



Network Detection of Worms and Viruses

Peter J. Welcher and Carole Warner Reece

Introduction

I'm writing this article as Halloween approaches. Somehow, the topic seems timely, since it involves the threat of unseen but dangerous beasts. The challenge is to know when they're around!

In last month's article, <http://www.netcraftsmen.net/welcher/papers/worm01.html>, we reviewed the ground rules that seem to be changing. We also looked at controlling the rapid spread of viruses and worms, with an access list (ACL) configuration example. Because of space limitations, we deferred discussion of network-based virus/worm detection for this month.

This article delivers on that promise! Note that we are NOT going to talk about Intrusion Detection Systems (IDS's), since detecting and responding to events is what they are supposed to do. This article is focused on other approaches. We may come back to IDS's in a future article.

Let Us Count the Ways

Here's the short list of network methods for detecting a worm or virus attack. The assumption is that the worm is actively trying to propagate itself to other machines. There are other similar methods not on the list -- once you've gotten the idea, you can invent your own variations on this theme.

- 1 NBAR discovery
- 1 RMON2 at L7 with RMON probe or NAM
- 1 NetFlow (show commands or w/ a NetFlow collection/reporting system)
- 1 Switch MLS statistics
- 1 ACL or VACL
- 1 Sinkhole router
- 1 Packet analyzer software
- 1 Host firewalls
- 1 IDS or PIX reporting (CW V/SMS Monitoring for Security; SIMS)

In the discussions I've seen online, various network administrators have historically been first tipped off by high levels of traffic, broadcasts, etc. These tips came from routine performance reporting they were doing, or occasionally from fortuitous router crashes. When they started digging for the cause, they then often used one of the above techniques.

Lesson Learned #1: Get some performance management in place. It could be SolarWinds Orion, NetScout Performance Reporting, Concord, MRTG, Cricket, or another tool. But get it, use it, and look at the reports! It can help you make your network perform better, and it can tip you off to virus and worm attacks.

We notice that the above detection methods all have something in common. They look at the anomalous traffic, and use that to spot unusual activity. The source of the unusual activity is a compromised or infected host. That's it!

How do you find out what port(s) to watch? Either from reading online about the worm, or from noticing unusual activity on some port(s). Once the worm behavior is characterized, you've got definitive ports. But until then, and until you can get through to the virus vendor web pages, you may need to proceed on your own. That's where hyperactive ports play a role.

NBAR Discovery

The first item on the above list is NBAR discovery. This tells you about #packets and bytes to various TCP and UDP ports. If you know what's "normal", then this can tip you off to the presence of a virus. NBAR discovery can have performance impact on your router. NBAR discovery only produces highly aggregated data. In particular, it's not going to show who is transmitting to whom. It isn't going to tell you who may have the virus. But NBAR can also be used to rate-limit and control or drop the traffic the worm uses to propagate itself.

Here's a sample. The format shows data in sets of triple lines. The first line of three is the Packet Count, next Byte Count, and the third line is bits per second. The following shows the top 5 (plus unknown), you might want the top 10 or 20 in a real network.

```
Voice11#show ip nbar proto top-n 5
```

Protocol	Input Packet Count Byte Count 5 minute bit rate (bps)	Output Packet Count Byte Count 5 minute bit rate (bps)
telnet	1531 91860 0	0 0 0
netbios	78 9783 0	0 0 0
custom-01	8 486 0	0 0 0
napster	4 240 0	0 0 0
bgp	0 0 0	0 0 0
unknown	6 360 0	0 0 0
Total	1627 102729 0	0 0 0

The "napster" is in fact from trying to telnet to my PC on port 5555, so you have to be flexible in interpreting the results. I used a program called netcat to send some UDP traffic to port 5678, which is what was showing up as "unknown". After I configured `Voice11(config)#ip nbar port-map custom-01 udp 5678`, it showed up as "custom-01", as in the above results. Admittedly, for this to be very useful, you have to know what ports you're interested in, which means you already know something about what's biting you.

If you only want bps, for which the command would be: `show ip nbar proto stats bit-rate top-n 5`. We didn't do this above since the bps rates are all 0 (due to using short-lived traffic bursts!).

RMON2

If you have RMON2 probes (or NAM blades!) in key parts of the network, they can show the traffic mix for various TCP and UDP ports, similarly to NBAR discovery. (Except that most reporting tools show nice bar charts instead of just numbers). If the probe is collecting source address and destination address and port information, you can then drill down, to find out who is transmitting high volumes to certain ports. We'd love to show you sample screen captures, but our home labs don't have a NAM.

NetFlow

NetFlow can provide similar data. If you know what percentage of your traffic normally goes to various ports, then you can spot unusual levels of traffic to some port(s). This would be from data aggregated by port. You can then examine the unaggregated traffic going to those ports, looking first at source addresses sorted to show high volume to the selected port (s). You can also look for abnormal traffic patterns, such as devices systematically scanning all devices in a subnet. For example, for at one customer's site we saw Nachi infected hosts displayed as the source of many flows that are destined to random destinations. By working with the MIS folks, these hosts were quickly isolated.

This technique assumes you've put NetFlow collection and reporting in place ahead of time. See <http://www.cisco.com/warp/public/732/Tech/nmp/netflow/index.shtml> for more information. Cisco sells a NetFlow collection tool. See <http://www.cisco.com/en/US/products/sw/netmgtsw/ps1964/index.html>. Cisco's strategy is for partners to provide the analysis and reporting. Pete hears that Concord, InfoVista, NetQoS and others have some NetFlow reporting capabilities (but hasn't delved into what they can and cannot do). Some are mentioned in the slightly old document at http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/netfl_wp.htm.

See also <http://www.caida.org/tools/measurement/cflowd/> for free tools for working with NetFlow. See also <http://www.caida.org/tools/utilities/flowscan/> and some of the other CAIDA tools.

Cisco has proposed NetFlow version 9 to the IETF as a way to standardize NetFlow data formats and schemas across vendors to allow network management companies to develop cross-vendor traffic reporting tools. The IPFIX working group is pursuing this, taking into account technologies from other vendors as well. See <http://www.ietf.org/html.charters/ipfix-charter.html>.

If you don't have the tools in place, you still may be able to use NetFlow. If you have a lot of activity going on, it's going to show up in the active flow cache.

```

Voice11(config)#interface Ethernet0/0
Voice11(config-if)#ip route-cache flow
Voice11(config-if)#end

<Generated some traffic>

Voice11#show ip cache flow
IP packet size distribution (313 total packets):
  1-32   64   96  128  160  192  224  256  288  320  352  384
416   448  480
  .000 .881 .047 .060 .000 .000 .000 .003 .006 .000 .000 .000 .000 .
    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
  .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000

IP Flow Switching Cache, 278544 bytes
  3 active, 4093 inactive, 11 added
  242 ager polls, 0 flow alloc failures

```

```

Active flows timeout in 30 minutes
Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 17032 bytes
 0 active, 1024 inactive, 0 added, 0 added to flow
 0 alloc failures, 0 force free
 1 chunk, 1 chunk added
last clearing of statistics never
Protocol          Total    Flows   Packets Bytes   Packets Active
(Sec) Idle (Sec)
-----
Flows      /Sec      /Flow  /Pkt      /Sec      /Flow      /Flow
TCP-Telnet          3      0.0      80       40       0.3
27.8      15.6
UDP-other          3      0.0       6      107       0.0
5.5       15.6
ICMP              2      0.0       4      100       0.0
0.0       15.4
Total:            8      0.0      33       47       0.3
12.4      15.5

SrcIf          SrcIPAddress      DstIf          DstIPAddress      Pr SrcP
DstP  Pkts
Et0/0          192.168.1.1      Local          192.168.1.200     01 0000
0000          5
Et0/0          192.168.1.101   Local          192.168.1.200     01 0000
0000          5
Et0/0          192.168.1.101   Local          192.168.1.200     06 0881
0017          36

```

If you've looked at the above really carefully, you'll notice the DstP (destination port) must be in hex, since 0x0017 is 23, the usual telnet port.

You may well want to be more specific than this. Use the '|' modifier to filter so as to see just what you need to see. This probably does mean looking at the full output at least once. (Big scroll buffer or logging for offline examination in a text editor may help.)

Switch MLS Statistics

If you have Catalyst 6500's running CatOS, consider configuring "set mls flow full". You can then do things like "show mls statistics entry ip dst-port 135" to see what's sending to that port. (This idea is from the Cisco TAC guide on W32.Blaster Worm, URL below. See that document for sample output.)

For 6500's with Native IOS, "mls flow ip full" configures it, "show mls ip" and variations display the data.

ACL or VACL

Be careful! Access lists can seriously impact router performance!

We created an access list 101 in our lab router and applied it to the interface in use. We then can immediately see:

```

Voice11#show ip access-lists
Extended IP access list 101
 10 permit udp any any eq 5678 (15 matches)
 20 permit ip any any (69 matches)

```

Ok, so if you know what ports you care about, that shows if there's any traffic on them.

If you beef up the access list by adding "log" at the end of selected lines, you'll get console or telnet session logging of matches. (Be very careful, doing this for all packets will kill your router!). If you include a rule specifying a UDP or TCP port, then the log entries will also show port numbers. If you have configured syslog logging to CiscoWorks (or a Windows or UNIX/Linux host running a version of syslogd), you can capture the log messages in a file for further analysis. (With CiscoWorks, just check the tail end of .../CSCOPx/log/syslog.log).

You can also use the 'log' keyword with VACL's in Catalyst 6500 switches (PFC needed; MSFC not needed). Similar cautions apply.

A sample follows:

```

Oct 29 10:20:44 192.168.1.200 54: 01:41:56: %SEC-6-IPACCESSLOGP:
list 101 permitted udp 192.168.1.101(137) -> 192.168.1.200(137), 1
packet

Oct 29 10:20:48 192.168.1.200 55: 01:42:00: %SYS-5-CONFIG_I:
Configured from console by console

Oct 29 10:22:47 192.168.1.200 56: 01:44:03: %SEC-6-IPACCESSLOGP:
list 101 permitted udp 192.168.1.101(138) -> 192.168.1.255(138), 1
packet

Oct 29 10:23:28 192.168.1.200 57: 01:44:44: %SEC-6-IPACCESSLOGP:
list 101 permitted tcp 192.168.1.101(2599) -> 192.168.1.200(23), 1
packet

Oct 29 10:23:37 192.168.1.200 58: 01:44:53: %SEC-6-IPACCESSLOGP: list 101
permitted udp 192.168.1.101(2621) -> 192.168.1.200(5678), 1 packet

```

Bearing in mind our performance cautions, there are a couple of ways to use this trick. One is to live dangerously, by applying and then removing the ACL quickly. Make sure you have console logging off before you do this. (Log to buffer or remote syslog). You did say you enjoy playing Russian roulette, didn't you?

Another tip is to restrict what you log. If you know what ports the worm is exploiting, log only those ports. One way to render such logging even safer is to use spare or lab gear, where if you make the router busy it doesn't matter. Enough warnings!

What does this buy us? The big thing to notice here is that we can see the source addresses. So if we're logging a port that only a worm is likely to be using, then the source is an infected computer (unless the worm spoofs source addresses, of course).

Sinkhole Router

One related idea is to create a so-called "sinkhole router". The key assumption is that worms randomly select addresses to try to infect. Some such addresses may well be bogus, as in unassigned or reserved address space. The idea is therefore to set up a dedicated (old?) router that is a "magnet" for such packets. This router may well see packets from

many if not all of the infected hosts in the network. And if you turn on access list logging on this sinkhole router, it sooner or later spots each infected host for you.

This idea solves two problems. You really don't want to be messing around with access list logging on live routers. And you don't really want to be doing it all over your network. The sinkhole router localizes the work and the risk on a spare router. Much less work, much less risk!

To set up a sinkhole router, create static routes to Null 0 for bogus destination networks. These can include all private network address space not in use within your network. Then redistribute these static routes into your internal routing protocol. Add your favorite access list with logging, set up syslog logging (or use terminal monitor), and you're gathering info.

A variation on this lets you reduce network-wide routing table impact. I like to have Internet gateway routers configured to originate default. So packets to bogus destinations will normally head for the gateways. Put the prospective sinkhole router on a LAN switch connecting to one if not both gateways. Then put the static routes on the gateways, to send packets to bogus destinations to the sinkhole router.

This begs the question of what network destinations are bogus (unassigned). There is a list of registries and other address assignment info at http://www.cert.mil/techtips/whois_by_ipaddr.htm. In particular, you may find the following document useful: <http://www.iana.org/assignments/ipv4-address-space>. That certainly gets you in the right ballpark. It may not be worth trying to be more precise than that. For a specific router template with static routes to NULL 0 for bogus destinations, see also <http://www.cymru.com/Documents/secure-ios-template.html>. If being careful, check it against the IANA list as it may be a bit old / out of date.

Packet Analyzer Software

If you have packet analyzer software such as Network Associates' Sniffer (TM), WildPackets' EtherPeek (TM), or Fluke Networks' OptiView Protocol Expert (TM), great. If not, consider the free Ethereal program for Windows or Linux, or tcpdump for UNIX/Linux. <http://www.ethereal.com/>. If you use VNC, you can use these applications for remote packet capture. Pull the capture file back and analyze it at your desk in comfort. <http://www.tightvnc.com/>. So using this software, you can hook dedicated or spare computers to switch SPAN ports and be ready to protocol analyze. This can be a great tool for troubleshooting.

The previous paragraph could give you an idea of how to use a packet analyzer in a worm storm. If you know that a certain host is infected, watch its traffic to see what the worm or virus does to spread. You can then set up a capture filter and capture all traffic to the ports used by the worm. Source addresses tell you who else might be infected.

Host Firewalls

OpenBSD systems can filter inbound traffic using the built-in packet filter "pf". If you log the blocked packets, they can easily be viewed by running tcpdump on a pseudo-interface. If you watch this, any unusual traffic to the system or probes by worms or viruses may be visible.

Personal firewalls on some computers are another early warning system. When they start popping up dialog boxes repeatedly, you know a worm storm is going on!

Loosely fitting under this heading is the Cisco CSA product. <http://www.cisco.com/en/US/products/sw/secursw/ps5057/>. This product is a hot seller recently, I hear. The claim is that without any tweaking, it stopped all the recent attacks. One attraction is that it does not use signatures that need constant updates, the way virus scanners do. Since the attacks are now getting faster than the turn-around time on the signatures, that's an attractive feature! The tie-in to the present article: like most personal firewall products, it lets you know when you're under attack.

You might be thinking "Yes, but those will only report if one system is under attack. What are the odds the randomly generated address will match that one system?" Good point! But consider that you would probably be using tools like this to protect all the servers, and possibly some laptops or desktops. If the alerts go to a central console, now you've got another source of info about what sources might be infected, etc.

IDS or PIX Reporting (CW V/SMS Monitoring for

Security; SIMS)

As noted earlier, reporting on things like worm or virus activity is what we'd hope an IDS would do. The PIX and even the security / firewall code in the routers can do some attack detection and alarming. With other vendors' firewalls, YMMV (Your Mileage May Vary). Since this article is long enough already, we won't go into this topic further at this time.

Links

The following are some of the links that I've gleaned the above ideas from. The other documents I've referred to which is not shown is the Cisco Networkers 2003 presentation SEC-2004. It's great to see so many creative ideas coming from the minds at Cisco. And since great ideas don't do any good unless you tell someone about them, thanks to the folks that published these items. We're mainly contributing to the effort by summarizing and focusing the discussion and passing on the links!

Combating Internet Worms: An Integrated Security Approach: Cisco Internet seminar on 9/4/03	http://www.cisco.com/go/semreg/blaster/123542_8
Uses of Network Management for Monitoring the IP Packet Blocks Input Queue PSIRT Advisory	http://www.cisco.com/en/US/products/hw/routers/ps133/products_white_paper09186a00801a5a68.shtml
Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks	http://www.cisco.com/warp/public/707/newsflash.html
Characterizing and Tracing Packet Floods Using Cisco Routers	http://www.cisco.com/warp/public/707/22.html
Cisco Security Notice: W32.BLASTER Worm Mitigation Recommendations	http://www.cisco.com/warp/public/707/cisco-sn-20030814-blaster.shtml
Cisco Security Notice: Nachi Worm Mitigation Recommendations	http://www.cisco.com/warp/public/707/cisco-sn-20030820-nachi.shtml

Conclusion

Increasing competence at firewalls may shift attacker focus to viruses, worms, Trojan horses, web sites that implant Trojan horses, and social engineering. Although there seem to be enough sites that still need to tighten up their firewalls or make better use of their DMZ's, so that there's still no shortage of easy targets. We'd rather be a hard target. What do you think?

Next month we'll continue the technical discussion by looking at Best Practices, for servers, routers, and switches. We think there's some new things to be said on that front as well. Yeah, some potential for it to get dry, so we'll try to provide extra lively words to make up for that! Stay tuned!

Please contact Chesapeake NetCraftsmen (<http://www.netcraftsmen.net>) if you'd like us to come do a security risk assessment, help with virus effects mitigation, help with network management or security management software (CiscoWorks!), etc.

Dr. Peter J. Welcher (CCIE #1773, CCSI #94014, CCIP) is a Senior Consultant with Chesapeake NetCraftsmen. NetCraftsmen is a high-end consulting firm and Cisco Premier Partner dedicated to quality consulting and knowledge transfer. NetCraftsmen has eight CCIE's, with expertise including large network high-availability routing/switching and design, VoIP, QoS, MPLS, IPSec VPN, wireless LAN and bridging, network management, security, IP multicast, and other areas. See <http://www.netcraftsmen.net> for more information about NetCraftsmen. Pete's links start at <http://www.netcraftsmen.net/welcher>. New articles will be posted under the Articles link. Questions, suggestions for articles, etc. can be sent to [pjw <at> netcraftsmen <dot> net](mailto:pjw@netcraftsmen.net).

Carole Warner Reece (CCIE #5168) is a Senior Consultant with Chesapeake NetCraftsmen with over 15 years of experience in internetworking. Her prior assignments range from technical to sales and marketing and managing technical staff. Carole recently designed challenging new labs for version 5 of the Cisco CIT Troubleshooting course. She previously developed and worked with other authors to produce networking skills based labs at a wide variety of levels.

10/29/2003

Copyright (C) 2003 Peter J. Welcher