Evil networks: BotNets

HOW TO GET REALLY RICH USING JUST COMPUTERS

(side effect: how to build secure and resilient P2P applications)

Bart Polot

Technische Universität München

May 27, 2014

Keywords

- BotNet
- C&C
- Resilience
- FastFlux
- Randomization

BotNet?

Infected Computer

BotNet?

Infected Computer x A LOT

BotNet?

Infected Computer x A LOT

+ Anonymous Botmaster

BotNet?

Infected Computer x A LOT

+ Anonymous Botmaster

=

BotNet?

Infected Computer x A LOT

+ Anonymous Botmaster

=

FUN

BotNet?

Infected Computer x A LOT

+ Anonymous Botmaster

=

~~FUN~~

BotNet?

Infected Computer x A LOT

+ Anonymous Botmaster

=

TROUBLE

BotNet?

Infected Computer x A LOT

+ Anonymous Botmaster

=

TROUBLE

(seriously, don't try this at home)

BotNet?

- SPAM
- DDoS
- ID Theft
- IP Theft
- Theft
- Phishing
- Scareware
- Virus distribution
- Anonymous VPN

BotNet?

- Money
- Money
- Money
- Money
- Money
- Money
- Money
- Money
- Money

BotNet?

- Requirements
  - Availability: ready for business
  - Stealth: don't show up on the radar
  - Anonymity: jail bad place to enjoy money
  - Authentication: private botnet
  - Size estimation: marketing counts
  - Confidentiality, Latency, Ease of use...

BotNet?

- Requirements
  - **Availability: ready for business**
  - **Stealth: don't show up on the radar**
  - Anonymymity: onion routing
  - Authentication: asymmetric crypto
  - Size estimation: timestamp algorithm
  - Confidentiality, Latency, Ease of use...

Regular Activity: Web, etc

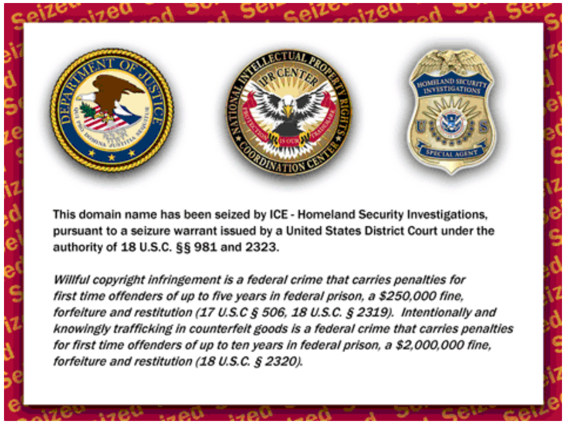
- Attacker
  - DDoS

- Defense
  - CDN

Akamai

Forbidden Activity: SPAM, etc

- Attacker
  - DDoS
  - Law
  - Experts
  - A / V
  - ISP



This domain name has been seized by ICE - Homeland Security Investigations, pursuant to a seizure warrant issued by a United States District Court under the authority of 18 U.S.C. §§ 981 and 2323.

Willful copyright infringement is a federal crime that carries penalties for first time offenders of up to five years in federal prison, a $250,000 fine, forfeiture and restitution (17 U.S.C § 506, 18 U.S.C. § 2319). Intentionally and knowingly trafficking in counterfeit goods is a federal crime that carries penalties for first time offenders of up to ten years in federal prison, a $2,000,000 fine, forfeiture and restitution (18 U.S.C. § 2320).

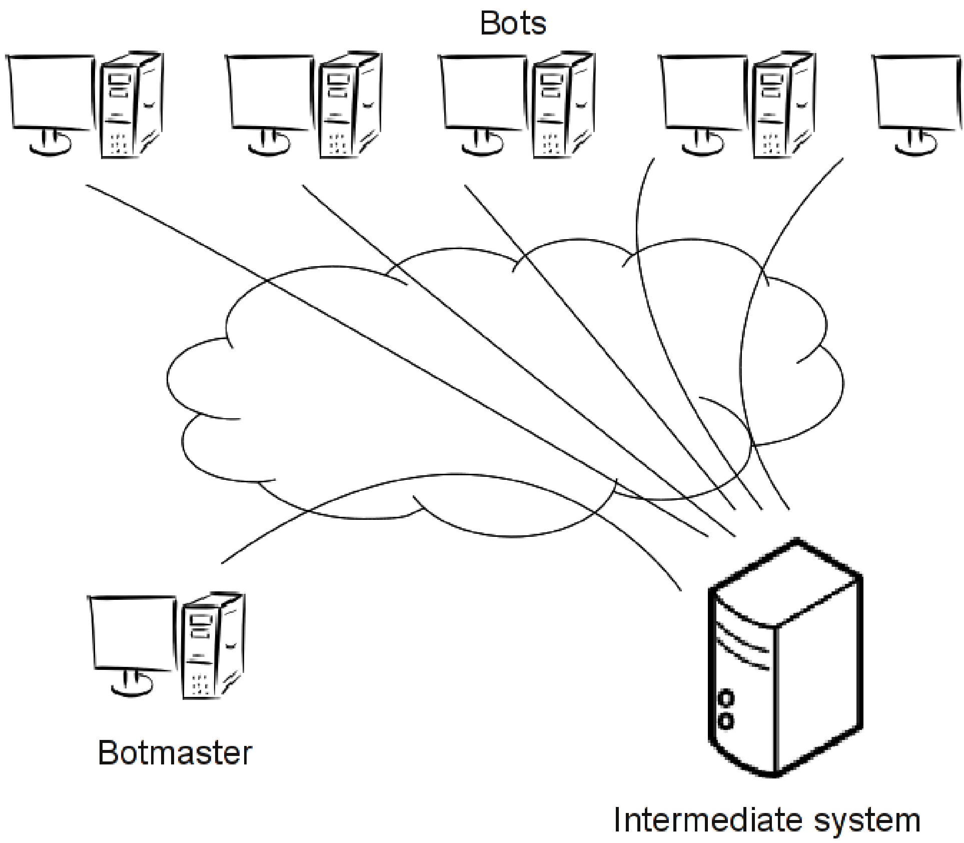
- Defense
  - ???

Pre - History

- Remote control of individual PC
  - NetBus
  - BackOriffice2000
  - Novelty / Spyware

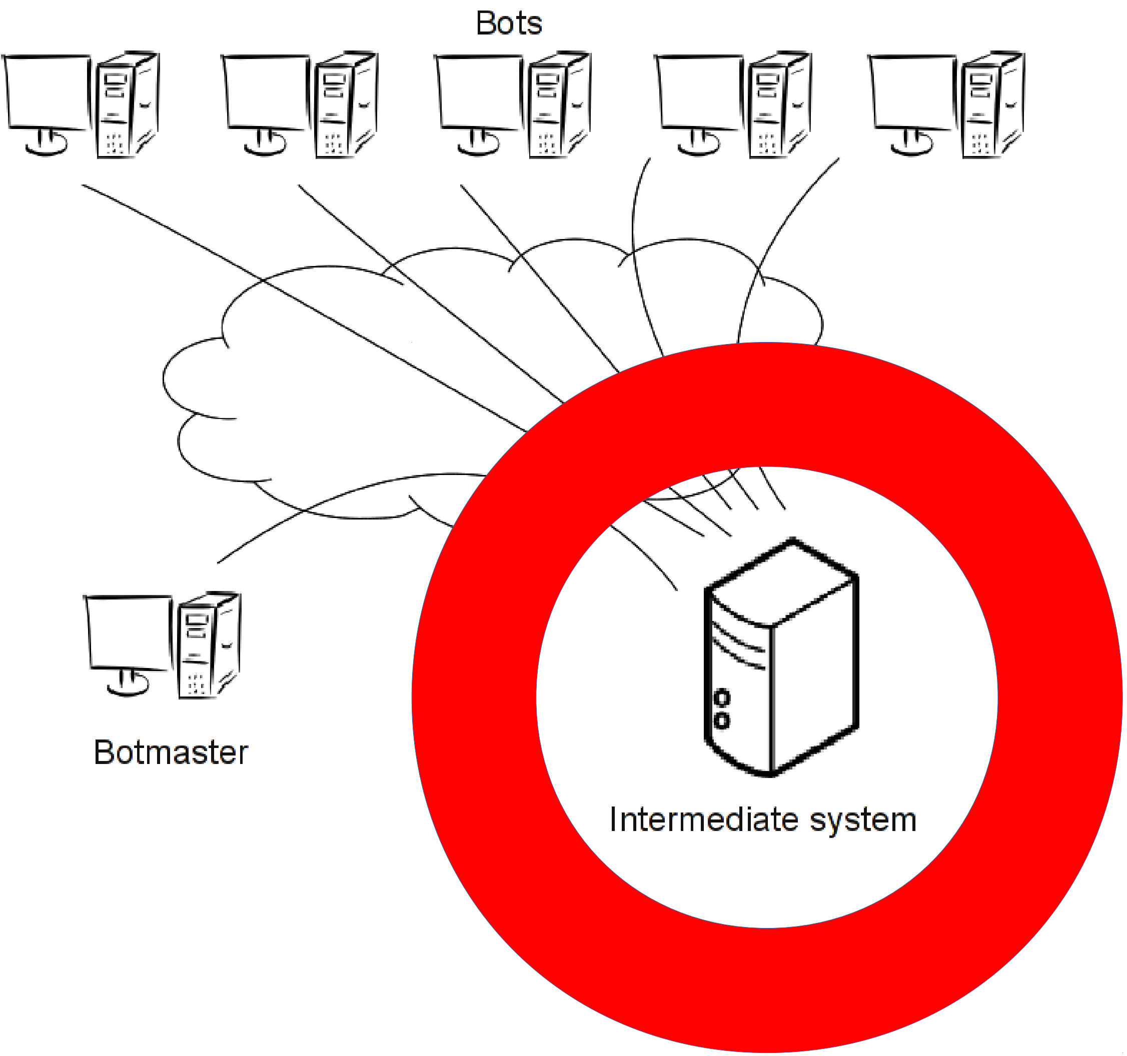
Ancient History

- Centralized server
  - Hacked server
  - Botmaster owned



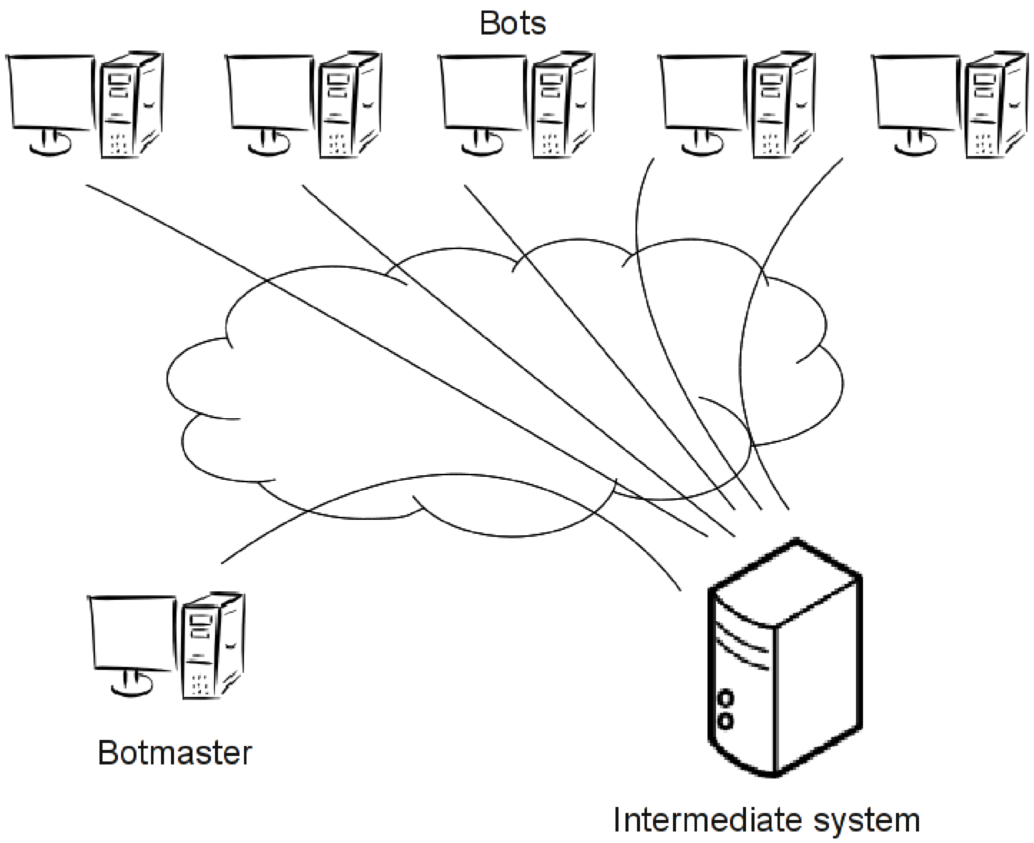
Ancient History

- Centralized server
  - Hacked server
  - Botmaster owned
- Easy to attack
  - Clean server
  - Disconnect server
- Trivial to implement



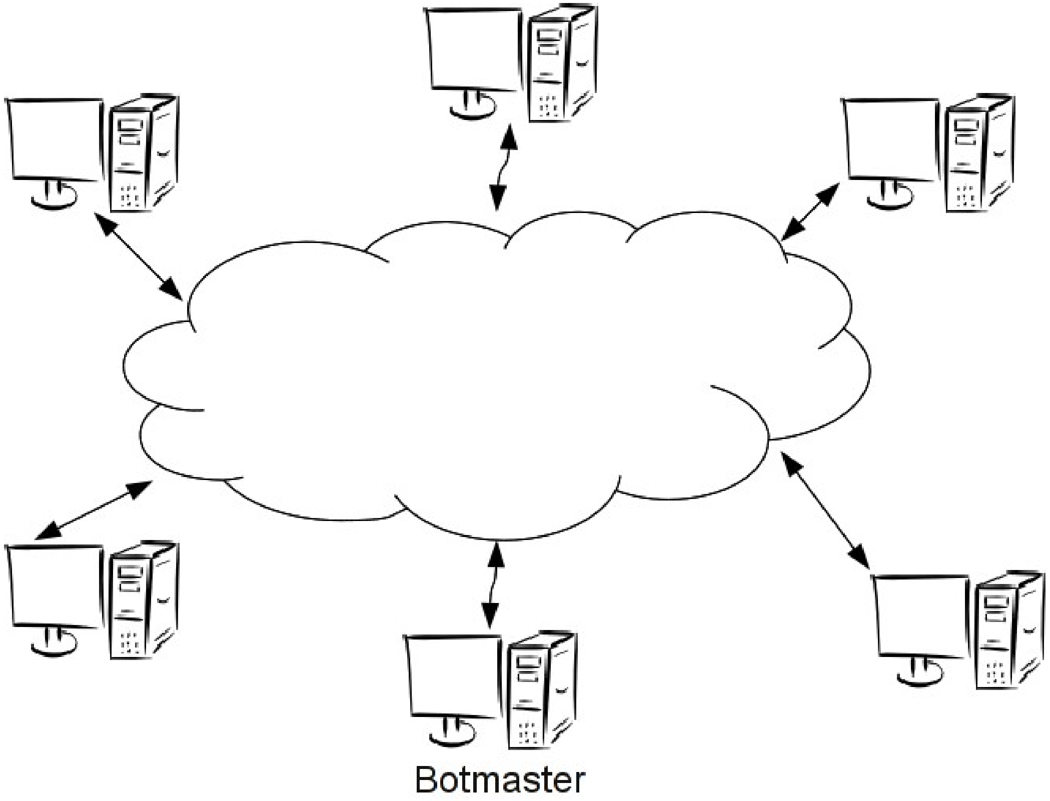
Ancient History

- IRC server
  - IRC resilience
  - Password
  - Botmaster via Tor
- Easy to attack
  - Clean server
  - Disconnect server
- Easy to implement



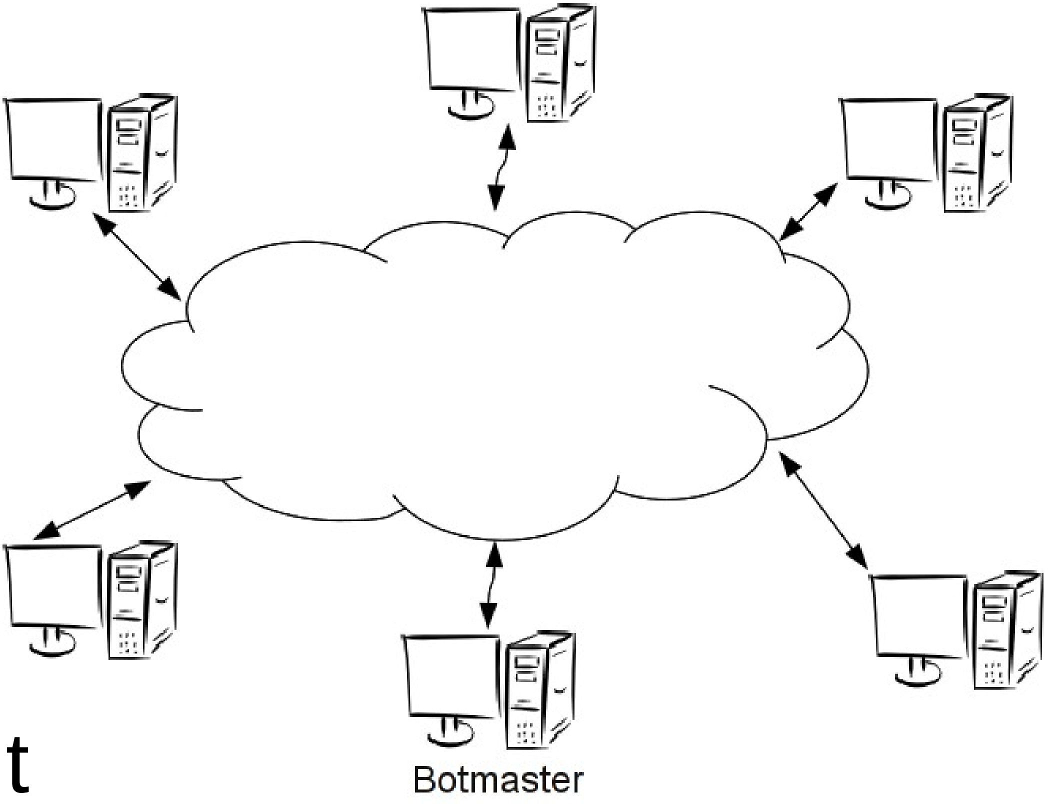
Modern History

- P2P networks
  - P2P resilience
  - Botmaster peer
- Harder to attack
  - No server
  - Exploit bot software



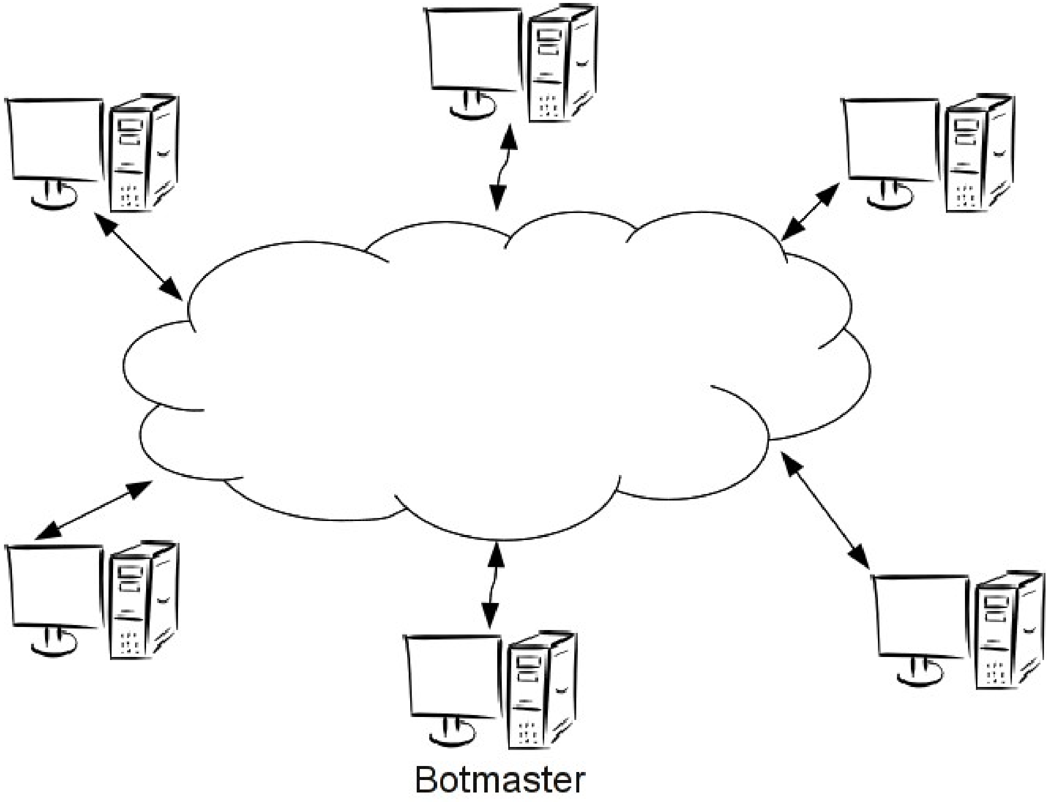
Modern History

- P2P networks
  - P2P resilience
  - Botmaster peer
- Harder to attack
  - No server
  - Exploit bot software
- Difficult to implement



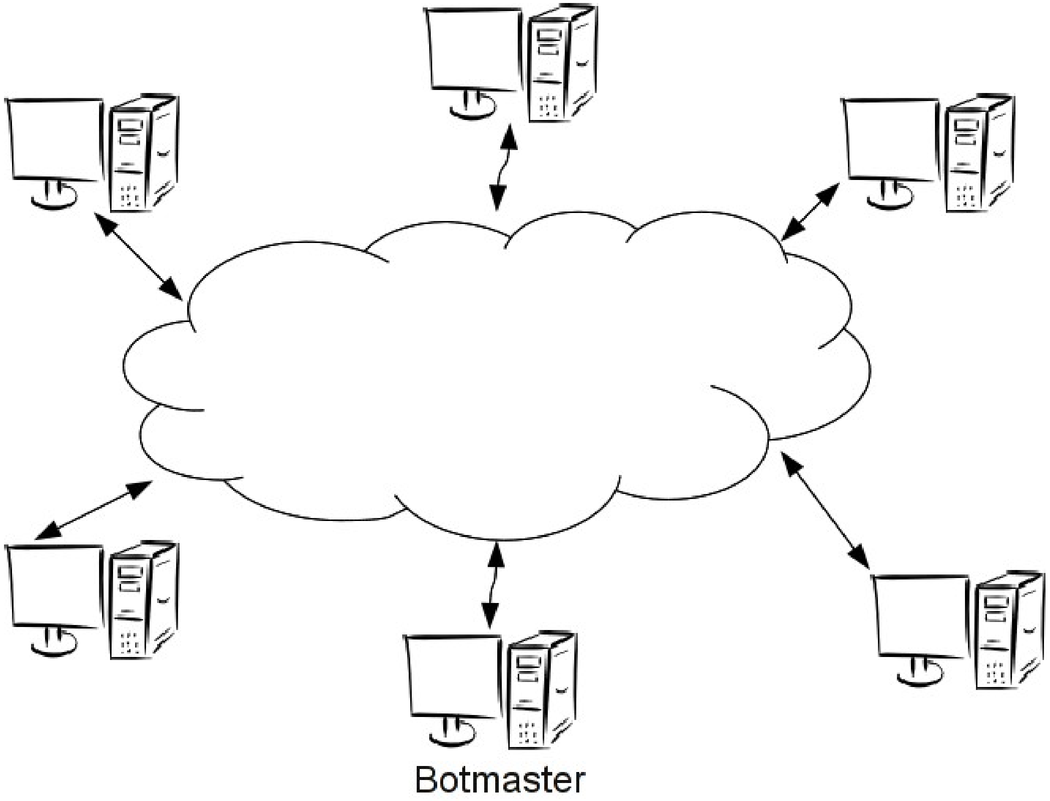
Modern History

- Storm Worm
  - Jan 2007
  - P2P C&C
  - Up to 50 million
  - Computing power
    - Top 500
  - Bandwidth
    - Country
  - Revengeful



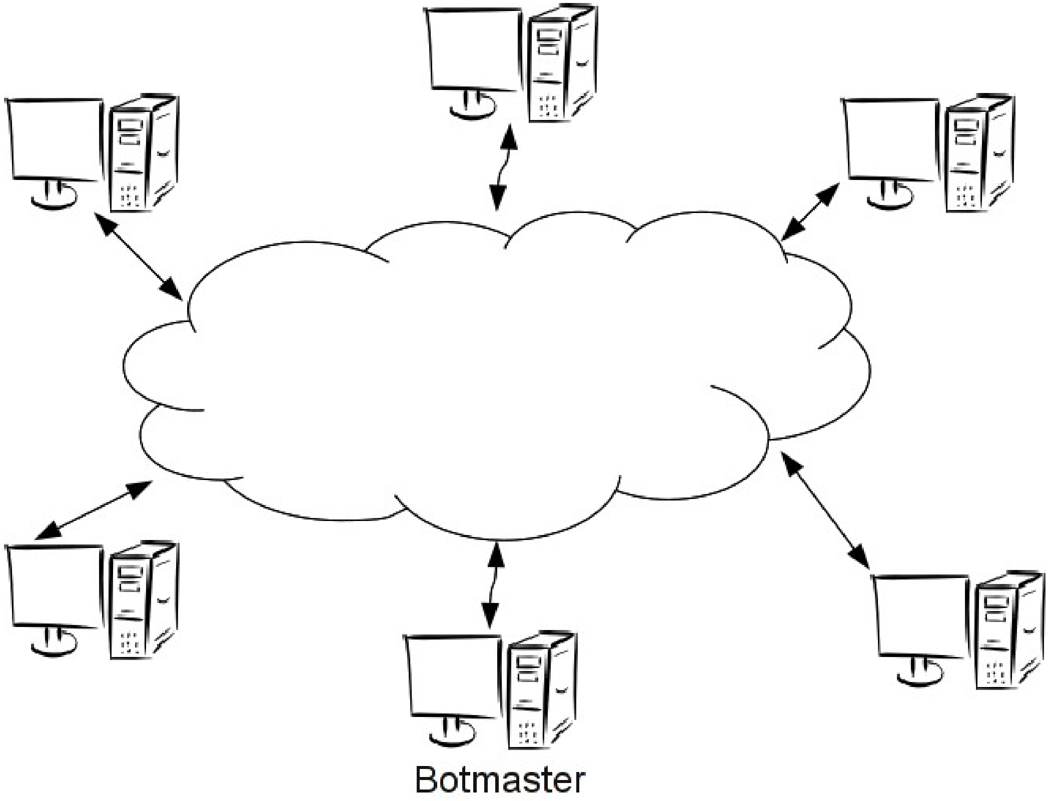
Modern History

- Storm Worm
  - Overnet
    - Kademlia
  - Cell structure
    - Hide size
  - Fast Flux



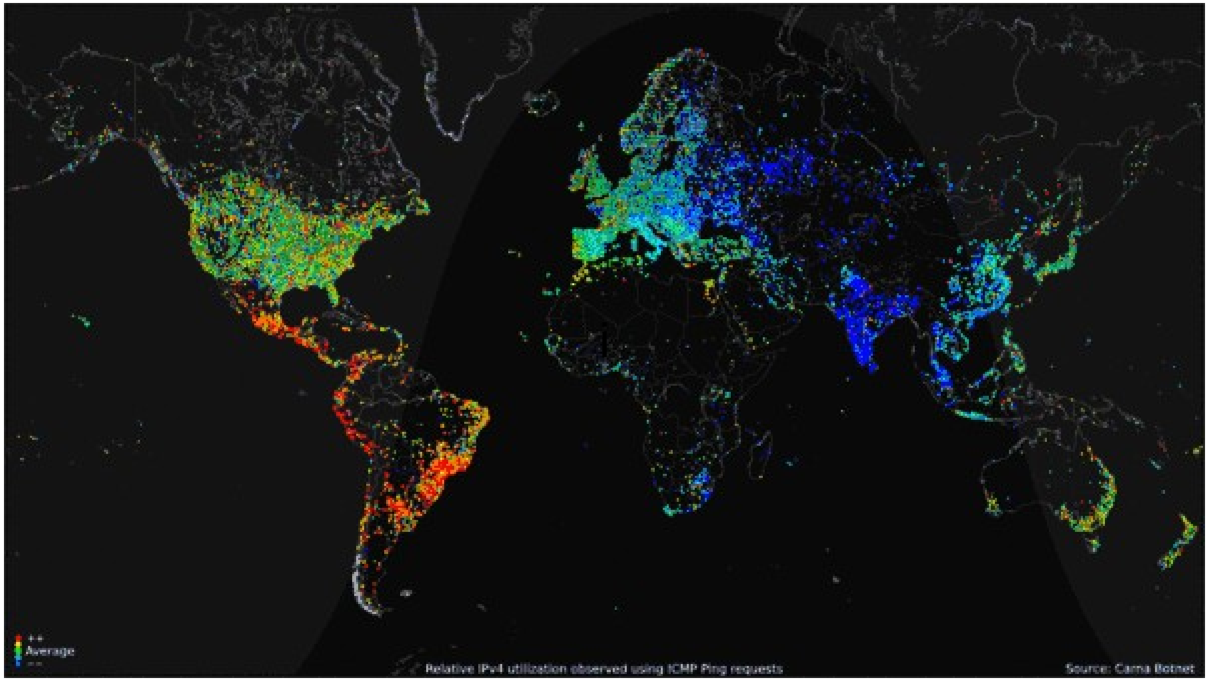
Modern History

- Storm Worm
  - Stormfucker
  - Poor crypto
  - No authentication
    - 4 byte XOR
    - 64 bit RSA



Modern History

- Carna Botnet
  - Routers
  - Default credentials
  - Internet Census 2012
  - Polite Botnet



Relative IPv4 utilization observed using ICMP Ping requests

Source: Carna Botnet

First vulnerability: Content Server

- Content Server is taken down: SPAM is useless
- Hide Content Server
- Use bots as proxies

DNS Round Robin

- Anatomy of a DNS request: google.com
  - Get NS . (root level) → 13 root servers
  - Get NS com. → 13 “.com” servers
  - Get NS google.com → Google's DNS server
  - Get A google.com → Google WEB server

DNS Round Robin

- Anatomy of a DNS request: google.com
  - . 14922 IN NS a.root-servers.net.
  - com. 172800 IN NS a.gtld-servers.net.
  - google.com. 172800 IN NS ns2.google.com.
  - google.com. 300 IN A 173.194.44.4

DNS Round Robin

- Return a list multiple results
- Each query return a different list

DNS Round Robin

- Load Distribution
- Avoid dead machines
- Simple and effective
- Not perfect: Distributon vs Balancig
- CLI Example (run twice)
  - $ dig google.com +trace

DNS Round Robin

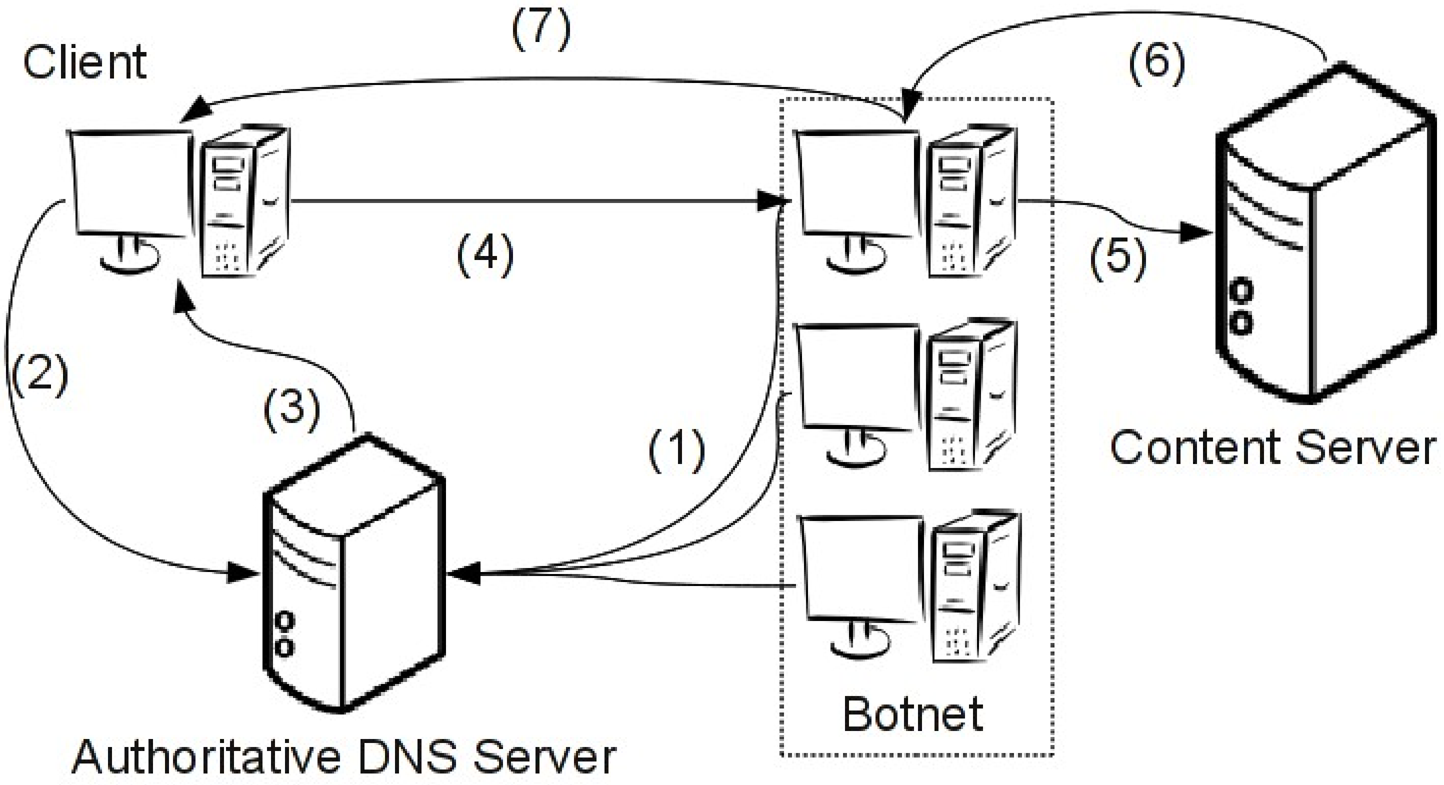
- Example: google.com
-
  ```
  google.com.          300    IN    A    173.194.44.41
  google.com.          300    IN    A    173.194.44.36
  google.com.          300    IN    A    173.194.44.37
  google.com.          300    IN    A    173.194.44.33
  […]
  ```
-
  ```
  google.com.          300    IN    A    173.194.44.33
  google.com.          300    IN    A    173.194.44.39
  google.com.          300    IN    A    173.194.44.40
  google.com.          300    IN    A    173.194.44.41
  [...]
  ```

Fast Flux

- Very aggressive Round Robin
- Thumb rule: TTL < 300 s
  - High load domains conflict with this
    - Yahoo: TTL 1800
    - Facebook: TTL 900
    - Google: TTL 300
    - Amazon: TTL 60 (!)
- Updated by destinations themselves

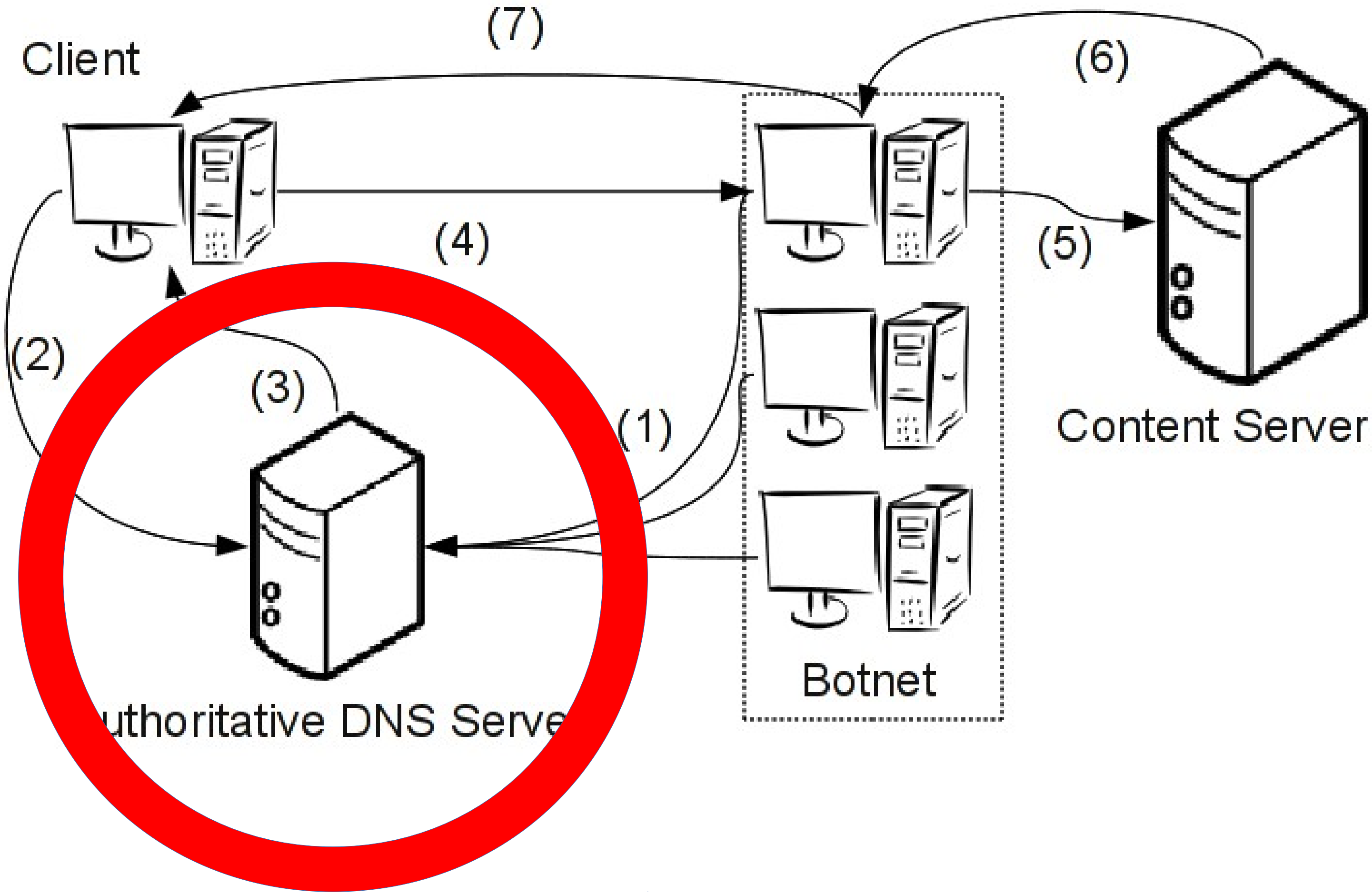
Fast Flux

1) Registration
2) Query
3) Response
4) Request
5) Forward
6) Content
7) Forward



Fast Flux

Single failure point: DNS server

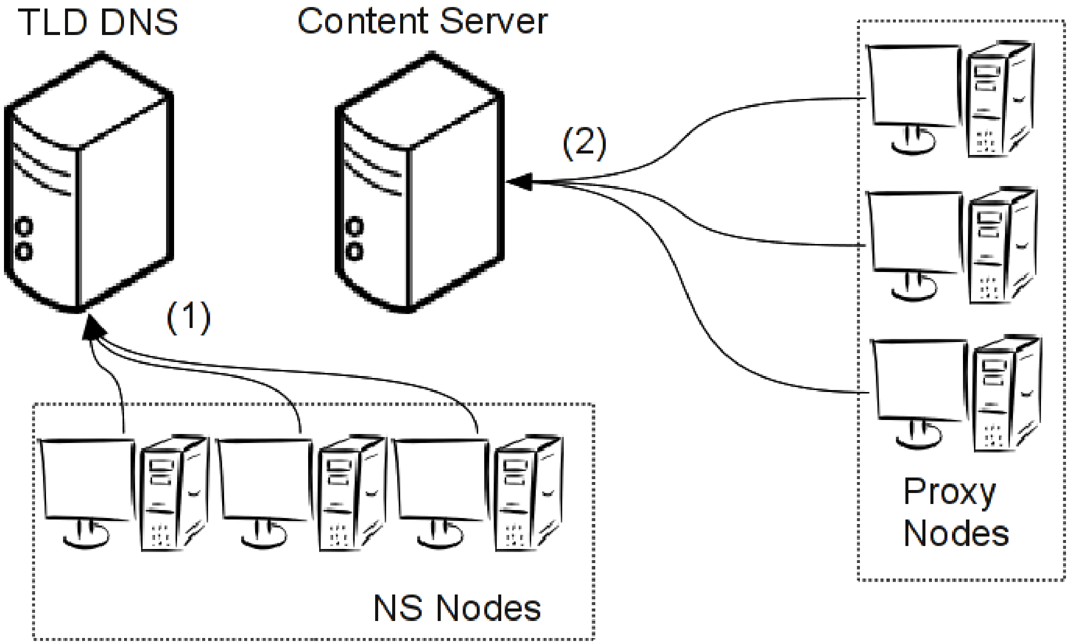


Double Fast Flux

- Fast Flux: Single A result → Multiple A (proxies)
- Double FF: Signle NS result → Multiple NS
- Do Fast Flux on both A and NS records
  - Different sets of nodes (specialized)

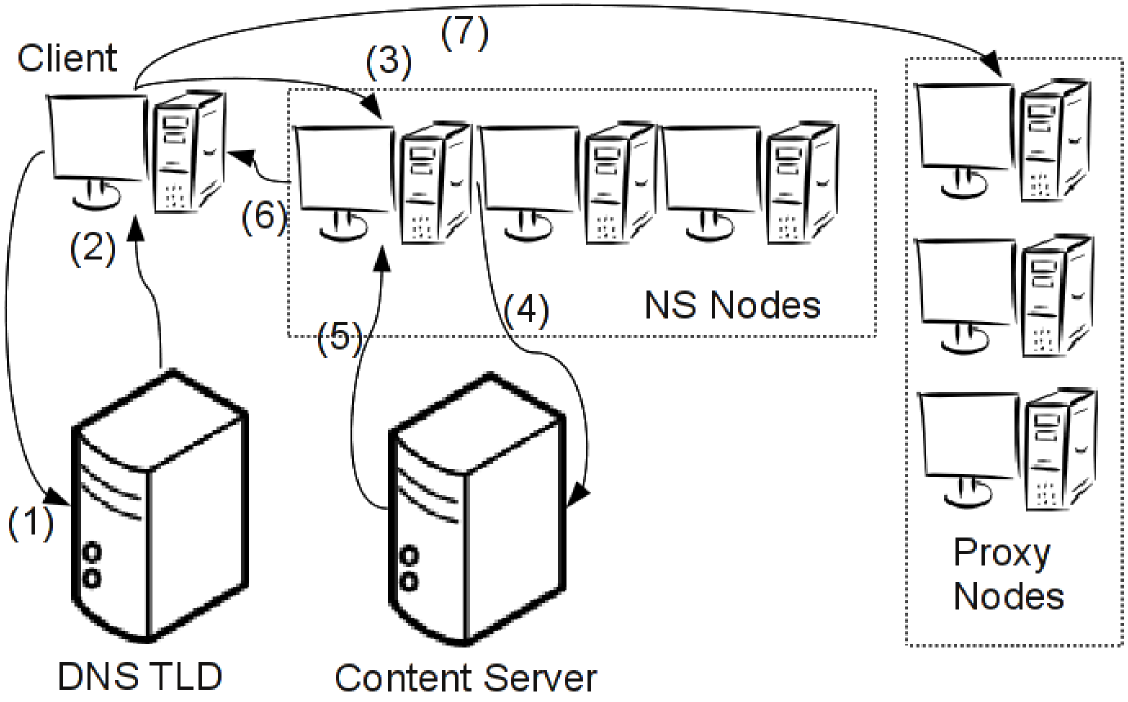
Double Fast Flux

- Stage 1: Registration
  - NS nodes to TLD
  - Proxy Nodes to NS content server



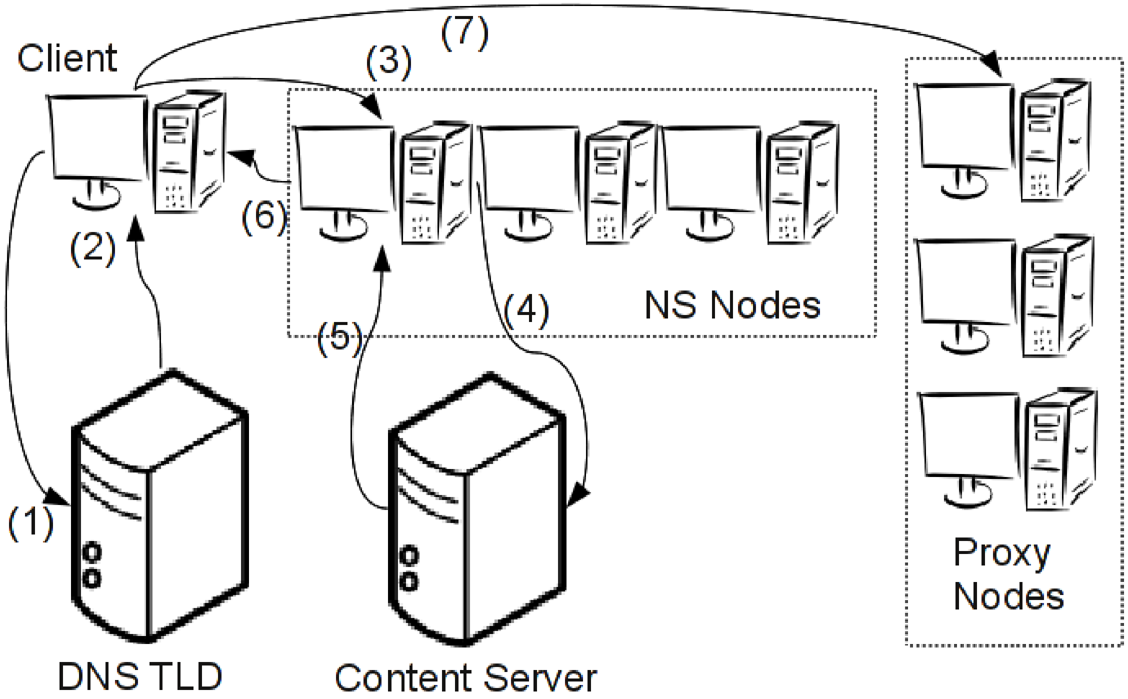
Double Fast Flux

- Stage 2: Operation
  1) Get NS for domain
  2) Reply: NS proxy
  3) Get A for domain
  4) Forward
  5) Reply A
  6) Forward
  7) HTTP request



Double Fast Flux

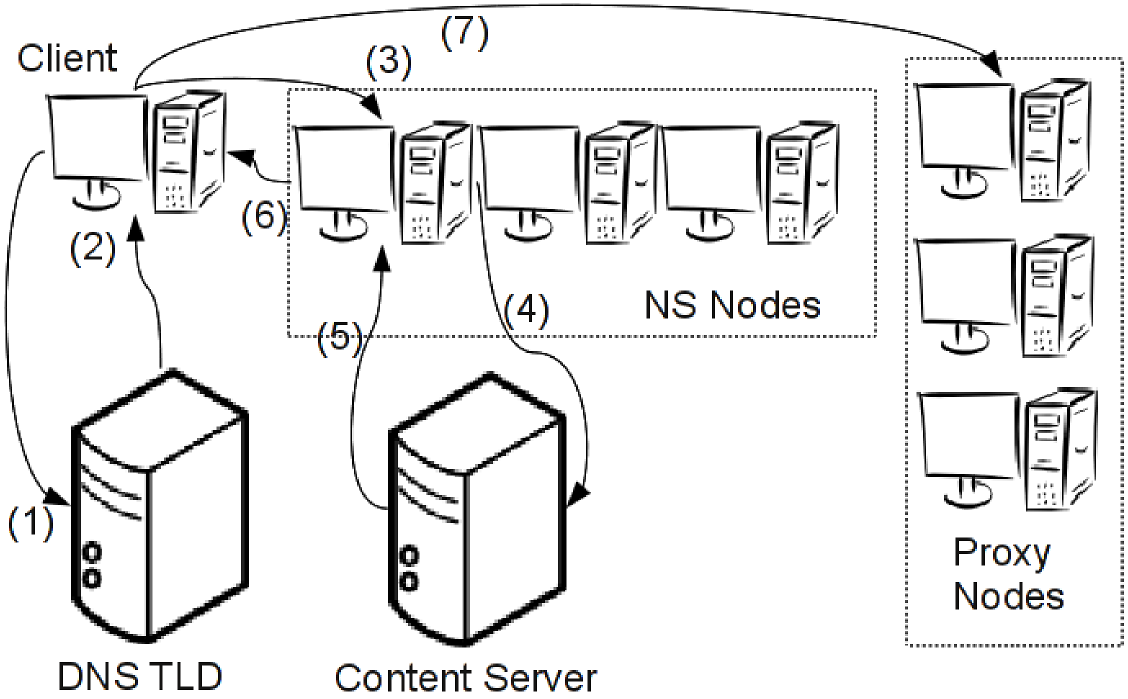
- Perfect? No single point of failure?



Double Fast Flux

- Perfect? No single point of failure?

  evildomain.com



Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains
  - Different every day
  - Download signed content
  - Affects up to 15 million Microsoft SERVER systems

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains
  - Different every day
  - Download signed content
  - Affects up to 15 million Microsoft SERVER systems
    - French Navy and Airforce
    - UK Ministry Defence (submarines, warships)
    - Bundeswehr
    - Police, Hospitals

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains
  - Different every day
  - Download signed content
- Response
  - Dec 16, 2008 Patch from Microsoft
  - Feb 12, 2009 “Conficker Cabal”
  - Feb 13, 2009 Microsoft offers 250.000 USD

Domain Name Randomization

- Conficker Cabal
  - ICANN
  - Microsoft
  - Verisign
  - Symantec
  - F-Secure

Domain Name Randomization

- Conficker Cabal
  - ICANN
  - Microsoft
  - Verisign
  - Symantec
  - F-Secure
  - Afilias, Neustar, China Internet Network Information Center, Public Internet Registry, Global Domains International, M1D Global, America Online, ISC, Georgia Tech, The Shadowserver Foundation, Arbor Networks, Support Intelligence

Domain Name Randomization

- Conficker Cabal
  - Pre-register all Conficker A domains
  - Starts in March, 2009
  - Finishes by mid-April, 2009

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains
- Response
  - Dec 16, 2008 Patch from Microsoft
  - Feb 12, 2009 “Conficker Cabal”
  - Feb 13, 2009 Microsoft offers 250.000 USD

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains
- Response
  - Dec 16, 2008 Patch from Microsoft
  - Feb 12, 2009 “Conficker Cabal”
  - Feb 13, 2009 Microsoft offers 250.000 USD
  - Feb 20, 2009 Conficker C

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains each day
- Conficker C: Feb 2009
  - Pseudorandom list of 50.000 domains each day

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains each day
- Conficker C: Feb 2009
  - Pseudorandom list of 50.000 domains each day
  - Try to connect to 500 of them
    - Success chance: ~1%
  - Distribute payload via P2P

Domain Name Randomization

- Conficker A: Nov 2008
  - Pseudorandom list of 250 domains each day
- Conficker C: Feb 2009
  - Pseudorandom list of 50.000 domains each day
  - Try to connect to 500 of them
    - Success chance: ~1%
  - Distribute payload via P2P
  - Game over

That pesky DNS

- DNS is controlled by authorities
  - Registration can be risky / expensive

That pesky DNS

- DNS is controlled by authorities
  - Registration can be risky / expensive
- Solution: no DNS!

That pesky DNS

- DNS is controlled by authorities
  - Registration can be risky / expensive
- Solution: no DNS!
- Zer0n3t
  - Use TOR hidden service to host C&C

That pesky DNS

- DNS is controlled by authorities
  - Registration can be risky / expensive
- Solution: no DNS!
- Zer0n3t
  - Use TOR hidden service to host C&C
  - Back to IRC!

Stealth communication

- Twitter / Facebook
  - Base64 encoded bit.ly pastebin hosted CMD
  - Koobface: spread via Social Networks
- HTTPS
  - Traffic on unknown ports: suspicious
  - Cleartext on know port: easy fingerprinting
  - Encrypted traffic on known ports: suspicious to DPI
  - Encrypted traffic on port 443: bingo!

Stealth communication

- Jabber/XMPP
  - For users: Modern and flexible IRC replacement
  - For botnets: Modern and flexible IRC replacement
  - More complicated account creation
- DNS
  - Morto, Feeder
  - TXT requests
  - Base64 → bit.ly → pastebin → zip → exe, dll

Other Features

- Rootkit
  - Bot is module of OS
- Bootkit
  - OS is module of Bot
- Integrated Antivirus
  - Less competition, less attention
- GPL license violation

Other Attacks

- White Hat Botmaster
  - Exploit vulnerabilities in Bot code
  - Exploit vulnerabilities in BotNet design
  - Send autodestruction commands
  - Ethical and legal concerns
  - Defense: Learning to program.

Other Attacks

- Sinkholing
  - Sybil attack: Impersonate control nodes
  - Isolate and disconnect nodes
  - Sybils must be responsive to avoid bootstrapping
  - Defense
    - Reputation systems
    - Smart FF re-bootstrap

Other Attacks

- Enumerate and block
  - Add bots to spam blacklist
  - Defense: Brute force (have millions of bots)
- Spamming
  - Insert bogus data (theft botnet)
  - Defense: ??

Other Attacks

- Size estimation
  - Crawl P2P network: recursive queries of peer lists
    - Inefective (sometimes as low as 2% discovered)
  - Emulate protocol and join
  - Defense: clustering

The Perfect Botnet

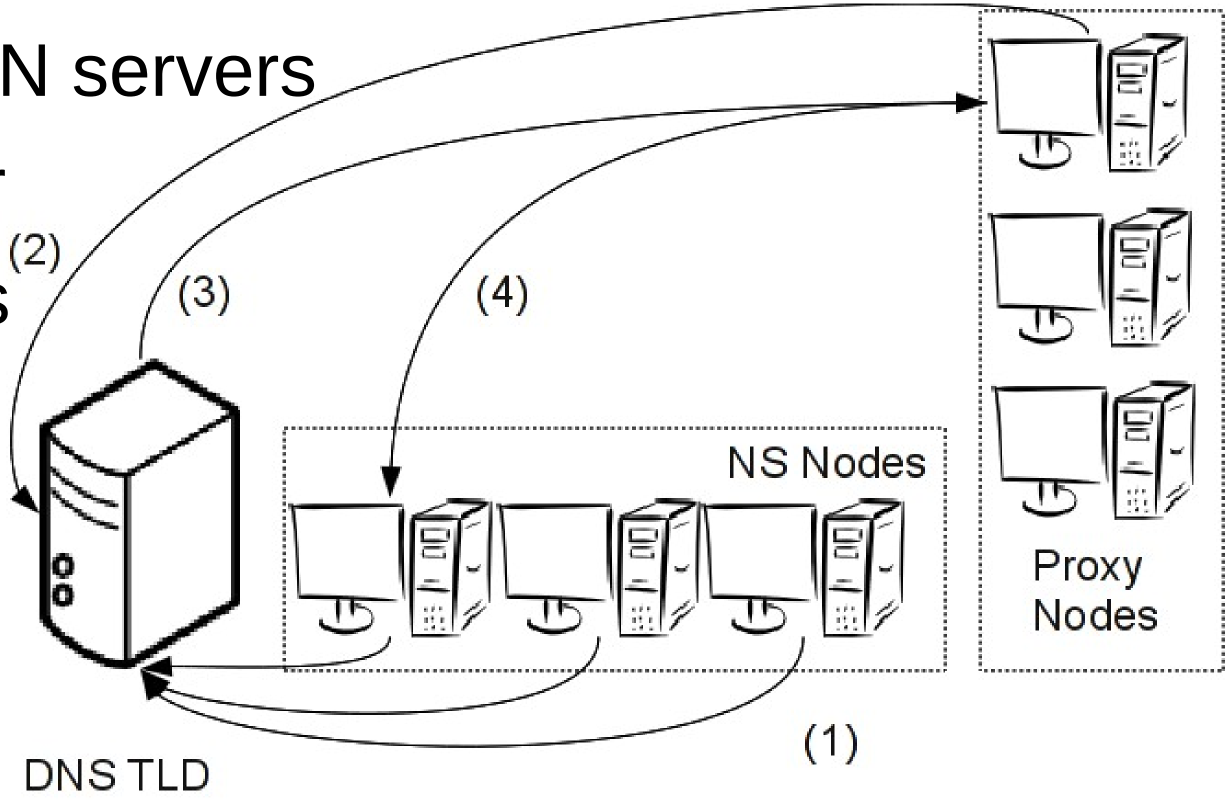
- No C&C: pure P2P based
  - No special nodes
- Domain name randomization
  - Instead of time, based on random but public data
    - Weather
    - Stock Market
  - Use Fast Flux for bootstrap
- Sign (and verify!) commands with proper crypto

The Perfect Botnet

- Use port 22, 443 for communication
  - Use <u>proper</u> crypto!
- Extra restricted situation: DNS
  - 8 'A' responses: 256 bits → DHT key
    - Google uses 11 'A' respones
    - Avoid invalid IP (127. - 10. - 172.16. - 5. - 224.)

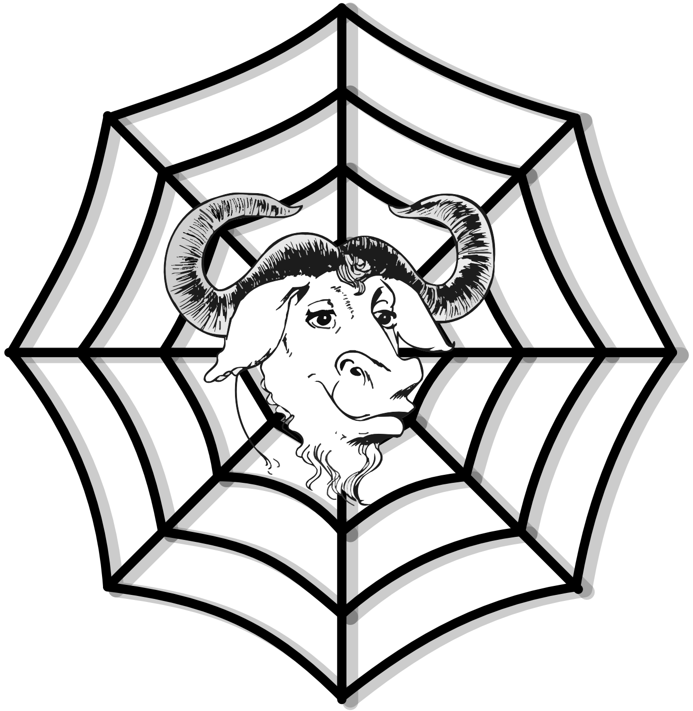
The Perfect Botnet

- Improve Fast Flux
  - NS proxies → DNS servers
  - 'A' proxies → CDN servers
  - 'A' nodes register with DNS servers



The Perfect Botnet

- Too much work? Find a framework!



The End

DON'T TRY THIS AT HOME

(IF YOU DO TRY, I DEMAND MY SHARE)

Questions?