Leveraging Honest Users:

Stealth Command-and-Control of Botnets

Diogo Mónica
INESC-ID/IST
diogo.monica@ist.utl.pt

Summary

- Motivation
- Problem statement
- Stealth C&C using browsers
- Final remarks

Motivation

- Botnets continue to evolve
- New strategies must be employed to avoid takedown and detection
- Our objective is to explore new directions future C&C infrastructure might take

Problem Statement

- Create a botnet that:
  - Avoids infiltration, size estimation
  - Reduces the likelihood of detection of individual bots
  - Maintains Botmaster anonymity

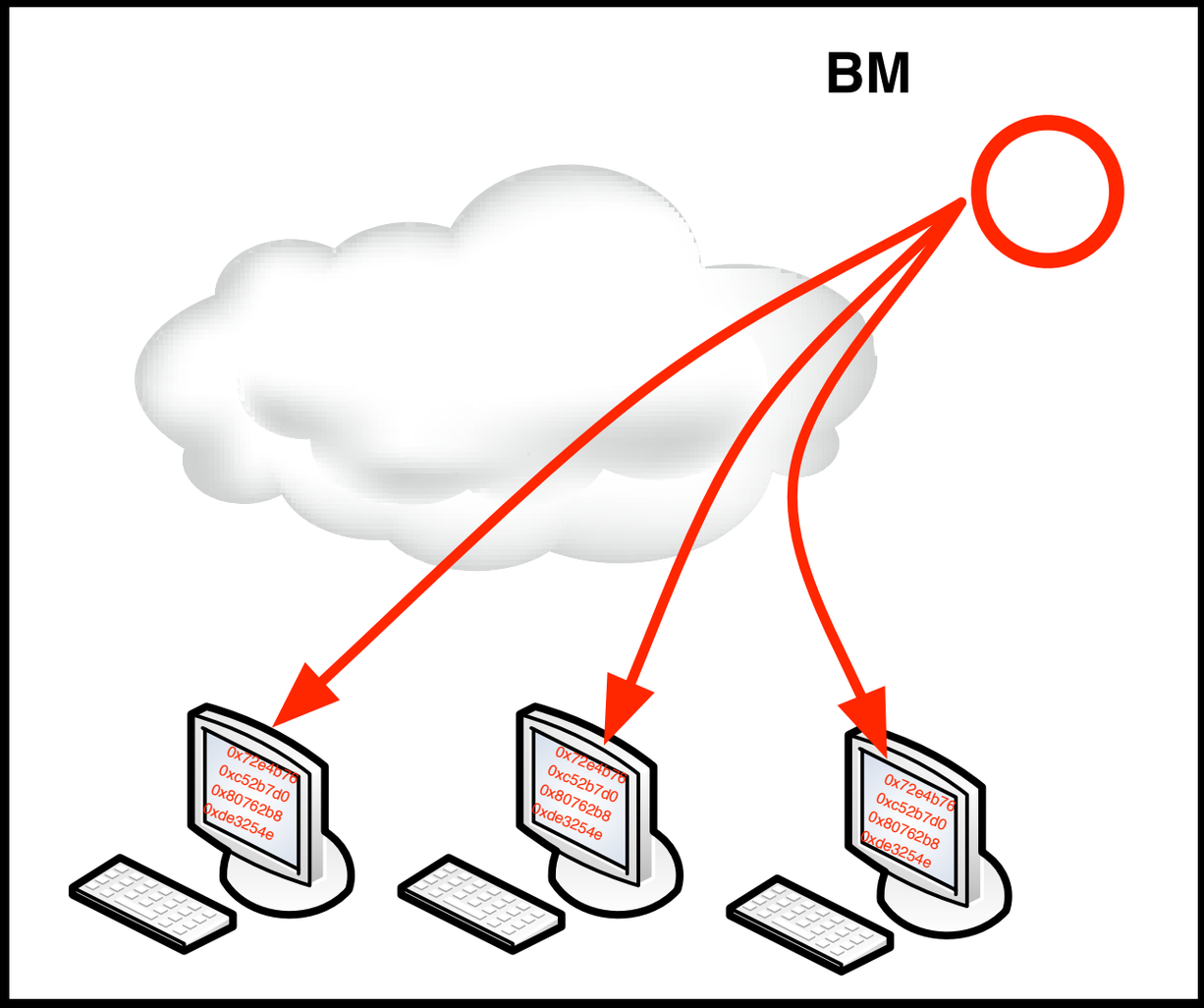
Assumptions

- Pre-existing population of infected hosts
- Trust anchor in the binary (public key)
- Bots can receive commands from bot master through some open port

Basic Architecture

- No active participation from bots in a botmaster owned C&C
- Bots passively listen for commands
- Commands are signed by the botmaster and pushed out to all the bots

Basic Architecture



Basic Architecture

- No C&C means:
  - no infiltration
  - no size estimation

Problems

- Command dissemination
  - Botmaster doesn't know IPs of bots
  - Direct dissemination exposes the botmaster
  - Disseminating commands takes too long
- Information retrieval
  - Bots don't know the IP of the botmaster

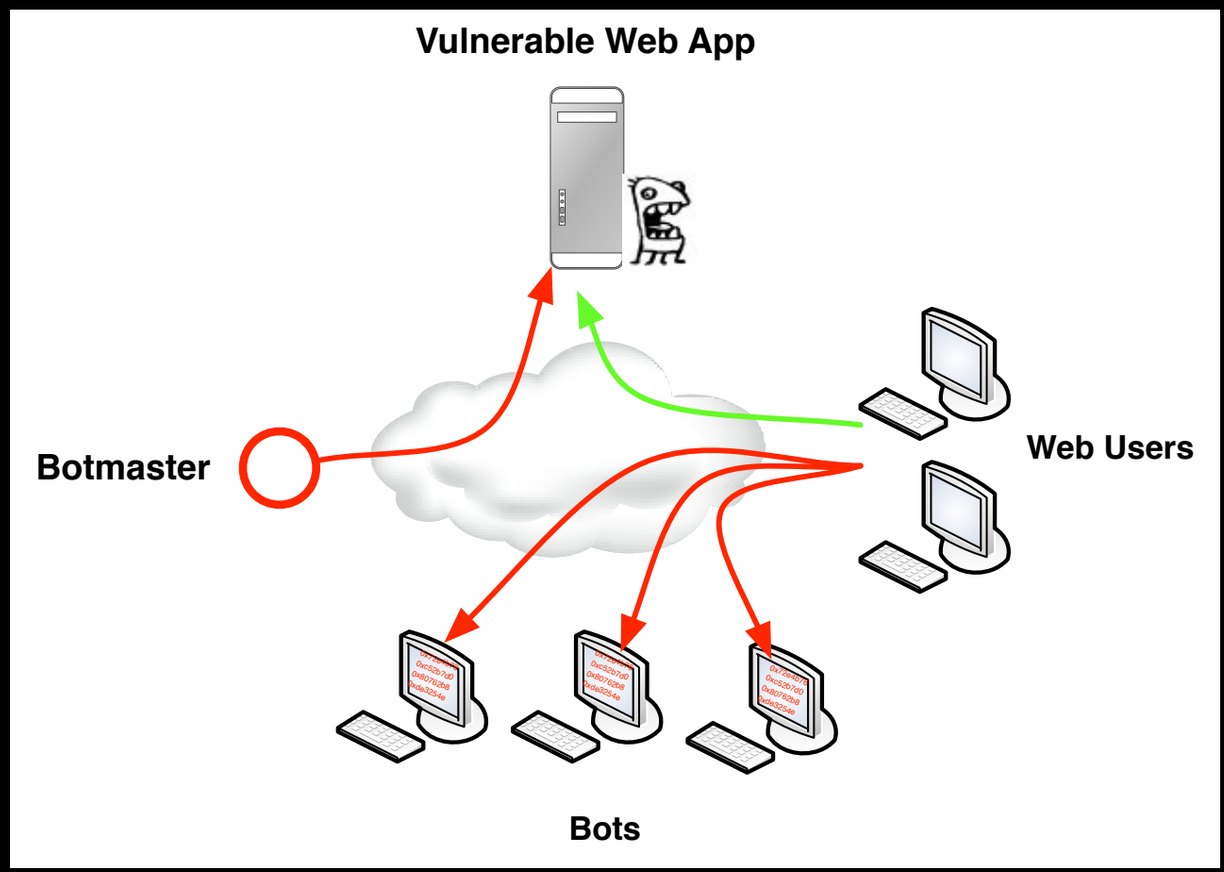
Command Dissemination

- Expendable layer of hosts
- No knowledge about the botmaster
- Do the “heavy lifting” of disseminating commands for the botmaster

Browsers!

- Browsers were created/optimized to do large number of requests per second
- Available crypto libraries in Javascript
- HTML5 brings new capabilities to the table

“Honest” intermediate layer



- Botmaster deploys (or infects) website with malicious code

“Honest” intermediate layer

- Command dissemination is not done by botmaster
  - Reduces the vulnerability to detection
- Visitors of the infected website propagate commands
  - Dissemination speed increase x #Web Users
- Detecting the existence of a bot is difficult
  - Commands are received but not acknowledged

“Honest” intermediate layer

- Replaying the commands will only further spread the botmaster’s orders
- Intermediate layer is expendable and can expire quickly
- Once the page is closed, all traces of “infection” of the web-browser disappear
  - It is hard for researchers to find the original malicious page

Analysis of Command Dissemination

- We created Javascript PoC
- Measured the number of AJAX requests per second
- Used EasyXDM to bypass Same-Origin-Policy
- Implemented public-key signatures for commands in Javascript

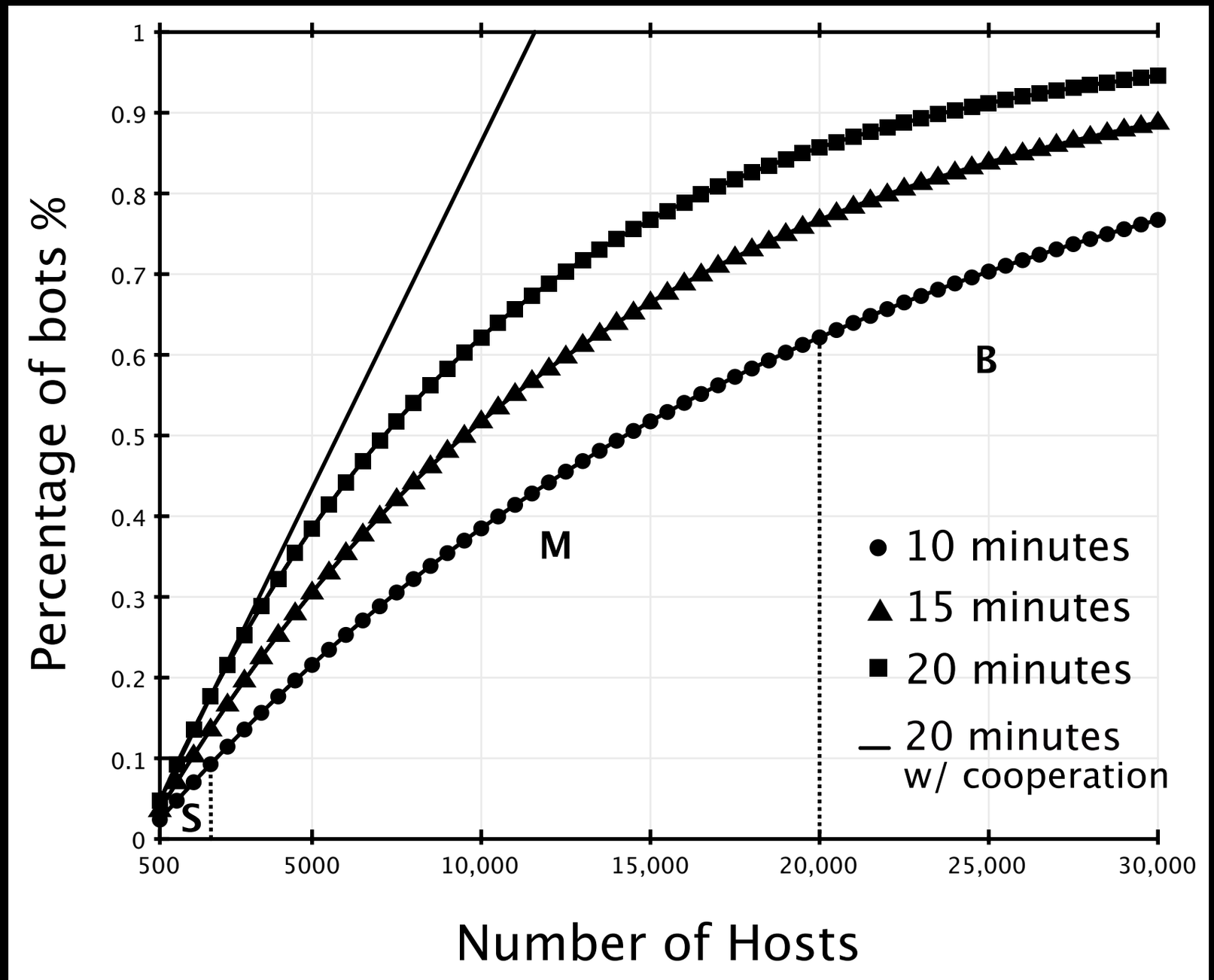
Analysis of Command Dissemination

- N = #bots
- S = #ips in the address space
- r = #requests / second a browser can make
- d = #days the malicious website is active
- v = #visitors per day the website receives
- m = #minutes a user spends on the website

Analysis of Command Dissemination

- N = 150000 bots
- S = 3086889768 (2^32 - Bogons)
- r = 250 requests/second
- d = 1 day

Analysis of Command Dissemination



Getting Visitors

- Create malicious website
  - Advertise through spam email, twitter, search engine poisoning, abuse URL shortener, etc
- Infect existing website:
  - XSS or SQL injection sufficient to get malicious code on legitimate websites
- Keeping users on the websites
  - Tabnabbing, clickjacking

Information Upstream

- Botmasters want to send stolen data upstream (credit-cards, email accounts, SSN's, etc)
- Our command dissemination infrastructure isolates each bot for robustness and stealthiness, but makes it difficult to create an upstream channel

Information Upstream

- For spamming-only botnets a simple solution, send information encoded along with spam
  - All information is encrypted with the botmaster's public key, ensuring confidentiality of data
  - The bot only has to do one thing: send spam

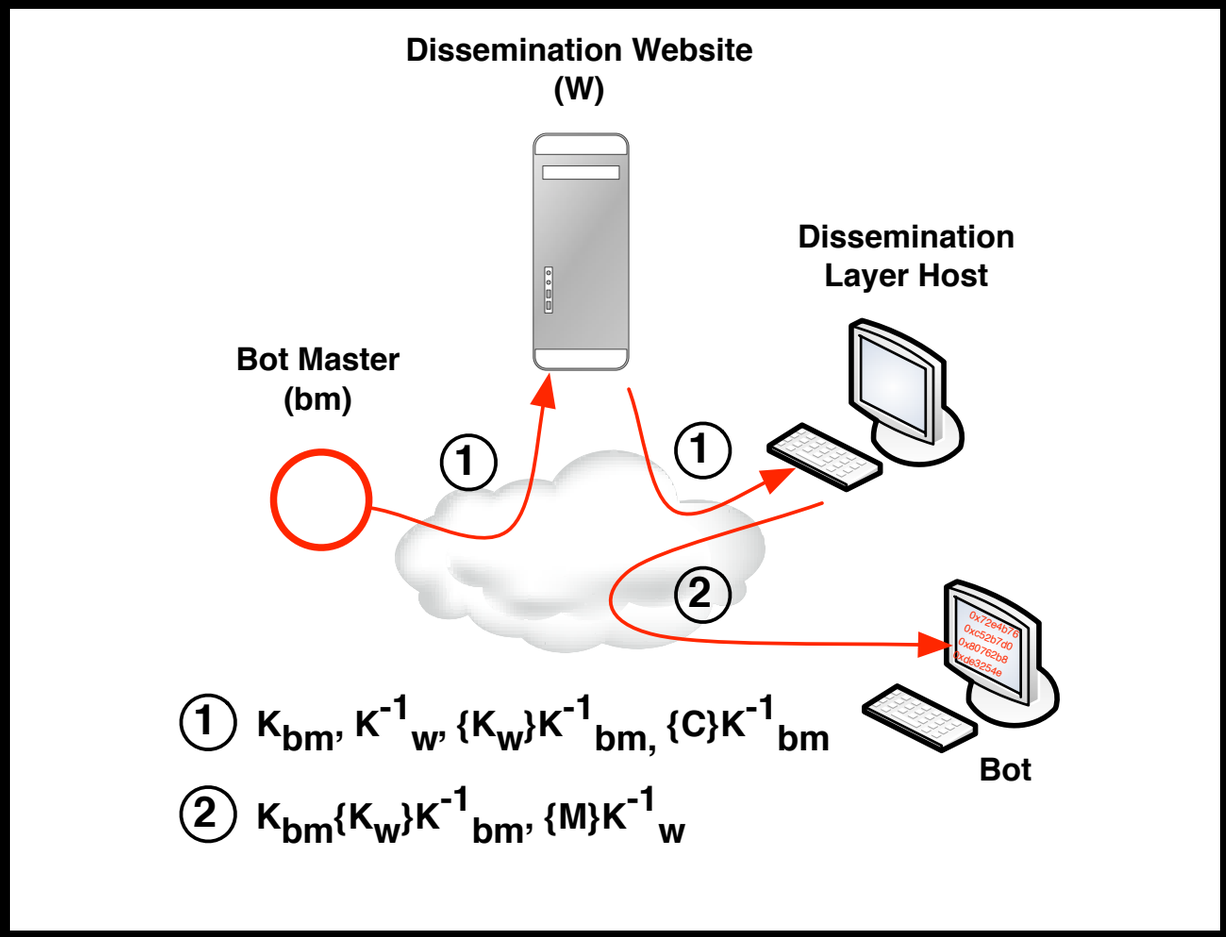
Information Upstream

- Does not expose the botmaster
- Stealth operation
- Only the botmaster can extract data from the bots

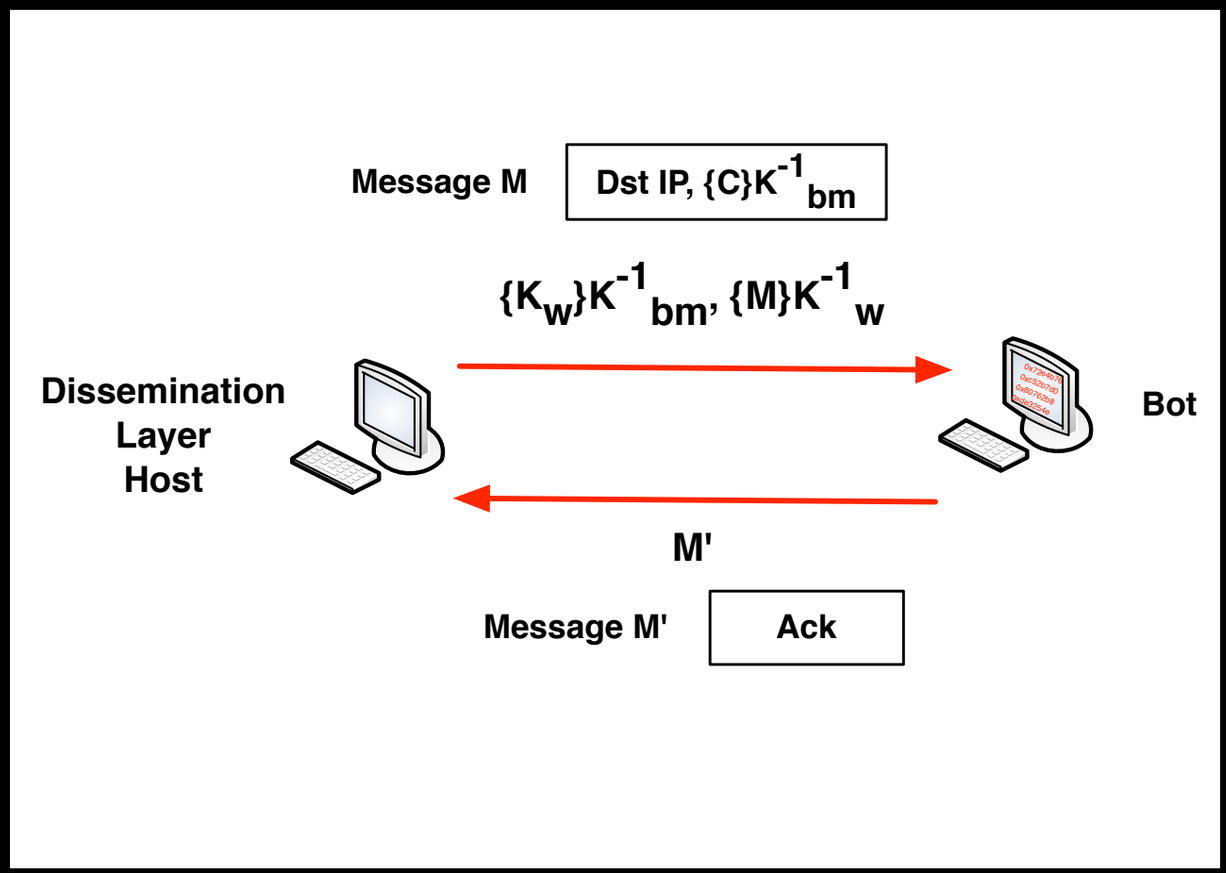
Information Upstream

- Botmaster creates website private/public key-pair and signs it with it's own public key
- The malicious code sent to the browsers includes this key-pair
- Browsers can prove themselves as originating from a "legitimate" dissemination website

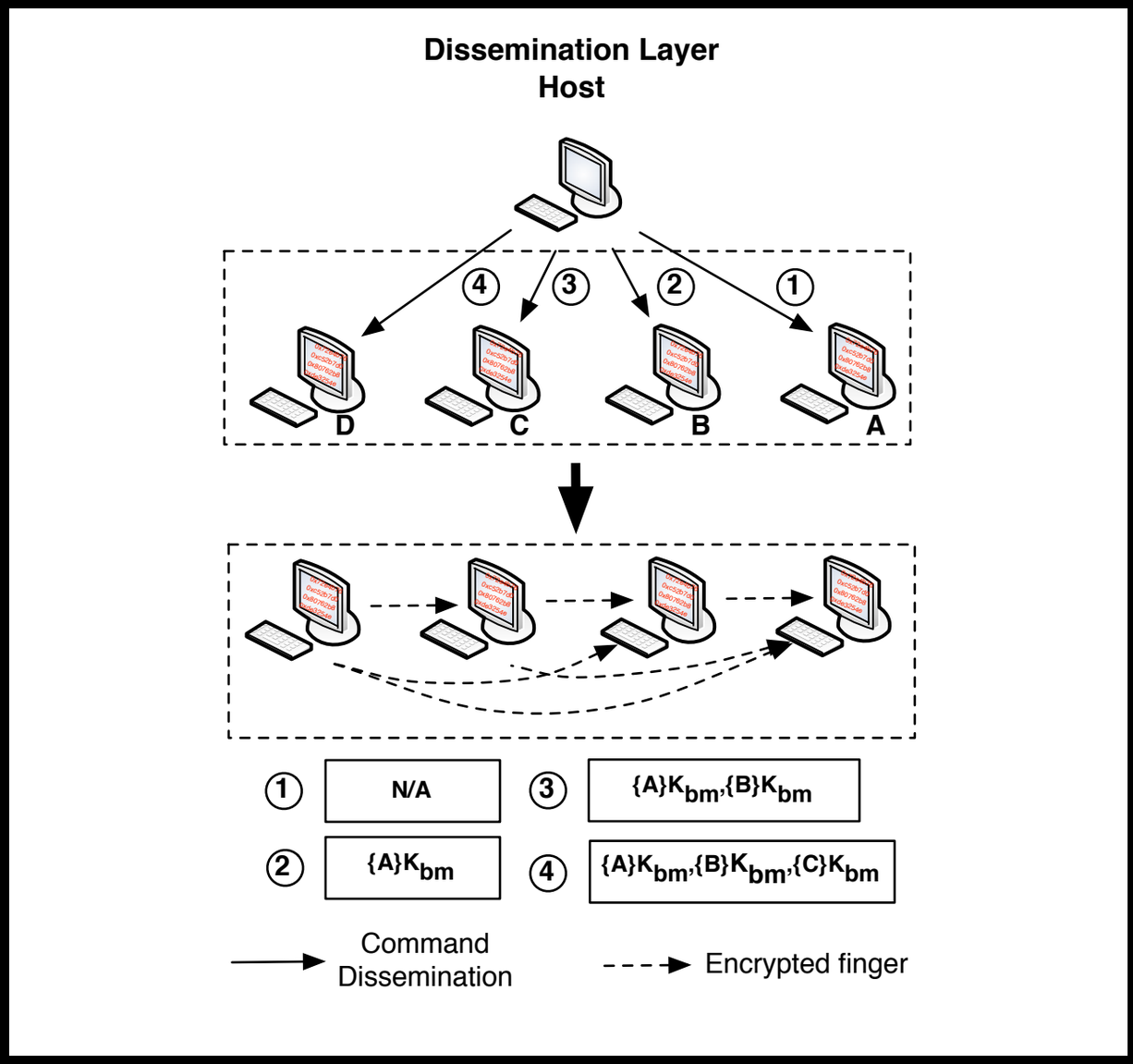
Information Upstream



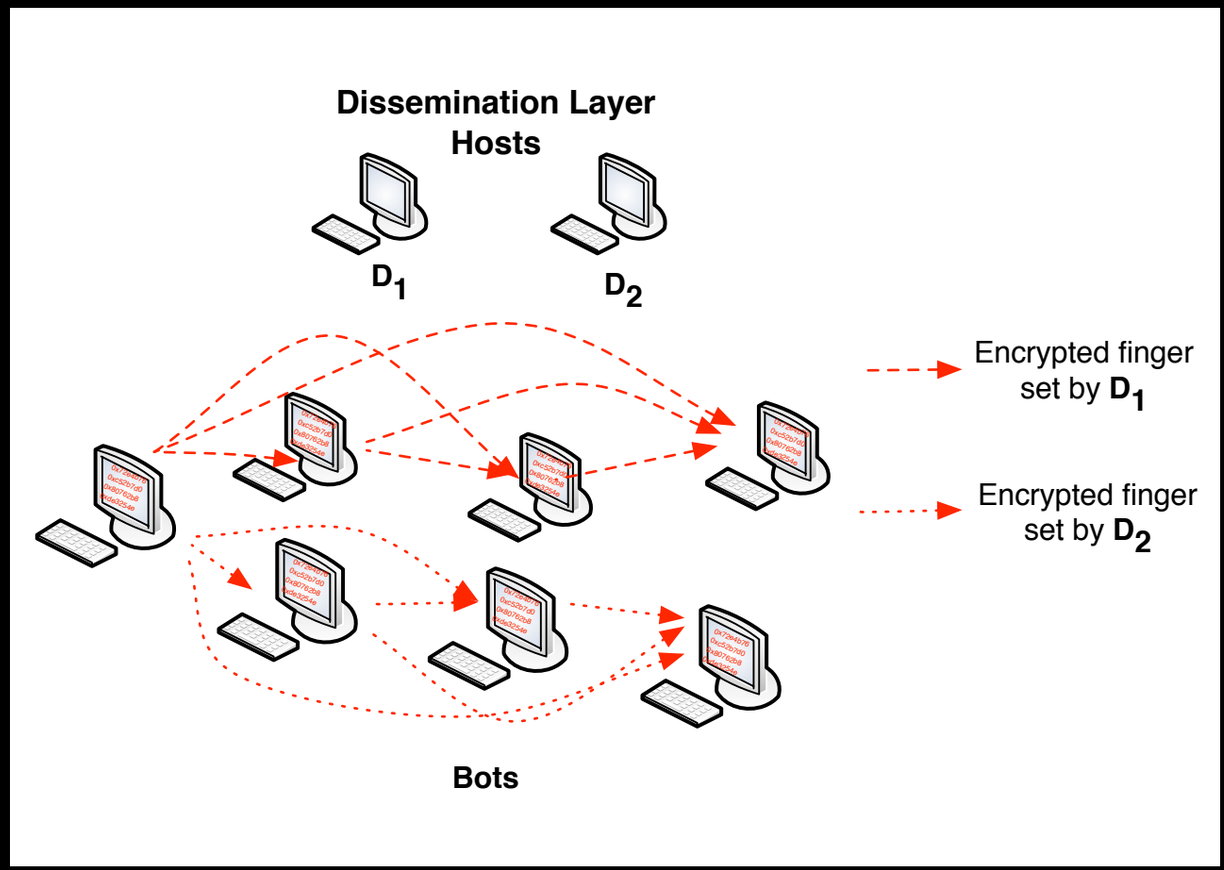
Information Upstream



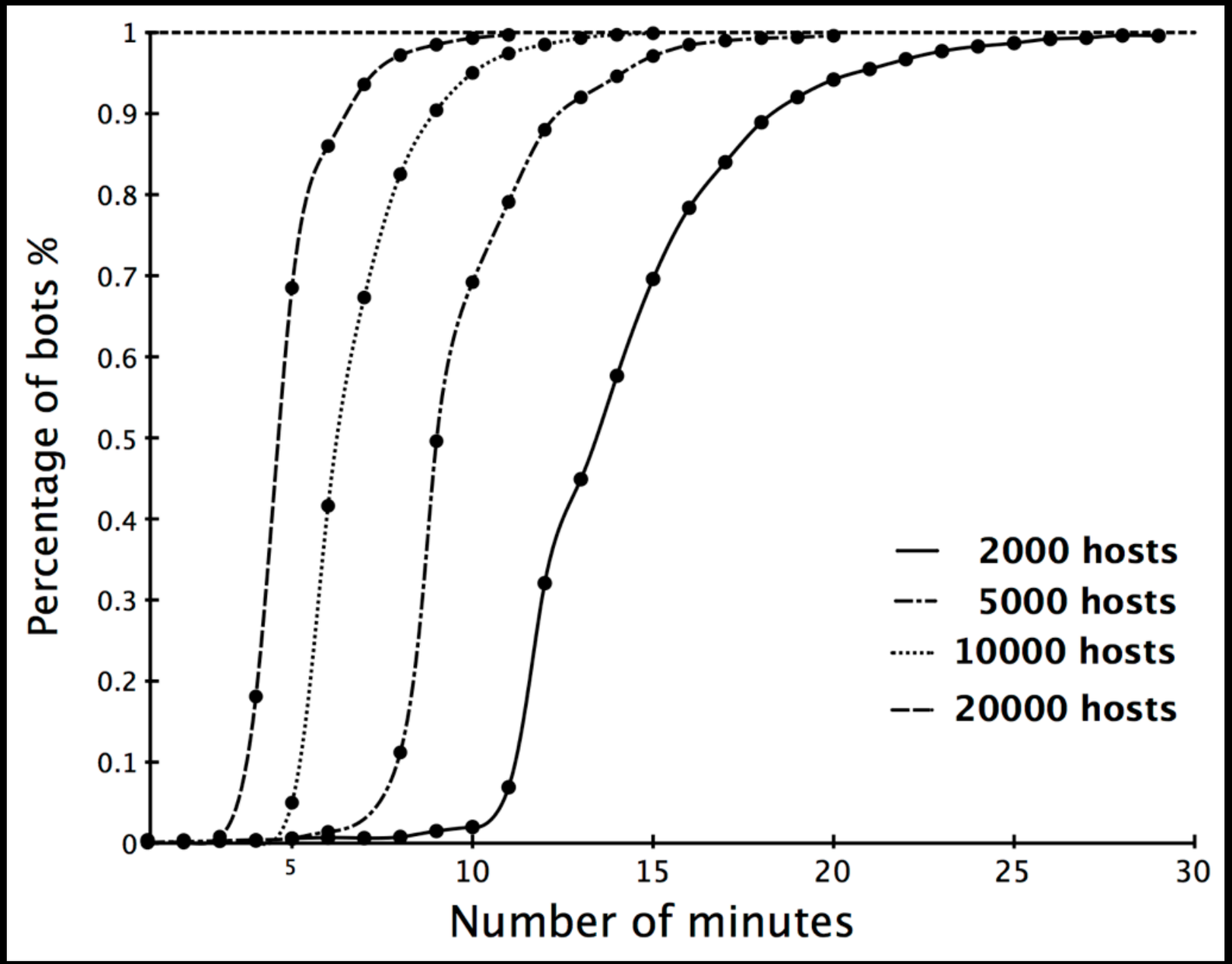
Information Upstream



Accessing the overlay



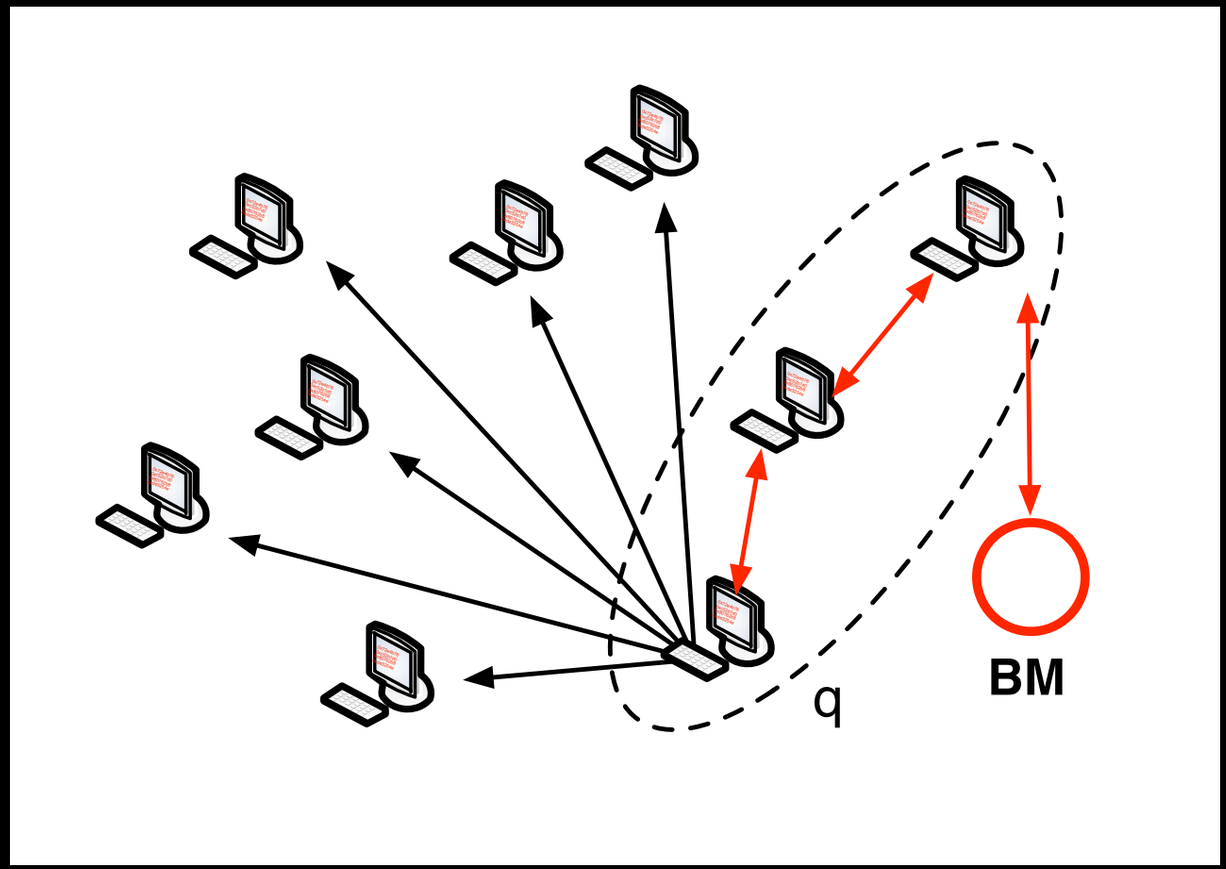
Overlay connectivity



Accessing the overlay

- Botmaster randomly scans the internet until it finds one host.
- Uses the encrypted fingers of this host to start crawling through the overlay.
- But...

Accessing the overlay



- Botmaster still needs to bounce through some nodes to guarantee anonymity when retrieving data

Final remarks

- Stealth C&C using browsers are feasible
- Increasing role of browsers in the malware landscape
- We should focus some IDS effort on the browsers
- We aren't good enough at detecting malicious websites

Thank you

Questions?

diogo.monica@ist.utl.pt