Analysis of an Internet-wide Stealth Scan from a Botnet

A. Dainotti, A. King, K. Claffy, F. Papale*, A. Pescapè*

alberto@caida.org

CAIDA - University of California, San Diego
*University of Napoli Federico II, Italy
THE “SIPSCAN”
Feb 2011

• A “/0” scan from a botnet
• Scanning SIP Servers with a specific query on UDP port 5060 and SYNs on TCP port 80

2011–02–02 12:15:18.913184 IP (tos 0x0, ttl 36, id 20335, offset 0, flags [none], proto UDP (17), length 412) XX.10.100.90.1878 > XX .164.30.56.5060: [udp sum ok] SIP, length: 384

REGISTER sip:3982516068@XX.164.30.56 SIP/2.0
Via: SIP/2.0/UDP XX.164.30.56:5060; branch=1F8b5C6T44G2CJt; rport
Content-Length: 0
From: <sip:3982516068@XX.164.30.56>; tag
    =1471813818402863423218342668
Accept: application/sdp
User-Agent: Asterisk PBX
To: <sip:3982516068@XX.164.30.56>
Contact: sip:3982516068@XX.164.30.56
CSeq: 1 REGISTER
Call-ID: 4731021211
Max-Forwards: 70
DARKNET
The UCSD Network Telescope

UCSD Network Telescope
Darknet xxx.0.0.0/8

Infected Host Scanning the Internet
OVERVIEW

isolating the “SipScan”

• Thanks to the unique payload fingerprint we could isolate it without inferences
OVERVIEW

isolating the "SipScan"

• Thanks to the unique payload fingerprint we could isolate it without inferences

![Graph showing packets per second of the scanning event and UDP port 5060 traffic over time.](image-url)
# OVERVIEW

## some quick statistics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># of probes (1 probe = 1 UDP + multiple TCP pkts)</td>
<td>20,255,721</td>
</tr>
<tr>
<td># of source IP addresses</td>
<td>2,954,108</td>
</tr>
<tr>
<td># of destination IP addresses</td>
<td>14,534,793</td>
</tr>
<tr>
<td>% of telescope IP space covered</td>
<td>86.6%</td>
</tr>
<tr>
<td># of unique couples (source IP - destination IP)</td>
<td>20,241,109</td>
</tr>
<tr>
<td>max probes per second</td>
<td>78.3</td>
</tr>
<tr>
<td>max # of distinct source IPs in 1 hour</td>
<td>160,264</td>
</tr>
<tr>
<td>max # of distinct source IPs in 5 minutes</td>
<td>21,829</td>
</tr>
<tr>
<td>average # of probes received by a /24</td>
<td>309</td>
</tr>
<tr>
<td>max # of probes received by a /24</td>
<td>442</td>
</tr>
<tr>
<td>average # of sources targeting a destination</td>
<td>1.39</td>
</tr>
<tr>
<td>max # of sources targeting a destination</td>
<td>14</td>
</tr>
<tr>
<td>average # of destinations a source targets</td>
<td>6.85</td>
</tr>
<tr>
<td>max # of destination a source targets</td>
<td>17613</td>
</tr>
</tbody>
</table>
REL WORKS

• Analyses of botnet scans
  - Z. Li, A. Goyal, Y. Chen, “Honeynet-based Botnet Scan Traffic Analysis”, Book Botnet Detection (Adv. in Inf Sec.) 2008

• Botnet code analysis

• Coordinated scans
  - S. Staniford, V. Paxson, N. Weaver; “How to Own the Internet in Your Spare Time”, Usenix Sec. Symp. 2002
  - Carrie Gates, “Coordinated Scan Detection”, NDSS 2009

Cooperative Association for Internet Data Analysis
University of California San Diego
COORDINATION
(lack of)


• “By analyzing the source code of five popular families of bots we studied different dimensions of scan strategies employed by botnets. […] Overall, we find they employ simple scanning strategies.”

• “Our dataset analysis accords with the above capabilities: most scanners we observe either use simple sequential scanning (IP address increments by one between scans) or independent uniform random scanning.”
COORDINATION
..and Redundancy


• “Redundancy. Since the bots in a botnet can readily be lost due to detection or due to the host computer going offline, the botmaster will prefer instructing multiple bots to scan the same addresses.”

• A simple and effective approach is to ask each bot to independently scan the specified range in a random uniform fashion. [...] In the source code analysis we find the most popular such one implemented to date (four out of five bot families implemented this strategy).

• Assumptions in the extrapolation of global properties:
  “[..].. second. each sender has the same global scan scope.
  [...] We argue that these two fundamental assumption likely apply to any local-to-global extrapolation scheme.
UNSPOOFED

Because...

- It seems to be a scan (UDP requests + TCP SYNs). No purpose in spoofing
- No IPs from our /8 or from unassigned space
- IPIDs and src ports from scanning hosts are consistent for the same host
- Egyptian outage: we were actually not seeing “egyptian” IPs when the Egypt was isolated from the rest of the Internet
UNSPOOFED

the “Egyptian Killswitch” (Feb 2011)

- No SipScan pkts are geolocated to Egypt during the Egyptian outage!

Animation created with an improved version of Cuttlefish, developed by Brad Huffaker
http://www.caida.org/tools/visualization/cuttlefish/
/0 SCAN

DShield

http://www.dshield.org
• We identified flow-level properties (e.g. 1 pkt + PS size) that allowed to spot the same traffic in MAWI/WIDE traces, which are anonymized.
• A few different /8 networks were found in the MAWI traffic associated with the Sipscan

http://mawi.wide.ad.jp/mawi/
SOURCE PORT CONTINUITY

(in theory)

• consider a single host
• using standard sockets for opening each new TCP connection or UDP session
• a new source port is assigned to each new connection/session
• on some operating systems of the Microsoft Windows family, the source port assigned is obtained by incrementing a **global** counter: `Src_port++` in range 1025 - 5000
• At the telescope: by looking at the “difference” between the source ports of two subsequent packets from the same bot we can infer how many connections/sessions it opened in between them
• If the bot probes at each round all the 256 /8 networks then we expect this difference to be 512
/0 SCAN

Exploiting source port continuity

- Src_port++ in range 1025 - 5000
- ~512 average increments between 2 “visits” to the telescope
HILBERT CURVE

http://xkcd.com/195

xkcd
HILBERT CURVE

Heatmaps

- The 1-dimensional IPv4 address space is mapped into a 2-dimensional image using a Hilbert curve.
- CIDR netblocks always appear as squares or rectangles in the image.

Software for hilbert-based IP heatmaps @ http://www.measurement-factory.com
REVERSE BYTE ORDER

progression

000.140.100.000
Most significant byte during the scan. However, the analysis of botnet activity is available at both the sources and the targets of the sipscan. This composite animation is available at both the sources and the targets of the sipscan.

We also created a composite animation which combines the world map animation into a single synchronized view of both the natural and reverse byte order heatmaps with the channel.

The network's distributed architecture of the botnet, which allows for a large number of hosts to scan in parallel, is utilized to probe the address space. When a host is found, it has a short time to respond before the next host in line begins the scan. One of these hosts is dedicated to probing the entire address space in reverse order. The progress is available at .

The animation proves the reverse byte order progression is rigorously followed by the sipscan throughout the 0.6 days of the scan. This animation of target progression shows the square uniformly filling in their natural byte order does not reveal a particular pattern in the target progression. The reverse byte order animation also shows the progression is available at .

The curve highlights pixels corresponding to IP addresses that have been covered by the scan. In contrast, reversing the original sequential order used in this scan has significant implications for understanding the behavior of bots during the entire 0.6 days. The linear slope indicates a constant arrival of new bots participating in the scan. To partially take into account the effect of dynamic IP address assignment, we also plot the cumulative sum of unique source IPs scanning per 5-minute interval, representing the evolution of the scan over time. The dashed line represents the number of unique source IPs scanning per 5-minute interval, representing the evolution of the scan over time. The continuous line with circles is the cumulative percentage of probed up to that time (February 00–00 GMT) in this frame. The animation represents the progression over time in Figure 00– Snapshot of our animation representing the progression over time.

For each frame, the pixels correspond to a 9×9 bitmap of size with each pixel being as a bitmap of size 9×9. The curve is displayed through all the possible combinations. The animation shows the progression in Figure 09– Snapshot of our animation representing the progression over time.

A significant finding was that reversing the assignment of bot activity—the progression is strictly observed by all available at . The bots during the entire 0.6 days. During the botnet's active periods, the large number of global targets close to 20,000 between two subsequent probes to the darknet. This finding indicates that each bot may have been assigned the entire set of possible values simultaneously. Each host then probes a location continuously for several days, probing an average of about 250,000 probes (about 6500) and the average number of destinations for the most significant byte is incremented by one. The least significant bytes of each address is reversed before mapping it into the address space.

Highlighting many pixels corresponding to IP addresses that have been probed prior to that point in time are highlighted. We also added a brightness decrement, which indicates the reverse sequential order of the least significant two bytes yields the progression is available at .

In contrast, reversing the original sequential order used in this scan has significant implications for understanding the behavior of bots. For example, the cumulative sum of unique source IPs 1continuous line with squares shows the cumulative percentage of bots that probed our darknet throughout the 0.6 days. The continuous line with square symbols shows the cumulative percentage of bots that probed our darknet throughout the 0.6 days. Figure 00– Arrival of new bots. The continuous line with squares shows the arrival of new bots that probed our darknet throughout the 0.6 days. The continuous line with square symbols shows the arrival of new bots that probed our darknet throughout the 0.6 days. Figure 00– Arrival of new bots. The continuous line with squares shows the arrival of new bots that probed our darknet throughout the 0.6 days.

The scanning statistics in Table 0 show that the number of global targets close to 20,000 between two subsequent probes to the darknet. This finding indicates that each bot may have been assigned the entire set of possible values simultaneously. Each host then probes a location continuously for several days, probing an average of about 250,000 probes (about 6500) and the average number of destinations for the most significant byte is incremented by one. The least significant bytes of each address is reversed before mapping it into the address space.
BOT TURNOVER
most src IPs leave constantly
BOT TURNOVER

few src IPs stay for a while

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># of probes (1 probe = 1 UDP + multiple TCP pkts)</td>
<td>20,255,721</td>
</tr>
<tr>
<td># of source IP addresses</td>
<td>2,954,108</td>
</tr>
<tr>
<td># of destination IP addresses</td>
<td>14,534,793</td>
</tr>
<tr>
<td>% of telescope IP space covered</td>
<td>86.6%</td>
</tr>
<tr>
<td># of unique couples (source IP - destination IP)</td>
<td>20,241,109</td>
</tr>
<tr>
<td>max probes per second</td>
<td>78.3</td>
</tr>
<tr>
<td>max # of distinct source IPs in 1 hour</td>
<td>160,264</td>
</tr>
<tr>
<td>max # of distinct source IPs in 5 minutes</td>
<td>21,829</td>
</tr>
<tr>
<td>average # of probes received by a /24</td>
<td>309</td>
</tr>
<tr>
<td>max # of probes received by a /24</td>
<td>442</td>
</tr>
<tr>
<td>average # of sources targeting a destination</td>
<td>1.39</td>
</tr>
<tr>
<td>max # of sources targeting a destination</td>
<td>14</td>
</tr>
<tr>
<td>average # of destinations a source targets</td>
<td>6.85</td>
</tr>
<tr>
<td>max # of destination a source targets</td>
<td>17,613</td>
</tr>
</tbody>
</table>
targets which we believe is likely a function of a parameter.

But we discovered a correlation between coverage and overlap in rate of the scan or specific subnets being scanned. However, the overall coverage and overlap of target addresses are independent of the number of bots active at any given time. These properties – the representation of the address bytes – are independent of the probed IP addresses being scattered all over the address space without clusters or holes in both the standard and reverse-byte order representation.

Distinct bots whether probed zero, one, or multiple times, make the scan impressive than one bot, and on average a targeted IP is probed by 850 bots. About 1.5 million IP addresses were probed by more than one bot, but they would most likely arrive from 0.5 million IP addresses. Not only would an automated intrusion detection system be unable to detect these probes, but they would most likely arrive from 0.5 million IP addresses. In a 69 network, see only 0.5 million IP addresses. In a 69 network, see only 0.5 million IP addresses. In a 69 network, see only 0.5 million IP addresses. These properties make detection highly unlikely.

The high bot turnover rate makes the scan impressive. Most bots sent few packets, for example, over a third of the bots sent up to 87 packets. More than 8 million bots over a third of the unique source IPs over a third of the unique source IPs over a third of the unique source IPs sent a single packet during the entire scan. The high bot turnover rate makes the scan impressive. The number of bots that sent more than 877 packets is two orders of magnitude smaller. This suggests rapid turnover of bots during the scan.

The correlation between coverage and the high bot turnover rate makes the scan impressive. The high bot turnover rate makes the scan impressive. The high bot turnover rate makes the scan impressive. The correlation between coverage and overlap is evident, and is consistent with a probabilistic mechanism in the choice of the targets that can be mediated by the botmaster to trade off completeness and redundancy of scanning. The finding illustrated in Figure 8 corresponds to different configurations of the scan. The figure shows the distribution of the number of packets sent by each bot. The diagram on the left uses a log-log scale to show all the data, whereas the diagram on the right uses a linear scale.

The high bot turnover rate makes the scan impressive. More than 8 million bots over a third of the unique source IPs over a third of the unique source IPs over a third of the unique source IPs sent a single packet during the entire scan. More than 8 million bots over a third of the unique source IPs sent a single packet during the entire scan. The high bot turnover rate makes the scan impressive. The number of bots that sent more than 877 packets is two orders of magnitude smaller. This suggests rapid turnover of bots during the scan. The high bot turnover rate makes the scan impressive. More than 8 million bots over a third of the unique source IPs over a third of the unique source IPs over a third of the unique source IPs sent a single packet during the entire scan. The high bot turnover rate makes the scan impressive. The number of bots that sent more than 877 packets is two orders of magnitude smaller. This suggests rapid turnover of bots during the scan.
COVERAGE & OVERLAP

different phases w/ different parameters?
COVERAGE & OVERLAP

“probes sent to reverse /16 subnets”

• Example of a reverse /16: *.*.45.123
• From the UCSD Telescope we can see only pkts to xxx.*.45.123
SIPSCAN FEATURES

• Operated by a botnet
• Global vs Global
• Observed by a /8
• No inferences on pkts: unique payload “signature”
• Lasting 12 days
• Sequential progression in reverse byte order
• Continuous use of new bots
• Stealth: IP progression, speed, use of new bots
• Coordination between sources (global sequential progression and small redundancy)
• Targeting SIP
THANKS