Analyzing and Exploiting Network Behaviors of Malware

Jose Andre Morales[1], Areej Al-Bataineh[2], Shouhuai Xu[1,2], and Ravi Sandhu[1]

[1] Institute for Cyber Security, University of Texas at San Antonio
{jose.morales,ravi.sandhu}@utsa.edu
[2] Department of Computer Science, University of Texas at San Antonio
{aalbata,shxu}@cs.utsa.edu

Abstract. In this paper we address the following questions: From a networking perspective, do malicious programs (malware, bots, viruses, etc...) behave differently from benign programs that run daily for various needs? If so, how may we exploit the differences in network behavior to detect them? To address these questions, we are systematically analyzing the behavior of a large set (at the magnitude of 2,000) of malware samples. We present our initial results after analyzing 1000 malware samples. The results show that malicious and benign programs behave quite differently from a network perspective. We are still in the process of attempting to interpret the differences, which nevertheless have been utilized to detect 31 malware samples which were not detected by any antivirus software on Virustotal.com as of 01 April 2010, giving evidence that the differences between malicious and benign network behavior has a possible use in helping stop zero-day attacks on a host machine.

1 Introduction

The ever growing sophistication of malware, especially zero-day attacks, with faster distribution and stealthier execution has forced signature based detection in an uphill battle that is difficult to win. Behavior based detection is increasingly being used by commercial software vendors with some success but is partially reliant on understanding the behavior of known malware to attempt detecting future attacks.

This research analyzes known malicious and benign samples in an attempt to exploit differences in their network behavior to accomplish accurate behavior based malware detection. The data set consisted of 1000 malware samples, including 31 not detected by any antivirus software on Virustotal.com on 01 April 2010 and 123 benign samples. The analyzed data included DNS, NetBIOS, TCP, UDP, ICMP and other network traffic. For each analyzed malware and benign sample, we collected occurrence amounts of basic network functions such as total number of DNS queries and NetBIOS query requests. Observations of captured network activity and occurrence amounts were analyzed and correlated to identify network behaviors occurring mostly in malware. We use clustering and classification algorithms to evaluate how effectively our observed network behaviors can differentiate malware from benign samples.

Given our observed network behaviors, our clustering and classification produced minimal false positives and false negatives. In addition, 31 malware samples not

S. Jajodia and J. Zhou (Eds.): SecureComm 2010, LNICST 50, pp. 20–34, 2010.

identified by any antivirus software on Virustotal.com on 01 April 2010 were correctly clustered and classified using our observed network behaviors. These results give evidence that the observed differences between malicious and benign network behavior can be useful in stopping zero-day attacks on a host machine.

The principal contributions of this research are:

1. Identification of network behaviors occurring mostly in malware usable in behavior based malware detection.
2. Discovery of novel malicious uses of network services by malware.
3. Evaluating the effectiveness of observed network behaviors in identifying malware and benign processes with clustering and classification.

This research presents early results of one perspective of an ongoing project dealing with malware behavior based on a sample size of 1000 malware and 41 benign processes. The benign processes were executed three times each for a total of 123 instances which were used as samples for our analysis. The goal of this ongoing research is a real time behavior based malware detection system incorporating several perspectives capable of detecting known and unknown malware on host machines.

The rest of this paper is organized as follows: Section 2 gives related work, Section 3 describes our data set, Section 4 presents our network behaviors, Section 5 gives our clustering and classification results, Section 6 discusses our approach and results, Section 7 gives limitations and Section 8 is conclusions and future work.

2 Related Work

The research of Bayer et. al. [3] presents a dynamic malware analysis platform called Anubis which is used to collect behaviors of known malware samples in a controlled environment. This system inputs a binary executable and records API invocation, network activity, and data flows. The results are used to report observed behaviors principally on the file system, registry, and network activity. The reported network activity only provides data on usage of protocol traffic, connections to remote servers, file downloads, port scanning and other typical network tasks. The results give direction as to which forms of network activity should be monitored closely for malicious events. Using Bayer et. al. as motivation, we analyzed and produced occurrence amounts of basic network functions which were used to aid in defining our set of network behaviors.

Malware binary analysis platforms such as Anubis [1], Malheur [13], Bitblaze [4] and CwSandbox [24] are designed primarily to run known malware samples in a controlled environment and record execution behavior. Recording is done via various techniques from API hooking to monitoring modifications and data flows in various OS components. These platforms record general network activity behavior which are reported to the user. The reports do not include sufficient detailed information to identify malware's precise implementation and use of network services making it difficult to discover novel malicious acts captured in network activity. Our research fills this gap by capturing finely grained network behavior facilitating detailed analysis which was key in our discovery of novel network behaviors that successfully detected several malware samples.

The research presented by Morales et. al. [15] analyzes a specific form of network activity behavior called RD-behavior which is based on a combination of DNS activity and TCP connection attempts. The authors found bot processes often use reverse DNS queries (rDNS) possibly to harvest new domain names. The rDNS often fails and is then followed by a TCP connection attempt to the input IP address of the failed rDNS, the authors regard this as an anomalous behavior. This anomalous behavior is successfully used by the authors to detect bots and non-bot malware. The approach in [15] was limited when the authors removed one of their defined behavior: a failed connection attempt to the returned IP address of a successful DNS query. Our results revealed an almost total absence of rDNS usage and several instances where malware used the removed behavior. Using this behavior helped raise our detection accuracy.

The research of Zhu et. al. [28] detected bots with a host-based technique based on high number of failed connection attempts. Measuring the connection failure rate of bots and benign processes showed that successful bot detection is achievable using only this metric. Measuring failed connection attempts may only be effective with bots that are totally or partially inactive while fully active up to date bots and other malware with little or no failed connection attempts may go undetected by this approach. Our research relates failed connection attempts with DNS, NetBIOS and other network behaviors creating a more robust approach to malware analysis and detection.

A broad corpus of research exists analyzing and detecting malware samples, families and categories [8,17,11,9,22,6,14,12,16,18,7,2,23]. All use different perspectives to measure, analyze and detect malware using host-based, network-based and hybrid approaches. Our research enhances the current literature by relating different specific network activities together to define network behaviors mostly used by malware.

3 Data Set Analysis

Our analysis is based on 1000 known malware samples and 41 benign samples. The benign samples were executed three times each for a total of 123 instances which were used as samples for our analysis. The malware samples were acquired by downloading the first 969 samples from the CWSandbox sample feed on 27 October 2009 [24]. The upload date was arbitrarily chosen. The set contains a broad range of malware types including: bots, backdoors, malware downloaders, keyloggers, password stealers and spyware amongst others, Table 1 lists prominent malware in the data set. Uploading the MD5 sums to Virustotal.com provided malware names from Kaspersky, McAfee and Symantec. We also downloaded 31 malware samples from the 31 March 2010 upload on CWSandbox malware repository. These 31 were chosen because their MD5 sums, listed in Table 2, were reported as undetected by all antivirus software used by Virustotal.com on 01 April 2010 and we were capable of executing and capturing their network behavior in our testing environment. The majority of our malware samples had successful network activity during the collection period connecting with remote hosts and conducting malicious deeds.

The benign test set, also listed in Table 1, covered a wide range of popular and daily used network active applications including: web browsers, FTP clients, RSS readers, social network clients, antivirus software, Peer-to-Peer (P2P) clients and standard network tools amongst others. We captured network activity in VMWare Workstation with

Table 1. Prominent malware and benign samples in data set

Prominent malware samples in data set			
Downloaders	**Bots**	**Worms**	**Hybrids**
Bifrose.bmzp	Koobface.d	Iksmas.bqs	Krap.n
PcClient.ahqy	Padobot.m	Mydoom.m	PolyCrypt.b
Poison.pg	Virut.by	Allaple.a	Refroso.qj
Turkojan.il	Zbot.acnd	Bacteraloh.h	Scar.hez
Genome.cehu	Buzus.amsz	Palevo.ddm	
CodecPack.ill			
Lipler.fhm			
Adware	**Scareware**	**Rootkits**	**Viruses**
FenomenGame	SystemSecurity.cc	Tdss.f	Sality.aa
BHO.nby	XpPoliceAV.apd		
Monderd.gen			
Benign samples in data set			
Adobe Reader	Ares	Avant	BitTorrent
Chrome	CuteFtp	DeskTube	Facebook Desktop
FileZilla	FireFox	FlickRoom	Flock
Google Talk	Google Update	IE explorer	Kaspersky Security
K-Meleon	LimeWire	Ping	PPLive
PPStream	RSSBandit	Skype	Snarfer
Snitter	SopCast	Spyware Dr.	Stream Torrent
Streamer radio	TortoiseSVN	Traceroute	TVants
Tvkoo	TVUPlayer	TweetDeck	Twhirl
uTorrent	UUSee	Win Player	Win Update
Zultrax			

Windows XP SP2 using Windows Network Monitor along with proprietary network layer monitors to record the network activity for an execution period of 10 minutes for each data set sample. The individual samples were manually executed one at a time in VMWare Workstation with our monitors collecting all network traffic and the captured data was saved to a local repository for analysis. The benign processes were installed, used under normal conditions and updated (when available) during testing.

The network activity of the group of 969 malware samples was collected between 27 October 2009 and 01 November 2009, the network activity of the group of 31 malware samples was collected on 01 April 2010, and the network activity of the group of 41 benign samples was collected between 01 April 2010 and 03 April 2010. Collecting network behavior of the malware samples was done immediately after downloading the samples to assure the samples were still active, meaning the malware would still connect with remote hosts and conduct malicious deeds producing network traffic. The vast majority of our malware samples, over 95%, produced network traffic which was the basis of our analysis.

Table 2. MD5 sums of data set malware samples not detected on VirusTotal.com

31 malware not detected on Virustotal.com - 01 April 2010	
732e014e309ffab8ed9a05198d060a0b	ce1cd380910e28092f880643ec1f809d
94004413140e2022c0880f3828a1c0ee	cbed573de18b900cd91cc9e4558fb645
bcebf381a36099f697d2e00f3ec4f26e	7a84fd3ff0aa487ae2142e7130c78d9f
2fbea182c4c7d47419b2c25d72eb64bc	6d25e4a5db130cda772e09d458afacad
8a98176d289e099ccf359aaed06daf9e	bdd7bd56d65471b594c0822dd434a84f
037629b54b5714457ff2abefdab0c349	6b24b3779730f4add8d562daa1bc0ddf
7407c24f17d7c582901623c410ab7a91	8189e6f967b612e5ee7a74981278de4a
36a256686620fa7d3b9433af19cf57a2	5cfb57eac56c8639329d9ecab7b7f4ac
cde17b3c02d6143a9c1fa22eedad65ac	fbc377f7010b6a3216f7fd330dcfe69e
2e3108689a758c629286ef552e89b858	0b15d6658f306cfea3fe20bd32c91a0d
ae7d5ad001c26bbda2f32610f28484b9	9207e79e1f2191d3d44343482ab58a4e
25181c8ed97357b52ea775bc5dca353c	2bbb004cc926a071bda327ca83bf03fb
b0c89519569ce2e310958af0e5932ed1	e73da6feae4fabd251bb19f39c1a36d3
d2ebbc7609672d46e7bb8b233af585aa	e38c4a027b5a570eae8c57de8e26fcbb
bc8aa3e072fbec4045bf94375ac53be9	018197ab7020625864e6f4ff65611fc7
5dae2c8bf87e6a9ad44e52d38ac3411e	

4 Network Behavior

This research analyzes known malware and benign samples in an attempt to exploit differences in their network behavior to accomplish accurate behavior based malware detection. Differences in network behavior were identified through manual post analysis of collected network traffic. The captured network activity of our data set contained typical protocols such as TCP, UDP, and DNS but they were not always used in the normal expected way, most notably in our malware samples. We were able to collect occurrence totals of basic network functions and correlate together different occurrence amounts of specific network activity to identify network behaviors which, according to our results, occurred more often in malware than benign samples. The identified network behaviors, defined as B_n where n is an identification number, are described below.

4.1 DNS and NetBIOS

The Domain Name System (DNS) and Network Basic Input/Output System (NetBIOS) provide services to acquire IP addresses when a domain name is provided and vice versa [5,19]. Coarse-grain occurrence amounts of both protocols by known malware has been previously shown [3,15]. Table 3 summarizes our occurrence amounts for DNS queries, reverse DNS queries and NetBIOS name requests. The analysis revealed 100% of benign processes and 77% malware issuing DNS queries mostly due to malware's use of other network services, such as NetBIOS and ICMP, to acquire IP addresses for connection attempts. The benign samples with failed DNS queries were web browsers unable to reach third party content and P2P video and audio streamers unable to locate

remote hosts for a specific stream. Several malware samples had failed DNS queries, most were domain names of malware servers that were either not active or previously discovered and shut down. Reverse DNS queries (rDNS) were notably absent with only 2% of malware and no benign samples. This contradicts the findings of [15] which documented bots and non-bot malware performing rDNS and conjectured these queries were an essential component to establish malicious network activity. It can be inferred, from testing our samples, that the current generation of malware may possibly be less reliant on rDNS in favor of other techniques providing the same IP address and domain name related information.

Analyzing the occurrence totals of NetBIOS name requests (NBTNS) revealed 56% of malware and 4% of benign samples implemented this activity. The benign samples with NetBIOS name requests were the web browsers Google Chrome with fifteen name requests and Firefox with six name requests. Further analysis revealed the domain names used in the NetBIOS name request of Google Chrome and Firefox had first been used in a DNS query with some failing and others succeeding. The malware samples revealed two distinct forms of NetBIOS name request usage: (i) expected usage, same as benign, and (ii) performing NetBIOS name requests on domain names that were not part of a captured DNS or rDNS query. To our knowledge, the second form is a novel observation of NetBIOS use by malware not presented in previous research. Of the 1000 malware samples, 49% exhibited the second NetBIOS usage described here. We concluded this was a network behavior occurring mostly in malware and usable for detection. Based on this, we define the following network behavior:

– B_1: A process performs a NetBIOS name request on a domain name that is not part of a DNS or rDNS query.

Table 3 shows B_1 occurring only in malware, with 49%. Using online malware databases such as MalwareURL.com, we found many domain names used by our malware samples in B_1 identified as malware servers, but several other domains did not show up leading us to believe they were recently created and registered, inactive, had avoided detection, were infected hosts, or newly activated servers. We conjecture malware uses behavior B_1 in an attempt to acquire remote host information while avoiding detection by anti-malware that may not monitor NetBIOS but most probably does monitor DNS.

Table 3. Samples with DNS, NetBIOS, & B_1

Samples with	**Malware 1000 samples**	**Benign 123 samples**
DNS queries	77%	100%
Reverse DNS queries	2%	0%
NetBIOS name requests	56%	4%
Behavior B_1	49%	0%

4.2 RD-Behavior

This network behavior as originally defined [15] was primarily based on frequent usage of reverse DNS queries (rDNS) by bots. The authors defined four network behavior paths of which three included rDNS. Their results implied rDNS combined with TCP connection attempts was sufficient to detect malware and eliminated false positives by omitting the only behavior path dealing solely with DNS queries. Our analysis revealed a notable absence of rDNS and a high occurrence of DNS queries, see Table 3, many of which exhibited the omitted behavior. We conjecture better detection can be achieved by including all four behaviors from [15] redefined as follows:

- B_2: Failed connection attempt to an IP address obtained from a successful DNS query.
- B_3: Failed connection attempt to the input IP address of a successful rDNS query.
- B_4: Connection attempt to the input IP address of a failed rDNS query.

In [15] behavior path P_5 is defined as: A successful connection to an IP address used in a failed rDNS query and behavior path P_6 is defined as: A failure to connect with an IP address used in a failed rDNS query. We reduced the number of network behaviors by combining behavior paths P_5 and P_6 into one network behavior B_4. Behavior B_2 implies a successful connection should occur to IP addresses obtained in successful DNS queries, a failed connection attempt indicates something is not right and should be investigated. Malware can exhibit this behavior when domain names have been shut down or taken offline and their DNS records have not been updated or removed. Behavior B_3 has the same implication as B_2 but with the input IP address of rDNS queries. Behavior B_4 is assumed to only occur in malware. We assuem an input IP address failing an rDNS query as unreachable and should not be used for connection attempts. Table 4 shows total number of processes with behaviors B_2, B_3 and B_4. Our occurrence amounts showed 21% of malware and no benign samples with B_2 and no occurrences of B_3 and B_4 due to very low rDNS usage. These results imply rDNS may be used less often by malware in favor of other techniques providing the same information in a more clandestine manner.

Table 4. Samples with behaviors B_2, B_3 & B_4

Samples with	**Malware 1000 samples**	**Benign 123 samples**
Behavior B_2	21%	0%
Behavior B_3	0%	0%
Behavior B_4	0%	0%

4.3 UDP and ICMP

Traffic between local and remote hosts using captured User Datagram Protocol (UDP) [25] did not serve a significant role, except for DNS and rDNS, in our analysis due to similar occurrence amounts of network activity in both malware and benign. Previous research [3] has documented coarse-grain UDP occurrence amounts by malware,

but does not include a comparison with benign processes. Identifying network activity behaviors in the UDP protocol is part of our ongoing research.

The occurrence amounts of Internet Control Message Protocol (ICMP) [10] activity, which focused on ICMP echo requests and replies, revealed an elevated usage by the malware samples in comparison to the benign samples. Further analysis concluded that malware was using ICMP echo requests in the same manner as the Ping network utility [20] to decide if a remote host was reachable, thus being a candidate for a connection attempt. Malware use of ICMP has been previously observed [27] but was not distinguished as a behavior frequently used by malware in comparison to benign. Our analysis showed malware never attempted connections to IP addresses not receiving a reply to an ICMP echo request and almost always attempted to connect with IP addresses that did have a successful reply. Furthermore, the input IP address of the echo requests were never part of a DNS or rDNS query or NetBIOS name request leading to conclude these IP addresses were hardwired, dynamically generated, or downloaded from a malware server. Based on these observations, we define two network behaviors as follows:

- B_5: ICMP only activity, ICMP echo requests for a specific non-local network IP address with no reply or a returned error message.
- B_6: TCP/ICMP activity, TCP connection attempts to non-local IP addresses that received a successful reply to their ICMP echo requests.

We assume the IP addresses used in B_5 and B_6 are never part of DNS, rDNS or NetBIOS activity. This assumption is supported by our observations of the captured network activity. The results of this analysis are listed in Table 5. B_5 occurred more often in benign than malware but the benign samples also used ICMP less than malware, perhaps favoring other similar and more conventional services such as DNS queries, see Table 3. B_6 was exhibited in 11% of malware and only 2% benign samples. This supports our claim that malware frequents ICMP use to identify IP addresses for connection attempts. Our observations of B_5 and B_6 are, to our knowledge, novel in the literature not being previously reported.

Table 5. Samples with behaviors B_5 & B_6

Samples with	Malware 1000 samples	Benign 123 samples
Behavior B_5	3%	4%
Behavior B_6	11%	2%

4.4 Other Network Activity

This encapsulates other less occurring activities which were considered significant since they rarely occurred in any of our data set samples or were implemented in a non-conventional way. We consider these network activities to be anomalous and not necessarily malicious behaviors. The value of recording occurrences of these behaviors is in cases where a novel and never before observed, or rarely used malicious behavior occurs in a malware sample. We encompass this idea with the following behavior:

Table 6. Samples with behavior B_7

Samples with	Malware 1000 samples	Benign 123 samples
TCP connection attempts to IP addresses never used in DNS, NetBIOS, ICMP	10%	2%
Listen connections on non-typical port numbers	2%	7%
Successful DNS queries returning local network IP addresses	1%	0%
Use of non-typical network protocols and commands	4%	0%
Behavior B_7	18%	9%

– B_7: Network activity that is rarely occurring or implemented in an anomalous manner.

Table 6 lists the amount of samples exhibiting the different types of observed network activity and B_7. TCP connection attempts to IP addresses which were not part of DNS, NetBIOS or ICMP activity were the most prominent in this group with 10% in malware and only 2% in benign. These malware, upon initial execution, immediately attempted connections to IP addresses ranging from a few to over one hundred different addresses which appeared to have been hardwired or dynamically generated. The benign sample with this activity was the video chat program Skype which connected to a server during installation.

Second most prevalent network activity was use of non-typical protocols and network commands with 4% in malware none in benign. The malware attempted connections using either FTP or SMB or RTCP. These were the only samples from our data set using these protocols except for FTP which is a typical protocol; the reason we documented FTP usage is the malware had a very small amount of FTP activity download from a remote server along with a much lager amount of TCP and UDP traffic.

One malware sample used the authentication system KerebosV5 and one other malware sample used the network command suite Andx. Interestingly, the Andx commands were attempting to authenticate and access local network IP addresses in search of a file server perhaps to host inappropriate content. Listening TCP connections using non-typical port numbers occurred in 2% malware and 7% benign samples. Malware listened on non-typical or private ports [21] such as port numbers: 19178, 24450, 25254, 27145 and 36975; benign also listened on non-typical or private ports such as port numbers: 19396, 33680, 36363 and 58480. Two malware samples performed successful DNS queries on domain names returning local network IP addresses: gogog029.100webspace.net - 127.0.0.1 and probooter2009.no-ip.org - 0.0.0.0. It is unclear if these DNS query results were modified by the malware or if these were intentionally returned by the DNS server. B_7 was exhibited in 18% malware and 9% benign, suggesting rarely or anomalous occuring network activity may be useful in differentiating malware and benign.

5 Clustering and Classification

To evaluate how effectively our observed network behaviors can differentiate between malicious and benign samples, we input the data through clustering and classification algorithms using the Weka data mining software [26]. Clustering and classification algorithms are extensively used in the literature to evaluate proposed host, network and hybrid detection approaches and are well established as accurate indicators of effectiveness and efficiency of a proposed detection approach. Our data set consisted of the occurrence amounts of network behaviors B_1 through B_7, discussed in Section 4, for each malware and benign sample. The complete data set was used for clustering; for classification, the training set contained the first 700 malware samples and 40 benign with the test set containing the remaining samples. The 31 undetected malware samples were not part of the training set. Some of the samples in the test set not found in the training set are listed in Table 7.

Table 7. Some of the malware and benign samples in test set and not in training set

Malware samples	Benign samples
BHO.nby	Adobe Reader
Mabezat.b	BitTorrent
Monderd.gen	Chrome
Poison.pg	CuteFtp
Swizzor.a (2)	Facebook Desktop
Turkojan.il	FlickRoom
VB.bfo	Kaspersky Security
VB.vr	Skype
31 undetected malware	SopCast
	TVants

Table 8. Top three clustering results with 1000 malware and 123 benign samples

Clustering algorithm	Number of clusters	True positives	True negatives	False positives	False negatives	FP rate	FN rate
DBScan	8	119	1000	4	0	0.4%	0%
Expectation maximization (EM)	4	123	988	0	12	0%	1%
Xmeans	3	123	1000	0	0	0%	0%

5.1 Clustering Results

The data set was input to the complete suite of clustering algorithms in Weka. The top three results are listed in Table 8. False positives and false negatives were determined by observing if the majority of a cluster was composed of malware or benign samples. If malware was the majority then the benign samples were classified as false positives; if benign was the majority then the malware samples were classified as false negatives.

DBScan and EM algorithms produced encouraging results with no false negatives in the first and no false positives in the second algorithm. The four false positives produced by DBScan were SopCast, TVUPlayer, UUsee media center, and TVants. All of these are video streamers whose content source comes from several IP addresses which are constantly changed and removed, making it difficult to keep up to date. This is very similar to IP addresses used by malware authors, especially in botnets [18], which constantly change primarily to avoid detection. All four were grouped in one cluster with many different classes of malware, the samples in this cluster exhibited many instances of behaviors B_1, B_2 and B_7. The main reason why the four false positives were grouped in this cluster was due to having between 3 and 8 instances of behavior B_2. In each case, we attempted to access several video streams. Many of these were unreachable and analyzing the network activity showed the failed connection attempts to IP addresses of successful DNS queries. Further investigation into these IP addresses revealed they were temporary video content servers where the specific video streams were no longer available. The IP was taken offline but the records pointing to them had not been removed from the software's database of active streamers.

The twelve false negatives produced by the EM algorithm consisted of nine malware downloaders, three of which belong to the packed.win32.krap family, one worm, one bot (koobface) and one of the 31 undetected malware samples with MD5 hash value *7407c24f17d7c582901623c410ab7a91*. Three samples: koobface and two malware downloaders were seemingly inactive having no successful connection attempts with remote hosts and only four samples exhibited at most a single instance of just one of the following behavior symptoms: B_1, B_2, B_6, B_7. The small amount of network behaviors produced by these malware led to their false negative production since their network traffic was very similar to the benign samples.

The Xmeans algorithm produced no false positives and no false negatives, with all malware grouped in two clusters and benign in one cluster. The 31 undetected malware samples, see Table 2, were correctly clustered by both Xmeans and DbScan while EM correctly clustered 30 implying our network behaviors can detect malware missed by commercial antivirus software and may be usable in stopping zero-day attacks. Overall, the clustering suggest our network behaviors are capable of detecting malware with minimal false positives and false negatives.

5.2 Classification Results

Several classification algorithms were applied on the test set with BayesNet, NNge, Random Forest and Rotation Forest producing the best results listed in Table 9. The false negative rates for all four algorithms were low ranging from 0.6% to 1%, the false positives were also very low ranging between 0% to 2%. All the algorithms had the same two malware samples, VB.vr and one of the 31 undetected malware (MD5 hash value $25181c8ed97357b52ea775bc5dca353c$) as false negatives. Both of these malware were not part of the training set, exhibited 3 or less instances of behavior B_5 with different IP addresses and had successful network activity with remote hosts whose IP addresses were acquired through successful DNS queries. The third false negative produced by BayesNet was one of the 31 undetected malware (MD5 hash value

$cbed573de18b900cd91cc9e4558fb645$) which was active, had two instances of behavior B_5 on two different IP addresses and was not in the training set.

TVants and SopCast were the only two processes flagged as false positives. These two samples were also clustered as false positives. The reason was again their failed connection attempts to IP addresses which were no longer online hosting a video stream thus producing instances of behavior B_2. Only one of the 31 undetected malware was flagged as false negative by all four of our algorithms, with one more being flagged by BayesNet. The other 29 undetected malware were all correctly classified by all four algorithms. This result further confirms the capability of our behaviors and occurrence amounts to detect malware not detected by commercial antivirus software and gives further evidence to their use in helping stop zero-day attacks. Overall, the classification results further suggest our network behaviors can correctly classify both known and unknown malware.

Table 9. Top four classification test set results with 300 malware and 83 benign samples

Classification algorithm	False positives	False negatives	FP rate	FN rate
BayesNet	1	3	1%	1%
NNge	1	2	1%	0.6%
Random forest	0	2	0%	0.6%
Rotation forest	2	2	2%	0.6%

6 Discussion

According to our results in Section 4, of the seven defined behaviors, B_1 occurred the most in the malware samples with 49% followed by B_2 with 21% and B_7 with 18%. All three are considered behaviors more likely to occur in malware than in benign processes with B_7 initially assumed anomalous and not necessarily malicious. Behaviors B_1, B_5 and B_6 are, to our knowledge, novel observations implemented by malware to locate active remote hosts for connection attempts and, in our tests, occurred more in malware than benign. Behavior B_7 is particularly interesting due to its subjective nature which can encapsulate any network activity considered significant and rarely occurring. Therefore it is easy to add activities which degrade detection accuracy. A knowledge expert is best suited to compose activities which comprise this behavior.

Our clustering results were better than expected with perfect results in the case of Xmeans, implying our network behaviors are capable of providing accurate malware detection. Our data set covered a wide spectrum of known malware and benign classes and was able to train our classifiers to correctly identify the majority of malware in the test set with minimal false positives and false negatives.

The most interesting aspect of the results was the highly accurate clustering and classification of the 31 undetected malware. The MD5 sums of all 31 samples were not detected by any antivirus software on Virustotal.com on 01 April 2010 yet our testing correctly identified them with minimal exceptions. This detection accuracy gives strong evidence that our behaviors can help stop zero-day attacks on a host machine, especially in cases where signature-based detectors fail to identify a zero-day attack.

A robust detection system encompasses several malware detection perspectives. This research has only studied one of these perspectives, network activity, in a behavior based way to avoid implementing a detection methodology dependent on malware signatures. Part of our ongoing research is to combine our findings of the network activity perspective with other perspectives to produce a more complete behavior based malware detection system.

7 Limitations

Several protocols such as ARP and SMB were not studied. Their value to enhance our detection accuracy is being analyzed and added to current results. All analysis was done in a virtual machine which forcibly excluded interesting malware samples that are VM aware and ceased to execute or masqueraded as benign upon VM detection. The data set consisted only of malware samples which are initially executed by a mouse double click. Malware packaged as a dll file, kernel system service, or non-executable were not used. We are developing tools allowing the execution of any malware sample regardless of its format.

8 Conclusion and Future Work

This research analyzes known malware and benign samples in an attempt to exploit differences in their network activity behavior to accomplish accurate behavior based malware detection. By analyzing and comparing known malware and benign processes, we have successfully exploited differences in their network activity behavior and produced accurate and effective malware detection with minimal false positives and false negatives. This was accomplished by producing a set of behaviors which occurred most often in our analyzed malware samples during which two novel behaviors frequently used by malware were discovered.

Our analysis results successfully clustered a diverse group of malware and benign process with very high accuracy and minimal false positives and false negatives. Classification algorithms correctly detected newly introduced malware samples also with minimal false negatives and false positives. Most interestingly, our data set included 31 malware samples whose MD5 sums were not detected by any antivirus software on Virustotal.com on 01 April 2010. These undetected malware were correctly identified using our analysis in both clustering and classification algorithms with few exceptions. This provides strong evidence that our identified behaviors can be used together with existing anti-malware solutions, especially signature-based antivirus software, to help stop zero-day attacks on a host machine. This research has presented early results on one perspective, namely network activity, of a larger ongoing project to develop a behavior based malware detection system.

Future work includes examining a suite of protocols for yet-to-be observed activity usable in creating new behaviors and refining our current behavior set and evaluation methodology to further increase detection effectiveness. Alos implementing our network behaviors in a real time detection prototype to measure the efficiency of such an approach including resource usage in collecting data in heavy traffic flows and precise measurements of elapsed time used to detect a malicious process.

Acknowledgement

This work is partially supported by grants from AFOSR, ONR, AFOSR MURI, and the State of Texas Emerging Technology Fund.

References

1. http://anubis.iseclab.org/
2. Balatzar, J., Costoya, J., Flores, R.: The real face of koobface: The largest web 2.0 botnet explained. Technical report, Trend Micro (2009)
3. Bayer, U., Habibi, I., Balzarotti, D., Kirda, E., Kruegel, C.: A view on current malware behaviors. In: LEET 2009: Usenix Workshop on Large-scale Exploits and Emergent Threats (2009)
4. http://bitblaze.cs.berkeley.edu/
5. http://tools.ietf.org/html/rfc1034
6. Ellis, D.R., Aiken, J.G., Attwood, K.S., Tenaglia, S.D.: A behavioral approach to worm detection. In: WORM 2004: Proceedings of the 2004 ACM workshop on Rapid malcode, pp. 43–53. ACM Press, New York (2004)
7. Gu, G., Perdisci, R., Zhang, J., Lee, W.: BotMiner: Clustering analysis of network traffic for protocol- and structure-independent botnet detection. In: Proceedings of the 17th USENIX Security Symposium, Security 2008 (2008)
8. Gupta, A., Kuppili, P., Akella, A., Barford, P.: An empirical study of malware evolution. In: COMSNETS 2009: Proceedings of the First international conference on COMmunication Systems And NETworks, pp. 356–365. IEEE Press, Piscataway (2009)
9. Holz, T., Steiner, M., Dahl, F., Biersack, E., Freiling, F.: Measurements and mitigation of peer-to-peer-based botnets: a case study on storm worm. In: LEET 2008: Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, pp. 1–9. USENIX Association, Berkeley (2008)
10. http://tools.ietf.org/html/rfc792
11. Jiang, X., Xu, D.: Profiling self-propagating worms via behavioral footprinting. In: WORM 2006: Proceedings of the 4th ACM workshop on Recurring malcode, pp. 17–24. ACM, New York (2006)
12. Kolbitsch, C., Comparetti, P.M., Kruegel, C., Kirda, E., Zhou, X., Wang, X.: Effective and efficient malware detection at the end host. In: 18th Usenix Security Symposium (2009)
13. http://www.mlsec.org/malheur/
14. Moore, D., Shannon, C., Claffy, K.: Code-red: a case study on the spread and victims of an internet worm. In: IMW 2002: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, pp. 273–284. ACM, New York (2002)
15. Morales, J.A., Al-Bataineh, A., Xu, S., Sandhu, R.: Analyzing dns activities of bot processes. In: MALWARE 2009: Proceedings of the 4th International Conference on Malicious and Unwanted Software, pp. 98–103 (2009)
16. Morales, J.A., Clarke, P.J., Deng, Y., Kibria, B.G.: Identification of file infecting viruses through detection of self-reference replication. Journal in Computer Virology Special EICAR conference invited paper issue (2008)
17. Moskovitch, R., Elovici, Y., Rokach, L.: Detection of unknown computer worms based on behavioral classification of the host. Comput. Stat. Data Anal. 52(9), 4544–4566 (2008)
18. Nazario, J., Holz, T.: As the net churns: Fast-flux botnet observations. In: 3rd International Conference on Malicious and Unwanted Software, MALWARE 2008, pp. 24–31 (2008)
19. http://tools.ietf.org/html/rfc1001#ref-2

20. http://en.wikipedia.org/wiki/Ping
21. http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers
22. Rabek, J.C., Khazan, R.I., Lewandowski, S.M., Cunningham, R.K.: Detection of injected, dynamically generated, and obfuscated malicious code. In: WORM 2003: Proceedings of the 2003 ACM workshop on Rapid malcode, pp. 76–82. ACM Press, New York (2003)
23. Stinson, E., Mitchell, J.C.: Characterizing bots' remote control behavior. In: Hämmerli, B.M., Sommer, R. (eds.) DIMVA 2007. LNCS, vol. 4579, pp. 89–108. Springer, Heidelberg (2007)
24. http://www.sunbeltsoftware.com/Malware-Research-Analysis-Tools/Sunbelt-CWSandbox/
25. http://tools.ietf.org/html/rfc768
26. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)
27. Yin, H., Song, D., Egele, M., Kruegel, C., Kirda, E.: Panorama: capturing system-wide information flow for malware detection and analysis. In: CCS 2007: Proceedings of the 14th ACM conference on Computer and communications security, pp. 116–127. ACM, New York (2007)
28. Zhu, Z., Yegneswaran, V., Chen, Y.: Using failure information analysis to detect enterprise zombies. In: 5th International ICST Conference on Security and Privacy in Communication Networks, Securecomm 2009 (2009)