Routing Worm: A Fast, Selective Attack Worm based on IP Address Information

Cliff C. Zou[†], Don Towsley[‡], Weibo Gong[†], Songlin Cai[†]
[†]Department of Electrical & Computer Engineering
[‡]Department of Computer Science
University of Massachusetts, Amherst MA 01003
{czou,gong,scai}@ecs.umass.edu, towsley@cs.umass.edu

Abstract

Most well-known worms, such as Code Red, Slammer, Blaster, and Sasser, infected vulnerable computers by scanning the entire IPv4 address space. In this paper, we present an advanced worm called "routing worm", which implements two advanced attacking techniques. First, a routing worm uses BGP routing tables to only scan the Internet routable address space, which allows it propagate three times faster than a traditional worm. Second, and more importantly, the geographic information of BGP routing prefixes enables a routing worm to conduct pinpoint "selective attacks" by imposing heavy damage to vulnerable computers in a specific country, company, Internet Service Provider, or Autonomous System, without collateral damage done to others.

Because of the inherent publicity of BGP routing tables, attackers can easily deploy routing worms, which distinguishes the routing worm from other "worst-case" worms. Compared to a traditional worm, a routing worm could possibly cause more severe congestion to the Internet backbone since all scans sent out by a routing worm are Internet routable (and can only be dropped at the destinations). In addition, it is harder to quickly detect a routing-worm infected computer since we cannot distinguish illegal scans from regular connections without waiting for traffic responses. In order to defend against routing worms and all scanning worms, an effective way is to upgrade the current Internet from IPv4 to IPv6, although such an upgrade will require a tremendous effort and is still a controversial issue.

1. Introduction

Computer worms are malicious programs that self-propagate across a network exploiting security or policy flaws in widely-used services [26]. Most previously wide-spreading worms, such as Code Red, Slammer, Blaster, and Sasser [7], are scanning worms that find and infect vulnerable machines by probing IP addresses in the entire IPv4 Internet address space. How fast a worm can propagate is determined by many factors. Among them, three major factors could be improved by attackers:

(1). The number of initially infected hosts;
(2). A worm's scan rate η, defined as the number of scans an infected computer sends out per unit time;
(3). A worm's hitting probability p, defined as the probability that a worm's scan hits any computer that is either vulnerable or already infected.

"Hit-list worm" presented by [23] exploits the first factor above to improve a worm's propagation speed by containing a large number of IP addresses of vulnerable hosts in the worm code. The second factor, worm scan rate, is determined by the efficiency of a worm's code and also the network bandwidth. If attackers want to improve a worm's propagation speed, another effort is to increase the worm's hitting probability p, i.e., to waste fewer scans on obviously empty IP space.

In order to defend against Internet worm attacks, we need to anticipate and study how attackers will improve their attacking techniques. In this paper, we present an advanced scanning worm called "routing worm", which increases its propagation speed by removing many empty IP addresses from its scanning space based on information of BGP *routable* addresses. We define two types of routing worms — one based on "/8" prefix (x.0.0.0/8) address allocation, another based on BGP routing prefixes. We call them "/8 routing worm" and "BGP routing worm", respectively. Without missing any potential target in the Internet, a /8 routing worm and a BGP routing worm can reduce their scanning space to 45.3% and 28.6% of the entire IPv4 address space, respectively. In this way, attackers can increase

Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)
1087-4097/05 $20.00 © 2005 IEEE

the spreading speed of their worms by a factor of two to more than three without adding much complexity to the worm codes.

The IP address information of BGP routing prefixes provides geographic information about which IP addresses belong to which country, company, Internet Service Provider (ISP), or Autonomous System (AS). With such information, attackers could deploy a routing worm to selectively impose heavy damage to compromised hosts if they belong to a specific entity (country, company, ISP, or AS) and leave the compromised hosts belonging to others intact. Such a "selective attack" property makes a routing worm tremendous dangerous considering the potential attacks initiated by terrorists, revengers, or business rivals.

Because of the inherent publicity of BGP routing tables, attackers can easily deploy a routing worm without much extra effort — this distinguishes the routing worm from other theoretical "worst-case" worms. In addition, compared to a traditional worm that scans the entire IPv4 space, a routing worm could possibly cause more congestion trouble to the Internet backbone, and also makes it harder to quickly detect infected computers. We will explain these challenges in detail later in this paper.

To defend against the threat of routing worms and all scanning worms, we show that upgrading the current IPv4 Internet to IPv6 is an effective way, although such an upgrade will require a tremendous effort and is still a controversial issue.

The rest of this paper is organized as follows. Section 2 surveys related work. In Section 3, we discuss how routing worms can use various types of IPv4 address information to improve their spreading speed. In Section 4, we point out that attackers can use routing worms to conduct selective attack based on geographic information of IP addresses or BGP prefixes. Then in Section 5, we point out two additional challenges brought up by routing worms. In Section 6, we present modeling and analysis of routing worms based on uniform-scan worm model [31]. Then we propose to upgrade IPv4 to IPv6 to defend against scanning worms in Section 7. In the end, Section 8 concludes this paper.

2. Related work

At the same time when we proposed the "routing worm" in this paper, Wu *et al.* [28] independently presented a "routable scan" strategy that is similar to the reducing scanning space idea of the routing worm. However, the routing worm presented in this paper is not only a simple "routable scan" worm, but also a worm that could be used by attackers to conduct selective attack to a specific country or company (ISP, AS, etc), which is more dangerous and important to attackers than simply improving a worm's propagation speed. Staniford *et al.* [23] presented several possible fast spreading worms such as "Warhol" worm and "hit-list" worm right after the 2001 Code Red incident. [23][30][29][8][22][18][14] provided the major research work on how to model and analyze a worm's propagation under various situations.

Many people have studied how to derive the geographic information of ASes, ISPs, IP addresses, or domain names from public available information. The Skitter project provides detailed information of the AS number, name, longitude and latitude for every AS in the Internet [6]. In [5], CAIDA provides the mapping between AS number and the country it belongs to. Furthermore, there are location mapping commercial services, such as EdgeScape from Akamai [9] and the free IP-to-location service from Geobytes [17].

The Route Views project [20] and the Routing Information Service from RIPE NCC [19] provide detailed BGP routing information of the Internet. In 1997, Braun [3] first used BGP routing tables to determine the fraction of IP space that has been allocated. CAIDA also studied this issue in 1998 [4].

Some people have proposed upgrading IPv4 to IPv6 as a defense against scanning worms [25][24][29], but have not explained this issue in detail. Thus most people have not paid attention to the inherent capability of IPv6 in preventing attacks from scanning worms.

3. Routing Worm: A Fast Spreading Worm

The central idea of the spreading speed improvement of a routing worm is to make the worm's target-finding more efficient without ignoring any potential vulnerable computer in the Internet.

3.1. BGP routing worm

One simple way to reduce the scanning space is to use the information provided by Border Gateway Protocol (BGP) routing tables. Both the Route Views project [20] and RIPE NCC [19] provide complete snapshots of BGP routing tables several times per day. BGP routing tables contain all Internet routable IP addresses. A "*BGP routing worm*" is an advanced worm that contains BGP routing prefix information to only scans BGP routable IP addresses. In this way, the worm effectively reduces its scanning space without missing any target.

A BGP routing *prefix* is a chunk of IP addresses that have the same n most-significant bits in their addresses where n is called *prefix length* for this prefix. Because of multihoming, many prefixes in a BGP routing table overlap with each other — one prefix of shorter length contains all of the IP addresses in another prefix of longer length. To determine the percentage of IPv4 space that is BGP routable, we

Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)
1087-4097/05 $20.00 © 2005 IEEE

download BGP routing tables from Route Views [20], extract routing prefixes, and remove all overlapping prefixes that are contained by others. We illustrate in Fig. 1 how the utilization of IP space has evolved in the six-year period from 1997 to 2003.

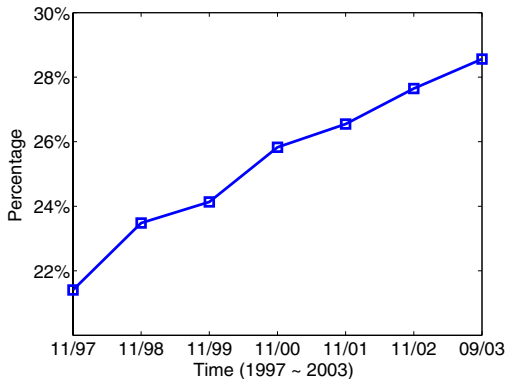


Figure 1. Percentage of BGP routable address space over the entire IPv4 space from 1997 to 2003 (data from Route Views project [20])

Although the number of computers connected to the Internet has increased greatly from 1997 to 2003, due to the usage of Classless Inter-Domain Routing (CIDR), Network Address Translation (NAT), and Dynamic Host Configuration Protocol (DHCP), the allocated routable IP space has not increased much. Fig. 1 shows that about 28.6% of IPv4 addresses are BGP routable on Sept. 2003. By including the information of BGP routing prefixes, a BGP routing worm can reduce its scanning space by 71.4% without ignoring any potential vulnerable computer.

3.2. /8 routing worm

BGP routing tables in September 2003 contain more than 140,000 prefixes. After removing overlapping prefixes, a BGP routing worm still needs to include about 62,000 prefixes. To avoid adding a big payload to a routing worm, attackers could possibly use IPv4 "/8" address allocation information instead of BGP routing prefixes.

The Internet Assigned Numbers Authority (IANA) provides public information about how "/8" prefix (x.0.0.0/8) of IPv4 has been assigned [12]. Each "/8" prefix contains 2^{24} IP addresses and there are 256 (2^8) "/8" prefixes in IPv4. By combining the IANA "/8" allocations with the information of BGP routing prefixes (BGP data from September 22, 2003), we find that 116 "/8" prefixes contain all BGP routable IP addresses. In other words, from an attacker's point of view, a worm does not need to waste its scans on IP addresses belonging to the other 140 "/8" prefixes.

A "*/8 routing worm*" is defined as an advanced worm that only scans those "/8" prefixes that contain BGP routable addresses. According to the BGP data from September 2003, a /8 routing worm only needs to scan 45.3% of IPv4 space by adding a small 116-byte prefix payload.

In fact, Code Red II [1] has already used part of IANA address allocations to reduce its scanning space: if an IP address generated by a Code Red II worm belongs to 127.0.0.0/8 (loopback addresses) or 224.0.0.0/4 (16 "/8" multicast addresses [12]), then the worm skips that address and generates a new address to scan. In this way, Code Red II scans 93.4% of the entire IPv4 space (239 out of 256 "/8" address space).

3.3. Routing worm based on prefix aggregation

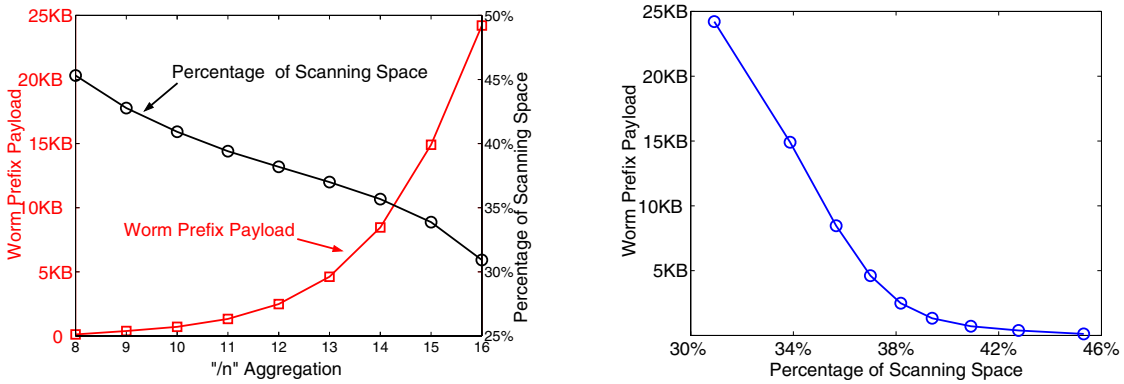
A BGP routing worm scans a potentially smaller space than a /8 routing worm, but its routing prefix payload is much larger. In order to have a good trade-off between the size of the scanning IP space and the payload requirement for a routing worm, attackers could aggregate BGP routing prefixes. Here "aggregation" means that many BGP prefixes can be combined into one that has a shorter prefix length by adding the empty IP space between those original ones. In this way, the newly generated prefix covers all the IP space in those original prefixes. Through aggregation, a routing worm would need to scan a larger IP space but store fewer prefixes in its payload.

One simple aggregation method is to aggregate all prefixes that have prefix lengths longer than n to be "$/n$" prefixes ($8 \leq n \leq 32$), which can be called "$/n$ aggregation". If $n = 32$, no prefixes need to be aggregated and a BGP routing worm is derived; if $n = 8$, a /8 routing worm is derived.

Fig. 2 shows the aggregation impact on a routing worm's scanning space and prefix payload. For clarity, we only show the aggregation results from "/16" aggregation to "/8" aggregation in this figure. It shows that, as a routing worm aggregates more BGP prefixes together, it increases its scanning space while reducing the size of its payload. For example, if a routing worm uses "/16" aggregation, the worm increases the scanning space from the original 28.6% of a BGP routing worm to 30.9% of the IPv4 space, while reducing the prefix payload dramatically from the original 175KB to 24KB.

By using prefix aggregation, attackers have the freedom to choose a suitable "$/n$" aggregation according to their needs, or to the desired spreading properties of their routing worms.

Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)
1087-4097/05 $20.00 © 2005 IEEE



(a). "/n" aggregation ($n = 8 \sim 16$)

(b). Payload and scanning space trade-off

Figure 2. Prefix aggregation impact on a routing worm's scanning space and prefix payload (In the left-hand figure, the left Y-axis represents a routing worm's prefix payload; the right Y-axis represents the percentage of scanning space over the entire IPv4 address space. In the right-hand figure, each point from left to right represents "$/n$" aggregation where $n = 16, 15, \cdots, 8$, respectively.)

4. Routing Worm: A Selective Attack Worm

By considering IP address information, a routing worm not only increases its propagation speed, but also can use such information to conduct selective attacks, which is a more important property to attackers. "Selective attack" means that hackers or terrorists can selectively impose heavy damage to vulnerable computers in a specific country, company, ISP, or AS with little collateral damage done to others.

4.1. Selective attack based on IP geographic information

IANA provides limited information about who owns a "/8" network [12]. For example, 214.0.0.0/8 and 215.0.0.0/8 are allocated to the US Department of Defense; 56.0.0.0/8 is allocated to "US Postal Service"; 43.0.0.0/8 is allocated to "Japan Inet", etc [12]. Such information can be possibly used by attackers in their /8 routing worms.

Meanwhile, BGP routing tables provide detailed information about what Autonomous System (AS) owns a specific network prefix. Since many people have studied how to derive geographic information from BGP routing prefixes [6][5], hackers, revengers, or terrorists can use routing worms to conduct pinpoint heavy attacks to vulnerable computers in a specific country, company, ISP, or AS with little collateral damage to others.

Attackers can program a routing worm to exhibit different behaviors based on the location of the compromised computers. For example, if a compromised computer belongs to a specific country or company, the routing worm can impose heavy damage to this computer; otherwise, the compromised computer will be simply used as a stepping stone to scan and infect other vulnerable computers without being destroyed. For another example, attackers can program a routing worm to have a higher scanning preference for IP prefixes belonging to a specific target — this "target preference" scanning method is an extension of the "local preference" used by Code Red II [7].

4.2. Selective attack: a simple but general attacking idea

In fact, "selective attack" is a simple but very general attacking idea for any large-scale spreading virus or worm. Viruses or worms can use any information they can get from compromised computers to conduct selective attacks. Such information of a compromised computer includes the computer's IP address, time zones, Operating System, installed software, CPU, memory, network connection type and speed, etc. For example, a worm can selectively impose heavy damage on compromised computers if they have installed illegal Windows Operating Systems, or a specific peer-to-peer file sharing program, or video cards from a specific manufacturer.

Besides inflicting damage, attackers can also use the "selective attack" idea to improve a worm's spreading speed. For example, on any compromised computer, Code Red always generates 100 threads to scan and infect others simultaneously [16]. However, some compromised computers that have a small-size memory or a slow network connection cannot support those 100 threads without crash; on the other hand, many compromised computers that have

Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)
1087-4097/05 $20.00 © 2005 IEEE

powerful CPU, large memory, and high connection speed may be able to support thousands of threads generated by the worm. Therefore, attackers can program a worm to generate different number of scanning threads based on computer resources to speed up the worm's overall spreading speed.

In fact, a primitive selective attack has already been implemented by Code Red II — the worm generates 300 threads if a compromised computer runs non-Chinese Windows and 600 threads if the computer runs Chinese Windows [1].

5. Other Challenges from a Routing Worm

Besides its fast spreading speed and selective attack properties as explained in the above two sections, a routing worm imposes two additional challenges to the Internet and our defense systems. In this section, we discuss these two challenges in detail.

First, when an ordinary scanning worm is transformed into a routing worm, it may cause more severe congestion to the backbone of the Internet. Since about 70% of IPv4 space is not BGP routable, around 70% of scan packets sent out by an ordinary worm, which target IP space that is not Internet routable, will be quickly dropped at "default-free" routers[1] before entering the backbone links of the Internet. Thus a major part of worm scan traffic will not appear on the Internet backbone. On the other hand, all scans sent by a routing worm are BGP routable, and hence, will travel across the Internet backbone and reach the routers of the destination ASes or local networks. Therefore, a routing worm, especially a bandwidth-limited routing worm, will cause more severe congestion trouble to the Internet backbone than current scanning worms that scan the entire IPv4 space. For example, Slammer worm has caused severe congestion in many parts of the Internet [15]. If this worm writer had changed the worm code to be a routing worm, it would possibly have caused severe congestion to the entire Internet instead of just congestion in some local area networks.

Second, a routing worm makes it harder to quickly detect and then quarantine internal infected computers in an enterprise network. As explained in [22], in order to defend an enterprise network against a fast worm attack, the defense system of the enterprise network must be able to identify and then quarantine an internal infected computer as quickly as possible. One general detection method is to detect the abnormal level of illegal traffic sent out by an infected computer due to the random scans generated by a scanning worm [13][22][27]. For an ordinary worm that scans the entire IPv4 space, because a large percentage of the worm's random scans target non-routable address space, we can quickly detect an internal infected computer based on its outgoing connection destinations without waiting for the traffic response. On the other hand, for a routing worm infected computer, we have to wait a while for its traffic response (such as TCP timeout or ICMP error messages from routers) in order to detect its illegal traffic. For the defense of a fast spreading worm such as the Slammer worm, such a detection time difference might be critical for shutting down the worm infection process before it is too late.

1 A default-free router is a router that "actively decides where to send packets with a destination outside the AS to which the router belongs, and not forward it, by default, to another router" [2].

6. Routing Worm Propagation Modeling and Analysis

In our previous paper [31], we have presented a uniform-scan worm model that is described by worm propagation parameters:

$$\frac{dI_t}{dt} = \frac{\eta}{\Omega} I_t (N - I_t) \quad (1)$$

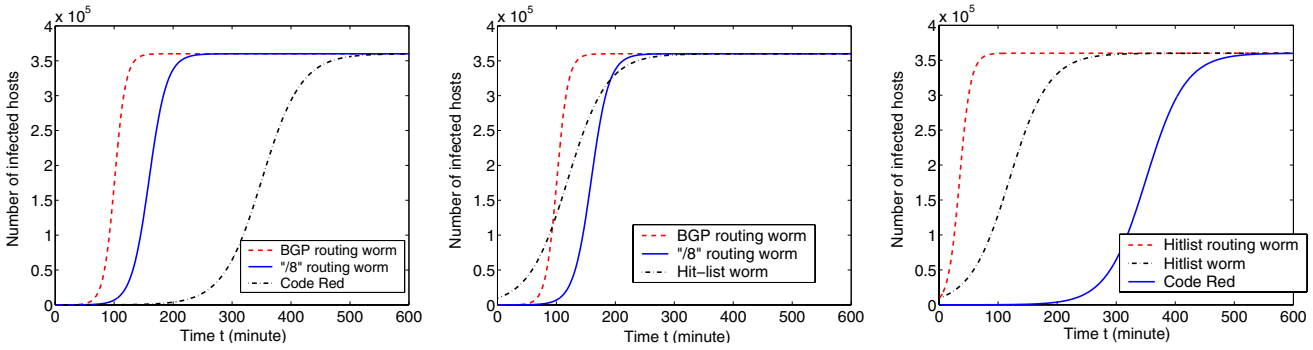
where I_t is the number of infected hosts at time t; N is the total number of vulnerable hosts in the system before the worm spreads out. At $t = 0$, I_0 hosts are infected and the remaining $N - I_0$ hosts are vulnerable. η is the worm's average scan rate and Ω is the size of the worm's scanning space.

If a routing worm uniformly scans its scanning space and has the same average scan rate as a traditional uniform-scan worm, then according to (1), a routing worm will propagate faster due to its smaller scanning space Ω. To show how much faster a routing worm can propagate, we use Code Red as the example of a traditional worm, which has a scan rate $\eta = 358$ per minute and a vulnerable population $N = 360,000$ [29]. We assume that there are $I_0 = 10$ initially infected hosts. Fig. 3(a) shows the numbers of infected hosts I_t of the Code Red worm, a /8 routing worm, and a BGP routing worm as functions of time t, respectively. It shows that by using IP routing information, routing worms clearly increase their spreading speed.

Staniford *et al.* [23] introduce a "hit-list" worm that has an address list of a large number of vulnerable hosts in the Internet. Since a hit-list worm can infect all vulnerable hosts in its hit-list within a few seconds [23], we ignore this hit-list infection time and assume that a hit-list worm begins to propagate with a large number of initially infected hosts where I_0 equals to the size of the hit-list. When a hit-list worm uniformly scans the Internet after its hit-list scanning phase, its propagation can be modelled by (1) with $\Omega = 2^{32}$.

To study the propagation differences between a hit-list worm and routing worms, Fig. 3(b) compares a BGP routing worm, a /8 routing worm, with a hit-list worm that has a hit-list of $10,000$ vulnerable hosts and the same scan rate $\eta = 358$/min as Code Red. This figure shows that the hit-

Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)
1087-4097/05 $20.00 © 2005 IEEE



(a). Comparison between two types of routing worms and Code Red

(b). Comparison between two types of routing worms and a hit-list worm

(c). Comparison between a hit-list routing worm, a hit-list worm, and Code Red

Figure 3. Worm propagation comparisons

list worm can infect a larger number of hosts in a short time, but its infection growth rate is smaller than routing worms.

A hit-list worm and a routing worm try to improve their spreading speed through two different approaches. These two approaches do not conflict with each other and can be easily combined together to generate a new worm, called a "hit-list routing" worm, that has both a large number of initially infected hosts and a fast propagation speed. Fig. 3(c) shows the propagation of a hit-list routing worm, which has a 10,000 hit-list and the BGP routing prefixes. Compared with a traditional worm and an ordinary hit-list worm, the hit-list routing worm spreads out much faster.

The famous *Warhol* worm presented in [23] is a hit-list worm that uses "permutation scan" instead of uniform scan. Permutation scan provides a form of coordination among infected hosts to avoid multiple scanning on the same IP addresses [23], which cannot be modeled by the uniform-scan worm model (1). Due to the coordination mechanism, Warhol worm propagates faster than a uniform-scan hit-list worm after most vulnerable hosts have been infected (as shown in Figure 3 in [25]). However, a routing worm and the original Code Red can also deploy the same permutation scan instead of uniform scan without any problem. If a routing worm and Code Red implement the same permutation scan as a Warhol worm, these three worms will have the similar propagation relationship as what shown in Fig. 3 (although the pattern of propagation curves will change slightly as shown in Figure 3 in [25]).

Authors in [23] also present a *flash* worm that contains IP addresses of all vulnerable hosts in the worm's hit list, which can infect all vulnerable computers in the Internet within tens of seconds [23]. However, it is very hard or impossible to collect up-to-date IP addresses of all vulnerable hosts in the global Internet, especially for the vulnerabilities of computers that do not advertise their addresses (such as SQL database servers attacked by Slammer, or the ISS security products attacked by Witty worm [21]). Therefore, flash worms exist in theory, not likely to be generated by attackers in an Internet scale attack (although it is possible for attackers to use a flash worm to attack a local area network).

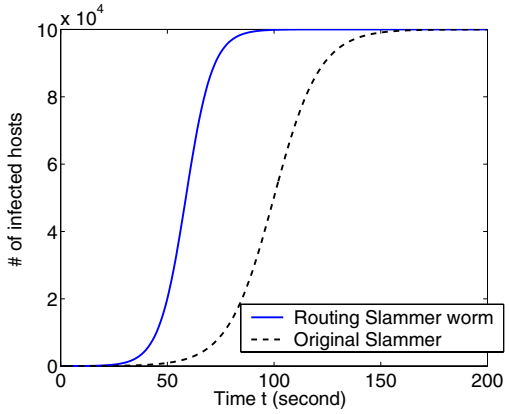


Figure 4. Worm propagation comparison of the original Slammer with the /8 routing worm transformed from Slammer

Due to its tiny payload requirement, a "/8 routing worm" might be used by attackers in their future bandwidth-limited worms. A "bandwidth-limited worm" is a worm that fully uses the link bandwidth of an infected host to send out infection traffic. For example, SQL Slammer is a bandwidth-limited worm with an average scan rate $\eta = 4000$ scans/second [15]. Because Slammer is a UDP-based worm that puts the complete worm code into one single UDP packet, the BGP routing worm idea is not realistic for this worm. Each UDP infection packet sent out by Slammer is 404 bytes [15]. if the worm author transformed Slammer into a /8 routing worm, which is called a "routing Slammer worm", the

Proceedings of the Workshop on Principles of Advanced and Distributed Simulation (PADS'05)
1087-4097/05 $20.00 © 2005 IEEE

UDP infection packet would be 520 bytes (by adding a 116-byte prefix payload). After transforming into a /8 routing worm, the routing Slammer worm would have an average scan rate $\eta = 4000 \times 404/520 = 3108$ scans/second. Fig. 4 shows the worm propagation of the original Slammer and the new routing Slammer worm as functions of time (the other parameters are $N = 100,000$, $I(0) = 10$, the same as what used in [29]).

7. Defense Against Routing Worms: Upgrading IPv4 to IPv6

It is very hard to prevent attackers from generating a routing worm due to the following two reasons: (1) both IANA "/8" allocations and BGP routing tables are public available information that are difficult or impossible to hide from attackers; and (2) a routing worm is very easy for attackers to implement, much easier than the hit-list worm presented in [23]. Once attackers obtain BGP routing prefixes, they can use the same BGP data for all scanning worms to attack various vulnerabilities. On the other hand, to program a hit-list worm, attackers need to put effort to collect a hit-list of vulnerable computers and have to repeat such work for different vulnerabilities. Such a hit-list is especially hard to collect for vulnerable hosts that do not advertise their addresses (e.g., Windows SQL servers attacked by Slammer). Because of the real threat coming from routing worms, we must find a way to prevent a routing worm from quickly spreading out.

A routing worm increases its propagation speed by reducing its scanning space. Fig. 3 shows how much faster a routing worm can propagate when the worm reduces its scanning space by only half to two thirds. Therefore, if we use the same principle to dramatically increase a worm's scanning space, we can significantly slow its propagation speed. For this reason, we believe that an effective defense against routing worms and all scanning worms is to upgrade the current IPv4 Internet to IPv6 — the vast address space of IPv6 (its BGP table does not reveal address information of networks with less than 2^{64} IP addresses) can prevent a worm from spreading through scanning.

IPv6 has dramatically increased IP space from 32-bit addresses to 128-bit addresses. Because of this huge IP address space, IPv6 implements a hierarchical addressing theme where the smallest network has 2^{64} IP addresses (with prefix /64) [10][11]. In other words, the smallest network in IPv6 BGP routing tables contains the number of IP addresses equal to that of *4 billion* IPv4 Internet.

Some people might think that allocating such a big address space for a smallest network wastes too much IP resource. Actually, it does not. Suppose there are *1000 billion* people on earth, then on average each person can own 2.3 million the smallest networks (/64) mentioned above for unicast usage.

Attackers can still use BGP routing tables to program a routing worm. However, they cannot know address allocation information inside any /64 network from BGP routing tables since the longest prefix in BGP routing table is /64. A local network might uses smaller address space for address allocation internally, but such local address allocation information will not show up in BGP routing table. Such local address allocation is confidential to the local network administrators that attackers cannot know without port-scanning beforehand.

Even one single /64 network in IPv6 will have sufficient IP space to defeat scanning worms. Suppose there are $N = 1,000,000$ vulnerable hosts in one single /64 network and a worm has a scan rate $\eta = 100,000$/second with $I_0 = 1000$ initially infected hosts. If the worm only scans and infects hosts in this /64 network, then $\Omega = 2^{64}$. Based on (1), the worm will need to spend *40 years* to infect half of the vulnerable hosts in this single /64 network.

Of course, upgrading IPv4 to IPv6 is not the omnipotent solution for defending all kinds of worm attacks. It is only useful for defending worms that find victims by random scanning, such as Code Red, Slammer, Blaster, Sasser and Witty worm. In addition, IPv6 is still a controversial issue and there are many important economic and technical details to be solved before we can upgrade the current IPv4 to IPv6.

8. Conclusions

In this paper, we present a new advanced scanning worm called "*routing worm*". Based on BGP routing prefix information, a routing worm not only propagates faster, but also is able to conduct pinpoint selective attacks to specific country, company, ISP or AS. Because of the inherent publicity of BGP routing tables, attackers can easily deploy routing worms in the future. Compared to a traditional worm, a routing worm could possibly cause more severe congestion trouble to the Internet backbone, and makes it harder to quickly detect infected computers. An effective way to defend against routing worms and all scanning worms is to upgrade the current IPv4 to IPv6, although such an upgrade will require a tremendous effort and is still argued by many people.

Acknowledgements

This work was supported in part by ARO contract DAAD19-01-1-0610, NSF Grant EEC-0313747, EIA-0080119, ANI-0085848 and CNS-0325868.

References

[1] eEye digital security: CodeRedII worm analysis. http://www.eeye.com/html/Research/Advisories/AL20010804.html, 2001.

[2] A. Antony and H. Uijterwaal. Routing information service R.I.S. design note. http://www.ripe.net/projects/ris/Notes/ripe-200/, 1999.

[3] H. Braun. BGP-system usage of 32 bit Internet address space. http://moat.nlanr.net/IPaddrocc, November 1997.

[4] CAIDA. IPv4 address space utilization. http://www.caida.org/outreach/resources/learn/ipv4space, 1998.

[5] CAIDA. IPv4 BGP geopolitical analysis. http://www.caida.org/analysis/geopolitical/bgp2country, 2003.

[6] CAIDA. Visualizing Internet topology at a macroscopic scale. http://www.caida.org/analysis/topology/as_core_network, 2003.

[7] CERT. CERT/CC advisories. http://www.cert.org/advisories/.

[8] Z. Chen, L. Gao, and K. Kwiat. Modeling the spread of active worms. In *Proceedings of the IEEE INFOCOM*, March 2003.

[9] Akamai service: EdgeScape. http://www.akamai.com/en/html/services/edgescape.html.

[10] R. Hinden and S. Deering. RFC-3513: Internet protocol version 6 (IPv6) addressing architecture. April 2003.

[11] R. Hinden, S. Deering, and E. Nordmark. RFC-3587: IPv6 global unicast address format. August 2003.

[12] IANA. Reserved IPv4 addresses. http://www.cidr-report.org/v6/reserved-ipv4.html, January 2004.

[13] J. Jung, S. E. Schechter, and A. W. Berger. Fast detection of scanning worm infections. In *Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID)*, September 2004.

[14] G. Kesidis, I. Hamadeh, and S. Jiwasurat. Coupled kermack-mckendrick models for randomly scanning and bandwidth-saturating internet worms. In *Proceedings of 3rd International Workshop on QoS in Multiservice IP Networks (QoS-IP*, Feburary 2005.

[15] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver. Inside the Slammer worm. *IEEE Magazine on Security and Privacy*, 1(4), July 2003.

[16] D. Moore, C. Shannon, and J. Brown. Code-Red: a case study on the spread and victims of an Internet worm. In *Proceedings of the second ACM SIGCOMM Workshop on Internet Measurement*, November 2002.

[17] Net world map project: IP address locator tool. http://www.geobytes.com/IpLocator.htm?GetLocation.

[18] D. Nicol and M. Liljenstam. Models of Internet worm defense. IMA Workshop 4: Measurement, Modeling and Analysis of the Internet. http://www.ima.umn.edu/talks/workshops/1-12-16.2004/nicol/talk.pdf, January 2004.

[19] RIPE NCC routing information service. http://www.ripe.net/ris.

[20] University of Oregon route views project. http://www.routeviews.org.

[21] C. Shannon and D. Moore. The spread of the Witty worm. http://www.caida.org/analysis/security/witty/, March 2004.

[22] S. Staniford. Containment of scanning worms in enterprise networks. *Journal of Computer Security*, 2003.

[23] S. Staniford, V. Paxson, and N.Weaver. How to own the Internet in your spare time. In *Proceedings of USENIX Security Symposium*, August 2002.

[24] M. H. Warfield. Security implications of IPv6. Internet Security Systems, Inc. White Paper, 2003.

[25] N. Weaver. Warhol worms: The potential for very fast Internet plagues. http://www.cs.berkeley.edu/∼nweaver/warhol.html, 2001.

[26] N. Weaver, V. Paxson, S. Staniford, and R. Cunningham. A taxonomy of computer worms. In *Proceedings of ACM CCS Workshop on Rapid Malcode (WORM'03)*, October 2003.

[27] N. Weaver, S. Staniford, and V. Paxson. Very fast containment of scanning worms. In *Proceedings of 13th USENIX Security Symposium*, August 2004.

[28] J. Wu, S. Vangala, L. Gao, and K. Kwiat. An efficient architecture and algorithm for detecting worms with various scan techniques. In *Proceedings of the 11th Annual Network and Distributed System Security Symposium (NDSS'04)*, February 2004.

[29] C. C. Zou, L. Gao, W. Gong, and D. Towsley. Monitoring and early warning for Internet worms. In *Proceedings of 10th ACM Conference on Computer and Communications Security (CCS'03)*, October 2003.

[30] C. C. Zou, W. Gong, and D. Towsley. Code Red worm propagation modeling and analysis. In *Proceedings of 9th ACM Conference on Computer and Communications Security (CCS'02)*, October 2002.

[31] C. C. Zou, D. Towsley, and W. Gong. On the performance of Internet worm scanning strategies. Technical Report TR-03-CSE-07, Umass ECE Dept., November 2003.