Tracking A Zombie Army

(2005 Update)

James Lick

Chair APCAUCE

jlick@jameslick.com

Evolution of Spamming

- Regular mail server on a T1 line
- Throw-away dialup/broadband accounts
- Open relays
- Open proxies
- Hacked servers
- Asymmetric routing over dialup
- Zombie Armies

Evolution of Malware

- For fun: seeing what is possible
- Prank for recognition
  - Get your trojan/virus/worm in the newspaper
  - DDOS a company and see it on TV
- Profit
  - Sell anonymous access to spammers
  - DDOS for extortion
  - Keylog/phishing for identity theft to steal money
  - Clickbots to generate fake advertising revenue
  - Use to distribute more malware
  - Sell malware removal products to victims

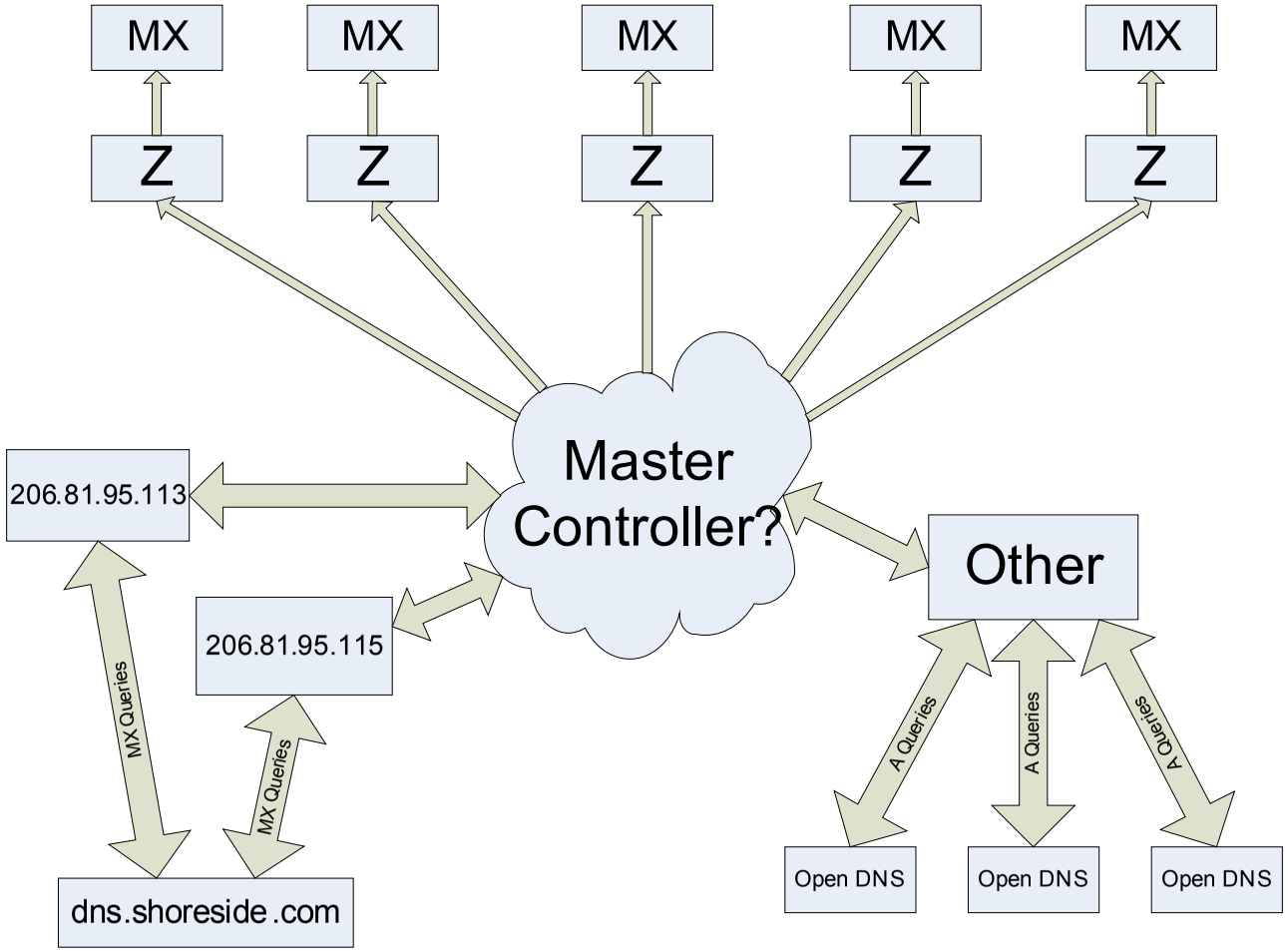
What Are Zombies?

- Various type of malware install programs to use your machine – also known as bots
  - Remote Admin Tools
  - Anonymous Proxies (password protected)
  - Keyloggers
  - Clickbots
  - Web servers
  - Virus and spam email distribution engines
  - Anonymous IRC servers
- Usually controlled via anonymous IRC servers or p2p methods

What is a Zombie Army?

- A collection of several Zombie PCs controlled by a single group – also called botnets
- 4-10 million compromised computers on the net actively used for abuse
- One group claimed to control 500k hosts
- More and more every day: 50-100k+
- Used for spamming since circa 2003

Zombie Army Architecture



Who controls Zombie Armies?

- Hard to say
- Best indication is Eastern European and Russian groups associated with organized crime
- Some programming contracted to Indian and Chinese programmers
- Zombies are usually compromised by one group and resold to another

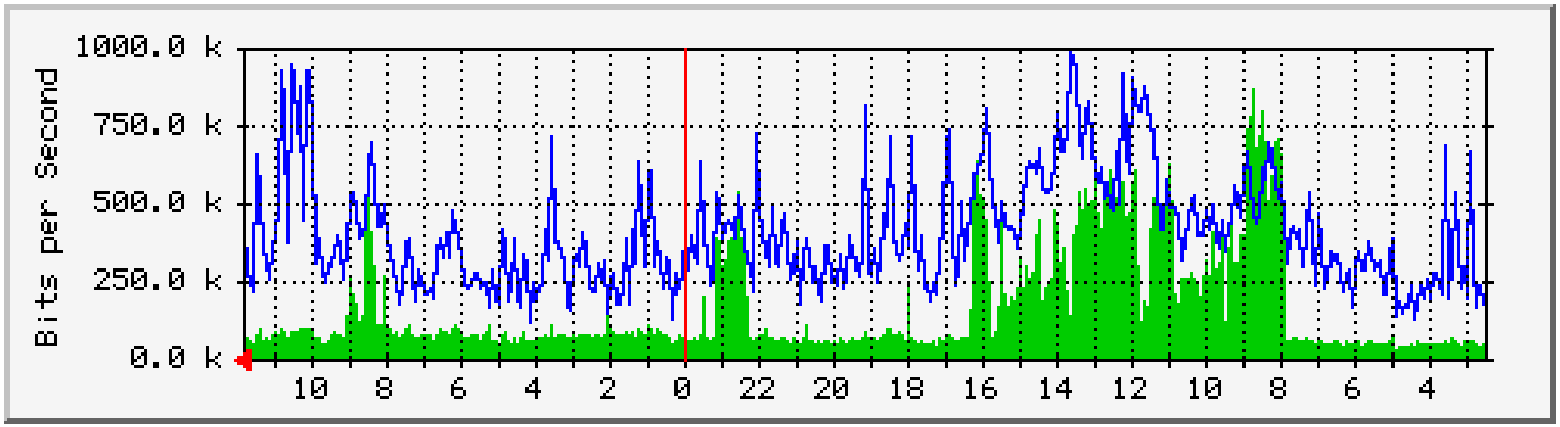
How do you track Zombie usage?

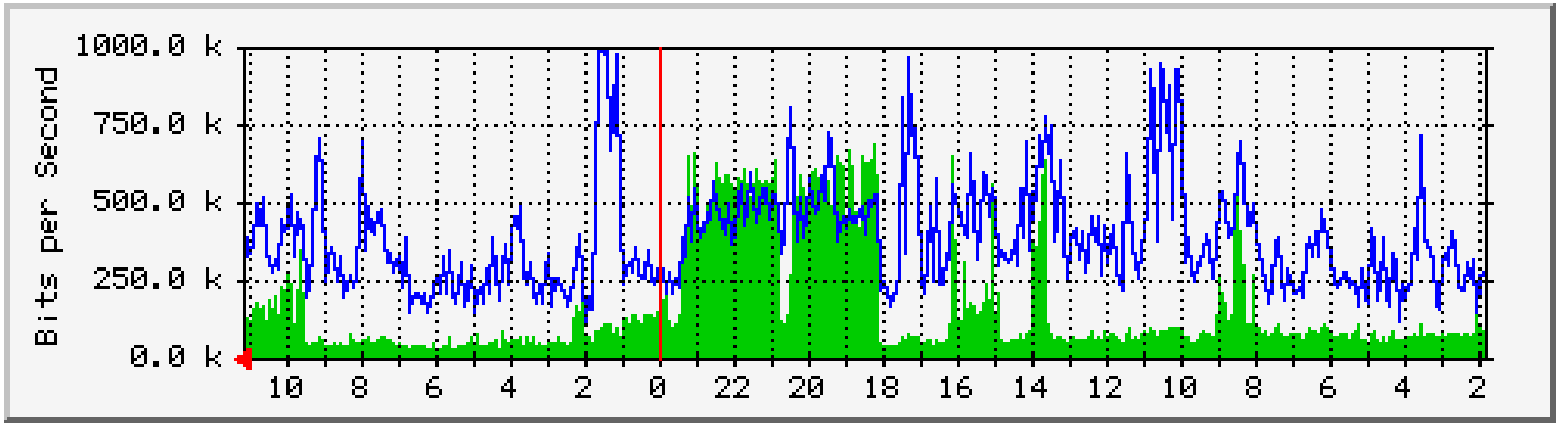
- It’s impossible!!!
- No, it’s just very, very difficult.
- The evidence is there if you have
  - Luck
  - A Search Warrant or Subpoena
  - Plenty of resources
- Lots of other crimes are very difficult to solve

My DNS server is attacked

- My in.named process was eating CPU
- My bandwidth usage was higher than normal
- Lots of DNS queries from 206.81.95.113 and 206.81.95.115
- All queries are MX queries
- Oops! I had an “Open DNS Server”
  - Recursion was open to all
- I was essentially being DNS DOS’d

Unusual amount of incoming traffic





Email delivery in a nutshell

- DNS MX lookup for domain
- DNS A lookup for MX (optional)
- Open SMTP port 25 connections to MX systems until one answers
- Send SMTP commands
- Mail server accepts message

Securing the server

- Turn off recursion to outside hosts:

```
acl internal {
        192.168.168.0/24;
        66.92.182.240/28;
};

options {
        allow-recursion { internal; };
};
```

- Firewall out the attackers:
  - block in from 206.81.80.0/20 to any

What just happened?

- Why all these lookups?
  - Attacking me?
  - Spamming people?
  - Distributing viruses?
- Who was attacking?
  - 206.81.80.0/20 is AceTech USA
    - Spamhaus says they are mortgage spammers
  - 206.81.95.0/24 is Clear Tech Services
- How can I find out more?

How to find out more

- Could I give bogus results and see what happens?
- How can I give bogus results to these clients?
- Could BIND 9's view feature help me lie?
- How do I make this all work?
- What will happen?

Configuring BIND to lie

- BIND 9 has ‘view’ feature
- ‘view’ is usually used for ‘Split DNS’
  - Hide internal hosts from the Internet without running multiple servers
- Make an ACL of who you want to lie to:

```
acl attackers {
        206.81.80.0/20;
};
```

Designing the fake view

- Make a ‘fake’ view first in the file:

```
view "fake" {
        match-clients { attackers; };
        recursion no;
        zone "." {
                type master;
                file "static/fake.named.root";
        };
       zone "com" {
                type master;
                file "static/fake.named.com";
        };
};
```

- Wildcards in root zone don’t work

Fake zone examples

- Add in a fake ‘root’ or ‘.’ zone:

```
.       IN      SOA     dns.shoreside.com. jlick-dns.drivel.com. (
                1 1800 900 259200 3600 )
        IN      NS      dns.shoreside.com
dns.shoreside.com       IN      A       66.92.182.248
```

- Add in a fake “com” zone:

```
$ORIGIN .
$TTL 86400
com     IN      SOA     dns.shoreside.com. jlick-dns.drivel.com. (
                11 1800 900 259200 3600 )
            IN      NS      dns.shoreside.com
dns.shoreside.com       IN      A       66.92.182.248
*.com   IN      MX      10 smx1.tcp.com
```

Make a view for regular zones

- Add your regular zones in a view last, so it will be the fall-through default:

```
view "real" {
        match-clients { any; };

        zone "." {
                type hint;
                file "static/named.root";
        };
…
};
```

See what happens next…

- Within seconds, hundreds of systems from all over the world start calling
- Most of them already known zombies on the CBL list
- sendmail on that system melts down
- Everything is bounced, so I don’t know the content of the mail

Collecting mail attack data

- Looking at available ‘honeypot’ software for collecting mail sessions, most is inefficient and fragile
- Chris Lewis of Nortel has a patched version of postfix’s smtp-sink which offers more logging
- Smtp-sink is a very efficient multi-threaded C program, and stood up to the challenge

Spamhaus was right!

- Typical spam sample collected:
  - HELO <my-ip-address>
  - “If you are paying more than 3.6% on your mortgage, we can slash your monthly payment!”
  - URL in gogetdealz.com domain
  - Not CAN-SPAM compliant
    - Forged headers, no opt-out, no mailing address, etc.

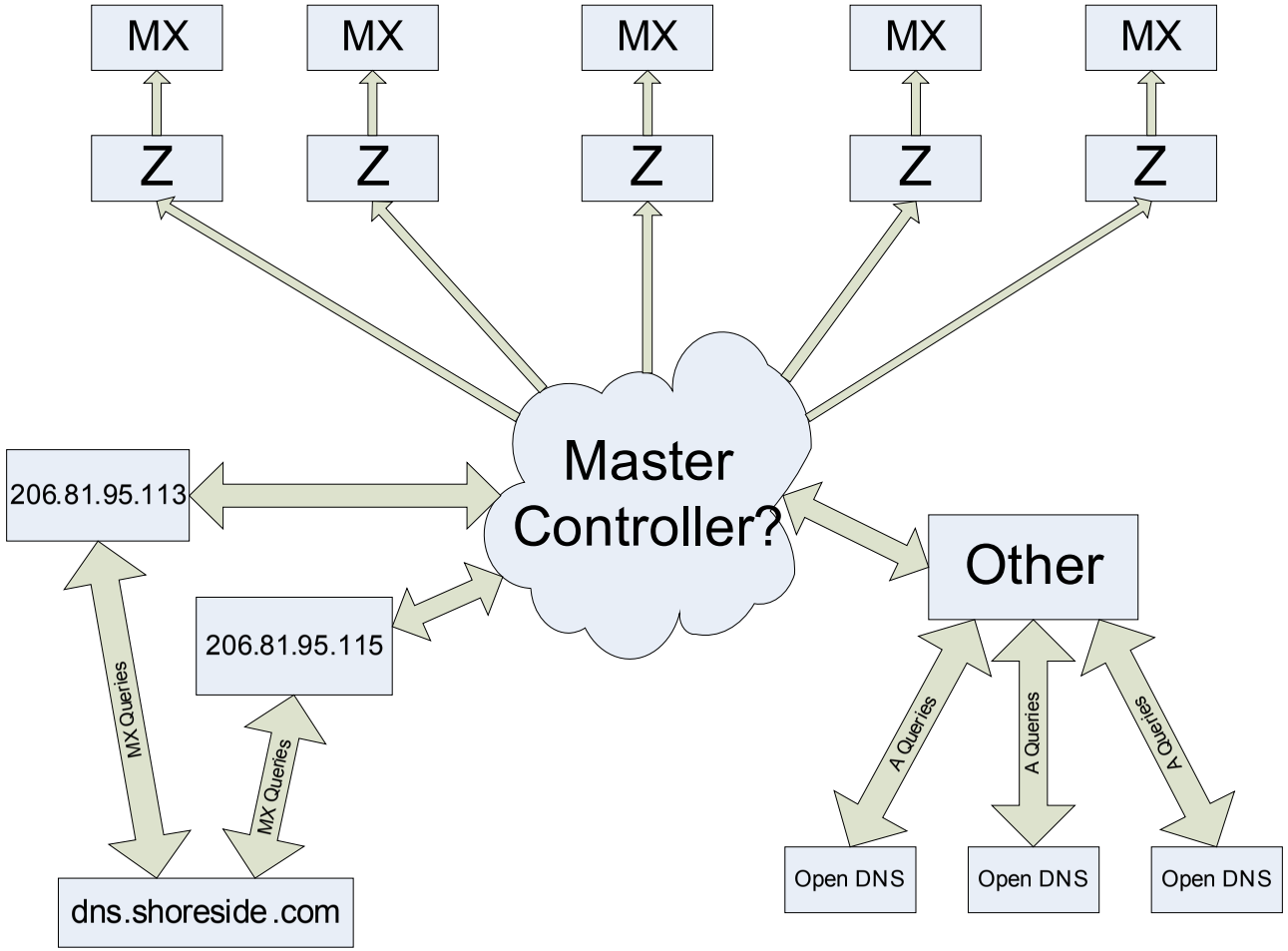
Some more questions arise

- Who was doing the DNS A queries?
  - Only one or two DNS A queries seen for my MX
  - DNS A queries coming from another “Open DNS” server
  - Zombie controllers are farming out all lookups and passing info on to the zombies
- Who is Clear Tech Services?
  - Servers in Spokane, WA, USA
  - Company in Columbia, TN, USA
  - Tried contacting by phone, but not interested in talking to me

More questions (cont)

- Who is gogetdealz.com?
  - On address 219.148.62.226
  - Located in Shijiazhuang, Hebei, PRC
  - On China Telecom network
  - Multiple Spamhaus SBL listings
    - Male anatomy pills, Mortgage ‘bank’, “#1 source for reliable bullet-proof services”
  - All domains linked to ns[123].33122.biz DNS servers

Zombie Army Architecture



Ideas for spam honeypots

- Track DNS lookups, not just SMTP
- Dedicated DNS honeypots can hand out tokenized MX records to track subsequent A record lookups
- With a large IP allocation, DNS honeypot can then hand out different A records to track subsequent SMTP session
- Better tracking of how infrastructure is abused

Identifying Spam

- Identifying spam by recipients is roughly broken into three criteria:
  - Identity
  - Reputation
  - Content

How to make it harder to infect systems?

- Most malware is distributed through email
- Client SMTP port 25 is hard to secure
- Client SMTP-AUTH port 587 is fairly mature and is easier to secure
- Maturing MTA auth schemes work on domain level
- Using MTA auth implies taking responsibility for resulting abuse issues
- Let’s use these arguments to increase security

Putting this into practice

- Decide who you are:
  - Sell/provide/support direct end-user access (consumer ISP, enterprise network manager, etc.): implement the following advice
  - Do not support end-user access, or provide business/power-user access (network backbone/transit, premium ISP, data center manager, etc.): don’t implement this, but encourage/require your customers to

Controlling Email Abuse

- Step 1: Block port 25 from leaving the end-user networks
  - Block in AND out, source AND destination to prevent asymmetric routing tricks
  - Vast majority of users don't need third party mail server access
  - Those that need it have many possible solutions: SMTP-AUTH, VPN, ssh tunneling
  - Recipient mail servers increasingly block client SMTP from dynamic addresses
  - DO NOT block MSP (587 and 466), ssh, VPN, pop2, pop3, ssl pop3, imap, or ssl imap ports

Controlling Email Abuse

- Step 2: Implement SMTP-AUTH
  - Start by offering it and encouraging use
  - After a transition, require it (block port 25 from your client systems to your mail servers)
  - SMTP-AUTH makes it easier to identify affected account than IP addresses
  - Will not completely stop zombies; assume they will be able to hijack credentials from the PC or be able to brute-force guess passwords

Controlling Email Abuse

- Step 3: Block internal hosts from talking to your incoming MXes (excluding authorized internal mailhosts)
  - Stops current spam distribution methods
  - Should have minimal impact
  - Just turning off relay still leaves you open to your own zombies sending to your own users
  - You can actually do this earlier if you want

Controlling Email Abuse

- Step 4: Identify, Contain, Correct Problems
  - Implement one or more of following
  - Establish per-account volume limits with either a cutoff or alert triggered
  - Track message-ids sent by each account so that complaints can be mapped back to an account easily

Controlling Email Abuse

- Step 4 (continued):
  - Send outgoing messages through spam and virus filters
    - Spam filters aren't completely reliable, so you can't block on this, but large amounts of spam through an account should trigger an investigation
    - Virus filters are reliable, so these messages MUST be blocked AND investigated
  - Require subscribers to use only their authorized address(es) through your servers (not necessary but very effective)

Yes! This is hard!

- But it must be done – Email abuse exceeds 90% in places and is growing
- Pain will diminish over time
- ‘Promiscuous’ networks will attract abuse
- Not doing it makes your reputation suffer
- You are not only the victim, you are also the abuser – Take Responsibility!

No! It isn’t perfect!

- Implementing these measures should decrease abuse by at least an order of magnitude
- Makes it too hard for abuse to be effective
- Yes, there are many other infection vectors; They can be addressed separately
- Don’t ignore the pretty good in your search for the perfect

Conclusions

- Careful monitoring of suspicious DNS patterns can reveal abuse
- Disinformation can reveal inner workings
- It is possible to track things to the source
- Honeypots can use this to find more detail
- ISPs can strengthen security and policy enforcement to make it more difficult to assemble Zombie Armies

Thank You!

Contact me at:
jlick@jameslick.com

More detail on controlling email abuse:
http://www.livejournal.com/users/jlick/10243.html