蜜罐，并且被判定异常，特征生成模组将此异常封包生成对应的攻击特征，立即应用到我们的异常监测模组，以筛选掉后续任何符合该特征的封包，避免重复产生相同的攻挚特征。

3.2. 正常网络流量搜集模块

从控制网络内搜集一定时间区间 t 内的封包，并且根据目的 IP-Port 将封包分组，例如 192.168.1.10:502 为一组，192.168.1.10:47808 为不同的一组，如图 6 所示。分组完毕的封包会送到正常行为模型训练模块。搜集封包的区间 t 会根据不同的工业控制系统环境有所不同，如果环境拥有明显的周期性，周期为 d，则 t 应取 d 的数倍，以训练出适当的正常行为模型，以智能电表为例，若读表的周期为 5 分钟一次，搜集封包的区间 t 建议取 15、20 或 25 分钟，也就是 3~5 倍的周期；若没有周期性，应尽可能搜集所有正常操作，t 可设定为一日。此模块搜集传感网络流量以 PCAP 档案的形式储存，可以使用 wireshark [12] [13] [14] 进行检查，如图 7 所示。

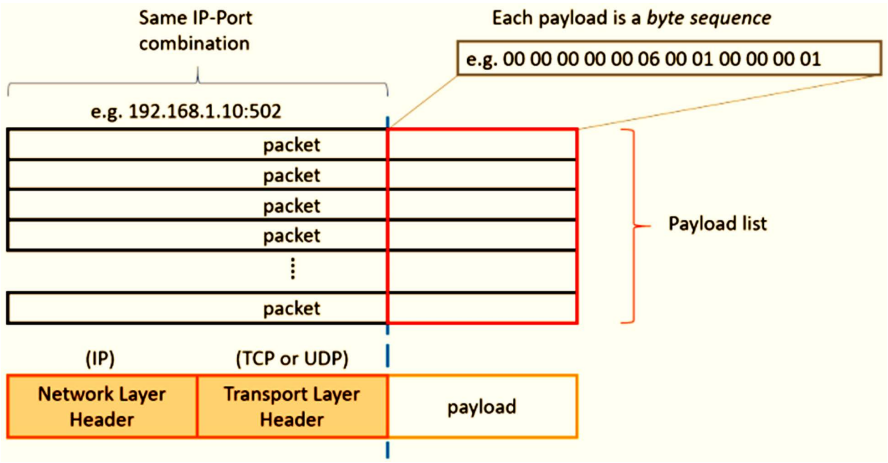

Figure 6. Normal network traffic
图 6. 正常网络流量

2 189.371149	Dell_31:ba:fd	Broadcast	ARP	60 Who has 10.11.101.1? Tell 10.11.101.156
3 190.335161	Dell_31:ba:fd	Broadcast	ARP	60 Who has 10.11.101.1? Tell 10.11.101.156
4 191.335165	Dell_31:ba:fd	Broadcast	ARP	60 Who has 10.11.101.1? Tell 10.11.101.156
5 192.621545	Dell_31:ba:fd	Broadcast	ARP	60 Who has 10.11.101.1? Tell 10.11.101.156
6 193.335225	Dell_31:ba:fd	Broadcast	ARP	60 Who has 10.11.101.1? Tell 10.11.101.156
8 194.335330	Dell_31:ba:fd	Broadcast	ARP	60 Who has 10.11.101.1? Tell 10.11.101.156
13 195.375671	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.1? Tell 192.168.50.5
14 195.383352	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.1? Tell 192.168.50.5
20 195.634678	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.1? Tell 192.168.50.5
21 195.835576	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.5? Tell 0.0.0.0
25 196.835410	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.5? Tell 0.0.0.0
33 197.835545	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.5? Tell 0.0.0.0
37 198.835514	Dell_31:ba:fd	Broadcast	ARP	60 Gratuitous ARP for 192.168.50.5 (Request)
38 198.844941	Dell_31:ba:fd	Broadcast	ARP	60 Who has 192.168.50.1? Tell 192.168.50.5

Figure 7. Check normal network traffic using wireshark
图 7. 以 wireshark 检查正常网络流量

3.3. 正常行为模型训练模块

每一组由正常网络流量搜集模块送来的封包，都预期产生一个相应的正常行为模型(NBM)。正常行为模型的产生分为两个部分，一是频繁方式(frequent pattern)，二是非频繁方式(non-frequent pattern) [15] [16]。每个封包的 payload 可以被视为一个序列数据，称之为字节序列，因为一组封包由多个封包组成，该模型的训练数据为多个字节序列。正常行为模型训练模块的算法 1，大致可以分为四个工作，依序是

数据准备(preparing data)、频繁特征提取(extracting frequent pattern)、非频繁特征获取(extracting non-frequent pattern)、非频繁特征分群(clustering on non-frequent pattern)。输入有四个，*PG*表示正常网络流量搜集模块输出的封包群组(packet group)，*MS*，*ML* 分别表示 Minimum Support 和 Minimum length，是用来调整频繁方式探勘的参数，*CBT* 表示 Cluster Boundary Threshold，是用来调整非频繁方式分群的参数。输出有两个，*FPSet* 表示探勘出的频繁方式组，*NFPCSet* 表示非频繁方式分群的结果。

3.3.1. 异常监测模块

异常监测模块会监视控制网络内的网络流量，根据每个封包的流向，使用对应的 NBM 进行检查，检查的程序如下：

1) 取得一个欲检查的封包，与特征池中所有的特征进行比对，若无一符合，进入步骤(2)；否则，判定该封包为攻击；

2) 从频繁方式集中取出一个尚未比对过的频繁方式，若皆已比对过，判该封包为异常；

3) 比对该封包的数据是否符合该频繁方式，若不符合，重复(2)-(3)步骤；

4) 根据符合的频繁方式的编号(sid)，从非频繁方式集中取出对应的群；

5) 从群落中取出一个尚未比对过的群心和其半径，若皆已比对过，判定封包为异常；

6) 比对该封包数据的后续部分是否被包含在群心和其半径的覆盖范围内，若否，重复(5)-(6)步骤；

7) 该封包判定为正常该程序可表示为算法 2，在算法的最后还加入了攻击特征生成的程序，若异常封包的目的地是蜜罐，就产生攻击特征。输入 *Pin* 表示欲检查的封包，*FPSet*，*NFPCSe* 分别表示 NBM 的频繁 set 与非频繁集群 set，*sigSet* 表示已知的攻击特征集(特征池中所包含的攻击特征)；输出 *isAnomalous* 是一布尔值，表示异常监测的判断结果。

3.3.2. 特征生成模块

流向蜜罐的封包若被判定为异常，会被送进此模块进行特征生成。此模块会将异常封包的数据纪录下来，制作成 Snort 规则的形式，并且称之为攻击特征。若该异常封包有通过频繁方式的检查，那么数据就会以该频繁方式进行切割，至多分成 3 个区段，采取这个做法的用意在于保留频繁方式的信息，让管理员在检查攻击特征时可以知道该攻击与哪个频繁方式有关。举例来说，若数据 0 × 0011223344 符合频繁方式如式(1)，因为该频繁方式位于数据的中间，所以生成的特征就会是三个区段，如下式所示，从中可以看出频繁方式的信息位于第二个区段。

alert tcp any any -> 192.168.1.10 502 (sid: 1000001; content:"|00|";
offset:0; depth:1; content:"|11 22|"; offset:1; depth:2; content:"|33 44|";
offset:3; depth:2;) (1)

产生攻击特征的算法 3，输入Pin表示被判定为异常的封包，$matchfp$表示该异常封包符合的频繁方式，产生的攻击特征会被切割成至多 3 个区段：频繁方式前、频繁方式和频繁方式后，若频繁方式从 offset0 开始，就不会有频繁方式前区段，因此只会有 2 段；若$matchfp$值为$null$，代表此封包不符合任何频繁方式，产出的攻击特征会使用完整的数据(full payload)，也就是只有 1 段。输出$signature$为一字符串，表示产生的攻击特征。为了避免同样的特征重复产生，此模块产出的 Snort 规则会被异常监测模块使用，作为所有封包的第一个检测项目。

4. 实验分析

实验包含 Modbus 和 BACnet 两个通讯协议，Modbus 的部分有 748 个正常封包，314 个异常封包；BACnet 的部分有 400 个正常封包，22 个异常封包。正常封包会以 8:2 比例切割成两组封包，8 成的部分当作正常

行为模型的训练数据，2 成的部分和异常封包合并为测试数据。频繁方式的分析应只需封包数据的前 15 个字节，因此本实验也就固定使用前 15 个字节进行频繁方式探勘。实验内容有 4 种，1) 不同的 MS 与 ML 对异常监测效能的影响，2) 固定 MS 与 ML 时集群边界阈值(Cluster Boundary Threshold, CBT) [17]对异常监测效能的影响，3) 不使用频繁方式，而以完整数据分群的异常监测效能，4) ROC 曲线的比较。实验结果如表 3 所示，虽然完整数据分群在 BACnet 的数据上表现得相当完美，但是在 Modbus 的数据上则相当差。提出的方法先排除频繁方式，能够稳定分群的效果，使分类的效果稳定达到 90%以上的精度。

Table 3. The experimental results
表 3. 实验结果

协议	方法	MS3-ML3-CBT02	MS3-ML5-CBT02
Modbus	频繁方式 + 非频繁方式分群	97.52%	92.06%
	完整数据分群	45.97%	45.97%
BACnet	频繁方式 + 非频繁方式分群	97.05%	93.62%
	完整数据分群	100%	100%

MS 的选择取决于网络中数据包传输的周期性，而 ML 的选择则取决于数据包内容的规律性。从目前的实验来看，似乎 MS 和 ML 均为 3 效果最佳，没有足够的数据来进行更深入的观察。CBT 的选择会影响分类的灵活性，当 CBT 过大时，异常报文会被判断为正常，从而降低召回率。如果 CBT 值太小，则将判断数据包中的任何细微变化为异常，提高误报率。为了平衡两者，倾向于将 CBT 设置为 0.2。尝试在数据中找到常用的方法，期望这种频繁的方式至少包含过程数据的长度信息，而较长的方式将包含功能控制信息，而非频繁的方式则等同于过程数据的一部分。不同之处在于，仅分析和比较单个数据包的数据内容，而不考虑连续数据包的顺序；此外，尽管它们将数据包的内容分为功能控制行为和过程控制行为，但分类是分开进行的。在本文的方法中，同时考虑频繁和不频繁的方法，在没有实际的工业通信数据的情况下，选择使用模拟控制系统来收集 Modbus 数据包作为正常流量，然后使用自制的 Modbus 数据包作为测试数据，通过功能控制行为进行分类的准确性为 92.75%，并且该行为基于过程数据。分类精度为 85.69%。

5. 结论

针对工控系统通信协议中的 TCP 和 UDP 数据，提出了一种通过序列模式探索和聚合层次分组建立正常行为模型并检测网络异常的方法。通过区分频繁和不频繁方式，可以将数据分为两部分：应用程序层标头和过程数据，并且可以提高分组的稳定性。如果不使用频繁方式，而仅使用聚类来构建正常的行为模型，则将其应用于 BACnet 时效果会更好，但在 Modbus 上效果不稳定。CBT 微幅提高就可能使 recall 大幅下降。频繁方式结合非频繁方式分群可以让 recall 提高，但频繁方式可能产生一定量的误报，有利也有弊。此外，提出一种结合蜜罐与特征生成的系统架构，在这个架构下，可以使用正常行为模型来做异常监测，也能在同时将恶意封包的数据以分段的形式记录为 Snort 规则，虽然这样的特征还需要进一步分析、精炼，才能得到较简短、精确的特征，但这种以分段的形式记录的攻击特征，可以说是已经替恶意封包进行了适当的分类，让后续的分析更为容易。但是，此方法用于基于 TCP 和 UDP 的工业控制系统通讯协定时，考虑可以将实验的范围扩展到其它网络协议中，以测试此方法是否普遍适用所有工业控制系统的通讯协定。其次，目前的系统架构虽然能产生攻击特征，但也仅限于将攻击封包的数据分段，如何利用这些分段的攻擊特征精炼出更简单、准确的攻击特征，是一个非常重要的课题。

基金项目

浙江省教育科学规划课题(2021SCG120)、衢州职业技术学院校级项目(QZYY2007、KGXM202009)和衢州市科技计划指导性项目(2020017)给予资助。

参考文献

[1] 尚文利, 杨路瑶, 陈春雨. 面向工业控制系统终端的轻量级组认证机制[J]. 信息与控制, 2019, 48(3): 344-353.

[2] 朱建军, 安攀峰, 万明. 工控网络异常行为的 RST-SVM 入侵检测方法[J]. 电子测量与仪器学报, 2018, 5(7): 8-14.

[3] Huang, K., Zhang, Q., Zhou, C., Xiong, N. and Qin, Y. (2017) An Efficient Intrusion Detection Approach for Visual Sensor Networks Based on Traffic Pattern Learning. *IEEE Transactions on Systems, Man, and Cybernetics*: *Systems*, **47**, 2704-2713. https://doi.org/10.1109/TSMC.2017.2698457

[4] Zheng, Z. and Reddy, A.L.N. (2017) Safeguarding Building Automation Networks: THE-Driven Anomaly Detector Based on Traffic Analysis. 26*th International Conference on Computer Communication and Networks* (*ICCCN*), Vancouver, 31 July-3 August 2017, 1-11. https://doi.org/10.1109/ICCCN.2017.8038393

[5] Wan, M., Shang, W. and Zeng, P. (2017) Double Behavior Characteristics for OneClass Classification Anomaly Detection in Networked Control Systems. *IEEE Transactions on Information Forensics and Security*, **12**, 3011-3023. https://doi.org/10.1109/TIFS.2017.2730581

[6] Mantere, M., Sailio, M. and Noponen, S. (2014) A Module for Anomaly Detection in ICS Networks. *Proceedings of the* 3*rd international Conference on High Confidence Networked Systems*, Berlin, April 2014, 49-56. https://doi.org/10.1145/2566468.2566478

[7] Tonejc, J., Güttes, S., Kobekova, A. and Kau, J. (2016) Machine Learning Methods for Anomaly Detection in BACnet Networks. *Journal of Universal Computer Science*, **22**, 1203-1224.

[8] Mantere, M., Sailio, M. and Noponen, S. (2014) A Module for Anomaly Detection in ICS Networks. *Proceedings of the* 3*rd International Conference on High Confidence Networked Systems*, April 2014, 49-56. https://doi.org/10.1145/2566468.2566478

[9] Ye, T., Jiang, X., Wan, D., *et al.* (2016) Ultrafast Photogenerated Hole Extraction/Transport Behavior in a New Type $CH_3NH_3PbI_3$/Carbon Nanocomposite and Its Application in a Metal Electrode Free Solar Cell. *ChemPhysChem*, **17**, 1-9. https://doi.org/10.1002/cphc.201600817

[10] Caswell, B., Beale, J. and Baker, A. (2006) Snort Intrusion Detection and Prevention Toolkit. Syngress Publishing, Amsterdam.

[11] 岳洋. 基于 Snort 的蜜罐系统的设计与实现[D]: [硕士学位论文]. 哈尔滨: 哈尔滨理工大学, 2010.

[12] Kumar, D., Narwal, P. and Singh, S.N. (2019) A Hidden Markov Model Combined with Markov Games for Intrusion Detection in Cloud. *Journal of Cases on Information Technology*, **21**, 14-26. https://doi.org/10.4018/JCIT.2019100102

[13] 张文安, 洪榛, 朱俊威. 工业控制系统网络入侵检测方法综述[J]. 控制与决策, 2019(11): 2277-2288.

[14] Han, L., Zhou, M., Qian, Y., *et al.* (2019) An Optimized Static Propositional Function Model to Detect Software Vulnerability. *IEEE Access*, **7**, 143499-143510. https://doi.org/10.1109/ACCESS.2019.2943896

[15] Li, W. and Ren, J. (2018) Distributed Frequent Interactive Pattern-Based Complex Software Group Network Stability Measurement. *International Journal of Software Engineering and Knowledge Engineering*, **28**, 619-641. https://doi.org/10.1142/S0218194018500171

[16] Cho, D.J., Han, Y.S. and Kim, H. (2015) Frequent Pattern Mining with Non-Overlapping Inversions. In: Dediu, A.H., Formenti, E., Martín-Vide, C. and Truthe, B., Eds., *Language and Automata Theory and Applications*. LATA 2015. Lecture Notes in Computer Science, Vol. 8977, Springer, Cham, 121-132. https://doi.org/10.1007/978-3-319-15579-1_9

[17] Haslinger, J., Kučera, R., Šátek, V., *et al.* (2018) Stokes System with Solution-Dependent Threshold Slip Boundary Conditions: Analysis, Approximation and Implementation. *Mathematics and Mechanics of Solids*, **23**, 294-307. https://doi.org/10.1177/1081286517716222