Conficker and Beyond: A Large-Scale Empirical Study

Seungwon Shin
Success Lab, Texas A&M University
College Station, Texas, 77843, USA
seungwon.shin@neo.tamu.edu

Guofei Gu
Success Lab, Texas A&M University
College Station, Texas, 77843, USA
guofei@cse.tamu.edu

ABSTRACT

Conficker [26] is the most recent widespread, well-known worm/bot. According to several reports [16, 28], it has infected about 7 million to 15 million hosts and the victims are still increasing even now. In this paper, we analyze Conficker infections at a large scale, including about 25 millions victims, and study various interesting aspects about this state-of-the-art malware. By analyzing Conficker, we intend to understand current and new trends in malware propagation, which could be very helpful in predicting future malware trends and providing insights for future malware defense. We observe that Conficker has some very different victim distribution patterns compared to many previous generation worms/botnets, suggesting that new malware spreading models and defense strategies are likely needed. Furthermore, we intend to determine how well a reputation-based blacklisting approach can perform when faced with new malware threats such as Conficker. We cross-check several DNS blacklists and IP/AS reputation data from Dshield [6] and FIRE [7], and our evaluation shows that unlike a previous study [18] which shows that a blacklist-based approach can detect most bots, these reputation-based approaches did relatively poorly for Conficker. This raised the question, how can we improve and complement existing reputation-based techniques to prepare for future malware defense? Finally, we look into some insights for defenders. We show that neighborhood watch is a surprisingly effective approach in the Conficker case. This suggests that security alert sharing/correlation (particularly among neighborhood networks) could be a promising approach and play a more important role for future malware defense.

1. INTRODUCTION

Conficker worm (or bot) [26] first appeared in November 2008 and since then it has rapidly and widely spread in the world within a short period. It exploits a NetBIOS vulnerability in various Windows operating systems and utilizes many new, advanced techniques such as a domain generation algorithm, self-defense mechanisms, updating via Web and P2P, and efficient local propagation. As a result, it has infected millions of victims in the world and the number is still increasing even now [16, 28].

It is clear that the complex nature of Conficker makes it one of the state-of-the-art malware, and therefore the analysis of Conficker is very important in order to defend against it. A full understanding of Conficker can also help us in comprehending current and future malware trends. Existing research of Conficker analysis mainly falls into two categories. The first focuses on analyzing the Conficker binary and its behavior, revealing its malicious tricks such as the domain generation algorithm [23, 30]. In this direction, SRI researchers [23] and the Honeynet project [30] already provided excellent reports that analyzed Conficker in great detail. The second research category mainly focuses on analyzing the network telescope data [2] or DNS sinkhole data [13] to reveal the propagation pattern and victim distribution characteristics of Conficker on the Internet. There are very few studies in this direction, which is probably because it is very hard to obtain large scale real-world data of victims and the amount of data should be large enough to cover victims' global behavior. CAIDA [2] and Team Cymru [13] provided some initial reports which contain some very basic statistics on the scanning pattern and propagation information of Conficker. However, for a worm/bot that has infected so many victims and has so much potential to damage the Internet, it deserves a much deeper study. Such study is necessary because by analyzing this state-of-the-art botnet, we can gain more knowledge of current malware, e.g., how it differs from previous generation malware and whether such differences represent future trends or not. These deeper investigations could also provide new insights in developing new detection and defense mechanisms for current and future malware.

In this paper, we attempt to provide a deeper empirical measurement study of Conficker. We have collected a large-scale data set which contains almost 25 million Conficker victims with the help of *Shadowserver.org* (details on data collection are discussed in Section 3). We believe such scale is large enough to uncover Conficker's global patterns. We provide an extensive measurement of various distribution patterns of Conficker victims. Furthermore, we use a comparison- and cross-check-based methodology in our measurement study. We study the similarities and differences between Conficker and several other publicly reported worms/botnets. Then we analyze how these differences may affect existing reputation-based detection approaches. We

ACSAC '10 Dec. 6-10, 2010, Austin, Texas USA

also investigate possible aspects that may be useful for Conficker and future malware defense.

In short, this paper makes the following contributions:

- We provide a large-scale empirical study of almost 25 million Conficker victims. By analyzing this data, we reveal many interesting aspects that were previously unknown and show that Conficker victims exhibit a very different distribution pattern from many previously reported botnets or worms. This difference could be a new trend or some ignored facts that are potentially important for future malware defense. Detailed information is in Section 4.

- We evaluate the effectiveness of existing reputation-based approaches for detecting emerging malware threats. They are considered as promising in defending against unknown malware compared to traditional signature-based approaches [1]. Through cross-checking several DNS blacklists and reputation data from Dshield [6] and FIRE [7], our evaluation shows that these reputation-based approaches are not effective for Conficker defense. This suggests that these reputation-based approaches need to be significantly improved and complemented by other techniques. Detailed information is in Section 5.

- We study the Conficker data and find that neighborhood watch is surprisingly effective to detect or predict new victims. This could suggest that alert sharing/correlation (among distributed collaborators, particularly neighborhood networks) could be an effective and promising technique to defend against future emerging threats and it needs more attention for such research. Detailed information is in Section 6.

2. RELATED WORK

Conficker binary analysis. Porras et al. from SRI International provided a very extensive study of the Conficker binary analysis [23]. They analyzed several variants of Conficker and revealed how Conficker propagates, how it infects others, how it evades anti-virus tools and how it updates itself. This provided very detailed and valuable information of Conficker behavior. The Honeynet project [30] also provides a detailed analysis of Conficker binary. These studies also provide scanning tools for detecting Conficker victims in the network.

Conficker data analysis. With the use of the telescope data, researchers from CAIDA provided a simple analysis on Conficker propagation [2]. The Telescope data mainly contains scanning traffic from Conficker victims, which reveals Conficker victim location and timing information to display how Conficker emerges and spreads on the Internet. However, such data is not complete due to the size limit of (passive) monitoring networks. Recently, researchers started to use the DNS sinkholing technique [13] to collect much more accurate Conficker victim data. A report from Team Cymru[13] analyzed the behavior of Conficker victims and provided some general distribution and propagation information. However, there is still a lack of some deep analysis of Conficker victims such as how different the victims are from previous malware. This paper is a first attempt to provide an empirical deep study of Conficker victims, reveal how they are distributed differently from previous generation malware, and how this affects current reputation-based defense mechanisms. In addition, we want to understand if there are some effective techniques for early detection of future variations of Conficker.

3. DATA COLLECTION

An interesting feature of Conficker is the resilient function of updating itself. To avoid detection, it automatically generates new domain names (of updating servers) [23, 30] and connects to those domain names to download an updated version of itself. This function greatly supports Conficker to increase the survivability and resilience. However, once the domain generation algorithm was cracked by researchers, it also provides a way to sinkhole and track the victims. By registering new domain names that will be used by Conficker victims on controlled servers, defenders can collect visits from hosts infected by Conficker. This approach is widely known as DNS sinkholing and has been successfully adopted by researchers that study Conficker [13].

With the aid of *ShadowServer.org*, we have collected the Conficker sinkhole data captured from January 1, 2010 to January 8, 2010. During this period, we observed 24,912,492 unique IP addresses of Conficker victims. We note that the accurate counting of worm/botnet victims is not an easy task because of the existence of DHCP, NAT, and many other issues [31, 25]. For example, Stone-Gross et al. [25] pointed out that there is a slight difference between the number of IP addresses and the number of real infected hosts. This is the limitation of almost all existing worm/botnet measurement studies. We do not intend to solve this problem in this paper. We simply report our observations from our collected data. Although the number may not be exact, with such a large scale it at least provides an estimation of overall characteristics and statistics of the Conficker botnet.

To obtain more interesting results, we surveyed previous work [15, 14, 19, 18, 31, 32, 24] about the behavior of nefarious worms and bots/botnets. They are used to compare with our Conficker result and to help us track whether infection trends have changed. Based on the information they provide, we selected seven measurement studies, which are summarized in Table 1. Of these, three are well-known network worms [15, 14, 19] and four are botnets [18, 31, 32, 24]. Note that some studies of botnets do not specify botnet names in their work, but they show the result of malicious nodes that send spam emails. Since most spam emails are delivered by botnets [18], we can reasonably assume that their studies represent the behavior of some bots or malware.

4. WHO IS WORKING FOR THE CONFICKER BOTNET?

In this section, we provide a basic but important network-level examination, which demonstrates fundamental characteristics of Conficker victims. We review how Conficker victims are distributed over the IP address space and ASes. Also, we investigate the bandwidth of Conficker victims and domain names that Conficker victims belong to. Finally, we survey portions of countries where Conficker victims heavily exist. Some of them are already provided by other studies [2, 13], but our work is more than just providing basic measurement results. To comprehend the radical alteration of

Malware [Work]	*Type*	*Data Source*	*Data Collection Time*
Botnet 1 [18]	Botnet	Sinkhole server	Aug. 2004 ∼ Jan. 2006
Botnet 2 [31]	Botnet	Hotmail	Jun. 2006 ∼ Sep. 2006
Botnet 3 [32]	Botnet	Spamhaus	Nov. 2006 ∼ Jun. 2007
Waledac [24]	Botnet	Infilatrion into Waledac	Aug. 2008 ∼ Sep. 2009
CodeRed [15]	Worm	Measurement	Jul. 2001 ∼ Oct.2001
Slammer [14]	Worm	Measurement	Jan. 2003
Witty [19]	Worm	Measurement	Mar. 2004

Table 1: Data source of previous worms/bots for comparison.

malware, we compare Conficker victims' network-level characteristics with those of previous well-known bots or worms.

4.1 Distribution Over Networks

We plotted each victim's IP address to determine how Conficker victims are distributed over the IP address space and found that they are not uniformly distributed in the whole IP address space; instead the distribution is highly biased, mostly concentrated in some specific ranges.

Result 1. (Distribution over the IP address space) *Most of hosts infected by Conficker are concentrated in several specific IP address ranges.*

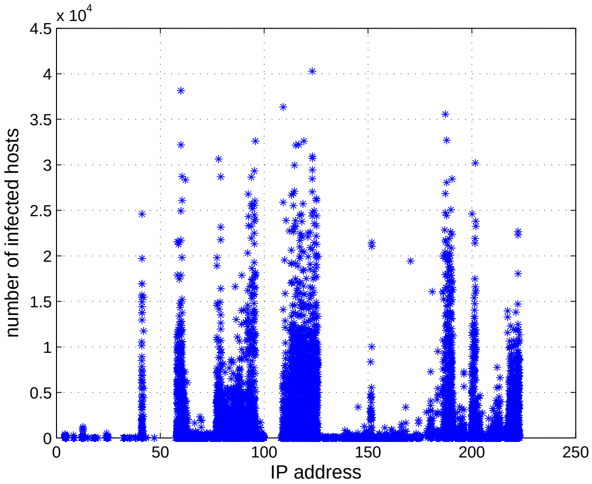


Figure 1: Distribution of infected hosts over IP address.

Figure 1 depicts the distribution of victims over the IP address space. The presence of several wide, sharp spikes, which represent densely infected areas, reveals that the victims are not uniformly distributed. Since the IP address ranges within these wide spikes could be regarded as more vulnerable, we inspected three notable wide spikes in detail. They are in the range of (109.* - 125.*), (77.* - 96.*), and (186.* - 222.*) and they cover around 87% of all victims. In particular, the widest and most prominent spike which is in the range of (109.* - 125.*), includes 9,303,423 infected hosts and accounts for 37.34% of the total number of Conficker victims. To get a more detailed view, we narrowed down the scope from the ranges to more specific networks. In the widest spike, we found that 123.* and 124.* networks are the main contributors. They comprise 1,701,438 infected hosts and account for 6.83% of all victims. We analyzed further and discovered that there are 40,278 Conficker victims in the 123.19.* network, which is around 61.9% of all possible IP addresses in that /16 subnet. Similar characteristics were observed in nearby networks such as the 123.22.* and the 123.23.*[1]

Result 1.1. (Distribution over IP address space - Comparison) *Some portions of IP address ranges were already affected by the previous botnets, but some ranges such as 109.* - 125.* are unique to Conficker.*

Comparing the distribution of Conficker victims over the IP address space with that of previous bots, we find that some ranges are similar to the previous results and some are unique to Conficker. The ranges of (77.* - 96.*) and (186.* - 222.*) are widely known as major locations of the Waledac bot [24]. Yet the interesting thing is that while the range of (109.* - 125.*) is one of the significant locations of Conficker, Waledac has no significant number of victims in that range. In addition, [18] investigated the IP address ranges of hosts infected by bots and they denoted that the ranges of (80.* - 90.*) and (210.* - 220.*) were major locations of bots, which is similar to Waledac analysis. However, both previous studies still did not point out the range of (109.* - 125.*) as a heavy contributor of bots. We tried to understand why the range of (109.* - 125.*) was not seen before. After investigating the data in this range, we concluded that the reason is most likely a change of infection trend, and we will elaborate on this in **Result 2.1**.

Since it is nearly impossible to monitor the *entire* Internet, it is more efficient to focus on specific (suspicious) networks that are more likely to contain commands directed by a botmaster. The IP address ranges within wide spikes, which are shown in Figure 1, can be good candidates that need to be focused.

Insight from Result 1 and 1.1 (Monitoring Networks more efficiently) *It is impossible to monitor all the IP addresses on the Internet, but we can monitor a limited number of specific ranges to efficiently detect commands and attacks in infected networks. Even though the ranges may be different for each botnet, there are still some common parts and they are good candidate ranges to monitor.*

[1] Since the 123.* network is in Class A network, it seems that there is no meaning in splitting it into subnetworks. However, people commonly split Class A networks into several /16 subnets to manage them efficiently. As in the case of 123.* network, we found that it is divided and assigned to several network providers. The 123.19.* network is one of them and it is assigned to *VietNam Post and Telecom Coorperation* and its *inetnum* is 123.19.0.0 - 123.19.255.255.

Representing identities of Conficker-infected hosts by IP address is often preferable in a way that it is precise and elaborate. However, the number of the infected IP addresses is so large that this makes it hard to grasp the global view of Conficker victims. Hence, we use the *Autonomous System (AS)*, which is a useful method for clustering hosts on the Internet for easier management and has been applied in previous measurement work, to group the hosts infected by Conficker.

Result 2. (Distribution over ASes) *Of all infected hosts, the top two ASes account for 28.37% of all victims and top 20 ASes cover 52.54% of all victims. In particular, most of the top rated ASes are located in Asia.*

Conficker victims are concentrated in a few ASes and most of the top infected ASes are located in Asia. As shown in Table 2, around 30% of infected hosts belong to one of only two ASes and more than 50% of infected hosts belong to one of the (top) 20 ASes. Most highly infected ASes are mainly distributed in Asia, particularly in China. This result also suggests that an approach to detect malicious hosts based on ASes would be practical.

ASN	*# Host*	*AS Name*	*Country*
4134	2825403	CHINA-BACKBONE	China
4837	1435411	CHINA169-BACKBONE	China
7738	385672	TELECOMUNICACOES	Brazil
3462	280957	HINET	Taiwan
45899	273577	VPNT-AS-VN	Vietnam
27699	260848	TELECOMUNICACOES	Brazil
9829	248444	BSNL-NIB	India
8167	237465	TELESC	Brazil
3269	231020	ASN-IBSNAZ	Italia
9121	207849	TTNET	Turkey
9394	195088	TELEFONICA	China
4812	182015	CRNET	China
4788	180876	CHINANET-SH-AP	Malaysia
8402	141130	TMNET-AS-AP	Russia
8151	138567	CORBINA-AS	Mexico
17974	137991	UNINET	Indonesia
4808	137672	TELKOMNET-AS2-AP	China
3352	135276	CHINA169-BJ	China
8708	128228	TELEFONICA-DATA-ESPANA	Romania
3320	126520	RDSNET	Germany

Table 2: Conficker victims in the top 20 ASes.

Result 2.1. (Distribution over ASes - Comparison) *Even though the top two ASes were also sources of previous botnets, most of other top rated ASes are newly emerged in the Conficker case.*

By comparing the result of the distribution over ASes with that of previous bots, we find that even if there are common ASes between Conficker and previous bots, there is a significant difference in the locations of infected ASes. Some studies [18, 31, 32] investigated which ASes are the major sources of the botnets that deliver spam emails[2]. We compare their findings with our result and denote it in Table 3. In [18], the authors analyzed data collected in 2004 - 2006 and pointed out that most of the bots are located in North America (particularly in USA), while in [31] and [32] in which data was collected in 2006 - 2007, it was emphasized that bots spread widely over the world. However, in the case of Conficker, ASes in the USA are no longer shown in the top 20 list. Instead, most highly infected ASes are located in Asia and South America.

From this result, we conclude that the trend of major locations of bot infected hosts is still changing; *(i) mainly located in North America*, *(ii) widely spread over the World*, *(iii) popular in Asia and South America.* This trend guides us to observe Asia and South America more closely than North America, which used to be the major source of spam email when we built blacklists to prevent spam at the time. It is important that the trend of major sources of bots is changing. Also, we find that four ASes in Conficker are never seen in previous results. Two of them are in Asia (Vietnam and India) and two of them are in South America (Brazil).

Insight from Result 2 and 2.1. (Change of Infection Trend) *North America used to be the main contributors of botnets, but now Asia and South America contribute more. This means that the locations of the main sources of botnets are changing and we may chase this trend (e.g., new malware spreading models and defense strategies are probably needed).*

4.2 Distribution Over Domain Names

In this section, we inspect the domain names of each victim using DNS reverse lookup.[3] A domain name indicates a group in which a host belongs and it can be a good way to reveal the host itself because domain names are expressed in easy and comprehensible words.

Result 3. (Distribution over Domain Name) *The .br, .net and .cn domains cover around 24.42% of Conficker victims. Interestingly, one of the third level domains covers around 7% of infected hosts, which means it contains more than 1,700,000 victims.*

As shown in Table 4, only a few domains account for about 20% of hosts infected by Conficker. This does not solely apply to top level domains but to all second level domains and third level domains as well. In the case of top and second level domain names, their scope is quite broad and it is hard to find any big advantage when compared to IP address range or AS number. However, for third level domain names, it is possible to focus on small sets of victims. It is useful to monitor victims because the top third level domain includes numerous Conficker victims. In particular, we find that domain *163data.com.cn* accounts for 6.88% of infected hosts. Also, more than 99% of victims in *163data.com.cn* include the word *dynamic* in their fourth level domain names. From this, we can guess that they are using dynamic IP addresses, as their names imply. This result is similar to [31] which uncovers dynamic IP addresses as a main source of most spam emails.

[2] In [32], they only present the top five of ASes, and that is why we could not compare the whole list.

[3] In our DNS reverse lookups, about 49% of victims did not return valid results and therefore we labeled them as "Unknown", shown in Table 4. Since previous studies also showed similar rates of "unknown" domains, we leave them in the table.

Conficker		Botnet 1 [18]		Botnet 2 [31]		Botnet 3 [32]	
ASN	*Country*	*ASN*	*Country*	*ASN*	*Country*	*ASN*	*Country*
4134	China	766	Korea	4134	China	4766	Korea
4837	China	4134	China	4837	China	19262	USA
7738	Brazil	1239	USA	4776	Australia	3215	France
3462	Taiwan	4837	China	27699	Brazil	4837	China
45899	Vietnam	9318	Japan	3352	Spain	4134	China
27699	Brazil	32311	USA	5617	Poland	no info.	no info.
9829	India	5617	Poland	19262	USA	no info.	no info.
8167	Brazil	6478	USA	3462	Taiwan	no info.	no info.
3269	Italia	19262	USA	3269	Italy	no info.	no info.
9121	Turkey	8075	USA	9121	Turkey	no info.	no info.

Table 3: Top 10 ASes hosting Conficker and Spamming Botnets.

Top Level	*Percentage*	*Second Level*	*Percentage*	*Third Level*	*Percentage*
Unknown	48.81%	Unknown	48.81%	Unknown	48.81%
br	8.83%	com.cn	6.89%	163data.com.cn	6.88%
net	8.65%	net.br	4.61%	veloxzone.com.br	1.96%
cn	6.94%	com.br	4.20%	dynamic.hinet.net	1.86%
ru	5.01%	hinet.net	1.91%	telesp.net.br	1.69%
it	2.36%	telecomitalia.it	1.55%	retail.telecomitalia.it	1.46%
ar	1.54%	corbina.ru	0.99%	brasiltelecom.net.br	1.39%
in	1.35%	ny.adsl	0.93%	broadband.corbina.ru	0.99%
com	1.21%	com.mx	0.90%	kd.ny.adsl	0.93%
mx	1.16%	com.ar	0.84%	prod-infinitum.com.mx	0.85%

Table 4: Top 10 Domain Names hosting Conficker Victims in each level.

Result 3.1. (Distribution over Domain Name - Comparison) *The .net domain is still prevalent, but new domains such as .br, .cn, and .ru have recently emerged as heavy resources of botnets. The .com and .edu domains used to be the major sources of worms, but now they cast off the yoke of malicious domains.*

Comparing the domain result with previous work, we found that a few domains that were not previously seen in Conficker. Also, we found that *.com* and *.edu* domains, which used to be nefarious domains, are now relatively clean. Unfortunately, because the previous work does not show second level and third level domain distributions, we could only compare top level domains. In previous studies, top contributors of infected domains are *.net, .com* and *.edu.* However, in the case of Conficker, things have changed. While the *.net* domain is still prevalent, there are newly emerged domains which are not shown in the previous work: *.cn, .ru, .in, and .mx.* All domains that are newly seen represent their countries and we call these ccTLDs (Country Code Top Level Domains). The report from Verisign [29] shows that the registration rate of above ccTLDs has increased explosively for the past three years. This implies that the number of hosts in newly registered domains have increased exponentially. Therefore we may monitor more closely whether they are infected by malware or not, since they may not be on any blacklists. The more interesting part is *.edu* and *.com* domains are no longer serious sources of malware. Of course, there are infected hosts which still belong to those domains, but its coverage is reduced to 1.21% in *.com* and 0.0096% in *.edu.* This result implies that the networks in *.com* and *.edu* domains are probably better managed and protected than before. The comparison result is summarized in Table 5.

Result 3.2. (Distribution over Domain Name - Sensitive Domain Name) *There are Conficker victims in government networks and companies listed in Fortune 100, even though the number of infected hosts is small.*

Besides sending DDoS packets and spam emails, a botnet can steal sensitive information from victims [11]. If hosts infected by a bot belong to critical networks such as government and military networks that contain sensitive information, a botmaster can steal important information from them. Using our Conficker data, we investigated how many victims are affiliated with government or military networks and we found 714 such victims. Surprisingly, victims in government networks are not limited to a few countries, instead they are spread around 70 countries including U.S.A., Parkistan, India and China. Also, we investigated how many victims are in well-known companies. To do this, we used the *Fortune 100 Company List* [8] and we found 2,847 such hosts. Conficker victims still exist within several reputable companies such as HP and IBM.

Insight from Result 3, 3.1 and 3.2. (Watch out for new and sensitive Domains!) *It is nearly impossible to monitor all domain names. However, we have observed that newly registered domains are more vulnerable and more easily infected by Conficker. Hence, it is necessary to closely monitor those recently registered domains. In addition, even though the number of victims is not large, a botmaster of Conficker can steal sensitive information from government and top rated company networks.*

4.3 Distribution over Bandwidth

Besides IP address, AS and domain names, bandwidth gives us information that shows us what kinds of networks Conficker victims belong to. It also helps to predict the

Conficker		CodeRed		Slammer		Witty	
Top level	*Percentage*	*Top level*	*Percentage*	*Top level*	*Percentage*	*Top level*	*Percentage*
Unknown	48.81%	Unknown	47.22%	Unknown	59.49%	net	33%
br	8.83%	net	18.79%	net	14.37%	com	20%
net	8.65%	com	14.41%	com	10.75%	Unknown	15%
cn	6.94%	edu	2.37%	edu	2.79%	fr	3%
ru	5.01%	tw	1.99%	tw	1.29%	ca	2%
it	2.36%	jp	1.33%	au	0.71%	jp	2%
ar	1.54%	ca	1.11%	ca	0.71%	au	2%
in	1.35%	it	0.86%	jp	0.65%	edu	1%
com	1.21%	fr	0.75%	br	0.57%	nl	1%
mx	1.16%	nl	0.73%	uk	0.57%	ar	1%

Table 5: Top 10 Domain Names hosting Conficker, Codered, Slammer and Witty.

power of the botnet. For instance, if we know there are one million Conficker victims in the world and most Conficker victims are in networks with bandwidth less than 1 Kbps, we deduce that it could generate 1 Gbps traffic in the best case. To measure the bandwidth, we use *Tmetric* [27] which sends ICMP packets to the target network and provides a measured bandwidth result. Since *Tmetric* needs to contact the target network to estimate the bandwidth, we can not get the bandwidth result without live target networks and hosts. It takes quite a long time to contact each host and measure the bandwidth, so we only contact one host in the subnetworks (/24) where Conficker victims exist. We reasonably assume that hosts in the same subnetwork (/24) have the same bandwidth.

Result 4. (Bandwidth Distribution) *About 99% of Conficker victims have bandwidth less than 1 Mbps and this means that most of them are ADSL or Modem/Dialup users.*

We find that most victims are using Modem/Dialup or ADSL networks. As shown in Figure 2 (a), about 90% of Conficker victims are in the network whose bandwidth is less than 200 Kbps and around 99% of victims are residing in the network whose bandwidth is less than 1 Mbps. This result is similar to [10] and [31] which denote most bots are using ADSL or Dialup networks. When we conducted this measurement, we found interesting patterns between the bandwidth of a subnet and the number of infected hosts in the subnet.

Result 4.1. (Bandwidth Distribution - relation with the numbers of victims) *The networks that have low bandwidth are likely to have more Conficker victims than those with high bandwidth.*

We suspect that there is a relationship between the bandwidth of a network and the number of infected hosts of the network. As shown in Figure 2 (b), the bandwidth of the subnet is inversely related to the number of infected hosts in the subnet. We think that this pattern is related to the manageability of each network. A network with high bandwidth indicates consuming high setup cost and it also means the network is that worthy. And we could infer that such worthy network is under reasonably good maintenance.

Insight from Result 4 and 4.1. (Examine ADSL or Modem/Dialup networks) *Hosts with ADSL or Modem/Dialup connections are still very vulnerable.*

4.4 Distribution over Geographic Location

Result 5. (Geographic Location) *34.47% of infected hosts are located in China, which is larger than the total number of Conficker victims from the next top eight countries.*

As shown in Table 6 on the distribution over countries, the top ten countries include over 70% of Conficker victims, China ranks number one by a large margin. Conficker victims are distributed over most of the world including Asia, Europe, and South America, but interestingly, only 1.1% of victims are located in North America. This result is somewhat different from previous infection patterns.

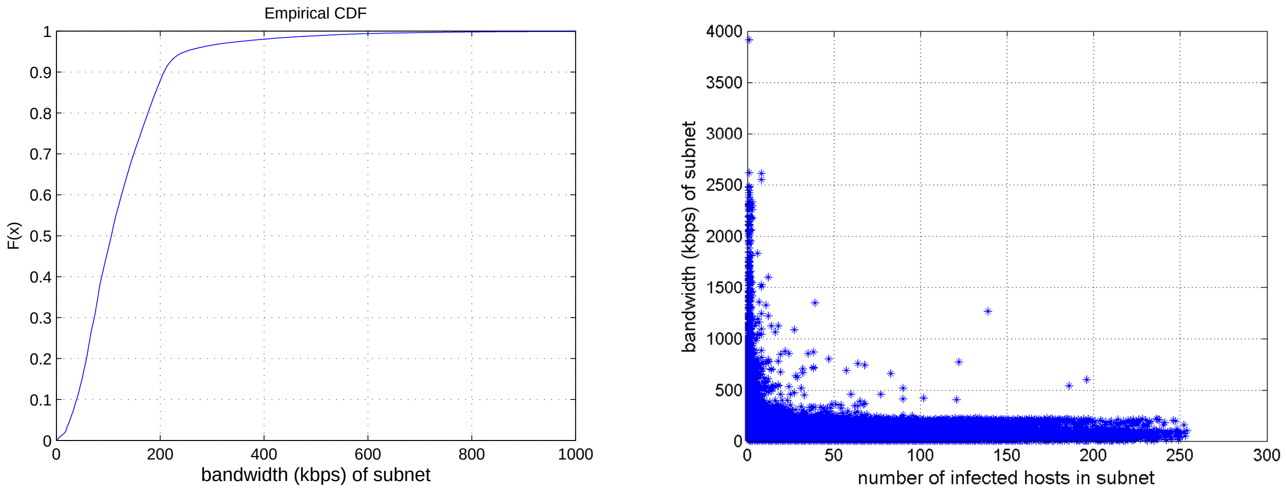
Result 5.1. (Geographic Location - Comparison) *In previous worms and botnets, most the infected hosts were located in North America - especially in USA, but in Conficker, most victims are located in the Asian region - especially in China.*

We compare the country distribution with that of other worms and bots to determine whether it is different or similar and we find that the location of heavy malware contributors is changing. Even though we could not get the exact country distribution from the previous work [18] [31], we are able to estimate which country had more victims based on their distribution over ASes. From Table 6 and 3, we observe that worms prevalent several years ago were mainly located in North America. In previous botnets, [31] and [32] show that victims are mainly located in both Asia and North America, but [18] and [24] denote that most victims are located in North America. However, contrast to the results of previous work, we find that Conficker victims are mainly located in Asia and not in North America, where only 1.1% of victims are located. Therefore, changing monitoring focus from North America to Asia seems reasonable.

Insight from Result 5 and 5.1. (From North America to Asia - Confirmed) *We clearly observe that the hosts infected by Conficker are mainly located in Asia and not in North America, as also shown in Result 2 and 2.1.*

5. HOW WELL DO REPUTATION-BASED DETECTION SYSTEMS DETECT CONFICKER?

In this section, we examine how well current reputation-



(a) (b)

Figure 2: Bandwidth measurement of Conficker victims.

Conficker		Waledac		CodeRed		Slammer		Witty	
Country	*%*	*Country*	*%*	*Country*	*%*	*Country*	*%*	*Country*	*%*
China	34.47%	USA	17.34%	USA	43.91%	USA	42.87%	USA	26.28%
Brazil	9.43%	U.K	7.76%	Korea	10.57%	Korea	11.82%	U.K	7.27%
Russia	7.39%	France	7.04%	China	5.05 %	Unknown	6.96%	Canada	3.46 %
India	4.45%	Spain	5.90%	Taiwan	4.21%	China	6.29%	China	3.36%
Italy	3.56%	India	5.50%	Canada	3.47%	Taiwan	3.98%	France	2.94%
Vietnam	2.81%	no info.	no info.	U.K.	3.32%	Canada	2.88%	Japan	2.17%
Taiwan	2.59%	no info.	no info.	Germany	3.28%	Australia	2.38%	Australia	1.83%
Germany	2.03%	no info.	no info.	Australia	2.39%	U.K.	2.02%	Germany	1.82%
Argentina	2.00%	no info.	no info.	Japan	2.31%	Japan	1.72%	Netherlands	1.36%
Indonesia	1.85%	no info.	no info.	Netherlands	2.16%	Netherlands	1.53%	Korea	1.21%

Table 6: Top 10 countries where Conficker, Waledac, Codered, and Slammer are located.

based detection systems detect Conficker. A DNS blacklist is an effective approach to detect malicious hosts and networks based on reputation [1]. We investigate how well it detects Conficker victims to verify its effectiveness. Also, we examine other reputation-based detection systems such as Dshield [6] and FIRE [7] to check if they could successfully detect Conficker victims.

5.1 DNS Blacklist

We have investigated several well-known blacklists such as DNSBL [5], SORBS [20], SpamHaus [22], and SpamCop [21] to see how many victims of Conficker are on their blacklists. We tested all 24,912,492 infected hosts and we found out that only 4,281,069 hosts are on blacklists which is only 17.18% of all victims.

Result 6. (DNS Blacklist) *DNS blacklists only cover a small portion of Conficker victims. More specifically, only 17.18% of Conficker victims are found on any of four DNS blacklists.*

Our investigation result is quite different from the previous work [18] which shows about 80% of bot infected hosts are already on some blacklists and we believe that the disparity is caused by the difference of distribution of infected hosts. As we mentioned in Section 4.1 and 4.2, the distribution of Conficker victims (over IP address space, ASes, Domain names and Countries) is different from the previous work, and this makes it hard to build effective blacklists for detecting emerging malicious hosts/networks, because blacklists highly depend on the reputation of hosts and networks obtained from their previous records (and currently heavily rely on spam activity records).

Insight from Result 6. (Unfortunately, blacklists can not help us all the time) *Only less than 20% of victims are on DNS blacklists, which means that we need better ways to detect future emerging malware.*

5.2 Dshield and FIRE

Some other reputation-based detection systems are also provided to complement DNS blacklists, and we need to investigate their performance of detection. Since most DNS blacklists are mainly to detect hosts or ASes sending spam, they may not detect other malicious behaviors (potentially) performed by (emerging) infected hosts. There are several studies that try to detect network scanning attacks or web-based attacks and Dshield [6] and FIRE [7] are good examples of them. Dshield provides information to detect hosts or ASes sending suspicious network scanning/attacking packets, and FIRE [7] lists malicious ASes which frequently host rogue networks by measuring their reputation. We plan to inspect how many Conficker victims are notified by Dshield and FIRE.

Result 7. **(Dshield)** *Only 0.33% of victims of Conficker are found on the list of malicious IP addresses reported by DShield, and most of the top ASes infected by Conficker are not on the malicious AS list of Dshield.*

Checking Conficker victims against the list provided by Dshield [4], we found that only a small portion of hosts and ASes are on the list. We investigated 588,797 IP addresses presented by Dshield, and they denoted world-wide attackers/scanners that were detected by all kinds of IDSs and reported to DShield. Since one of the infection vectors in Conficker is random IP scanning [17], we expect a large portion of Conficker victims to show up in Dshield. However, we only find 82,856 hosts from the list. This shows that these Conficker victim hosts are probably easy targets of many previous malware. However, DShield is still not good at catching major portions of new emerging malware such as Conficker. Similarly, we examined the malicious AS list provided by Dshield and we only observed 83 Conficker infected ASes out of 10,584 ASes given by Dshield. Only one of them (AS4812) is a serious contributor of Conficker (ranked 12th among infected ASes) but the rest are not as critical as *AS4812*. Most of them cover less than 0.02% of Conficker victims.

Result 8. **(FIRE)** *Most highly infected ASes by Conficker are not reported by FIRE.*

We compared our infection list of ASes with the results provided by FIRE as well and we want to know whether FIRE is helpful in detecting Conficker victims. Although FIRE denotes *AS4134* as the 8th most malicious AS in its list, most of other heavily infected ASes by Conficker are not shown in the top 500 malicious ASes of FIRE. Some of the main contributing ASes to Conficker have never shown up on FIRE's list.

Insight from Result 7 and 8. **(New and complementary detection approaches are needed)** *DNS blacklists, Dshield and FIRE detect only a small portion of Conficker victims. This means that these reputation-based approaches are not the perfect solution. We need to improve them significantly and complement them with other approaches.*

When we tested Dshield and FIRE, we expected that they could complement DNS blacklists, but the result is not very positive. This implies that these reputation-based systems alone are far from enough to protect the Internet from emerging threats. We believe that new detection systems based on anomalous behaviors of malware could be a good complementary approach to them.

6. CAN NEIGHBORHOOD WATCH HELP?

Conficker still uses network scanning to infect other hosts on the Internet as previous worms and bots did, and it also adopts several advanced skills to infect hosts efficiently. The spreading techniques of Conficker can be classified into two categories [3, 17]; *(i) infecting random hosts* and *(ii) infecting nearby hosts*. Conficker has a function of scanning randomly selected IP addresses. Although this will help Conficker to spread globally, it is not probably very efficient these days because most networks are protected by firewalls or Network Intrusion Detection/Prevention Systems. To propagate more efficiently, Conficker adopts several interesting techniques to infect hosts nearby; (1) an ability to infect other hosts in the same subnet, (2) an ability to infect hosts in the nearby subnets, and (3) an ability to infect portable storage devices.

The diverse infection techniques of Conficker lead us to ask this question: *"Which vector is more effective to infect hosts?"*. Some previous studies suggested that second approach - *(ii) infecting nearby hosts* - is probably more dominant in the Conficker case [17, 12]. We think that this seems reasonable, because even though most networks are protected well from outside threats, they are still open to internal attacks. However, they do not show concrete evidence to support it.

To determine whether this hypothesis is correct, we constructed a test. Prior to explaining our test, we declare that we will use /24 subnet as a basic unit in our test. And we make the following definition to simplify the test. We define two terms: *(i) "camp" is the group of /24 subnets whose /16 subnet is the same and locations are close together, and (ii) each /24 subnet is a "neighbor" of nearby /24 subnets in the same camp.* Sometimes, even if two /24 subnets are in the same /16 subnet, their physical locations could be far from each other. However, since our concept of *"camp"* is each /24 subnet with both nearby IP address and physical location, we should consider its location as well. Based on the above definition, we establish a hypothesis as follows. *Of the two infection vectors of Conficker, suppose the second infection vector plays a dominant role, the infection pattern*[4] *of a /24 subnet will be similar to that of its "neighbors" in the same "camp".* In other words, the hosts in nearby networks of infected host are more likely to be selected as future victims than randomly chosen hosts.

To evaluate this hypothesis, we have tested the following scenarios. First, we divide hosts into /24 subnets and assign each /24 subnet into a "camp" based on our definition. Second, we investigate the infection pattern of each /24 subnet to see whether the infection pattern of each /24 subnet is similar to its "neighbors". We use *Variance-Mean Ratio (VMR)* [9] for a numerical expression. In this test, we measure the mean and variance value of the numbers of infected hosts of each /24 subnet in each "camp", and calculate *VMR* for each "camp". If the value of *VMR* is less than one, distribution of the data set shows under-dispersion with mean value in the center, which means that infection patterns of /24 subnets in the "camp" are very similar to each other.

Result 9. **(Neighborhood)** *Most /24 subnets show similar infection patterns (numbers of infected hosts) with their "neighbors". The closer they are located with each other, the more similar in their infection patterns.*

We measured the *VMR* value of each "camp" and we found that more than 70% of "camps" denoted that their /24 subnet members are similar to each other. From this result, we reasonably infer that the dominant infection vector of Con-

[4] We use the number of infected hosts of /24 subnet as a feature to represent an infection pattern.

Within Distance	*# of all "camps"*	*# of "camps" whose /24 subnet members are similar to each other*
≈ 100km	85,246	62,121 (72.87%)
≈ 200km	65,748	44,633 (67.88%)
≈ 300km	54,415	36,495 (67.06%)

Table 7: The number of all "camps" and "camps" whose members are similar to each other.

ficker is to infect nearby hosts. The test result is shown in Table 7. When we did this test, we got three types of "camps" based on its geographical information. For instance, if we set the distance metric for the "camp" as 100km which means that all /24 subnets in the "camp" have the same /16 subnet and they are within 100km of each other, we found 85,246 "camps" from our data and we discovered 62,121 "camps" whose /24 subnet members are similar to each other. We observed that more than 67% of "camps" showed that their /24 subnet members are similar to each other. The closer their locations are, the clearer this pattern is shown. This result tells us that Conficker is more likely to select nearby hosts than randomly chosen hosts and this means Conficker victims are mainly infected by neighbor networks/hosts. We deduce from this result that infection from the inside could be more harmful than the threats from the outside. Usually, most enterprise networks and ISPs protect their internal hosts using firewalls and IPS/IDS from external attacks, but there are very few approaches to protect hosts from internal threats.

Result 9.1 (Detection based on neighborhood information) *We could detect unknown victims by sharing and correlating neighbor alert information, even if we only know small sets of families and its neighbors.*

Based on previous results, we propose an approach of detecting (or early warning) emerging (unknown) infected /24 subnets using neighborhood information and we show that the approach can detect unknown infected /24 subnets with more than 90% of accuracy. From the above test, we find that Conficker victims share their infection patterns with their neighbors, and this finding gives us an intuition that collecting and sharing neighborhood information would be helpful to detect unknown malware or provide early warnings. To validate this intuition, we have tested the simple scenario of *"We only have small portions of information of benign and malicious hosts, but we can gather neighborhood information. Then, how many unknown malicious hosts can we detect (or predict) based on neighborhood information?"*.

As a method of considering neighborhood information, we use the K-Nearest Neighbor (KNN) classification algorithm, because it is a very popular approach that classifies unknown examples using the most similar "neighbors" in the known examples. When we apply the KNN algorithm to our data, we need the following preparations.

- **define classes:** *in this test, we define two classes; benign (normal /24 subnet) and malicious (/24 subnet which has Conficker victims)*

- **collect data:** *we use our Conficker data for malicious data, and we collected the same number of benign /24 subnets as malicious /24 subnets.*[5]

- **divide data:** *we randomly select 20% of data from both data sets for training samples and other 80% of data is used for testing.*

After all preparation was completed, we used the KNN algorithm (we use 3 for K and use IP address to calculate the distance) to our data and found that it can detect unknown infected /24 subnets with a high accuracy. As shown in Table 8, we find that even if we only know a small part of Conficker data (20%), we can still predict other infected /24 subnets within more than 90% accuracy with reasonable True Positive (TP) and False Positive (FP)[6] rates. This detection result implies that if we share neighbor information, we could detect unknown victims or provide early warnings more efficiently.

Detection Accuracy	*TP rate*	*FP rate*
91.59%	91.65%	8.5%

Table 8: Accuracy, TP and FP rate of the Detection Approach based on Neighborhood Information.

Insight from Result 9 and 9.1. (Neighborhood watch) *We observe that a large portion of victims could be infected by nearby victims and find that it is very important to share threat information with neighborhood networks. And this insight implies that further research is needed for developing new detection/defending approaches based on cooperated/shared (alert) information (and probably in an efficient privacy-preserving way).*

7. CONCLUSION

In this paper, we have studied a large-scale Conficker infection data to discover (i) their distribution over networks, ASes and etc, (ii) difference from previous bots/worms (iii) the effectiveness of current reputation-based malware detection/warning systems, and (iv) some insight to help detect future malware.

Our analysis of Conficker victims and cross-comparison results allowed us to obtain profound insights of Conficker victims. They also guide us to understand the trends of malware infections and to find interesting ideas that can aid the design of future malware detecting systems. We revealed that current reputation-based malware detecting systems depending on previously known information are not enough to detect most Conficker victims. This result suggests that different kinds of (complementary) detection systems such as an anomaly-based detection system are needed.

[5] As a result, we have 1,300,000 malicious /24 subnets (infected by Conficker), and 1,300,000 benign /24 subnets (NOT infected by Conficker or other malware).

[6] TP denotes the rates that the detector classifies real malicious networks correctly, and FP denotes the rates that the detector classifies benign networks as malicious.

We provide a basis that proves the hypothesis of *"A Conficker bot is more likely to infect nearby hosts than randomly chosen hosts"* and we believe that it calls for more research of detection systems which are based on watching/sharing/correlating neighborhood information.

Acknowledgments

We greatly thank Chris Lee and ShadowServer.org for providing the data used in this paper. We also would like to thank our shepherd, Sven Dietrich, and anonymous reviewers for their insightful comments and feedback to improve the paper. This material is based upon work supported in part by the Office of Naval Research under Grant no. N00014-09-1-0776, the National Science Foundation under Grant CNS-0954096, and the Texas Higher Education Coordinating Board under NHARP Grant no. 01909. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Office of Naval Research, the National Science Foundation, and the Texas Higher Education Coordinating Board.

8. REFERENCES

[1] M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster. Building a Dynamic Reputation System for DNS. In *Proceedings of USENIX Security of Symposium*, Aug. 2010.

[2] CAIDA. Conficker/Conflicker/Downadup as seen from the UCSD Network Telescope. `http://www.caida.org/research/security/ms08-067/conficker.xml`.

[3] E. Chien. Downadup: Attempts at Smart Network Scanning. `http://www.symantec.com/connect/blogs/downadup-attempts-smart-network-scanning`.

[4] DHIELD. All suspicious Source IPs in DSHIELD. `http://www.dshield.org/feeds/daily_sources`.

[5] DNSBL. invaluement DNSBL (an anti-spam blacklist). `http://dnsbl.invaluement.com/`.

[6] DSHIELD. Cooperative Network Security Community. `http://www.dshield.org/`.

[7] FIRE. Finding Rogue Networks. `http://maliciousnetworks.org/`.

[8] Fortune. Fortune 100 companies. `http://money.cnn.com/magazines/fortune/`.

[9] U. G. and C. I. *Oxford Dictionary of Statistics (2nd edition)*. Oxford University Press, 2006.

[10] T. Holz, C. Gorecki, and F. Freiling. Detection and Mitigation of Fast-Flux Service Networks. In *Proceedings of NDSS Symposium*, Feb. 2008.

[11] N. Ianelli and A. Hackworth. Botnets as a Vehicle for Online Crime. 2005.

[12] S. Krishnan and Y. Kim. Passive identification of Conficker nodes on the Internet. In *University of Minnesota - Technical Document*, 2009.

[13] J. Kristoff. Experiences with Conficker C Sinkhole Operation and Analysis. In *Proceedings of Australian Computer Emergency Response Team Conference*, May 2009.

[14] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver. Inside the Slammer Worm. In *Proceedings of IEEE Security and Privacy*, May 2003.

[15] D. Moore, C. Shannon, and K. Calffy. Code-red: a case study on the spread and victims of an internet worm. In *Proceedings of ACM SIGCOMM Workshop on Internet Measurement*, Nov. 2002.

[16] B. N. Online. Clock ticking on worm code. `http://news.bbc.co.uk/2/hi/technology/7832652.stm`.

[17] P. Porras, H. Saidi, and V. Yegneswaran. A Foray into Conficker's Logic and Rendezvous Points. In *Proceedings of USENIX LEET*, Apr. 2009.

[18] A. Ramachandran and N. Feamster. Understanding the Network-Level Behavior of Spammers. In *Proceedings of ACM SIGCOMM*, Sep. 2006.

[19] C. Shannon and D. Moore. The Spread of the Witty Worm. In *Proceedings of IEEE Security and Privacy*, May 2004.

[20] SORBS. Fighting spam by finding and listing Exploitable Servers. `http://www.au.sorbs.net/`.

[21] SPAMHAUS. Spamcop.net. `http://www.spamcop.net/`.

[22] SPAMHAUS. The SPAMHAUS Project. `http://www.spamhaus.org/`.

[23] SRI-International. An analysis of Conficker C. `http://mtc.sri.com/Conficker/addendumC/`.

[24] B. Stock, M. E. Jan Goebel, F. C. Freiling, and T. Holz. Walowdac Analysis of a Peer-to-Peer Botnet. In *Proceedings of European Conference on Computer Network Defense (EC2ND)*, Nov. 2009.

[25] B. Stone-Gross, M. Cova, L. Cavallaro, B. Gilbert, M. Szydlowski, R. Kemmerer, C. Kruegel, and G. Vigna. Your Botnet is My Botnet: Analysis of a Botnet Takeover. In *Proceedings of ACM CCS*, Nov. 2009.

[26] M. S. Techcenter. Conficker worm. `http://technet.microsoft.com/en-us/security/dd452420.aspx`.

[27] Tmetric. Bandwidth Measurement Tool. `http://mbacarella.blogspot.com/projects/tmetric/`.

[28] UPI. Virus strikes 15 million PCs. `http://www.upi.com/Top_News/2009/01/26/Virus-strikes-15-million-PCs/UPI-19421232924206/`.

[29] Verisign. The Domain Name Industry Brief. `http://www.verisign.com/domain-name-services/domain-information-center/domain-name-resources/domain-name-report-sept09.pdf`.

[30] D. Watson. Know Your Enemy: Containing Conficker. `http://www.honeynet.org/papers/conficker`.

[31] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldzmidt, and T. Wobber. How Dynamic are IP Addresses? In *Proceedings of ACM SIGCOMM*, Aug. 2007.

[32] Y. Xie, F. Yu, K. Achan, R. Panigraphy, G. Hulte, and I. Osipkov. Spamming Botnets: Signatures and Characteristics. In *Proceedings of ACM SIGCOMM*, Aug. 2008.