Red Team Botnet: Creating and Maintaining Successful Minions

Streeterville Group
M. Aghajanian, M. Blackburn, T. Heller

Botnet Offensive Response

Offense	Compromise Hosts	Improve Survivability of Bots
Defense	• Block malware web sites. • Look for suspicious emails, keywords, attachments. • Keep OS and network software updated. • Monitor network for port probing.	• AV software updated and monitored. • Look for enumeration signatures. • Look for unnecessary files or registry entries. • Block suspicious processes. • Attempt detection of keyloggers (overloading “keypress” event throughout OS). • Warn user of suspicious system library calls.

Botnet Offensive Response (cont’)

Offense	Improve Quiet and Secret Communication	Improve the Survivability of C&C and Botmaster
Defense	• Watch localized and global traffic for abnormal, signature, and “loud” patterns. • Inspect packets for abnormal or signature payloads.	• Disrupt command and control. • Find and destroy C&C servers. • Find origins of the hackers.

Compromising Hosts

Strategy: Balanced Attack

Email

- Send emails exploiting human vanity and greed.
- Success of this technique helps profile users as *non-savvy* and gullible, therefore more-susceptible to future attacks.
- Respective emails can be archived for future use.
- Emails can either include a file or have a link to a well designed website which offers free and desirable "software" installation files.

Strategy: Balanced Attack (cont')

Target Discovery

- Once a host has been compromised, scanning can employed.
- Depending on network profile, a variety of scanning techniques can be used to maximize bot success and survivability.
- Topological scanning employed if network is more up-to-date and safeguarded.
- More aggressive host-scanning used otherwise.
- Vector/worm can employ a "back-off" strategy causing the botnet to grow at a pace which will cause less commotion.
- Network can be profiled by host so that attack reiterations are more quiet and successful.

Improving Bot Survivability

Functionality Through Commands:

- Can change the signature of the BAHbot software.
- Allows robust controls and defensive/offensive measures can be initiated dynamically.

Dynamic Storage Decisions:

- Store local data to maximize secrecy.
- Store options will be chosen from a list based on system profiling and the current defensive software

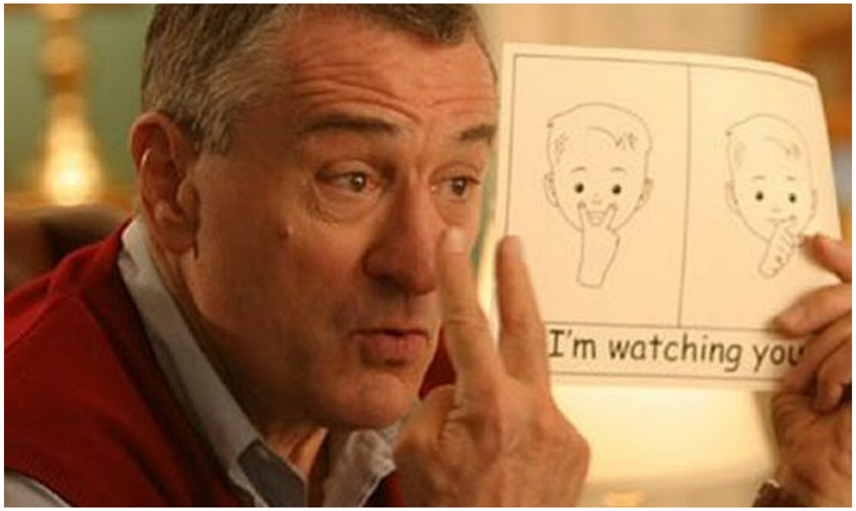
Anti-detection Techniques:

- Avoiding suspicious server nicknames.
- Throttling network activity to avoid detection.

Improving Quiet and Secret Communication

Goal:

How do we minimize visibility and maximize secrecy when communicating?



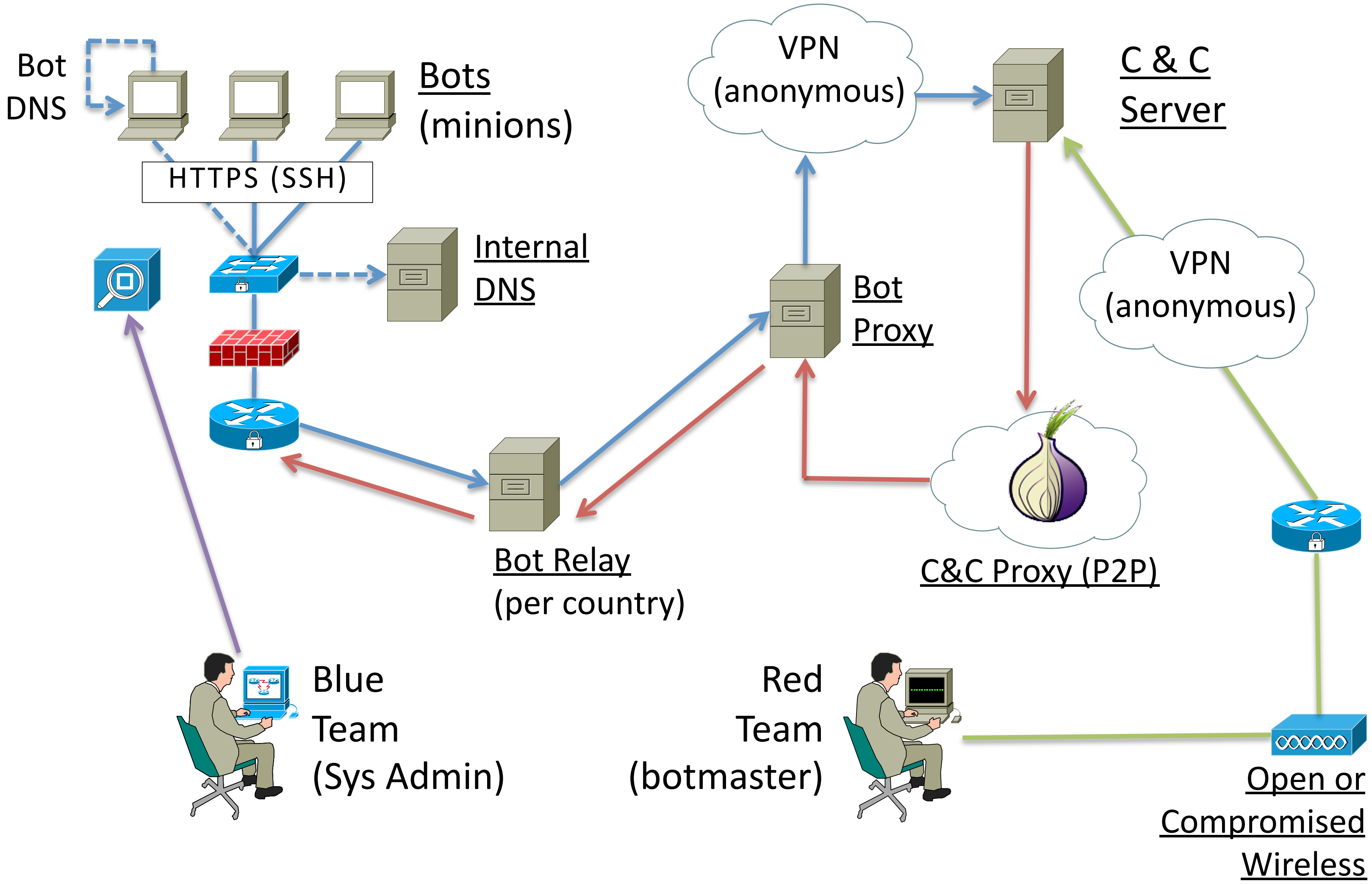
“I’m watching you!”

Network Communication Mappings

Source Function	Destination Port *	Protocol/ Technology *	Destination Function
Bots	HTTPS (443)	SSH	Bot relay → Bot proxy
Bot proxy	HTTPS (443)	VPN/SSH	C & C
C & C	HTTPS (443)	VPN/SSH	C & C proxy
C & C proxy	HTTPS (443)	SSH	Bot proxy
Bot proxy	HTTPS (443)	SSH	Bot relay → Bots (minions)
Master(s)	HTTPS (443)	VPN/SSH	C & C
Bots	25, 587	SMTP	Email servers (spam)
Bot	[randomize]	DNS	[loopback]
Host	53	DNS	Internal DNS server(s)

* Use common ports to connect and encryption to deliver.

Network Flow



Network Traffic Distinction

1. The communication necessary to implement the ultimate goal (spam) is “loud” and attracts attention more than any administrative task.

2. Complete all administrative communication that increases the probability of survival before launching the ultimate goal of the botnet (spam communication).

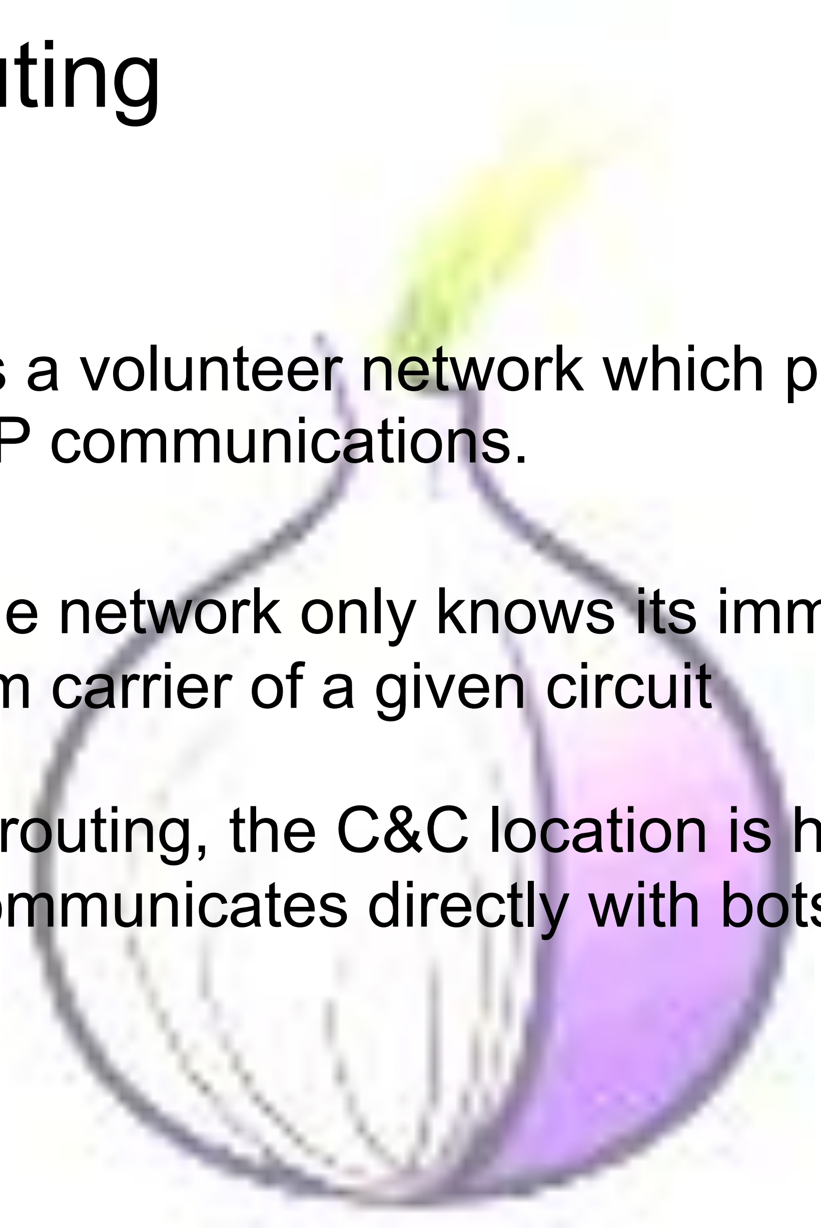
Network Footprint and Signature

Network Aspect	Response
Amount of traffic or connections.	Evaluate risks versus value.
Speed of traffic and making connections.	Throttle tasks, especially enumeration.
Size of traffic or payload.	Evaluate risks versus value.
Time of traffic or connections.	Randomize time of connections.
How the connections are made.	Use HTTPS (443), HTTP (80) to connect. Use SSH (ephemeral ports).
Type or content of traffic.	Encrypt communication; bot DNS; clean up after connections and spam.
Where or physical locations of traffic.	Use a Bot Relay per country.

Improving C & C Survivability

Or, “if you can keep your head when all around you are losing theirs…”

Onion Routing



Onion routing is a volunteer network which provides encrypted source-routed IP communications.

Each node in the network only knows its immediate predecessor and downstream carrier of a given circuit

By using onion routing, the C&C location is hidden from view, even when it communicates directly with bots.

Bot Masters VPN to the C&C

All communications to the C&C are encrypted end-to-end and use anonymous VPN or onion routing.

ButterflyUtopia.com

Mariposa botnet was only unraveled when botmaster connected to the C&C directly instead of using an anonymizing VPN.

Routing Around A Lost Proxy

If a C&C server goes down, all bots that connected to it are lost, They would need to be replaced or recompromised.

If the proxy is compromised using DNS blocking, as discussed in the paper, the C&C server can communicate a new proxy location directly to bots using the Tor network.

Disaster Recovery

What if **both** the Bot proxy and C&C are disabled?
Are all bots lost?

To allow connections to an update server, the Conficker worm used an algorithm to generate random URLs.

Algorithm was reverse-engineered, subsequent URLs could be predicted, and access to them blocked.

If this flexibility were combined with an external source,

The TWATTER©* algorithm

A random seed is needed to make the algorithm unpredictable

But how to distribute such a seed securely so that authorities cannot predict and preemptively disable the rally point?

1. Use a globally-accessible yet unpredictable key, such as the top hashtag on Twitter at 3pm.

2. Hash that key into a host name.

3. Servers can be established at that new location, where new code/configuration settings are distributed.

Smarter than the Average Bear

- Can we avoid honeypot bots rejoining?
- Only pick up key during normal user activity
- Honeypot host would “just sit there”