计算机科学与探索
Journal of Frontiers of Computer Science and Technology
ISSN 1673-9418,CN 11-5602/TP

《计算机科学与探索》网络首发论文

题目：利用序列分析的远控木马早期检测方法研究
作者：王晨，郭春，申国伟，崔允贺
网络首发日期：2020-09-29
引用格式：王晨，郭春，申国伟，崔允贺. 利用序列分析的远控木马早期检测方法研究. 计算机科学与探索.
https://kns.cnki.net/kcms/detail/11.5602.tp.20200929.0840.002.html

 doi: 10.3778/j.issn.1673-9418.2007087

利用序列分析的远控木马早期检测方法研究

王 晨[1,2]，郭 春[1,2+]，申国伟[1,2]，崔允贺[1,2]
1.贵州大学 计算机科学与技术学院，贵阳 550025
2.贵州省公共大数据重点实验室，贵阳 550025
+ 通信作者 E-mail: gc_gzedu@163.com

摘 要：远控木马是一类以窃取机密信息为主要目的的恶意程序，严重威胁着网络空间安全。现阶段基于网络的远控木马检测方法大多对数据流的完整性有较高的要求，其检测存在一定程度的滞后。本文在分析远控木马通信会话建立后初期流量的序列特性的基础上，提出了一种基于序列分析的远控木马早期检测方法。该方法以远控木马被控端和控制端交互中第一条 TCP 流为分析对象，重点关注流中由内部主机向外部网络发送且数据包传输层负载大于 α 字节的第一个数据包（上线包）及其后续数个数据包，从中提取包含传输负载大小序列、传输字节数和时间间隔在内的三维特征并运用机器学习算法构建了高效的早期检测模型。实验结果表明，该方法具备快速检测远控木马的能力，其通过远控木马会话建立后初期的少量数据包即可高准确率地检测出远控木马流量。

关键词：远控木马；序列分析；早期检测；网络通信行为
文献标志码：A **中图分类号：**TP309.5

王晨，郭春，申国伟，等. 利用序列分析的远控木马早期检测方法研究[J]. 计算机科学与探索
WANG C, GUO C, SHEN G W, et al. Research of Remote Access Trojan Early Detection Method Based using Sequence Analysis [J]. Journal of Frontiers of Computer Science and Technology

Research of Remote Access Trojan Early Detection Method Based using Sequence Analysis

WANG Chen[1,2], GUO Chun[1,2+], SHEN Guowei[1,2], CUN Yunhe[1,2]
1. School of Computer Science and Technology, Guizhou University, Guiyang 550025, China
2. Guizhou Provincial Key Laboratory of Public Big Data, Guiyang 550025, China

Abstract: Remote Access Trojans (RAT) is a kind of malware. The main intent of RAT is to steal confidential information and it seriously threatens the security of cyberspace. Most of current network-based RAT detection methods have high requirement on the integrity of the data stream, and their detection are delayed to a certain extent. Based on the analysis of the sequence characteristics of the initial traffic of RAT after the session is established, this paper proposes a RAT early detection method based on sequence analysis. The proposed method takes the first TCP stream in the interaction between the RAT's controlled and control ends as the analysis object, and focuses on the first packet that is sent from the internal host to the external network in the stream and its transmission layer payload is greater than α bytes (called information return packet) as well as several subsequent packets. In the proposed method, three-dimensional features including transmission payload size sequence, transmission byte and time interval are extracted, and a machine learning algorithm is used to construct an efficient early detection model. Experimental results show that this method has the ability to quickly detect RAT, and it can detect RAT traffic with a high accuracy through a small amount of data packets in the early stage.

The National Nature Science Foundation of China under Grant No. 61802081 (国家自然科学基金项目); The Science and Technology Foundation of Guizhou Province (贵州省科学技术基金 (黔科合基础[2020]1Y268，黔科合基础[2017]1051，黔科合重大[2018]3001)).

Key words: remote access trojan; sequence analysis; early detection; network communication behavior

1 引言

远控木马作为一类危害性极大且知名度极高的恶意程序，主要用于控制目标主机、监控受害者的主机行为以及窃取机密信息等[1]。CNCERT 发布的 2019 年度报告[2]指出，我国境内感染计算机恶意程序的主机数量约 582 万台；ProofPoint 在其 2019 年 Q3 威胁报告[3]中显示，尽管恶意程序数量呈现整体下滑趋势，远控木马的数量相比第二季度却增长了 55%。由此可见，远控木马相比于其他类型的恶意程序仍是在高速增长，俨然成为了互联网所面临的主要安全威胁之一。

与勒索软件等以破坏信息系统可用性为主的恶意软件不同，远控木马以破坏信息系统机密性为主，其集成了键盘记录、文件上传和下载、系统信息窃取、桌面/摄像头监控、进程与注册表修改以及硬盘读写等一系列危险操作。攻击者通过各种手段将远控木马植入到目标主机后即进行潜伏，待接收相关指令后搜集用户隐私数据并回传。由于远控木马具有高隐蔽性的特点，其常被用于 APT 攻击[4]的后渗透阶段以窃取机密信息。

为应对远控木马所引发的安全威胁，近年来国内外研究者提出了一系列远控木马检测方法。基于网络流量的检测方法是现阶段远控木马检测方法的主流[5-8]，但是所提方法中大多运用的是从整个流中提取的统计特征，对木马通信流的完整性要求较高，致使可能出现在检测到远控木马流量的同时，被控主机已然执行部分攻击指令而已经出现了隐私信息泄露的情况。因此，为实现对远控木马的有效防御，对检测方法的检测及时性提出了越来越高的要求。

为及时检测远控木马流量，从远控木马控制端和被控端建立会话后初期的流量中提取特征是一个可行思路。然而，若从远控木马会话建立后初期的流量中所提取的特征不能较好地反映木马的通信行为，则基于这些特征的检测方法容易出现较高的漏报率或误报率。针对上述情况，本文分析了远控木马会话建立后初期的网络流量行为，发现其控制端和被控端通常会在该时间段中出现不涉及人为操作而自动进行的、较为固定的数据包交互行为，且该行为与正常应用同时期的流量存在明显的序列差异。基于上述分析结果，本文提出了一种基于序列分析的远控木马早期检测方法。该方法针对远控木马会话建立后初期流量中“上线包”及其后少量数据包所提取的包负载大小序列、包时间间隔以及包负载上传下载比三个特征，运用机器学习算法建立木马检测模型实现对远控木马的早期检测。

本文主要工作如下：

1)通过对 35 个远控木马与 12 个正常软件的通信行为进行实验分析，发现远控木马和正常软件各自通信会话建立后连接初期的流量在数据包负载大小、数据包时间间隔等方面存在明显区别，并进一步分析了远控木马通信会话建立后流量中第一次出现、且具有较大负载的“上线包”及其之后的数个数据包所具有的特性；

2) 基于远控木马和正常软件在会话建立初期通信流量的差别 ,提出了一种基于序列分析的远控木马检测方法。该方法使用从“上线包”及其后续少量数据包的负载大小序列与时间间隔序列中提取的特征 ,结合机器学习算法实现对远控木马的早期检测；

3)搭建模拟环境对所提出的远控木马检测方法进行了实验测试 ,并探索了“上线包”后不同的数据包数量对于所提方法检测结果的影响。实验结果表明本文所提方法能够以通信会话建立后初期的数个数据包实现对已知和未知远控木马的高准确率检测，有助于及时检测出远控木马流量。

2 相关工作

现今远控木马检测方法主要分为两类，即基于主机的检测方法与基于网络的检测方法。

2.1 基于主机的检测方法

基于主机的检测方法主要通过分析木马源代码构建木马特征库以检测远控木马，亦或是在受控环境下（沙箱、虚拟机等）监测远控木马运行时的主机资源消耗情况、注册表编辑、端口隐藏以及关键 API 调用等敏感行为构建模型进行检测。

Ahmadi 等人[9]通过将程序运行中的 API 调用转化为灰度图并运用图像识别相关技术来检测木马。Matida 等人[10]通过收集恶意程序在沙箱中运行的早期行为来进行恶意程序判别。Canali 等人[11]基于 n-gram 的关联分析法提出了一种木马程序的检测模型。基于主机的检测方法需要将检测系统部署在主机上，除了占用一定主机资源之外，还需要涉及对系统的底层操作，可能对主机的稳定性造成一定影响。并且随着隐蔽技术在远控木马中的应用及发展，

基于主机的检测难度逐渐增加。

2.2 基于网络的检测方法

早期基于网络的检测方法通常采用报文负载匹配技术，提取数据包负载与木马特征库进行匹配以检测远控木马。该类检测方法检测速度快，同时还能具有较高的准确率，但是基于报文负载匹配方法的识别能力依赖于特征库的完备程度，且只能检测已知木马[12]。因此，基于通信行为分析的检测方法更受现阶段远控木马检测研究者的青睐，在网络入侵检测方法[13-14]中也常被运用。

李巍等人[15]通过分析远控木马的通信行为，将远控木马运行过程分为建立连接、命令控制与保持连接三个阶段，从每个阶段中提取不同的统计特征并结合 C4.5 算法实现检测。该方法需要使用完整的数据流，因而其检测存在一定程度的滞后，这也是现今很多远控木马检测方法都存在的问题。姜伟等人在文献[16]及文献[17]中设计了一种基于异常网络行为的远控木马检测模型，但该方法同样需要使用较长的通信流。Jiang 等人[18]将远控木马通信流量中出现在 TCP 三次握手后且相邻数据包时间间隔大于 1 秒之前的会话定义为早期阶段，从中提取了数据包数量、上下行数据包数量比等 6 个统计特征来快速检测远控木马，但该方法的漏报率较高；Adachi 等人[19]在 Jiang 研究的基础上将网络会话与主机进程进行关联，提取另外两个主机特征用于检测远控木马，但是其方法仍存在较高的误报率。宋紫华等人[20]从 TCP 会话的前 5 个数据包中提取了 14 个特征，并以此设计了一种木马快速检测方法，但此方法的检测对象是 TCP 会话，因此需要对多个 TCP 会话进行检测才能识别出具有主从连接的远控木马。吴双等人[21]受文献[22]的启发，以远控木马的人为控制特性为检测出发点，先对数据流进行切片，再以每个切片中前三个包的方向序列来判断数据流是否属于远控木马会话，但该方法需要分析每条流中 250 个数据包以检测远控木马会话。Pallaprolu 等人[22]从每个数据包中提取出特征向量，并运用集成学习对每个分类器的检测结果进行投票来获得高检测率，但由于该方法以会话中全部数据包为分析对象，所以需要很长的训练时间和检测时间。

通过对上述文献的分析可知，现阶段基于网络的检测方法大多对数据流的完整性有较高的要求，其检测存在一定程度的滞后；已有的远控木马早期检测方法则较少考虑数据流的序列特性而仅使用统计特征导致误报率较高。因此，本文重点关注远控木马通信会话建立后初期流量的序列特性，旨在高准确率的前提下及时地检测出远控木马流量。

3 木马通信行为分析

远控木马包括控制端与被控端，植入受害主机的被控端通常会伴随系统的启动而启动，而攻击者通过控制端发送指令与被控端进行交互。早期的远控木马通常由控制端发出连接请求以连接被控端。但是随着防火墙的广泛应用和发展，越来越多的远控木马采用反弹式连接（即由被控端发起连接请求以连接控制端）来避开防火墙的筛查。如图 1 所示，远控木马的运行过程可以划分为建立连接、命令交互和保持存活三个“阶段”。在建立连接阶段，被控端与控制端通过 TCP 三次握手完成连接，之后被控端会主动回传受害者的上线信息；命令交互阶段中部分木马会在主连接存在的情况下建立次连接用于执行指令与回传结果；而在保持存活阶段，攻击者通过设计心跳行为来保持持续的连接，部分木马的心跳包会以一定模式贯穿远控木马整个通信周期。

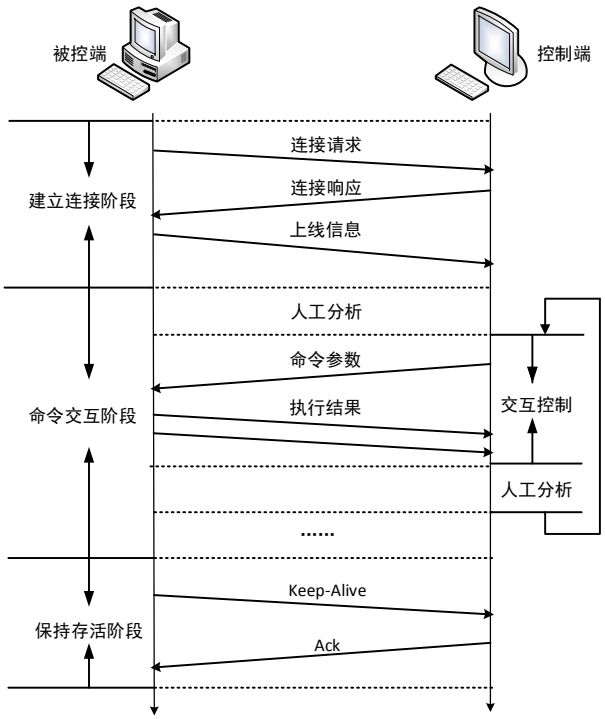

Fig.1 RAT communication flow chart

图 1 远控木马的通信流程图

据实验观察，远控木马的建立连接阶段流量具有以下特性：远控木马为了逃避网络监控软件的查杀，通常会尽可能减少被控端与控制端的交互来隐藏自己，这使得远控木马在会话建立后初期传输的数据量不会太多；建立连接阶段被控端需要反复发出连接请求直至控制端对其进行响应，之后被控端将收集到的受害者主机信息主动回传给攻击者，这种回传信息的数据包的负载通常较大，与其余早期交互的数据包负载大小存在明显区别。本文将由{源 IP 地址、目的 IP 地址以及传输层协议}三元组确定的网络通信流定义为 IP 会话，并将 IP 会话的第一条 TCP 流中由内部主机向外部网络发送且数据包传输层负载大于 α 字节的第一个数据包定义为信息回传包，也称为上线包。表 1 统计了 35 个远控木马在建立连接阶段的上线包的负载大小情况，这些远控木马均使用 TCP 协议进行信息回传。

由表 1 可知，上线包的负载在远控木马建立连接阶段的数据传输总量中占了极大比重，且 orion、mega 等木马在该阶段只进行了信息回传操作。如图 2 所示，不同远控木马的上线包负载大小因其传输内容不同而大小不一，为能够全部覆盖本文所分析远控木马的上线包，本文将 α 设定为 60。远控木马被控端向控制端主动发送上线包后，控制端会发送一个 ACK 包告知被控端已确认接收，在这之后的数个数据包可能出现在以下不同阶段从而具有不同特性：

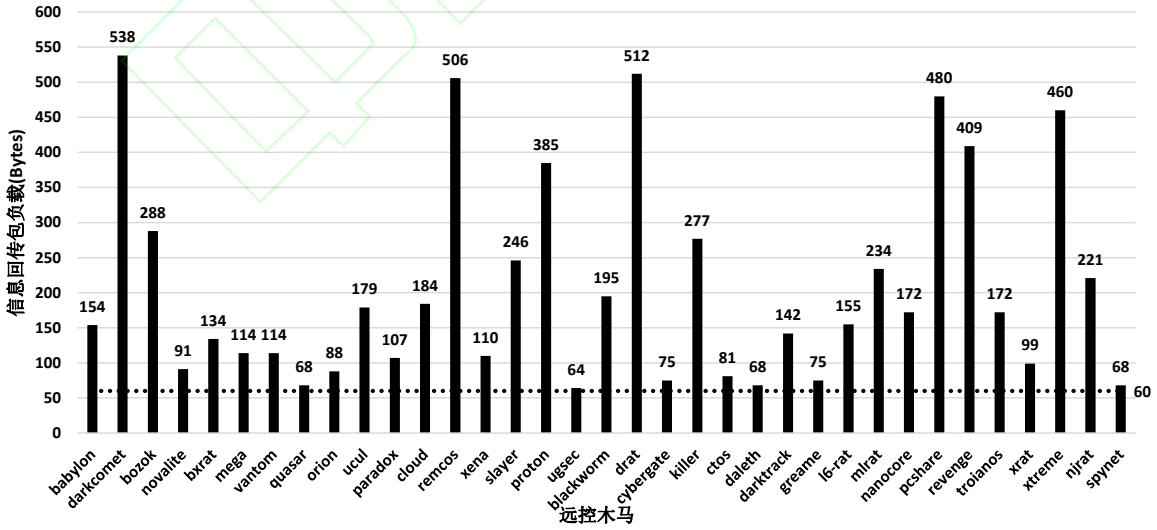

Fig.2 The payload distribution of the information return packet of the used RATs

图 2 实验所用远控木马的上线包负载分布

Table 1 The payload of the information return packet during the phase of connection establishment

表 1 上线包在建立连接阶段中的负载情况

木马	上线包/建立连接阶段数据传输总量(Bytes)	建立连接阶段数据包数	木马	上线包/建立连接阶段数据传输总量(Bytes)	建立连接阶段数据包数
babylon	154/162	6	darkcomet	538/618	6
bozok	288/306	3	novalite	91/105	3
bxrat	134/388	7	mega	114/136	4
vantom	114/136	3	quasar	68/68	2
orion	88/88	2	ucul	179/201	4
paradox	107/107	2	cloud	184/342	14
remcos	506/673	5	xena	110/110	2
slayer	246/246	2	proton	385/687	7
ugsec	64/68	3	blackworm	195/217	7
drat	512/592	11	cybergate	75/115	10
killer	277/277	2	ctos	81/873	83
daleth	68/68	2	darktrack	142/154	6
greame	75/115	10	l6-rat	155/155	2
mlrat	234/2714	17	nanocore	172/476	18
pcshare	480/480	2	revenge	409/409	2
troianos	172/284	4	xrat	99/110	4
xtreme	460/7421	30	njrat	221/221	2
spynet	68/103	10			

1) 建立连接阶段：这些数据包对应控制端自动发出的控制命令交由被控端执行亦或者发生心跳行为，但是由于该阶段下的所有行为均不涉及人为操作，远控木马发生大数据量交互的可能性较低，因而上线包之后数个数据包的负载大小往往小于上线包的负载大小；

2)命令交互阶段：攻击者观察到受害者上线，在经过一定的人为反应时间后将向被控端发送指令以执行相应攻击操作。由于指令对应的数据包大多只包含少许指令参数，其负载通常较小；

3) 保持存活阶段：出现在该阶段的原因是此时攻击者未在线，未能及时观察到被控端上线，通信双方在静默状态下进入了通过心跳包维持通信的保持存活阶段。由于心跳包在大多数情况下不涉及信息交互，其负载通常较小。

由以上分析可知，远控木马在上线包之后的数个数据包倾向于使用小包传输，而正常应用程序为实现资源快速交互会在建立连接后进行大量数据交换行为，其第一个传递数据的“大包”发生之后通常会继续传递大量由正常应用服务端发出的响应信息，由于不需要隐藏自身行为，这些数据的负载通常会很大，这与远控木马形成了鲜明对比。同样，此行为将使得正常程序在这几个数据包的上下行字节比值较小(其中，本文将由远控木马被控端(对应正常应用客户端)向远控木马控制端(对应正常应用服务端)的传输流量统称为上行流量，反之为下行流量)，远控木马则与之相反。另外，对于在建立连接阶段仅进行上线操作的远控木马，上线包与其后的数个数据包之间会因人为反应或心跳间隙造成较大的时间间隔，而正常程序在网络不发生堵塞时较少发生该情况。因此，本文在后续章节中将继续分析远控木马通信会话中上线包及其之后的数个数据包所具有的特性，对其提取特征并结合机器学习算法建立模型及进行检测。

4 基于序列分析的远控木马早期检测方法

4.1 检测方法

基于第三章的分析结果，远控木马在建立连接阶段中存在自动且相对固定的数据包交互行为，且该行为与正常应用存在明显区别。如果能在远控木

马实现命令交互以获取受害者隐私信息前发现其流量，则能够有效降低受害者隐私泄露的风险，达到早期检测的目的。因此，本文在对远控木马通信会话建立后初期的数据包序列进行分析的基础上，运用时序特征与统计特征，提出了一种基于序列分析的远控木马早期检测方法，该方法能够在远控木马会话建立初期及时且高准确率地检测出远控木马流量。由于部分远控木马存在主从连接的情况，为能够及时检测出木马流量，本文将检测单元设定为 IP 会话中的第一条 TCP 流，由{源 IP 地址、源端口、目的 IP 地址、目的端口、传输层协议}五元组确定。如图 3 所示，本文所提出的基于序列分析的远控木马早期检测方法首先对数据包进行过滤并抽取通信会话，选取每个会话中的第一条 TCP 流，以内部主机向外部网络发送的上线包为标志，加上其后数个数据包共同用于提取时序特征及统计特征，然后运用机器学习算法训练检测模型，最后交由训练好的检测模型对待检流量进行检测以判别该流量是正常应用流量还是远控木马流量。

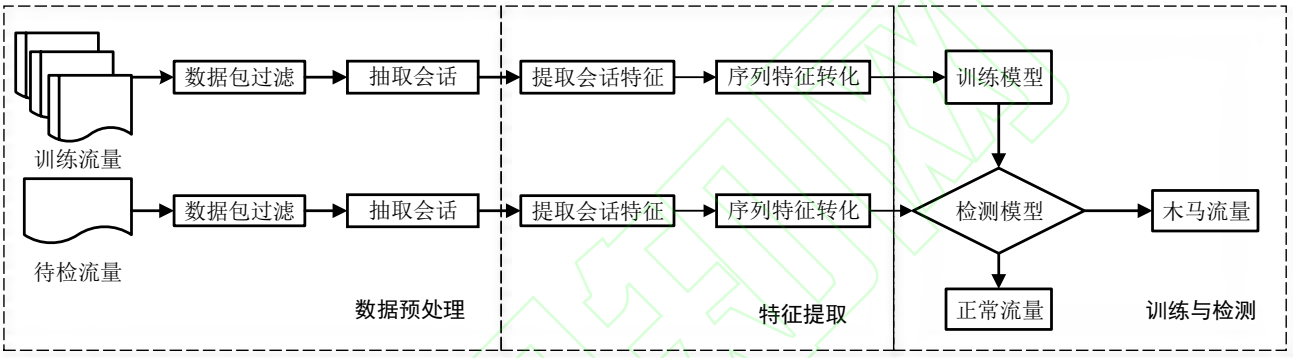

Fig.3　RAT Early Detection Method Based on Sequence Analysis

图 3　基于序列分析的远控木马早期检测方法

4.2　特征提取

本文选择上线包及其后续数个数据包的传输负载大小序列、传输字节数和时间间隔作为特征，具体介绍如下。

4.2.1　传输负载序列

远控木马在建立连接阶段的主要操作目的是向攻击者传输被控主机信息。据实验分析发现，部分远控木马会在连接建立后主动上传受害者信息；部分远控木马则是在接收到控制端自动发出的请求信息之后再发送上线包；少数远控木马还会在发送上线包之前进行确认版本以及握手等操作，但是相比于上线包，这些数据包由于只传递命令参数因而其负载往往较小。本文以上线包的大小作为阈值，将包负载字节数小于上线包大小（60 字节）的数据包称为小包并记为 0，反之则称为大包并记为 1。表 2 统计了 717 条正常会话与 985 条远控木马会话上线包之后 3 个数据包的负载大小序列(S_{Lenth})的分布情况，其中正常会话来源于实验室内部主机流量，木马会话包括 35 个远控木马的流量，表中序列个数小于 3 时表示网络流中上线包后出现的数据包数量不足 3 个。可以看出，正常应用所产生的网络流量的负载序列多集中在“011”与“11”类型，这与正常应用在建立连接后需要接收来自于服务器的资源响应有关；而远控木马出于对自身隐蔽性的考虑，上线包后的数个数据包多为小包。因此，我们将上线包后的数个数据包负载序列作为检测远控木马流量的特征之一。

Table 2　Statistics of the load sequence times of the following three packets after the information return packet

表 2　上线包后续 3 个数据包负载序列次数统计

负载序列类型	出现次数		占比/%	
	正常	木马	正常	木马
000、00、0	36	**722**	5.02	**73.30**
001、01、1	5	129	0.70	13.10
010、10	150	96	20.92	9.75
011、11	**468**	38	**65.27**	3.85
100	25	0	3.49	0
101	5	0	0.70	0
110	13	0	1.81	0
111	15	0	2.09	0
总计	717	985	100	100

为方便机器学习算法进行训练，本文将上线包后的数据包负载大小序列转化为十进制。对一个长度为 m 的数据包负载序列 $d=\{d_1,d_2,d_3.....d_m\}$，将其每个字符 d_i 乘以 2 的 $m-i$ 次方后再进行累加，即对应的特征值应表示为：

$$S_{lenth}=\sum_{i=1}^{m}d_i\times 2^{m-i}\quad 1\le i\le m\ And\ d=1or\,0 \tag{1}$$

4.2.2 传输字节数

如第三章所述，远控木马在上线包之后的数个数据包倾向于使用小包传输。因此，远控木马被动端所发出的上线包及之后交互的数个数据包的上下行字节比(R_{Lenth})通常也会较大。图 4 展示了 4 个远控木马被控端与 5 个正常应用控制端所发出的上线包及其后交互的 3 个包（4Pks）的上下行字节比值情况。由图 4 可知，这些木马流量的上下行字节比值在 10 左右，而正常应用的该值小于 1。这是因为上线包的负载较大，无论远控木马在上线包发出后进入命令控制状态或静默状态，其后产生的数个数据包对应的控制命令包或心跳包均多为小包；而正常应用行为如浏览网页、观看视频以及下载文件时产生的下行方向的数据包的负载较大，故正常应用在这几个数据包的上下行字节比值较小。

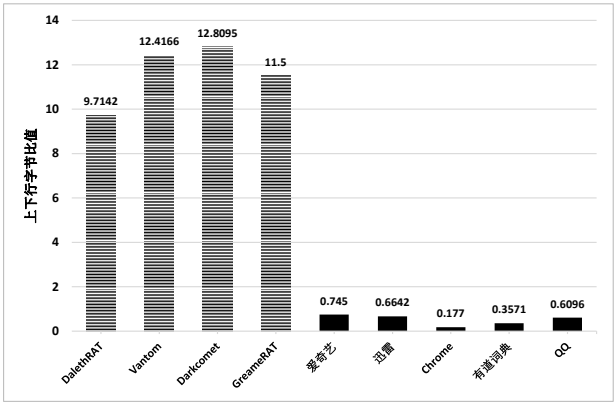

Fig.4 Comparison of upstream and downstream bytes of the information return packet and the next three packets (4Pks)

图 4 上线包与之后 3 个数据包(4Pks)上下行字节比

由于本文方法需要使用的数据包数量较少，有可能出现下行数据包负载之和为 0 的情况。本方法在下行数据包负载之和为 0 时将选择所有上行数据负载之和作为特征值。对一个长度为 m 的数据包负载序列 $d=\{d_1,d_2,d_3.....d_m\}$，当其上行数据负载总和为 U_{lenth}，下行数据负载总和为 D_{lenth}，则上下行数据包负载字节比特征值为：

$$R_{lenth}=\begin{cases}U_{lenth}\ if\ \ D_{lenth}=0\\ \dfrac{U_{lenth}}{D_{lenth}}\ if\ \ D_{lenth}\neq 0\end{cases} \tag{2}$$

4.2.3 时间间隔

有别于建立连接阶段，远控木马在命令交互阶段的行为更多包含攻击者的人为控制。该阶段攻击者可以通过控制端向被控端发送攻击指令执行恶意操作，包括但不限于下载文件、监控摄像头等，不同的恶意操作对应着不同的指令。通过第三章的分析可知部分远控木马在建立连接阶段中仅包含建立连接及上线包操作，之后便进入命令交互阶段或保持存活阶段，但由于远控木马的人为控制特性，故在建立连接阶段到进入命令交互阶段或保存存活阶段之间的数据包间出现相对较长的时间间隔，这是由攻击者需要一定的反应时间或者由攻击者自行设定的心跳间隔所导致的；而正常应用在连接建立初期其客户端和服务端交互频繁，因此处于该时期的

数个数据包的时间间隔会相对较小。该过程的示意图如图 5 所示，远控木马在上线包发出后到接收到控制命令之间需要 t 秒的攻击者反应时间，而正常应用客户端在发出资源请求后服务器会很快返回大量的响应数据包。

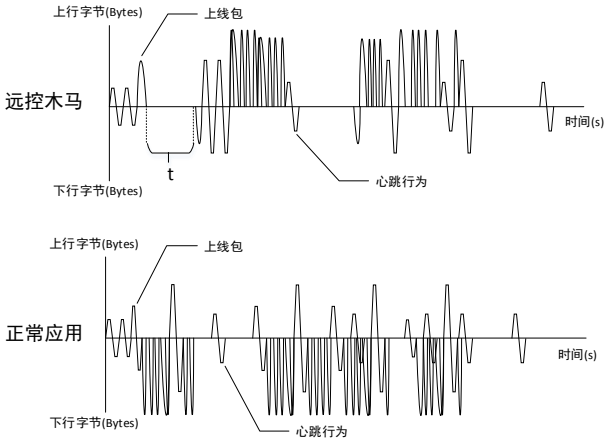

Fig.5 Comparison of RAT and normal application in data stream transmission

图 5 远控木马与正常应用数据流传输对比

因此，本文选取上线包及其后数个数据包中最大的一个时间间隔作为特征，记为 $T_{\text{int }erval}$ 。对一个长度为 $m-1$ 的数据包时间间隔序列 $T=\{T_1,T_2,T_3.....T_{m-1}\}$ ，其最大时间间隔特征值为：

$$T_{\text{int }erval}=MAX\{T_1,T_2,T_3.....T_{m-1}\} \quad (3)$$

综上所述，所提方法提取特征的对象为 n 个数据包(nPks)，即上线包及其后续 n-1 个数据包，共从会话时间与传输字节数 2 个方面提取了 3 维特征，具体描述如表 3 所示。

Table 3 Feature description

表 3 特征描述

类型	特征	描述
会话时间	$T_{interval}$	包含 n 个数据包序列中的最大时间间隔
传输字节数	R_{Lenth}	n 个包负载的上下行字节比
	S_{Lenth}	n 个包负载大小的序列

4.3 训练与检测

本阶段的主要工作是基于上述特征提取阶段所得到的特征向量构建一个检测模型用于区分远控木马流量与正常应用流量。为实现该目标，本文通过搭建实验环境采集了多个知名的远控木马流量和正常应用流量，从这些流量中提取传输负载大小序列、传输字节数和时间间隔等特征并添加类别标签（正常流量或远控木马流量）以构建训练集 D_{tr}（$D_{tr}=\{(x_1,Y_1),...,(x_M,Y_M)\}$，其中 $Y_i\in$ {正常流量，远控木马流量}，$x_i=(T_{\text{int }erval\,i},R_{Lenthi},S_{Lenthi})$）。然后通过 D_{tr} 结合分类算法建立一个远控木马检测模型。为得到高准确度的检测结果，本文分别运用了支持向量机(SVM)、贝叶斯(NB)、K 临近(KNN)、随机森林(RF)以及决策树(DT)等 5 种分类算法训练检测模型并对其检测结果进行对比。

在检测阶段，训练好的检测模型将对测试集中的每条待检流量进行检测，以判别各条流量属于正常应用流量还是远控木马流量。

5 实验

5.1 实验环境

在搭建实验环境时，为保护内网主机安全，本文将远控木马的控制端安装在具有公有 IP 的云服务器上，而被控端安装在局域网内的 Vmware 虚拟机中，虚拟机采用 windows 7 操作系统。检测模型所在主机的硬件配置为 8GB 内存，Intel i7-6700HQ 处理器，使用 Wireshark 抓包，编程语言采用 Python3.7，并使用 scikit-learn 0.22.1 库进行检测模型训练。实验拓扑如图 6 所示。

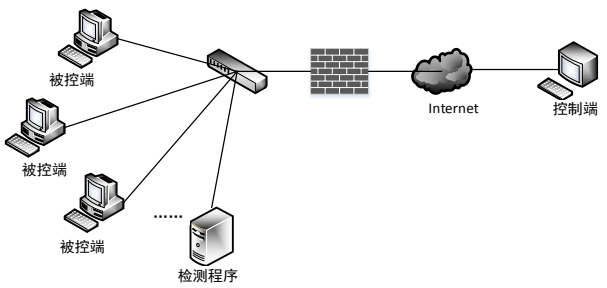

Fig 6 Experimental topology

图 6　实验拓扑

5.2　数据集

本文实验的远控木马程序包括 black_worm、daleth、darkcomet、darktrack、killer、nanocore、njrat、pshare、quasar、remcos、revenge、slayer、spynet、troianos、ugsec、babylon、bozok、bxrat、cybergate、greame、mlrat、mega、paradox、ucul、orion、novalite、drat、proton、cloud、l6-rat、ctos、vantom、xena、xrat、xtreme 等 35 个知名远控木马程序。为体现正常应用流量的覆盖率，本文从浏览器、电子邮件、即时通信、视频软件、P2P 应用、云服务以及游戏等七种不同类型的正常应用中共选取 12 款代表性程序。正常应用类型及选取程序如表 4 所示。实验中正常流量采集于实验室内部主机正常使用表 4 中的正常应用程序时所产生的网络流量。对于远控木马，由于本文所提的检测方法只涉及远控木马会话建立后的少量数据包，所以对每个远控木马每次采集约 5 分钟流量数据。由于正常应用流量在实际通信环境中通常远多于远控木马的流量，因此本文实验将正常流量与木马流量按照 10:1 的比例混合以模拟真实网络环境的流量占比情况。本文将收集到的正常流量和木马流量划分为训练集、已知远控木马测试集和未知远控木马测试集，其中未知远控木马测试集对应未在训练集中出现过的 killer、njrat、xrat、spynet 等 4 个远控木马的流量，用于测试本文方法对于未知远控木马流量的检测能力。具体而言，如表 5 所示，本次实验的训练集包括 5462 个正常应用会话和 452 个远控木马会话；已知远控木马测试集包括 5513 个正常应用会话和 449 个远控木马会话；未知远控木马测试集包含 717 个正常应用会话和 84 个远控木马会话。

Table 4　The normal applications used in experiment

表 4　实验所用正常程序

应用类型	正常程序
浏览器	IE、Firefox、Chorme
电子邮件	网易邮箱大师
即时通信	QQ、微信
视频软件	爱奇艺、QQ 音乐、网易云音乐
P2P 下载	迅雷
云服务	百度网盘
游戏	英雄联盟

Table 5　Experimental data

表 5　实验数据

类别	训练集	已知远控木马测试集	未知远控木马测试集
正常通信	5462	5513	717
木马通信	452	449	84
合计	5914	5962	801

5.3　评估标准

本文采用准确率（Accuracy）、漏报率（FNR, False Negative Rate）以及误报率（FPR ,False Positive Rate）这三个常用评价指标来衡量本文所提方法的检测效果。这些指标可以通过 TP、TN、FP、FN 计算得到。TP 表示远控木马流量被正确检测的数量，FN 表示远控木马流量被检测为正常应用流量的数量，FP 表示正常应用流量被检测为远控木马流量的数量，TN 表示正常应用流量被正确检测的数量。三个指标的含义及具体的计算方法为公式(4)-(6)。

Accuracy 表示预测的准确率：

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

FNR 表示所有预测为正的样本中实际为负的比例：

$$FNR = \frac{FN}{TP + FN} \quad (5)$$

FPR 表示所有预测为负的样本中实际为正的比例：

$$FPR = \frac{FP}{FP + TN} \quad (6)$$

5.4 实验结果及分析

5.4.1 实验结果

本文实验采用 SVM、NB、KNN、RF 和 DT 在内 5 种机器学习算法利用相同的训练集构建检测模型，再使用已知远控木马测试集与未知远控木马测试集来分别测试检测模型对已知远控木马和未知远控木马的检测能力，并进一步分析本文方法在不同数量数据包（上线包及其之后的数据包）下的检测效果。

Table 6 Results of 5 algorithms on the known RAT test set with different packets

表 6 5 种算法在不同数据包数下对已知木马测试集的检测结果

包数 / 算法	3Pks			4Pks			5Pks			6Pks		
	Accuracy	FNR	FPR	Accuracy	FNR	FPR	Accuracy	FNR	FPR	Accuracy	FNR	FPR
NB	0.918	0.129	0.079	0.967	0.176	0.021	0.912	0.176	0.081	0.882	0.206	0.111
SVM	0.924	0.149	0.071	0.926	0.165	0.067	0.958	0.162	0.032	0.945	0.368	0.030
DT	**0.989**	0.051	**0.007**	0.989	**0.026**	0.009	0.986	0.060	0.010	0.982	**0.071**	0.013
KNN	0.985	0.100	0.008	0.986	0.047	0.011	0.976	0.107	0.017	0.973	0.120	0.020
RF	**0.989**	**0.050**	0.008	**0.994**	0.029	**0.004**	**0.989**	**0.056**	**0.007**	**0.987**	0.090	**0.007**

表 6 显示了本文方法在分别使用 3、4、5 和 6 个数据包时(包含上线包)运用不同机器学习算法在已知远控木马测试集上获得的 Accuracy、FNR 以及 FPR 值，其中，使用 4 个数据包(4Pks)时运用 RF 算法获得的 Accuracy 最高，达到 99.4%，而对应的 FNR 与 FPR 分别为 2.9%与 0.4%。

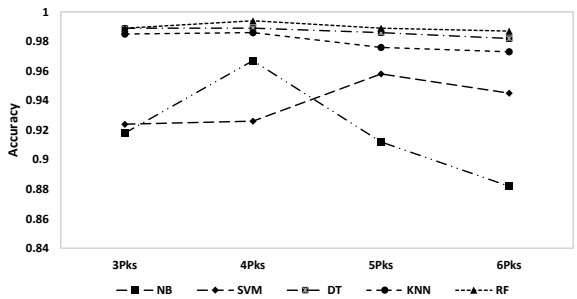

Fig.7 Accuracy of 5 algorithms on the known RAT test set

图 7 5 种算法在已知远控木马测试集上的 Accuracy

如图 7 所示，当本文方法在 3Pks 进行检测时，DT 与 RF 能够实现 98.9%的 Accuracy；当使用数据包数达到 4 时，RF 的 Accuracy 值能够达到 99.4%；RF 使用 5Pks 与 6Pks 分别获得了 98.9%与 98.7%的 Accuracy。值得注意的是，5 种算法中 SVM 与 NB 的 Accuracy 较低，其余 3 种算法均在 4Pks 时取得最佳的 Accuracy。经分析发现，这是因为方法所使用的特征 $T_{interval}$ 随着用于提取特征的数据包数量增加，其对于正常流量和远控木马流量的区分能力有所下降。图 8 给出了本文训练集与测试集中正常与远控木马流量在不同数据包下其特征 $T_{interval}$ 的值大于 1 秒的样本数量。可以看到用于提取特征的数据包数为 4Pks 时，$T_{interval}$ 的值大于 1 秒的木马流量样本多于正常流量样本，而当用于提取特征的数据包数为 5Pks 时，$T_{interval}$ 的值大于 1 秒的流量中正常流量样本占了大部分。因此本文所提检

测方法将检测数据包数设定为 4Pks。

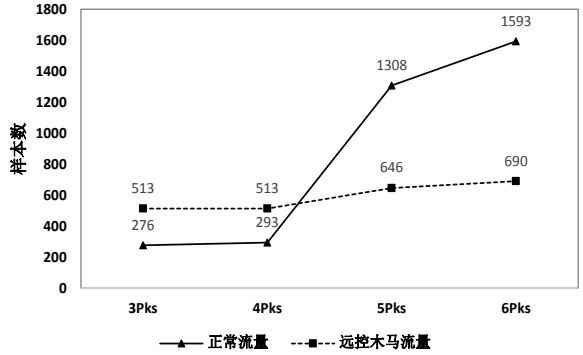

Fig.8 The number of samples whose eigenvalue of $T_{interval}$ above 1s for different packets

图 8 不同数据包下 $T_{interval}$ 的值大于 1 秒的样本数

表 7 为本文方法在 4Pks 时对未知远控木马测试集所得到的 Accuracy、FNR 与 FPR 值。表 7 的结果显示 RF 的检测效果最佳，其 Accuracy 达到 98.5%，而 FNR 和 FPR 分别为 1.2%与 1.5%。

Table 7 Detection results of 5 algorithms using 4Pks on the unknown RAT test set

表 7 4Pks 时 5 种算法对未知远控木马的检测结果

算法	Accuracy	FNR	FPR
NB	0.958	**0.012**	0.046
SVM	0.891	0.131	0.106
DT	0.980	**0.012**	0.021
KNN	0.965	0.071	0.031
RF	**0.985**	**0.012**	**0.015**

为进一步对比所提方法的检测效果，本文采用文献[18]与文献[23]的检测方法分别对相同的流量进行处理并提取特征，其中文献[18]使用 RF 算法训练模型并进行检测；文献[23]使用集成学习方法训练模型，其所用分类算法包括 KNN、J48 决策树与 SVM 三种。表 8 和表 9 分别给出了文献[18]与文献[23]的检测方法对已知与未知远控木马测试集的检测结果。

Table 8 Detection results of different methods on the known RAT test set

表 8 不同方法在已知远控木马测试集上的检测结果

方法	Accuracy	FNR	FPR
文献[18]	97.072%	12.731%	2.293%
文献[23]	98.035%	**0.020%**	2.533%
本文方法[RF+4Pks]	**99.379%**	2.895%	**0.435%**

Table 9 Detection results of different methods on the unknown RAT test set

表 9 不同方法在未知远控木马测试集上的检测结果

方法	Accuracy	FNR	FPR
文献[18]	95.717%	42.682%	0.586%
文献[23]	**98.521%**	**0.080%**	**1.519%**
本文方法[RF+4Pks]	98.502%	1.190%	1.534%

由表 8 的结果可知，本文方法在已知木马测试集上的 Accuracy 和 FPR 分别为 99.379%和 0.435%，优于文献[18]和文献[23]方法所得到的 Accuracy 和 FPR。在 FNR 指标上，由于文献[23]的方法是通过木马控制端和被控端的所有数据包来提取特征，所获得的 FNR 优于仅使用部分数据包来提取特征的文献[18]方法和本文方法；同时，本文方法在 FNR 指标上优于文献[18]的方法。由表 9 可知，在对未知远控木马数据集的检测结果上，本文方法在仅使用会话中少量数据包的情况下获得了 98.502%的 Accuracy 和 1.534%的 FPR，该结果与文献[23]方法使用会话中全部数据包所获得的 Accuracy 和 FPR 相近。因此，从实验结果可以看到，本文方法能够在远控木马控制端和被控端开始通信的初期，通过少量数据包来有效地区分正常应用的通信会话和远控木马的通信会话。

5.4.2 结果分析

本节给出了本文方法分别运用 5 种不同机器学习算法在 4Pks 时所需要的训练时间和检测时间。如表 10 所示，五种算法中最长的训练时长为 SVM 的 0.315 秒，最长的检测时长为 KNN 的 0.04

秒 ,而 RF 的训练时长为 0.045 秒 ,检测时长为 0.017 秒 ,表明本文方法使用这五种算法均能具有高的检测效率。

Table 10 The training and detection time of 5 algorithms on the known RAT test set using 4 Pks

表 10 4pks 时 5 种算法对已知远控木马的训练及检测时长

时间	NB	SVM	DT	KNN	RF
训练时长(s)	0.013	0.315	0.015	0.017	0.045
检测时长(s)	0.010	0.012	0.010	0.040	0.017

此外，为进一步分析本文方法对远控木马流量及时检测的能力，本文对比了文献[18]与文献[23]所用特征需要检测的数据包数量(从通信会话建立后开始统计)，并依据是否有攻击者操作木马分别进行统计，即静默状态与操作状态，结果如表 11 所示。

由表 11 可知，按照文献[18]中对远控木马早期的定义 ,由于正常应用早期不需要隐藏自己的行为 ,通信双方在会话建立后的短时间内即进行大量数据交互 ,导致该方法在正常程序流量中抽取数据包数量远多于本文方法；文献[23]由于其检测对象是数据包级，需要检测通信双方的所有交互数据包 ,而正常应用的交互流量中存在多达几万个数据包的长会话 ,因此该方法所需要检测的数据包数量大且效率较低，无法实现木马通信流量的早期检测；而本文方法关注于上线包 ,仅需要检测通信会话建立后初期的少量数据包。综上 ,本文方法所需的会话数据包数量较少 ,能够较早地检测出远控木马通信流量。

Table 11 Average number of packets need to detected by different methods

表 11 不同方法所需要检测的平均数据包数

方法	类型	所需检测的平均数据包数（个）
文献[18]	正常应用	377.98
	木马(静默状态)	8.6
	木马(操作状态)	8.7
文献[23]	正常应用	会话中全部数据包
	木马(静默状态)	会话中全部数据包
	木马(操作状态)	会话中全部数据包
本文方法 [RF+4Pks]	正常应用	**9.43**
	木马(静默状态)	**7.75**
	木马(操作状态)	**8.15**

6 总结

本文通过分析远控木马会话建立后初期的通信行为 ,发现木马程序与正常应用在该时期内存在数据包序列差异 ,然后进一步提出了一种基于序列分析的远控木马早期检测方法。本文实验通过提取上线包及其后续数个数据包并采用五种不同机器学习算法进行训练和检测 ,实验结果表明本文方法运用 RF 算法能够在 4Pks 时对已知远控木马测试集与未知远控木马测试集分别获得 99.379%和 98.502%的 Accuracy，表明本文方法能够及时且高准确率地检测远控木马流量。后续本文方法将在实际办公环境中进行测试 ,并研究如何进一步降低检测方法对未知远控木马检测的漏报率与误报率。

References:

[1] B. FARINHOLT et al. To Catch a Ratter: Monitoring the Behavior of Amateur DarkComet RAT Operators in the Wild[C]//IEEE. 2017 IEEE Symposium on Security and

Privacy (SP). 22-26 May 2017. San Jose, CA, 2017: 770-787.

[2] 国家互联网应急中心. 2019 年中国互联网网络安全报告[EB/OL]. [2020-07-20] https://www.cert.org.cn/publish/main/upload/File/2019%20CNCERT%20Cybersecurity%20analysis.pdf.

[3] Proofpoint. Proofpoint Q3 2019 Threat Report — Emotet's return, RATs reign supreme, and more[EB/OL]. [2020-07-20]. https://www.proofpoint.com/us/threat-insight/post/proofpoint-q3-2019-threat-report-emotets-return-rats-reign-supreme-and-more.

[4] ZIMBA AARON, Chen Hongsong, Wang Zhaoshun, et al. Modeling and detection of the multi-stages of Advanced Persistent Threats attacks based on semi-supervised learning and complex networks characteristics[J]. Future Generation Computer Systems, 2020: 501-517.

[5] SWE YIN, KHIN & KHINE. Network Behavioral Features for Detecting Remote Access Trojans in the Early Stage[C]//IEEE. Proceedings of the 2017 VI International Conference on Network, Communication and Computing (ICNCC). December 8, 2017. Kunming, China. Association for Computing Machinery, 2017: 92-96

[6] YAMADA M, MORINAGA M, UNNO Y, et al. RAT-based malicious activities detection on enterprise internal networks[C]//IEEE. 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST). 14-16 Dec. 2015. London, UK. IEEE, 2015: 321–325.

[7] Zhu Hongyu, et al. A Network Behavior Analysis Method to Detect Reverse Remote Access Trojan[C]// 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS). 23-25 Nov. 2018. Beijing, China. IEEE. 2018:1007-1010.

[8] YIN K S, KHINE M A. Optimal remote access trojans detection based on network behavior[J]. International Journal of Electrical and Computer Engineering, 2019, 9(3): 2177-2184.

[9] MANSOUR AHMADI, DMITRY ULYANOV, STANISLAV SEMENOV, et al. Novel Feature Extraction, Selection and Fusion for Effective Malware Family Classification[J]. Computer Science, 2016, 8(3): 183-194.

[10] RHODE M, BURNAP P, JONES K. Early Stage Malware Prediction Using Recurrent Neural Networks[J]. Computers & Security, 2018(77):578-594.

[11] CANALI D, LANZI A, BALZAROTTI D, et al. A quantitative study of accuracy in system call-based malware detection[C]//International Symposium on Software Testing and Analysis. Jul 2012. Minneapolis, MN, United States. ACM, 2012:122-132.

[12] VIDAL J M, OROZCO A L S, VILLALBA, LUIS JAVIER GARCÍA. Alert Correlation Framework for Malware Detection by Anomaly-based Packet Payload Analysis[J]. Journal of Network and Computer Applications, 2017:11-22.

[13] CHEN Hong, CHEN Jianhu, XIAO Chenglong, WAN Guangxue, XIAO Zhenjiu. Intrusion Detection Method of Multiple Classifiers Under Deep Learning Model. Journal of Frontiers of Computer Science and Technology, 2019, 13(7): 1123-1133.

[14] P. Santikellur T. Haque M. Al-Zewairi and R. S. Chakraborty, Optimized Multi-Layer Hierarchical Network Intrusion Detection System with Genetic Algorithms[C]//2019 2nd International Conference on new Trends in Computing Sciences (ICTCS). 9-11 Oct. 2019 Amman, Jordan. 2019:1-7.

[15] LI Wei, LI Li-hui, LI Jia, LIN Shen-wen. Characteristics Analysis of Traffic Behavior of Remote Access Trojan in Three Communication Phases[J]. Netinfo Security, 2015, 15(5): 10-15.

[16] Jiang Wei, Wu Xianda, Cui Xiang, et al. A Highly Efficient Remote Access Trojan Detection Method[J]. International Journal of Digital Crime and Forensics, 2019, 11(4):1-13.

[17] Jiang Wei,A Highly Efficient Remote Access Trojan Detection Method, CN107370752A,[P]. 2017-11-21.

[18] Jiang Dan and K. OMOTE. An Approach to Detect Remote Access Trojan in the Early Stage of Communication[C]//2015 IEEE 29th International Conference on Advanced Information Networking and Applications.

24-27 March 2015. Gwangiu, South Korea. IEEE, 2015: 706-713.

[19] ADACHI D, OMOTE K. A Host-Based Detection Method of Remote Access Trojan in the Early Stage[C]// 12th International Conference on Information Security Practice and Experience (ISPEC 2016). November 18, 2016. Zhangjiajie, China. Springer Verlag, 2016: 110-121.

[20] SONG Zi-hua, GUO Chun, JIANG Chao-hui. A Fast Trojan Detection Method Based on Network Traffic Analysis[J]. Computer and Modernization, 2019(06): 9-15

[21] Wu Shuang, Liu Shengli, Lin Wei, et al . Detecting Remote Access Trojans through External Control at Area Network Borders[C]//ACM/IEEE Symposium on Architectures for Networking and Communications. 18-19 May 2017. Beijing, China. IEEE, 2017:131-141

[22] BEAUCHESNE N, PRENGER R J. Method and system for detecting external control of compromised hosts: US, 9407647, B2[P]. 2015-9-17.

[23] PALLAPROLU S C, NAMAYANJA J M, JANEJA V P , et al. Label propagation in big data to detect remote access Trojans[C]//2016 IEEE International Conference on Big Data (Big Data). 5-8 Dec. 2016. Washington, DC, USA, 2016:3539-3547.

附中文参考文献:

[13] 陈虹，陈建虎，肖成龙，万广雪，肖振久. 深度学习模型下多分类器的入侵检测方法[J]. 计算机科学与探索, 2019, 13(7): 1123-1133.

[15] 李巍,李丽辉,李佳,林绅文. 远控型木马通信三阶段流量行为特征分析[J]. 信息网络安全,2015, 15(05):10-15

[17] 姜伟. 一种高效的远控木马检测方法：中国，107370752A,[P]. 2017-11-21.

[20] 宋紫华,郭春,蒋朝惠.一种基于网络流量分析的快速木马检测方法[J].计算机与现代化,2019(06):9-15.

WANG Chen was born in 1997. He is an M.S candidate at School of Computer Science and Technology, Guizhou University. His research interests include network security machine learning, etc.

王晨(1997-)，男，贵州大学计算机科学与技术学院硕士研究生，主要研究领域为网络安全、机器学习。

GUO Chun was born in 1986. He is an associate professor and M.S. supervisor at School of Computer Science and Technology, Guizhou University, as well as the member of CCF. His research interests include data mining, intrusion detection, malware detection etc.

郭春(1986-)，男，博士，贵州大学计算机科学与技术学院副教授、硕士生导师、CCF 会员，主要研究领域为数据挖掘、入侵检测、恶意代码检测。

SHEN Guowei was born in 1986. He is an associate professor and M.S. supervisor at School of Computer Science and Technology, Guizhou University, as well as the member of CCF. His research interests include network and information security and big data.

申国伟(1986-)，男，博士，贵州大学计算机科学与技术学院副教授、硕士生导师、CCF 会员，主要研究

领　　域为网络与信息安全、大数据。

CUI Yunhe was born in 1987.He is a lecturer and M.S. supervisor at School of Computer Science and Technology, Guizhou University. His research interests include network security, cloud computing, data centers etc.

崔允贺(1987-)，男，博士，贵州大学计算机科学与技术学院讲师、硕士生导师，主要研究领域为网络安全、云计算、数据中心。