botnets

Jim Binkley
PSU CS
outline

• intro – the problem
• how a botnet works
• tools and techniques for combat
• ourmon
• summary
Botnets

THE KILLER WEB APP

- Important Information on the Newest Internet Threat: Botnets, Zombie Armies, and Bot Herders
- Answers Your Questions: What are They? How Do They Spread? How Do They Work? How Can I Detect Them When They Don’t Want to be Seen? What Tools are Available to Fight this Menace?
- Complete Coverage of OurMon and Other Open Source Tools

Craig A. Schiller
Jim Binkley
• botnet definition
  – a distributed network of exploited clients
    • aka zombies
  – traditionally run from 1-N IRC servers
    • aka C&C or C-squared
    • DNS involved in interesting ways (fastflux)
  – run by individual known as botherder
  – may speak of botnet client or botnet server mesh
C&C

IRC protocol

botnet clients

100 to 100000
what’s it for?

- to make money e.g.,
- botnet clients send SPAM
  - drugs meaning pharmaceuticals
  - and other stuff
- botnet clients steal IDENTITY
  - phishing attacks
  - pharming attacks
  - key logging attacks
- extortion
  - we encrypted your files – want them back?
  - pay up or we DOS you
how is this different from before?

• evolution in hackery is true BUT
• hackery used to be immature male problem
• now it’s immature males working as a criminal enterprise
  – or working for a criminal enterprise
• often more international than before
  – which makes law enforcement very difficult
previous attacks

- not as well coordinated
- disruption was the motive – not profit
- Internet used more now for business
  - THEREFORE more business-related crime
- certain sacrosanct Internet protocols
  - infrastructure like DNS
  - DNS is now under attack – misuse is more and more common
one recent paradigm called “clicks for hire”

• google and others may pay money if advertising on web sites is “visited”
• therefore a botnet of 100000 computers may auto click on websites
• therefore a particular well connected host may become a server
  – with porn, viagra, loan web links on it
• google on: phpbb site:yoursite.com
how does it work

• how does a computer get infected?
  – you don’t run windows update
  – you practice poor password hygiene
  – you download something off of the Internet
    • via email
    • via the web
    • via a side effect of using the Web (IE)
    • Microsoft may not keep up the patches to the level necessary (see the 1st item above)
what happens to your computer

• you are now running a very complex app
  – agobot
  – sdbot
  – rbot
  – mytob
  – conficker, stormworm, zbot
• this may include a rootkit
• a backdoor (call it telnet but it might be ftp, tftp, dameware, etc)
• the new software tries to kill off AV software
• AV software may not be able to eradicate it
the new zombie

• “rallies” to the C&C
• contacts it via IRC
  – and/or an http GET
  – and/or possible newish P2P like protocols
• note that a zombie is just software
• you (the zombie) may be instructed to participate in a fan-out attack
typical bots

- contain some set of exploits
  - e.g., aimed at Microsoft file share exploits
  - SYN attacks at ports 445, 139 common
  - may be ordered to scan with TCP syns
- may make password attacks on local computers (user/password guesses)
- may take part in a DDOS attack
  - against microsoft
  - against an air force installation
  - against the White House
  - against Ma and Pa smallbiz.biz for extortion
what do the white hats do?

• they try to stomp out the C&C for the most part
• and if possible fix the infected ZOMBIE but
• many many zombies are to be found in broadband land
  – this is a serious problem
clear direction in black-hat land

• make c&c mechanism more better in terms of redundancy
• oddly paralleling making DNS itself more redundant (at least for roots)
what do the blackhats do to protect the C&C

- clients are programmed with DNS name/s not IP addresses for C&C
- multi-homed DNS
  - lsass.exploited.org is actually 3 IPs 1,2,3
- short DNS TTLs for clients
  - remap DNS often, check at boot
- dynamic DNS used thru commercial site
  - so change IP address
IRC – the protocol

- user logins to channel (chatroom name)
  - with optional password at a server on a port
    (TCP port 6667 but any port can work)
- server makes sure channel messages forwarded to all clients interested in channel
- IRC basic messages
  - join/nick/ping/pong/privmsg
IRC bot

• originally simply something automated that uses IRC
  – file xfer (so-called warez transfer)
  – might be benign like setup a scheduler “irc as meeting-maker”)
  – might be a game
  – might try to fool user into a “real” conversation with a computer robot (turing test)
  – might be agobot (not benign)
agobot and other bots overload

• IRC privmsg to simply include commands like
  – ddos this ip
  – scan these ip nets at this port with attack N
  – talk back if a scan succeeds
  – send spam
  – take new spam template as output
  – … it’s just software …
modern bots may not use IRC

• 1. may use p2p (a la emule/edonkey as does stormworm)
  – stormworm seems to use email (greeting card)
  – send email as spambot

• 2. may use http GET or POST for client to talk to server infrastructure
  – port 80 traffic may be “harder to spot”
zbot/zeus/wsnpoem

- http periodic update from client to server
- See for more info: https://zeustracker.abuse.ch/monitor.php
fast-flux DNS

1. given a bot server DNS name hardwired in the bot client,
   - the advisory makes the IP addresses change
   - to protect against one IP address being lost
   - mybot.info -> 10.0.0.1, 192.168.1.2, etc.

2. DNS servers themselves may be compromised and available for hire
   - therefore viagra1.info, viagra2.info may appear in spam as valid DNS addresses for a short period of time (long enough to be verified as real)

“we control the horizontal and the vertical”
ddos attacks

- tcp syns of course
- udp small packets
- DNS reflection attacks
  - open dns server - ns1.bar.com
    - resolves fooledyou.com for 3rd party
    - amplification attack if small query causes large response
    - sent to innocent 3rd party due to faked ip
    - modern version of “smurf”
    - fair number of DNS servers are open resolvers
tools and techniques

- are they distributed, network-wide, or per-host
- network tools and techniques
- per host tools and techniques
- intrusion detection
- anomaly detection
- “signature” detection
- preventative measures
some examples

• network-based
  – graphics to watch for anomalies
  – cricket and ourmon
  – signature-based IDS to watch for known attacks (like snort)
  – firewalls can keep them out

• honeypots and darknets
  – darknets can see scanning
  – honeypots can capture a bot and take it apart to see how it ticks (see nepenthes.mwcollect.org)
ourmon outline

• background
• experimental flow tuples
• botnet server mesh detection
• botnet client mesh detection
• conclusions
PSU’s network

• 26k students/faculty/staff
• 100s Ethernet switches, 10k lit ethernet ports
• wide-spread wireless “pubnet”, 802.11b/g
• typical daily traffic
  – 60k pps at peak periods
  – 200-300 mbits total, more to Internet, than from Inet
  – see next bullet item

• we have dorms (resnet) – easily infected – not centrally managed
  – massive p2p bittorrent/gnutella traffic
ourmon architectural breakdown

pkts from NIC/kernel BPF buffer

probe/FreeBSD

runtime:
1. N BPF expressions
2. + topn (hash table) of flows and other things (tuples or lists)
3. some hardwired C filters (scalars of interest)
4. PCRE tags for large-scale traffic analysis

30-second summaries

graphics engine/BSD or linux

outputs:
1. RRDTOOL strip charts
2. histogram graphs
3. various ASCII reports, hourly summaries or report period

ourmon.conf config file
scan count graph (worm count) in Jan. 2005

2k external host attack (DDOS) on infected host running IRC
recent large ddos attack

• fundamental pkts graph looks like this

![Packet Rate Graph](image-url)
that’s 869k pps – we have physical gE connection to Inet …
botnet situation

• over the last few years emerging picture
  – large percentage of our infections botnet related

• collateral damage common:
  – Jan 06/wireless subnet knocked off air due to DDOS attack
  – large and vicious DDOS attacks have occurred in OUS systems (previous pic)

• large amounts of TCP-based scanning aimed at ports 139/445

• decided to create IRC mesh detection module in ourmon to look for IRC-related malware

• goal: basic IRC statistics plus coupling of IRC to scanning module elsewhere in ourmon
infrastructure – 3 tuples in ourmon (irc new, tcp syn old)

• every thirty seconds extract 3 experimental flow tuples:
  • irc channel tuple:
  • irc host tuple:
  • tcp syn tuple
    – coupled with scan detection attribute called
      – tcp work weight
• IRC: we look at layer 7 IRC data, and use a snap size of 256 bytes.
irc tuples and stats

• we extract these 4 IRC messages:
  – JOIN, PRIVMSG channel-name
  – PING, PONG for client/server connectivity
• we want: IP addresses in channel names
• also client/server information taken from directionality of IRC messages
• per host and channel stats counters
• also per network stats counters, total message kinds of all 4 kinds – graphed with RRDTOOL
irc measures

• irc channel tuples:
  channel name, message counts, list of IPs

• irc node tuples:
  ip address, message counts, weak tcp ww, client/server flag

• TCP work weight: (comes from syn tuple)
  per IP ww = (Syns sent + Fins sent + Resets returned)/total pkts

view this as a **rude efficiency measurement**: 100% means you are sending control packets.
TCP ww

- we have years of experience with it
- < 50% is normal over some number of minutes
- not only attribute used for scan detection:
  - strength: typically use 1 syn/second at least
  - 2-wayness of data: typically look at this as additional attribute in 30-second scan determination
  - counts of L3 and L4 unique destinations
- strength and 2-wayness not used here:
  - IRC version of TCP work weight is weaker
- ww often affected by P2P lack of connectivity – especially with gnutella
high abnormal scanner count – ironically was the real alert

some kinda distributed tcp syn scan right?, wait … let’s look at the IRC data
bot server detection: uh-oh, irc RRD has ping/pong way UP!

![Graph showing IRC statistics]
hourly irc summary stats like so:

- channel  msgs  ips  scanners  evil
- f        157k  36k  1700  you tell me
- x        81k  13k  712
- normalirc  5k  20  0

- about 50k remote hosts with one campus botserver in several IRC channels
- a botclient “just changed” into a botserver Friday about 10 am, and acquired many friends fast
botserver conclusions

- from pure IRC POV:
  1. ping/pong counts
     - entire IRC nets at PSU 40/period, not 2k/period
  2. number of IPs in channel
     - biggest IRC channel 20 per day, not 10-50k
  3. total IRC server messages
     - pings/pongs/privmsgs elevate the server
- interesting: total number of high TCP wws
  - external hosts that cannot connect to on-campus bot server (running on windows system)
TCP syn point of view - stats

• 1. L3D/L4D: interesting but statistically weak result
• on the 2 days of the bot server
  – bot server IP had highest count of average L3 destinations per sample period for any campus host
  – 1100 versus next highest which was a web server
  – web server and/or p2p clients typically < 1000
  – all you really say: will score high for that attribute
L4 POV – more stats

• 2. Syn count per period
  – highest on day 1, less so (still bad) on day 2
  – but it was scanning on day 1 as a normal bot client

• 3. pkt count for sent/recv. pkts HIGHEST on day 2
  – RECV pkts/SENT pkts 10/1
botnet client detection

• typical IRC data gives us small meshes on campus of
  – max: 20, min: 2 IRC channels
  – ports used may be 6667, but may vary
  – some automated bots exist (devoted to traditional IRC phenomenon like audio/video dissemination)
  – we have dorms …

• what seems to happen though is that the botnet client meshes SCAN with greater than one host during the day

• we therefore need an hourly/daily summarization
# ubuntu channel - benign

<table>
<thead>
<tr>
<th>ip</th>
<th>tmsg</th>
<th>ping</th>
<th>pong</th>
<th>privmsg</th>
<th>ww</th>
<th>server</th>
</tr>
</thead>
<tbody>
<tr>
<td>net1.1</td>
<td>1159</td>
<td>1912</td>
<td>1910</td>
<td>6494</td>
<td>43</td>
<td>H</td>
</tr>
<tr>
<td>net1.2</td>
<td>7265</td>
<td>619</td>
<td>622</td>
<td>5086</td>
<td>0</td>
<td>H</td>
</tr>
<tr>
<td>net1.3</td>
<td>1721</td>
<td>4123</td>
<td>4100</td>
<td>7069</td>
<td>37</td>
<td>H</td>
</tr>
<tr>
<td>net2.1</td>
<td>2815</td>
<td>3913</td>
<td>3904</td>
<td>1711</td>
<td>0</td>
<td>S</td>
</tr>
</tbody>
</table>
### F7 - an evil client mesh

<table>
<thead>
<tr>
<th>ip</th>
<th>tmsg</th>
<th>ping</th>
<th>pong</th>
<th>privmsg</th>
<th>ww</th>
<th>server</th>
</tr>
</thead>
<tbody>
<tr>
<td>net1.1</td>
<td>1205</td>
<td>377</td>
<td>376</td>
<td>428</td>
<td>42</td>
<td>H</td>
</tr>
<tr>
<td>net1.2</td>
<td>113</td>
<td>39</td>
<td>43</td>
<td>25</td>
<td>96</td>
<td>H</td>
</tr>
<tr>
<td>net1.3</td>
<td>144</td>
<td>60</td>
<td>61</td>
<td>21</td>
<td>94</td>
<td>H</td>
</tr>
<tr>
<td>net1.4</td>
<td>46</td>
<td>12</td>
<td>14</td>
<td>17</td>
<td>90</td>
<td>H</td>
</tr>
<tr>
<td>net1.5</td>
<td>701</td>
<td>343</td>
<td>345</td>
<td>11</td>
<td>90</td>
<td>H</td>
</tr>
<tr>
<td>net2.1</td>
<td>1300</td>
<td>587</td>
<td>593</td>
<td>101</td>
<td>16</td>
<td>S</td>
</tr>
</tbody>
</table>
evil channel sort – rank channels based on simple metric

• f7 ahead of ubuntu –
  – given 4/6 scanners compared to none
• max work weight during day kept is important idea
  – out of set of N, how many were scanners at any time?
• key idea: > 1 scanner in channel
  – plus of course other attributes in logs help
    – including ports
    – length and intensity of scanning
more information

- see [http://www.cs.pdx.edu/~jrb](http://www.cs.pdx.edu/~jrb)
- [http://ourmon.sourceforge.net](http://ourmon.sourceforge.net)
summary

- botnets == scourge
- basic measures apply
  - if you run windows
  - run usoft update
  - run their firewall or get a better one (Zone Alarm is good)
  - run AV software (AVG is free)
  - sane password policy in an enterprise or at home
more information on the Internet

• www.shadowserver.org
• www.dshield.org
• if university affiliation
  – www.ren-isac.net