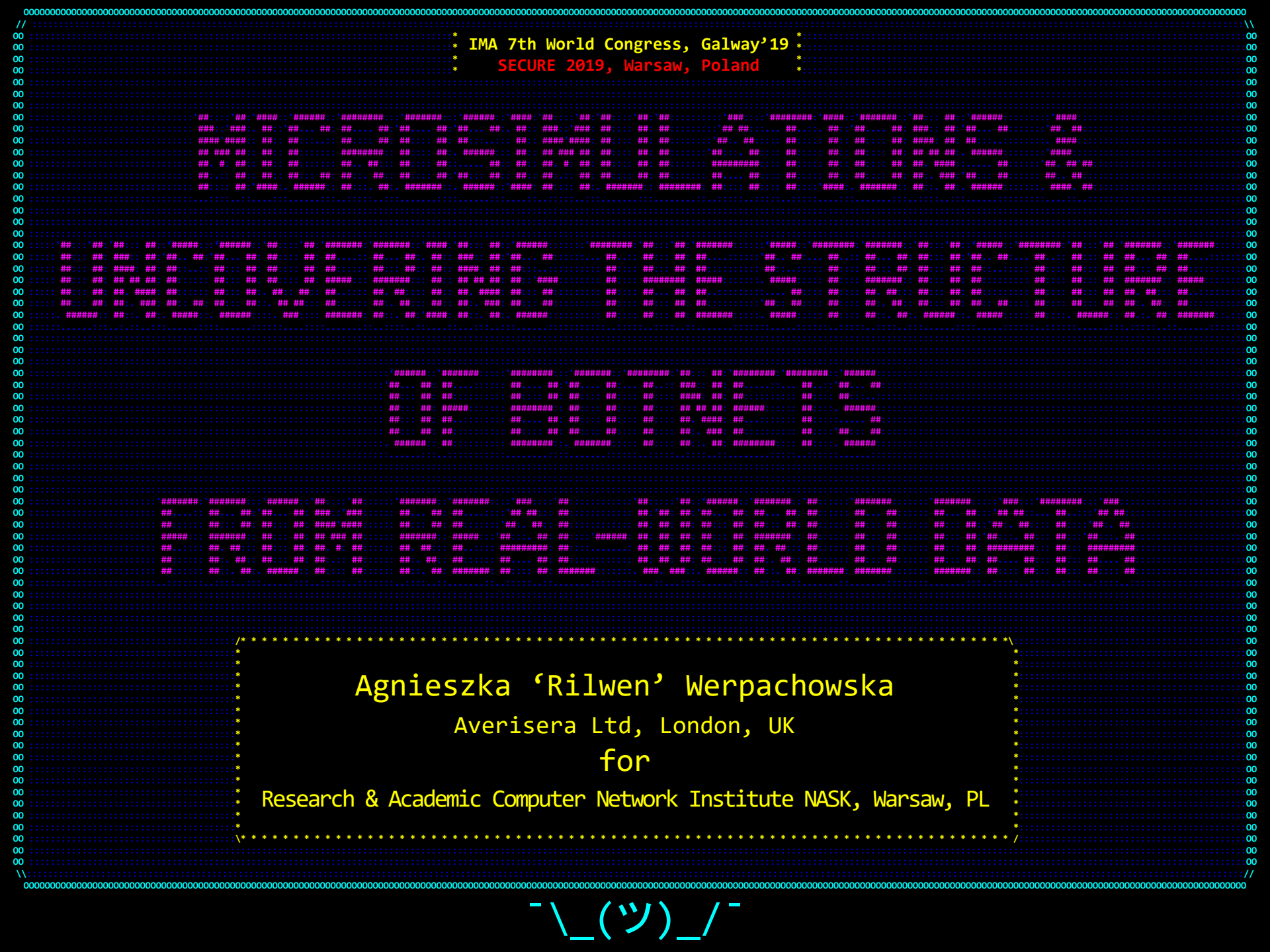
IMA 7th World Congress, Galway'19

SECURE 2019, Warsaw, Poland




Agnieszka 'Rilwen' Werpachowska

Averisera Ltd, London, UK

for

Research & Academic Computer Network Institute NASK, Warsaw, PL


¯_(ツ)_/¯

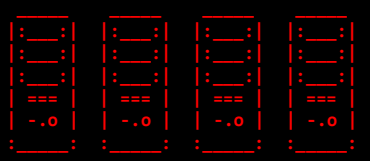
# /*** What are Botnets? ***/

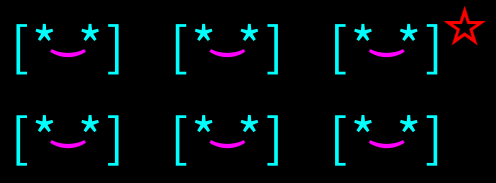
- **Botnet** is a network of compromised computers (zombies / bots) under the control of a remote attacker (bot master)
- Bots were originally developed as a useful tool – virtual agents helping the operators of IRC channels to monitor network traffic
- **Scrumping** – stealing computing resources as a result of a system being joined to a botnet
- Botnets are significant contributors to the malicious & criminal activities on the Internet today (spamming, hosting illegal materials, mining ₿, DDoS attacks, stealing sensitive data, penetrating corporate networks or strategic country infrastructures, and thus posing a national security threat).
- They form an **underground network whose size & scope is not fully known**

# /*** What are botnets? ***/

/* bot master */

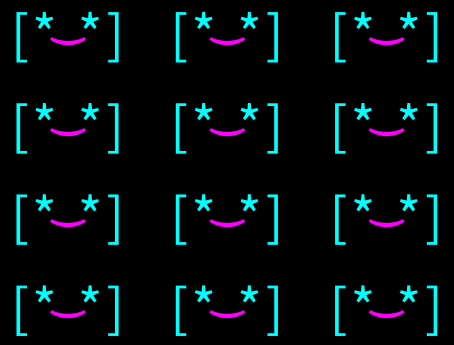
/* C&Cs */





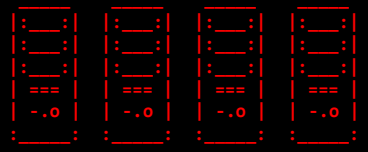
```
// spam email attachment
// infected media (e.g. USB stick)
// corrupted HTML website
// software vulnerabilities
// user naivety
```

# /*** What are botnets? ***/




/* bot master */

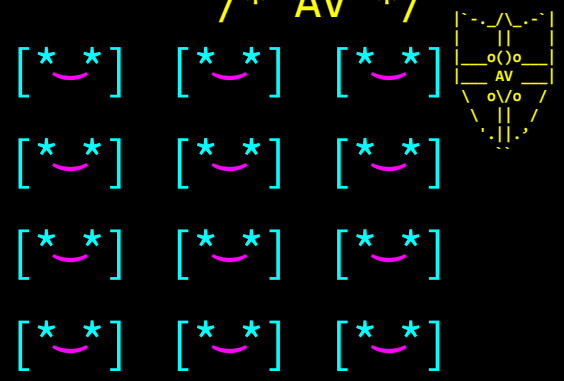
/* C&Cs */



// spam email attachment
// infected media (e.g. USB stick)
// corrupted HTML website
// software vulnerabilities
// user naivety
// social engineering

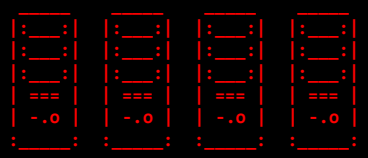
# /*** What are botnets? ***/

/* AV */



/* bot master */

/* C&Cs */

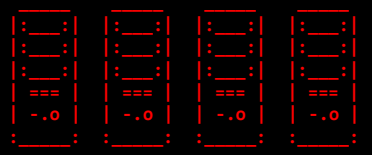


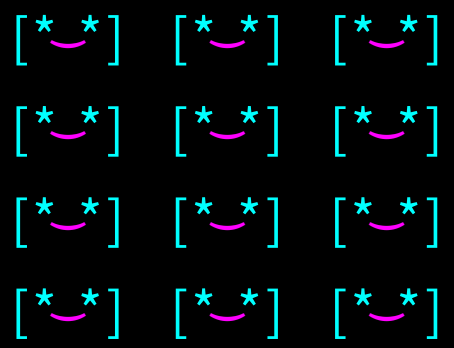
```
// household & corporate (different size distributions, user
                activity & behaviours, admins & maintenance schedules)
// desktop, laptop (can change networks), server (always on)
// different OS (types: Windows, MacOS, Linux; versions; releases)
// antivirus (releases) or not (source of data: AV benchmarks)
// user types (at home only, home & office, travelling laptop)
```

# /*** What are botnets? ***/

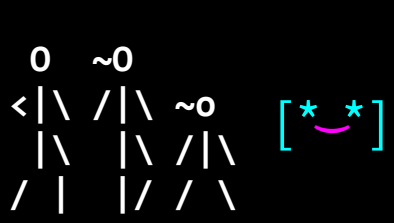
/* bot master */

/* C&Cs */

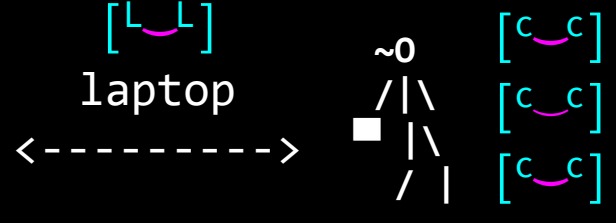




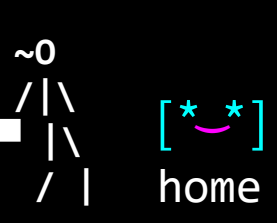
// user & admin time schedules



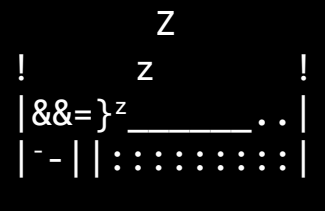
home network
IP:127.0.0.1



office network
IP:195.187.242.157



home



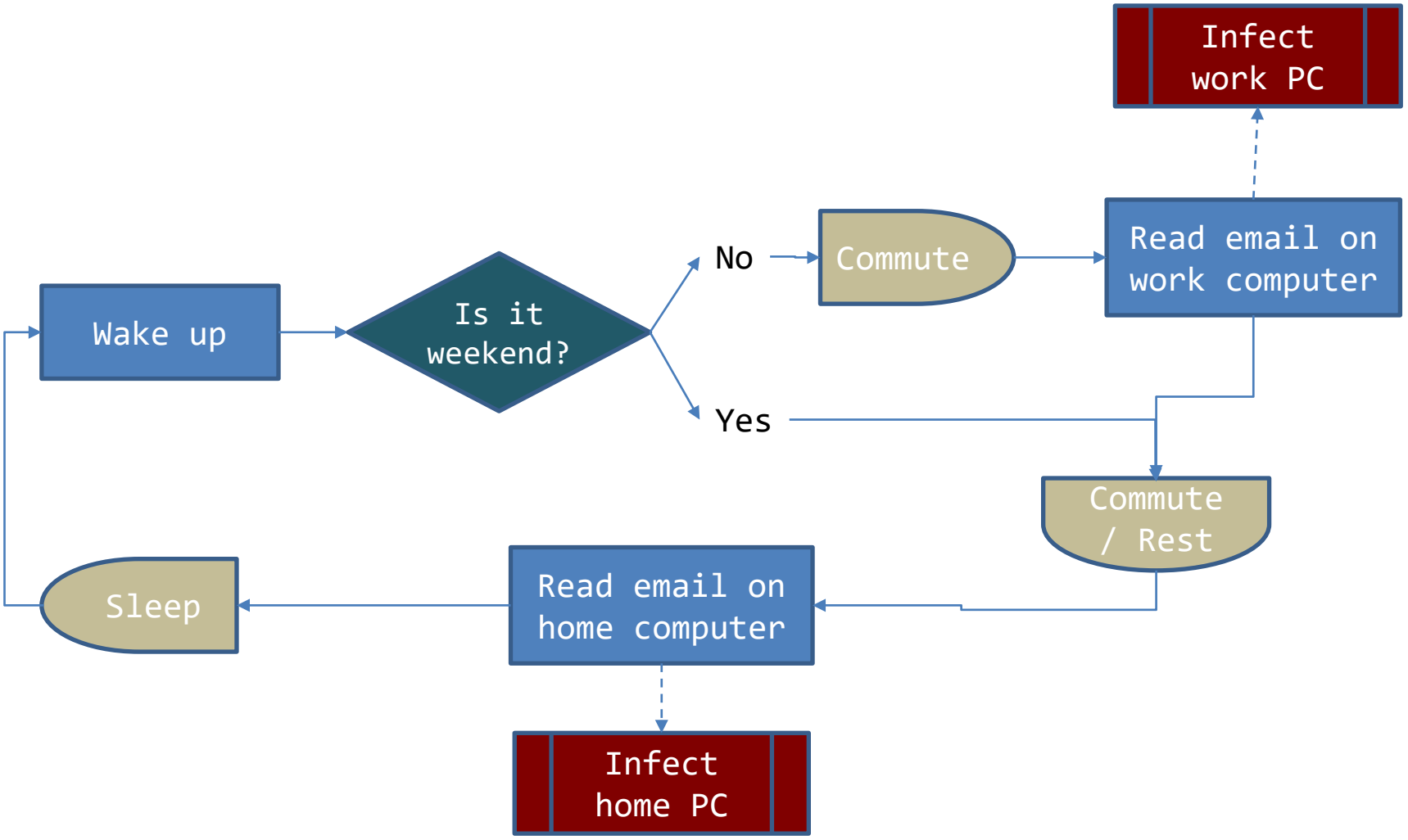
/* admin */

// maintenance
// system updates

(source of data:
e.g. ad click times)

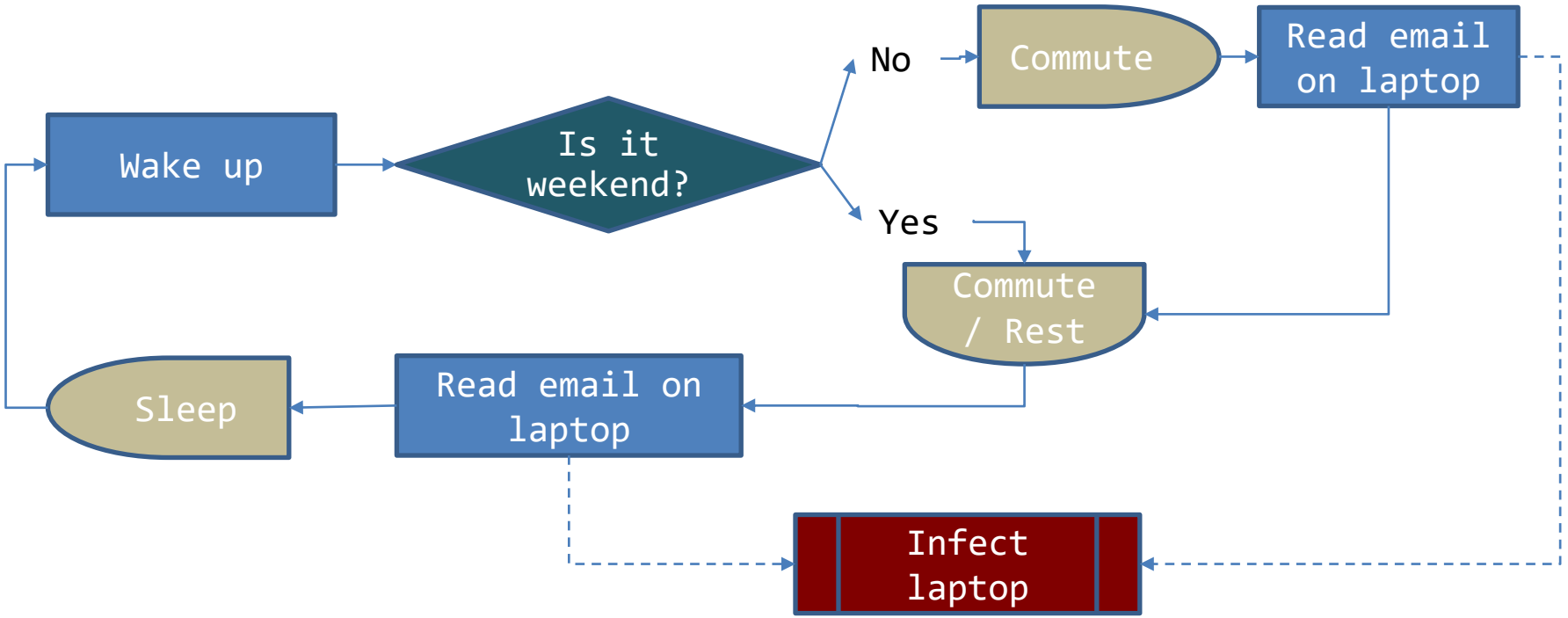
# /* Microsimulation of a botnet */

## Example of a desktop user schedule

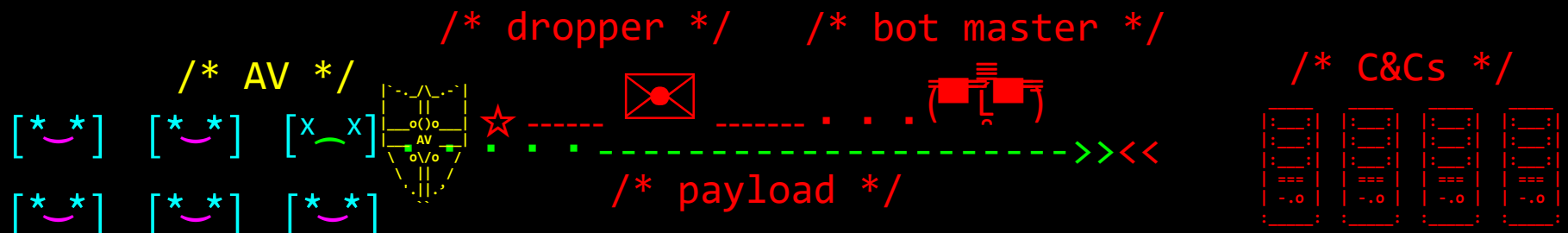


# /* Microsimulation of a botnet */

## Example of a laptop user schedule



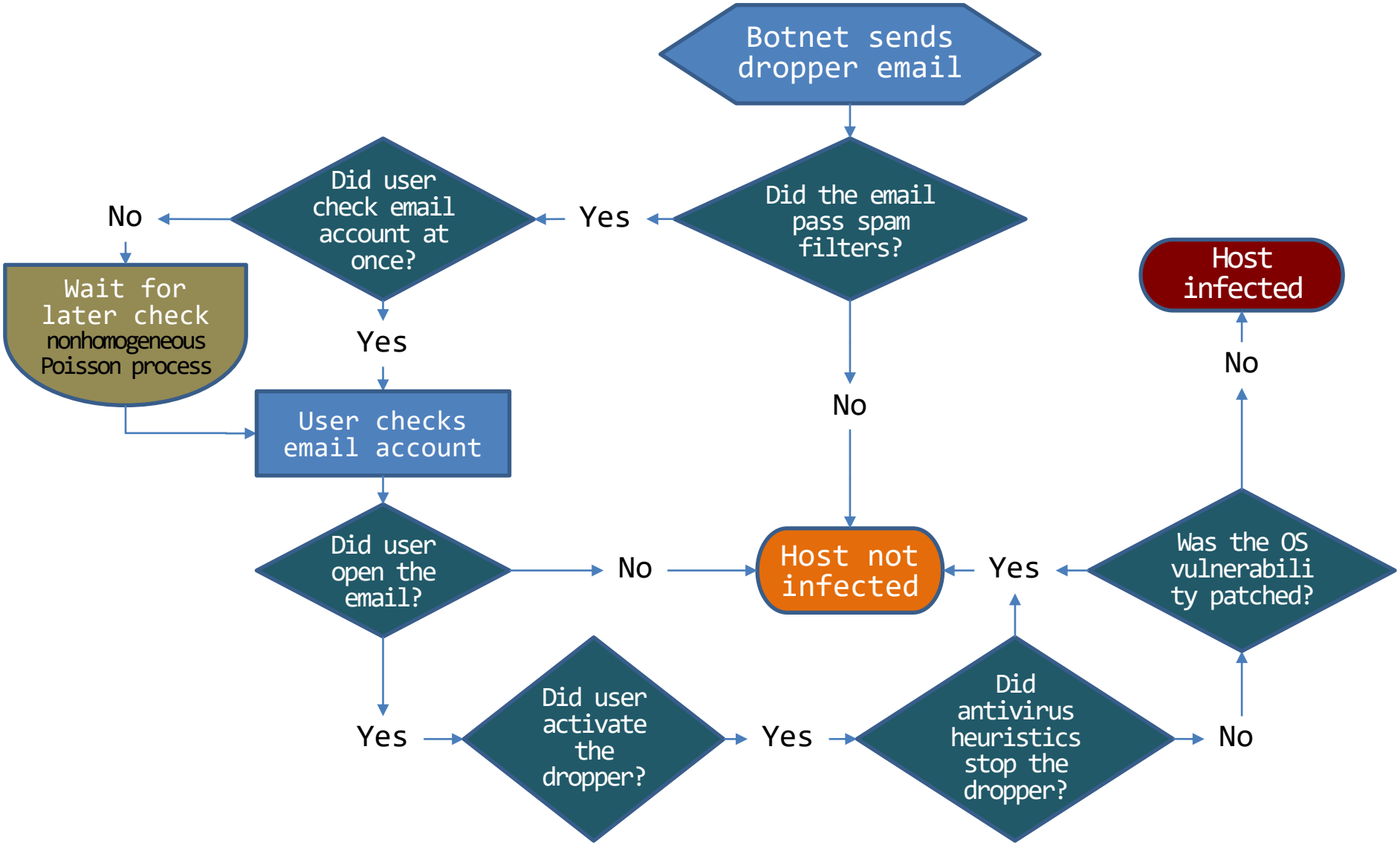
# /*** What are botnets? ***/



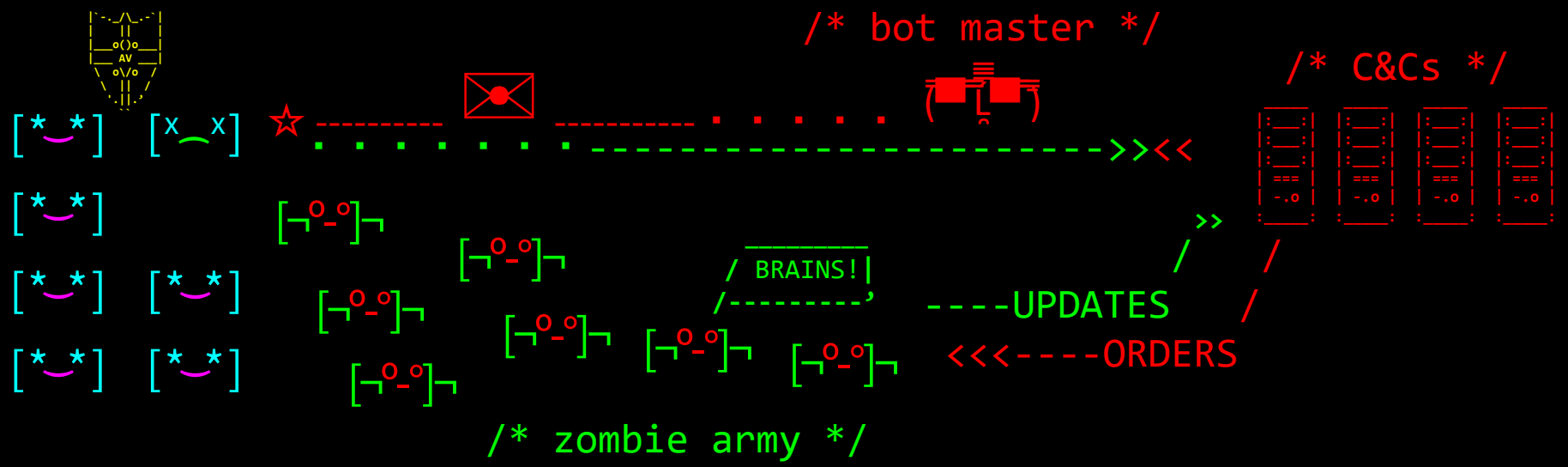

```
// established by bot master
// takes his/her instructions
// configures and updates bots
// sends orders to bots
```

# /* Microsimulation of a botnet */

## Infection proces

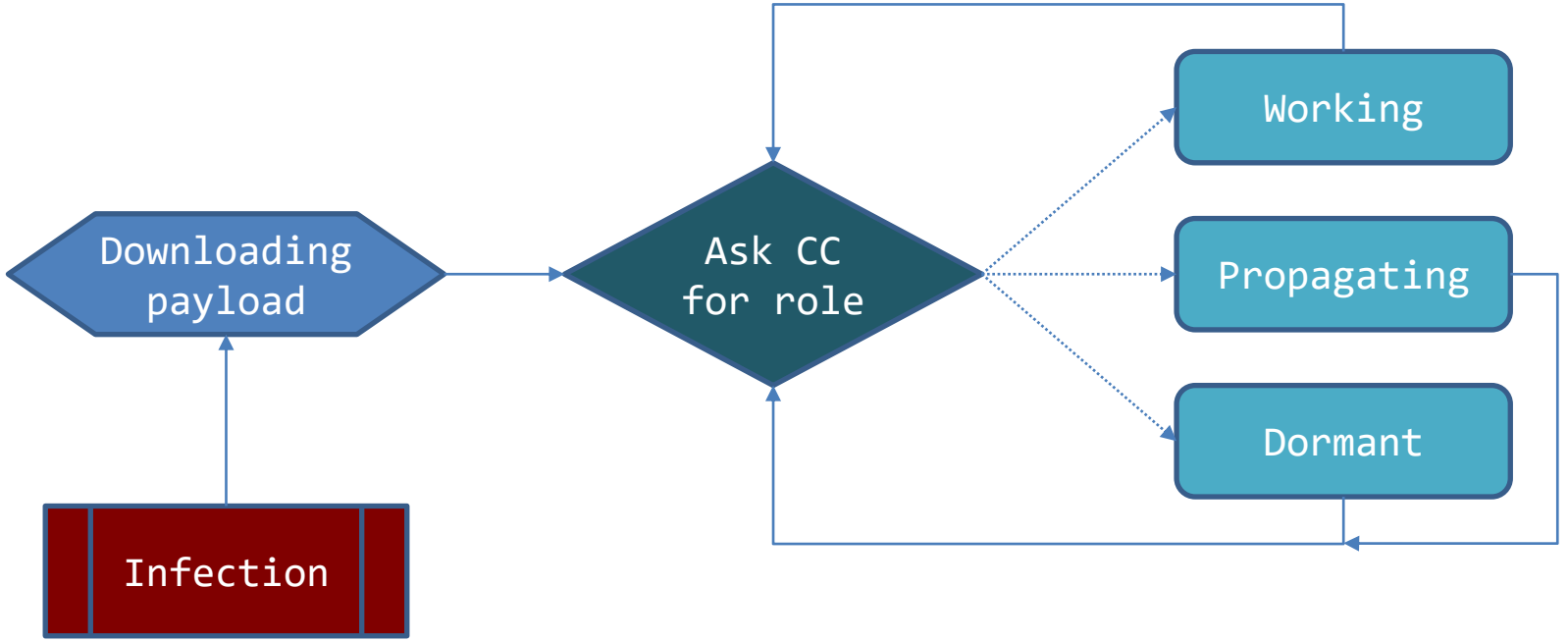


# /*** What are botnets? ***/

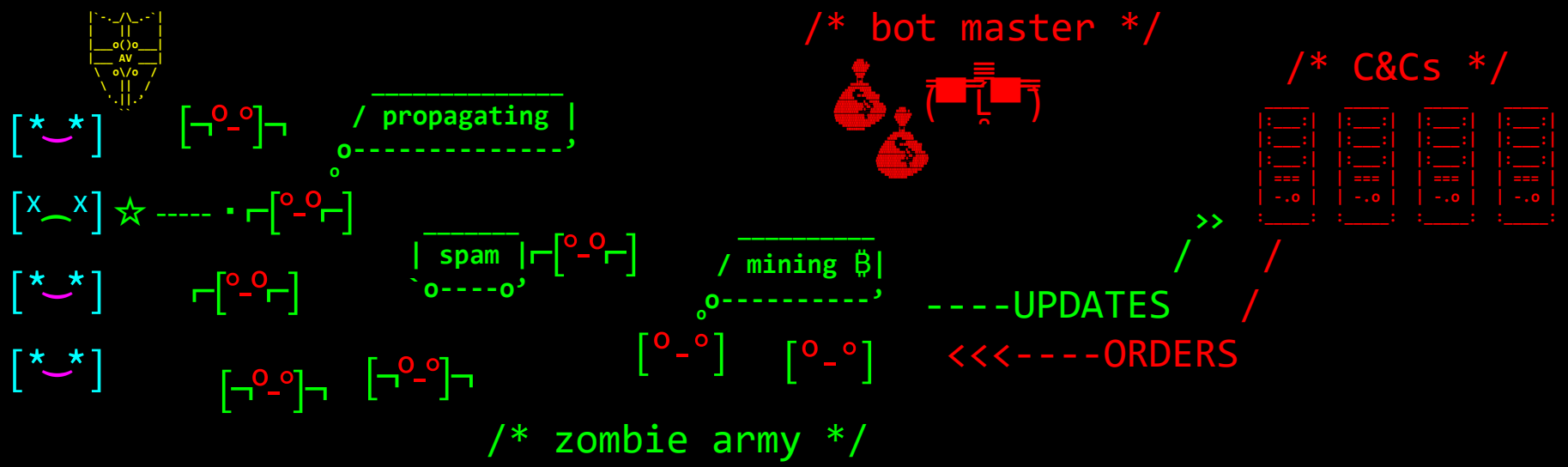


# /* Microsimulation of a botnet */

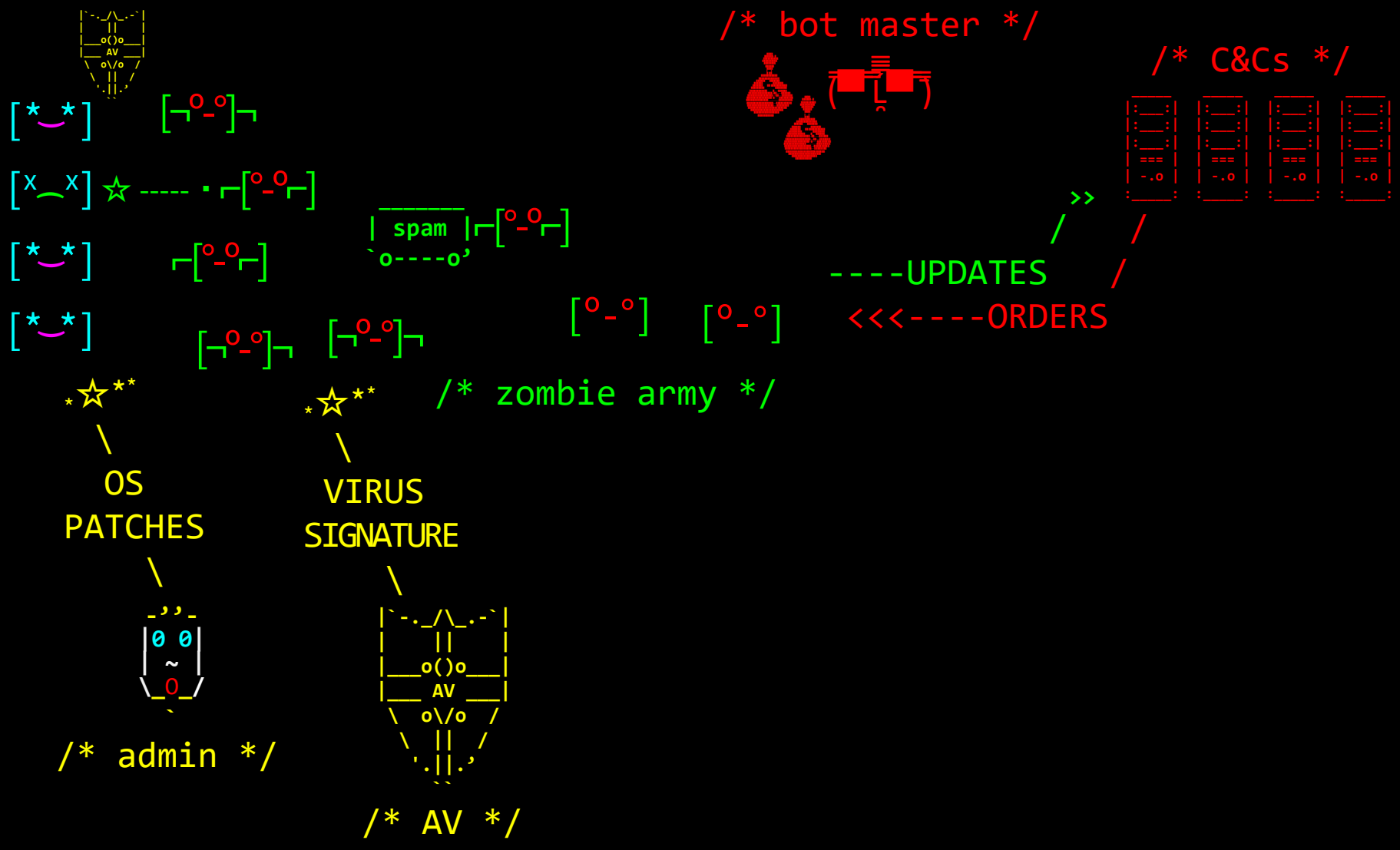
## Bot state evolution



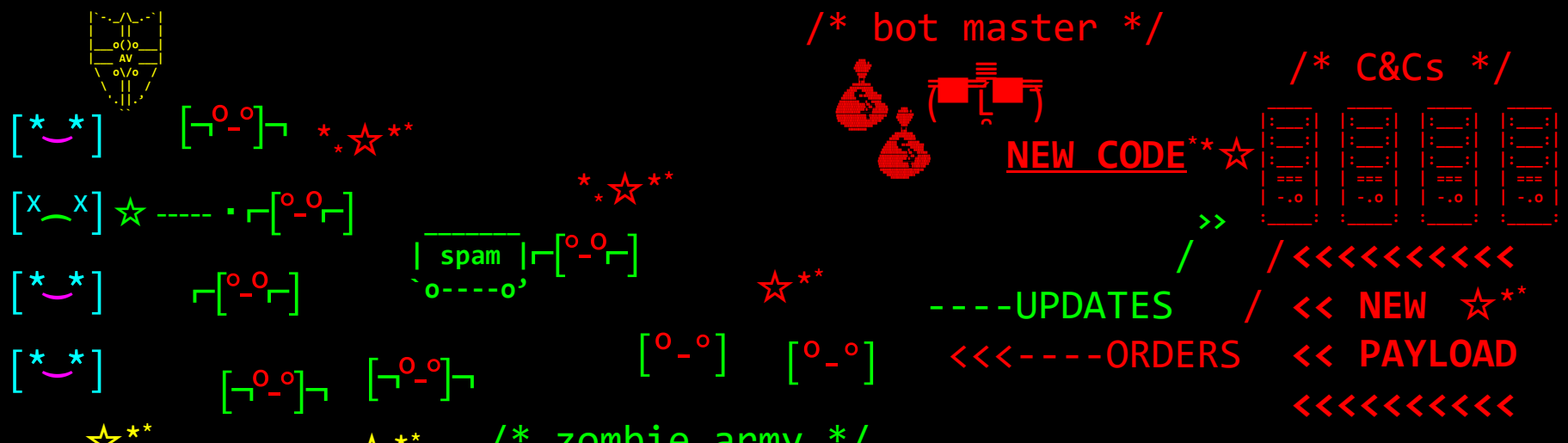
# /*** What are botnets? ***/



# /*** What are botnets? ***/



# /*** What are botnets? ***/




OS PATCHES

VIRUS SIGNATURE

/* admin */

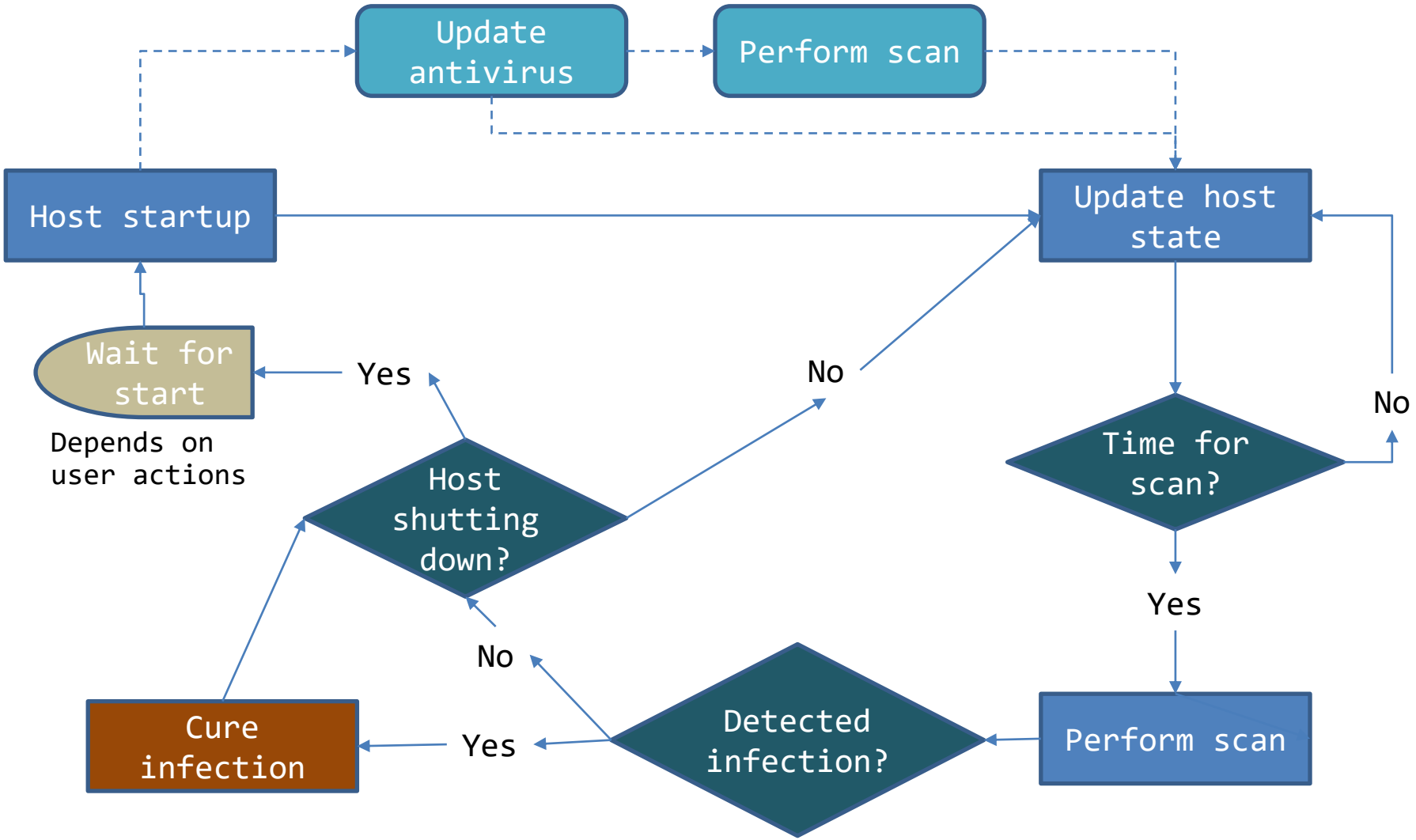
/* AV */

/* arms race */

// not 100% effective

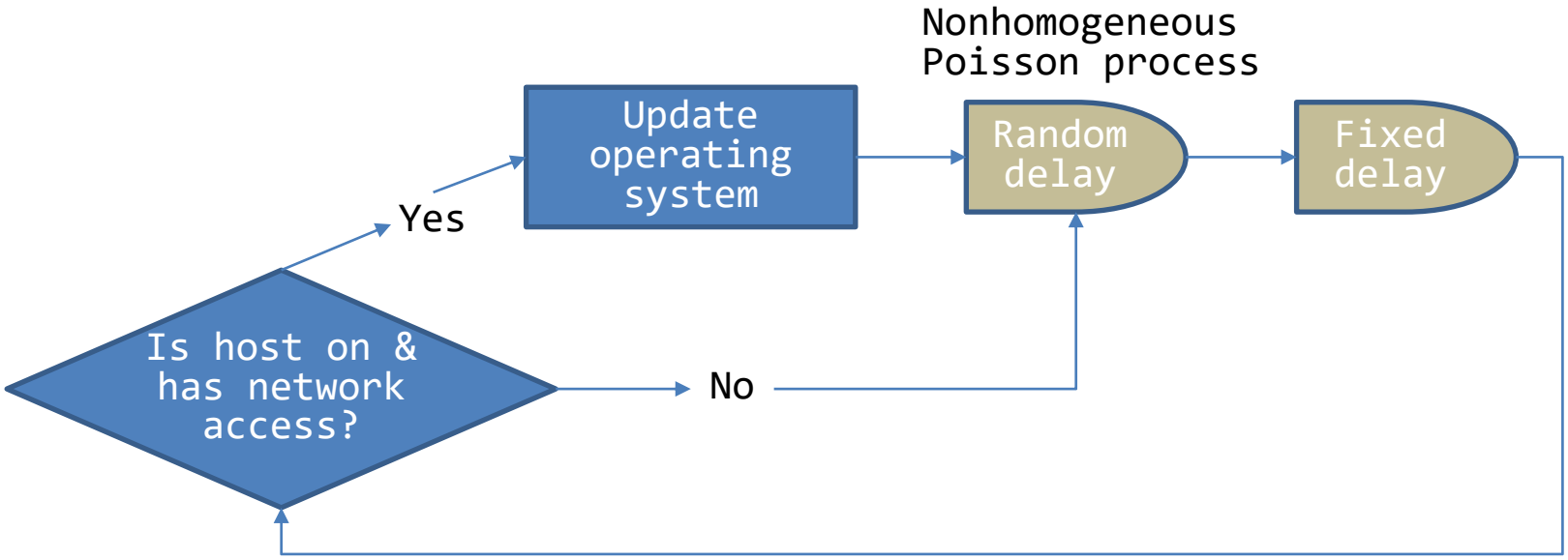
# /* Microsimulation of a botnet */

## Antivirus program operation

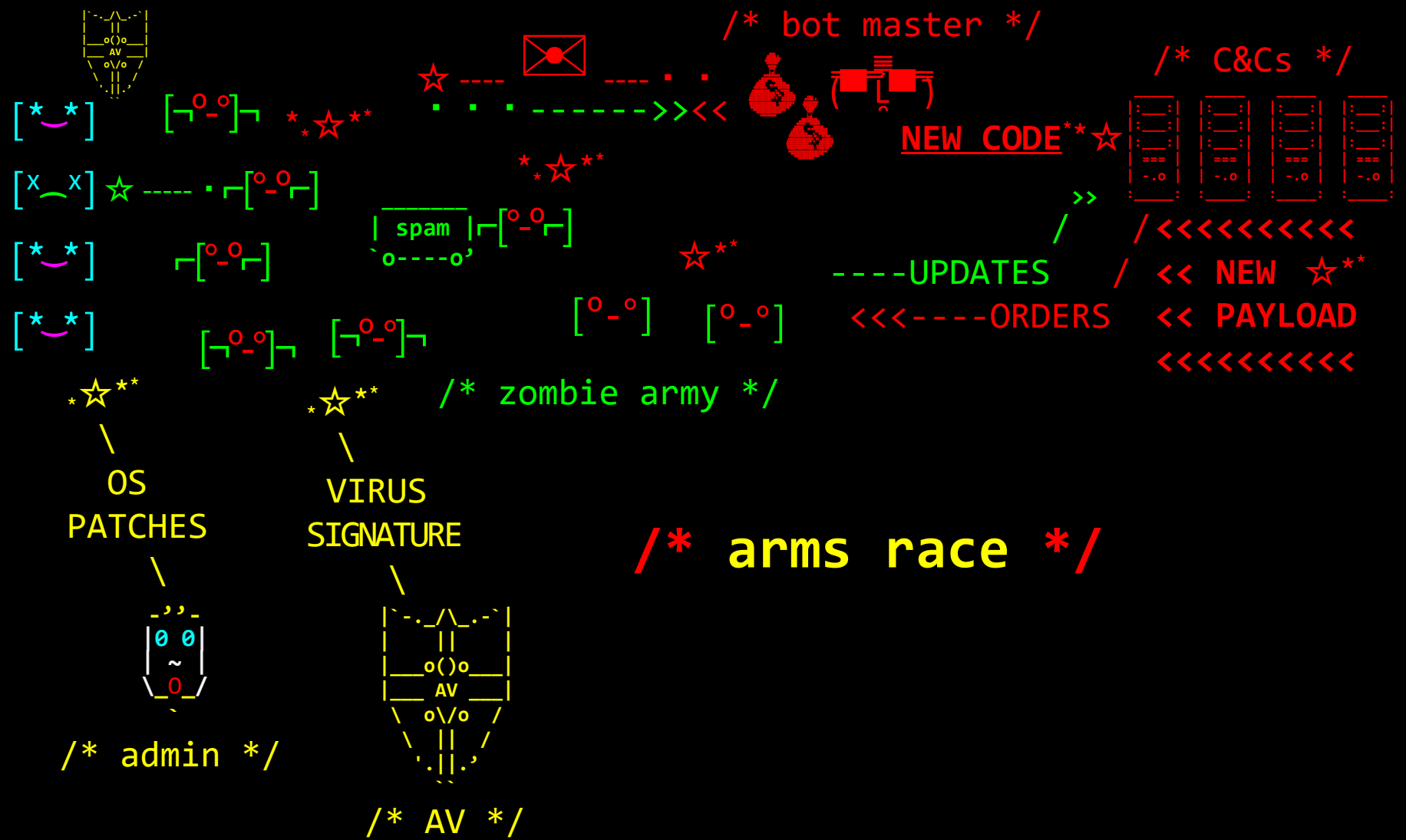


# /* Microsimulation of a botnet */

## Host maintenance by sysadmin

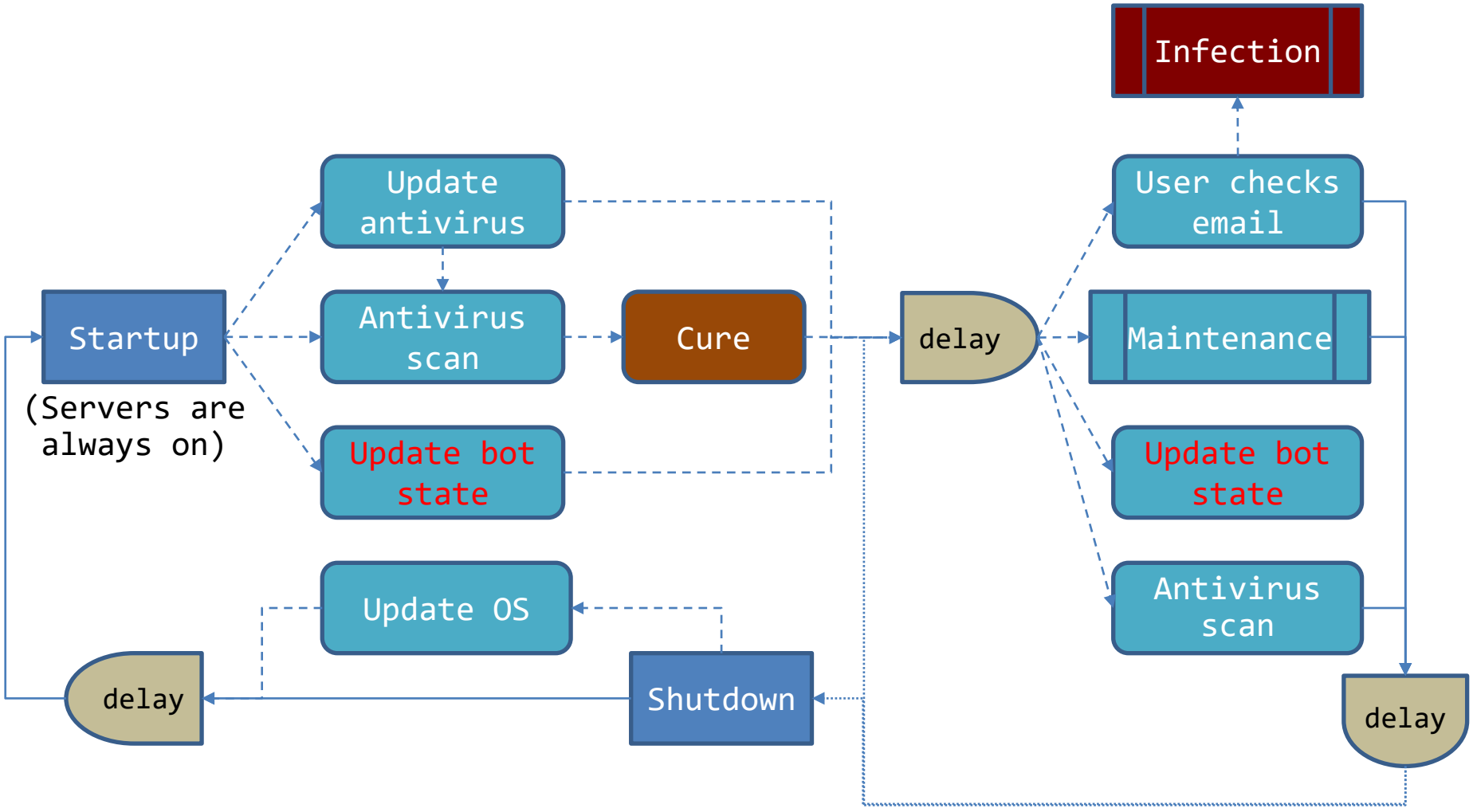


# /*** What are botnets? ***/



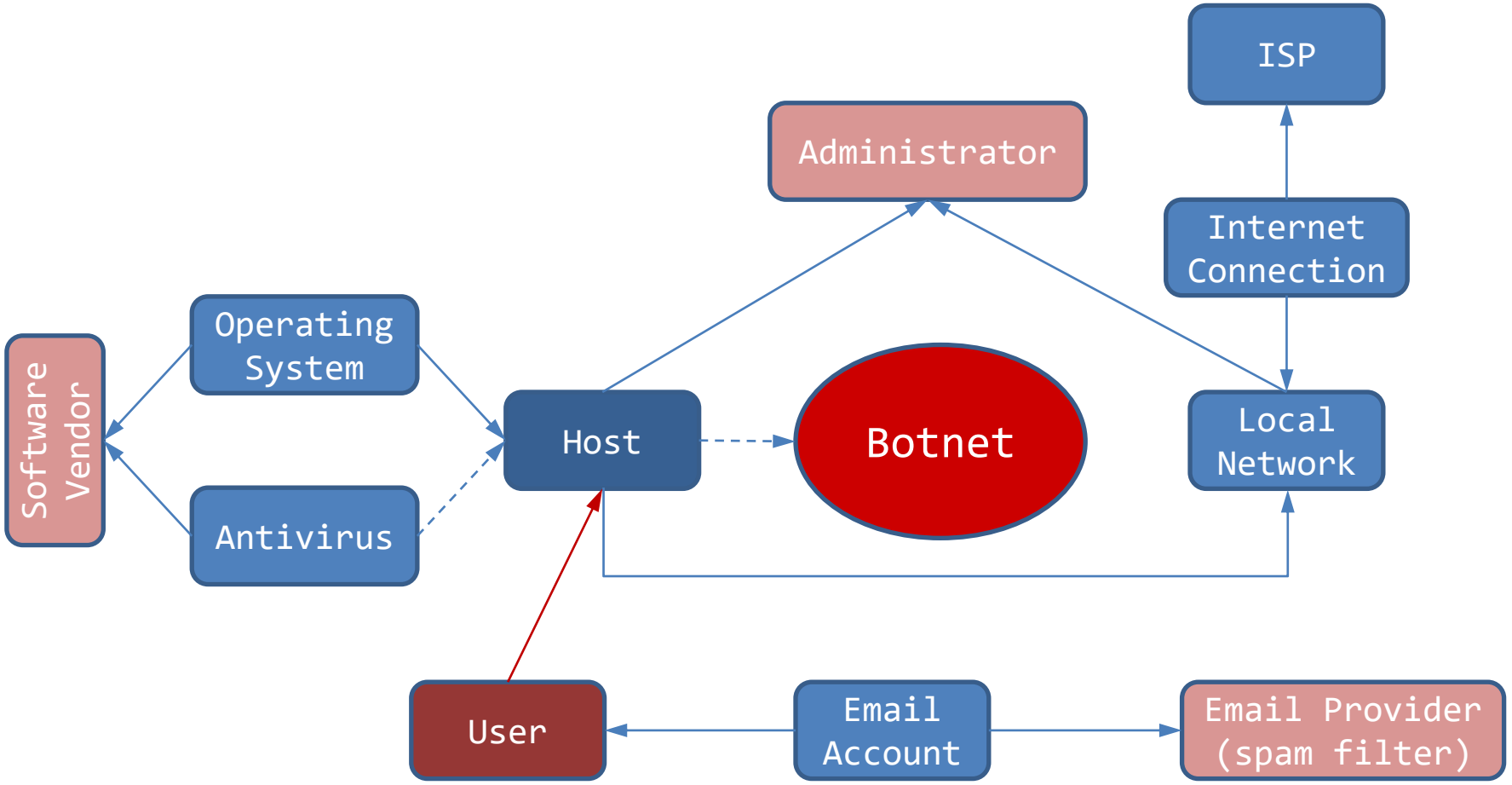
# /* Microsimulation of a botnet */

## Host activity cycle



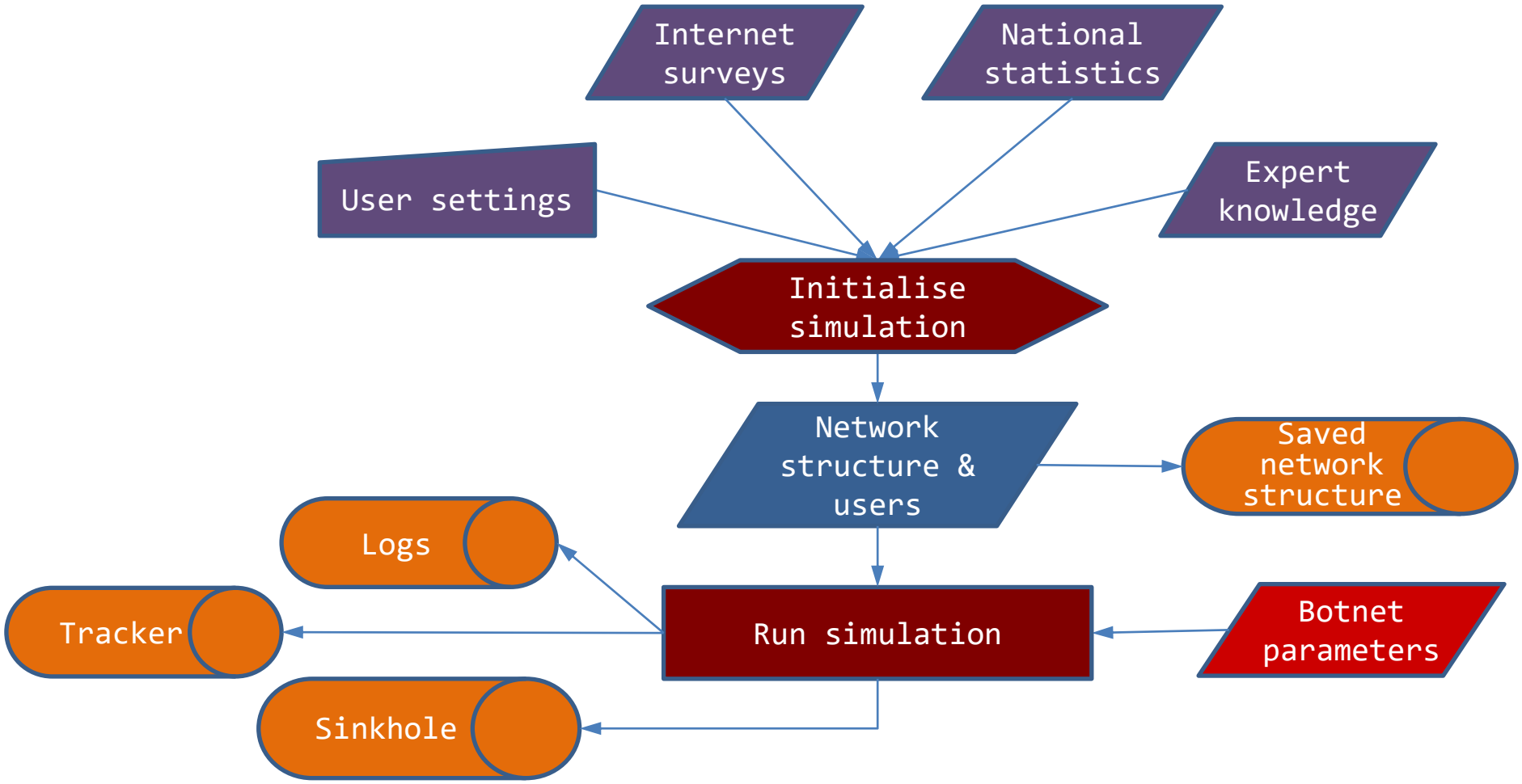
# /* Microsimulation of a botnet */

## Object model



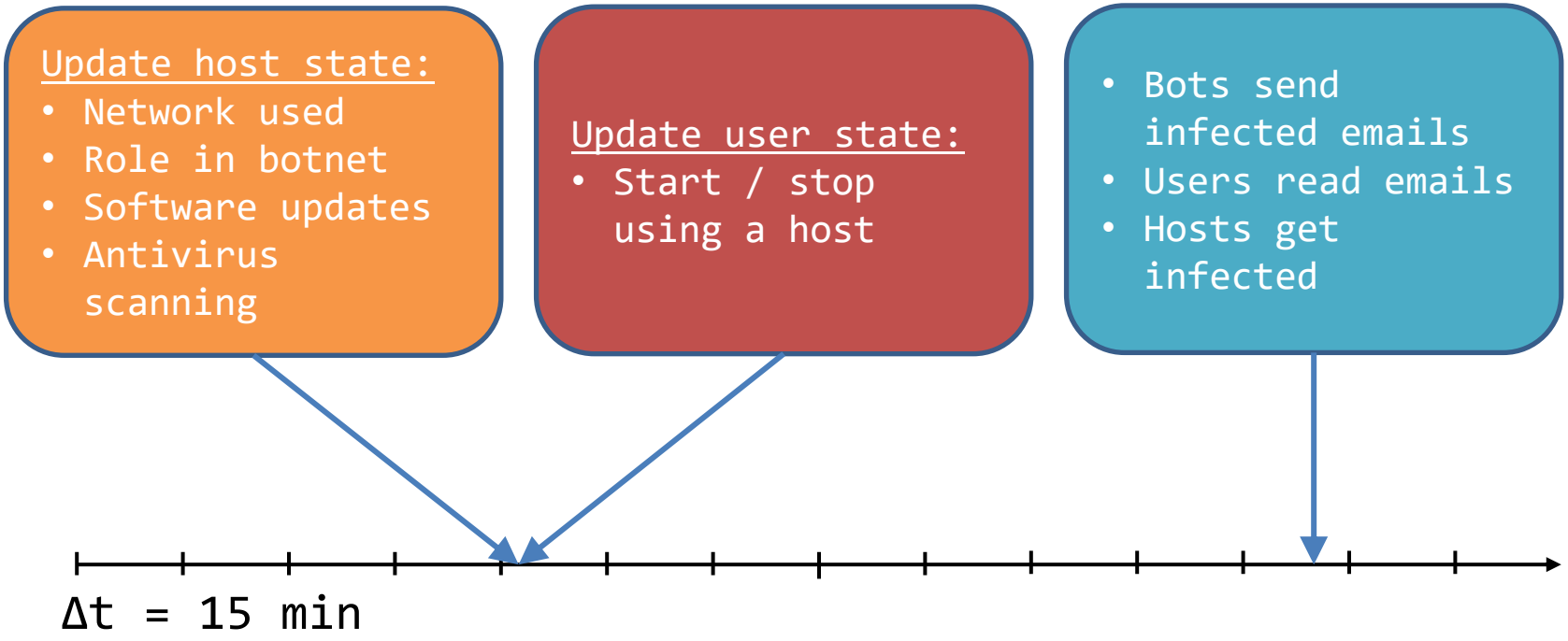
# /* Microsimulation of a botnet */

## Data flow

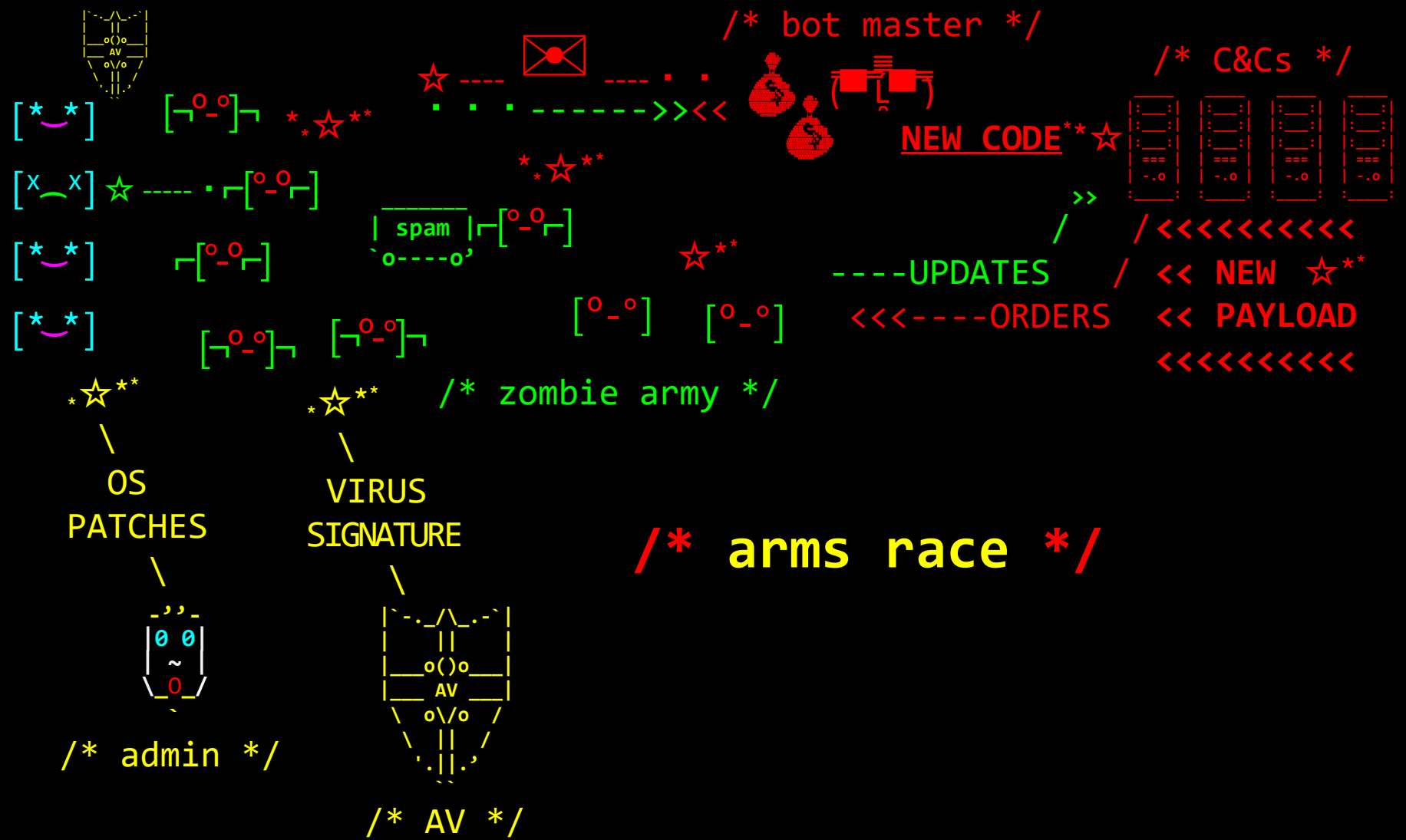


# /* Microsimulation of a botnet */

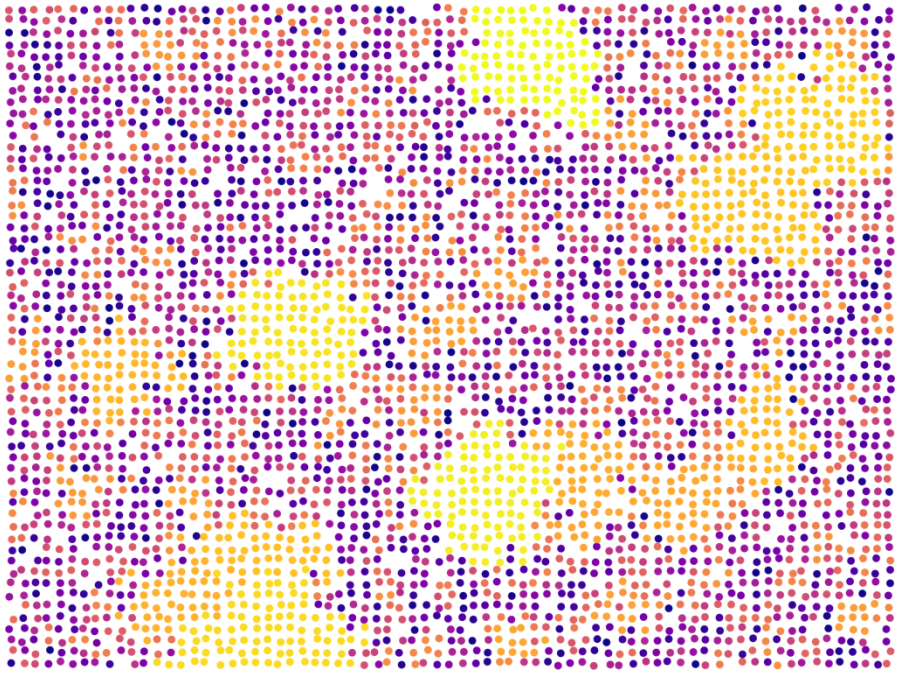
## Simulation timeline



# /*** What are botnets? ***/



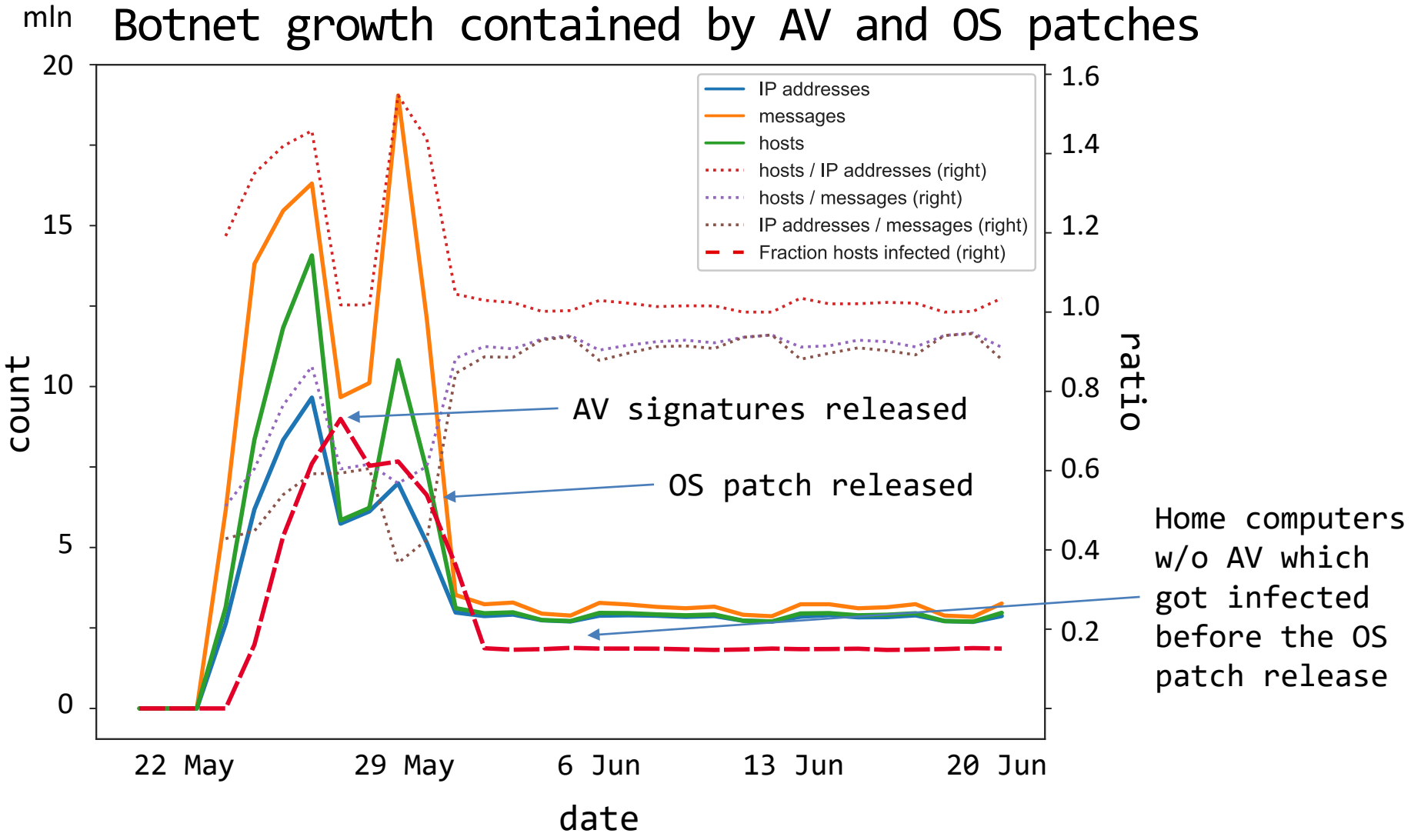
# Structure of Polish Internet



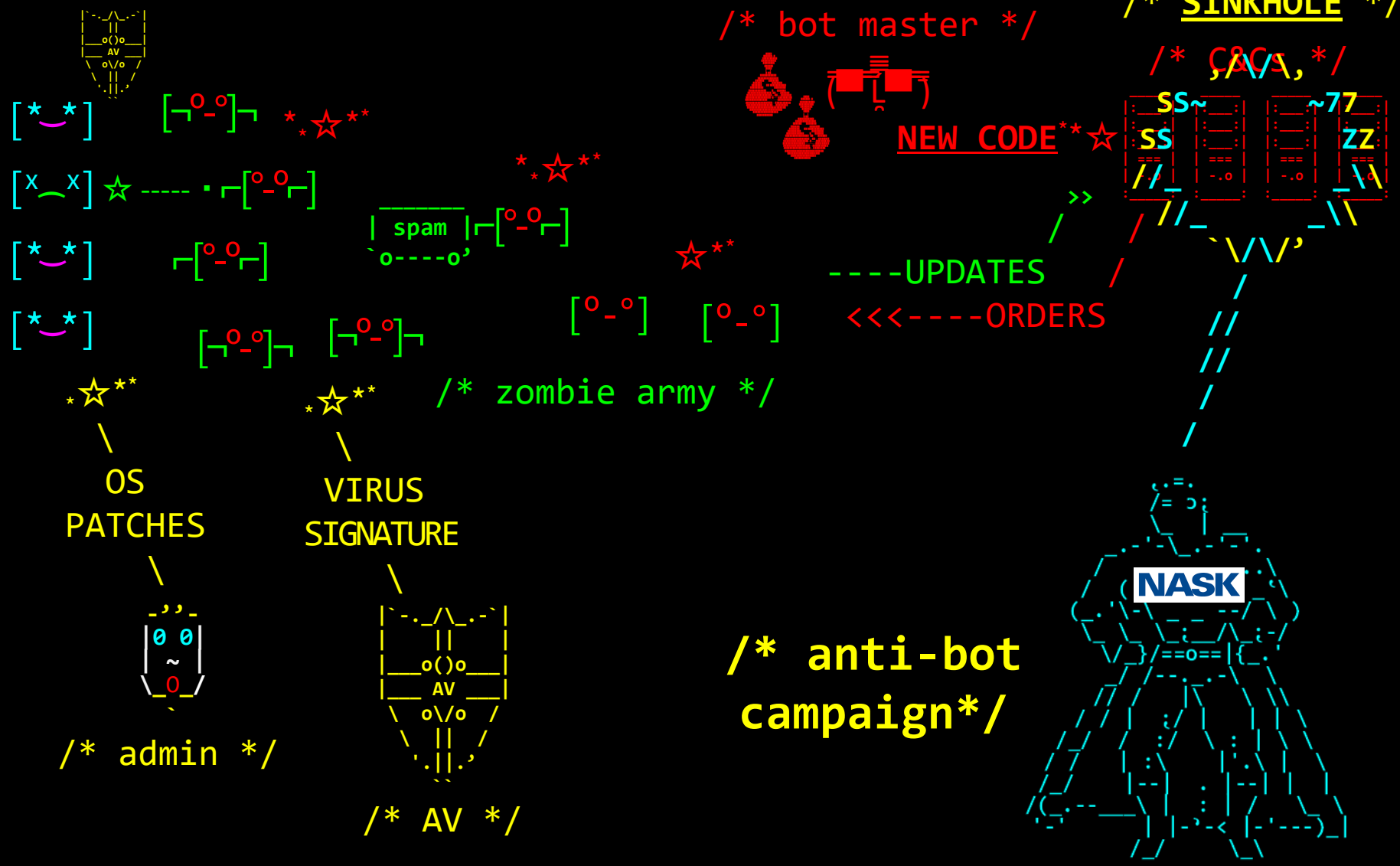
| | nbr of networks | nbr of desktops | nbr of servers |
|---|---|---|---|
| household | 10,453,728 | 1 | 0 |
| micro-enterprise | 2,073,600 | 2 | 0 |
| small enterprise | 52,366 | 21 | 0 |
| medium enterprise | 14,881 | 105 | 0 |
| large enterprise | 3,614 | 858 | 27 |
| laptops | 824,299 | - | - |

(Data source: Polish Office for Statistics, GUS)

# /*** Microsimulation of a botnet in PL network: growth & stabilisation ***/



# /*** Fighting botnets ***/



# /********** Virut **********/

# malware botnet operating at least since 2006

# **the 5th-most widespread threat to the Internet**, responsible for **5.5%** of computer infections (*Kaspersky Security Bulletin 2012):*
  * 2012: 300,000 computers in Egypt, Pakistan, India *(Symantec)*
  * 2013: 890,000 IP addresses in Poland *(CERT Polska)*

# **cybercrime activity**: DDoS attacks, spam, fraud, data theft, and pay-per-install activities

# **spreads via executable file** infection (email attachments, infected USB sticks and other media) and more, recently, via compromised HTML files (vulnerable browsers)

# **disrupted by NASK in January 2013**: takeover of 23 Virut C&Cs in attempt to shut it down **-> sinkholes (key information about botnet dynamics)**

# can’t be shut down completely, as some C&Cs are located at “.ru” domains, i.e. outside the reach of the Polish NASK. There’s a threat that it will reestablish itself (Virut’s alternate backup hosts mechanism)

# /* Sinkhole */

# Eavesdrops messages sent by bots to C&C together with bots' diagnostics (host ID, network type, IP, etc.)


```
{
"origin": "sinkhole",
"restriction": "need-to-know",
"confidence": "high",
"name": "virut",
"category": "bots",
"proto": "tcp",
"time": "2018-04-18T11:59:59Z",
"modified": "2018-04-18T12:00:29Z",
"until": "2018-04-18T16:13:00Z",
"source": "cert-pl.sinkhole",
"address": [
        {
      "cc": "CN",
      "ip": "WW.XXX.YYY.ZZZ",
      "asn": 4134
      }
    ],
"dport": 80,
"rid":
"45997e6910b19e38e443babdbd37a3d0",
"sport": 1592,
"dip": "AAA.BBB.CCC.DDD",
"id":
"e75c853670f6391b36c64e3c272848e9",
"count": 43
}
```

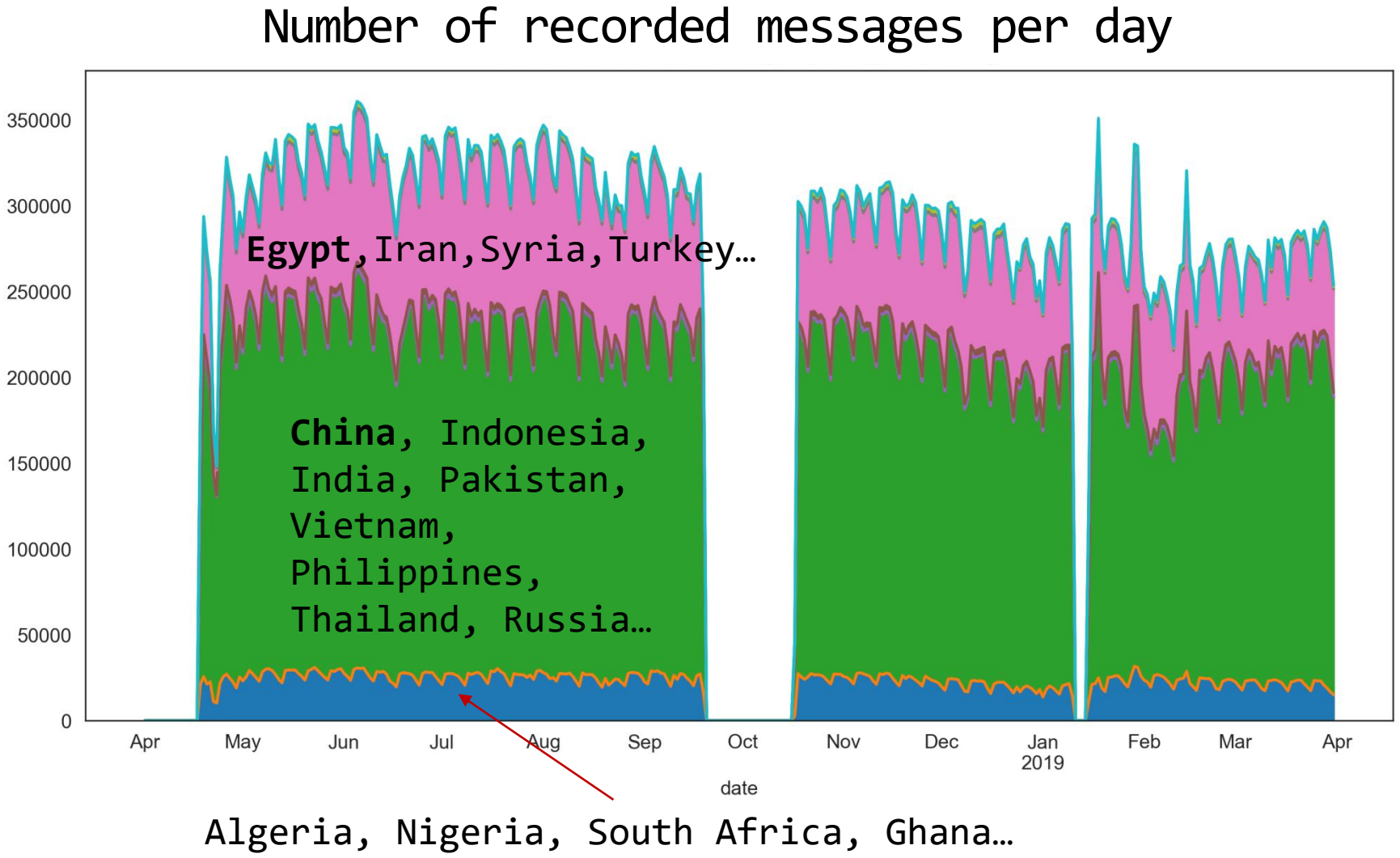

**PROBLEM:** one IP can be assigned to multiple hosts

**/* How many hosts in Polish computer network are infected? */**

CERT.PL

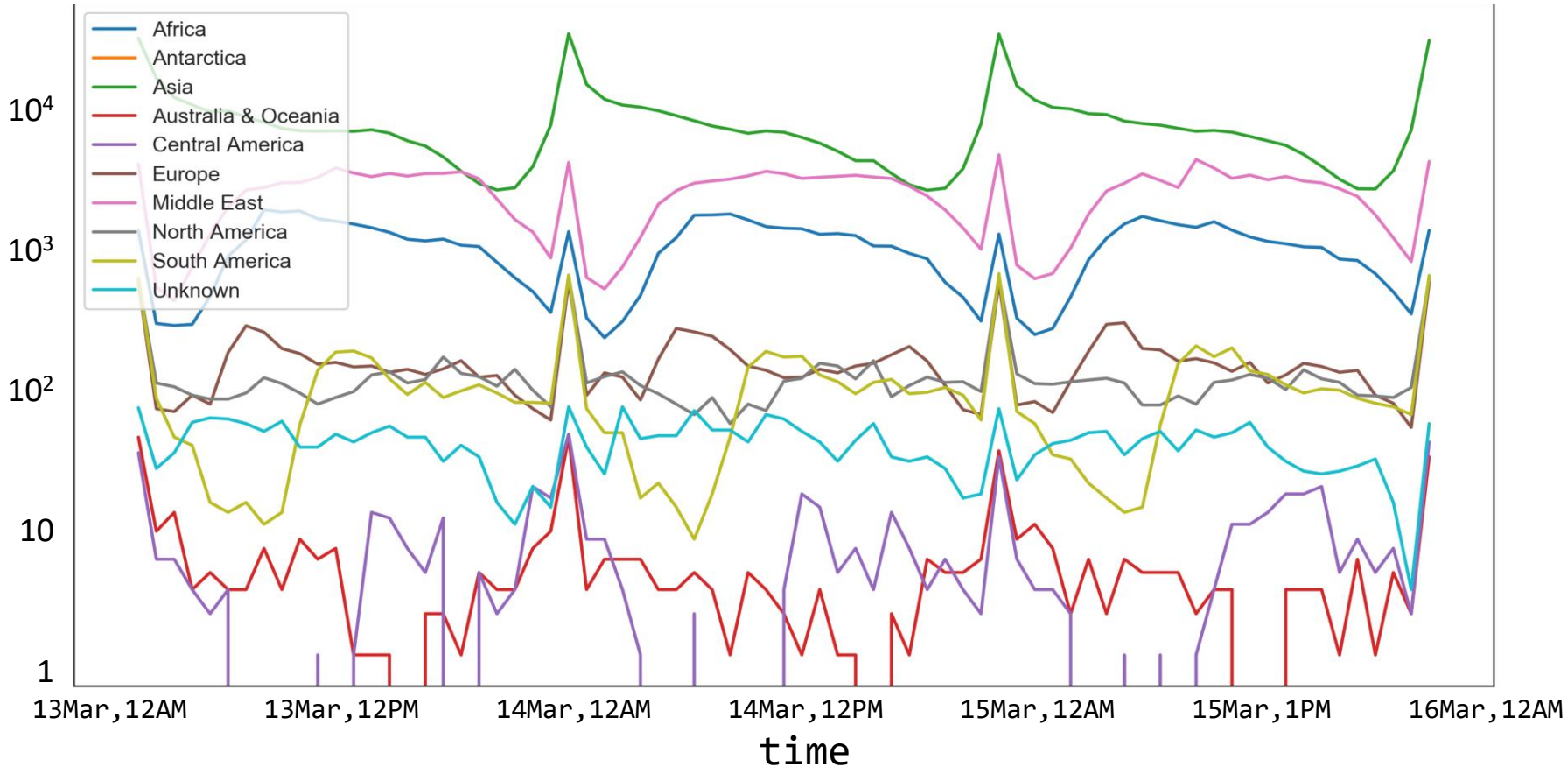
Computer Emergency Response Team

# /* Exploratory data analysis */



# /* Exploratory data analysis */

Number of recorded messages per world region per hour (13-16 Mar 19)




>> Messages from all regions peak every 24h **at midnight GMT**

>> We assume botnet messaging frequency is 24h by **wall clock time**

**EDA can help characterise botnet regime & detect changes**

***Our task*: based on sinkhole data, estimate the size and structure of Virut botnet in Polish Internet**

We know:

- *User behaviour* (studies and data, e.g. times of ad clicks)
- *Polish Internet structure* (government statistics, etc.)
- *AV* efficiency (available benchmarks)
- *Some botnet parameters:*
  - *typical botnet behaviours* (laboratory test)
  - **data from sinkhole**

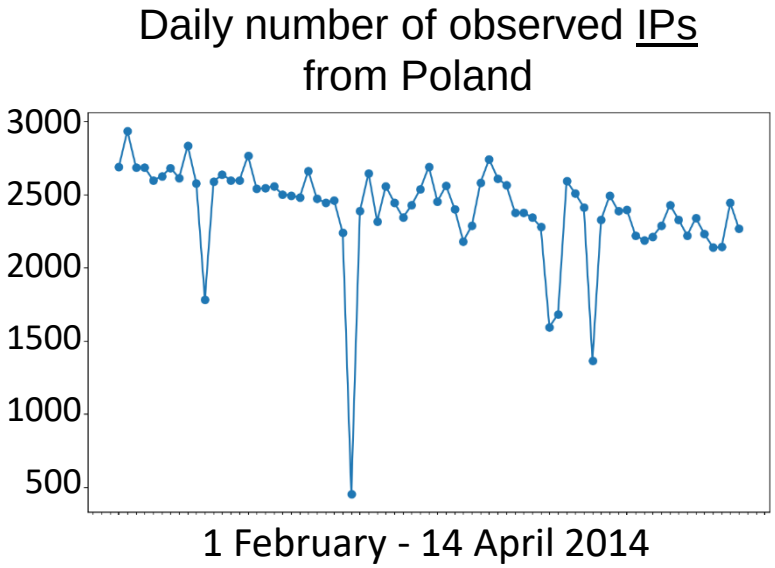
We can build (we already did!): efficient **botnet microsimulation**

***In other words, what parameters of our botnet microsimulation generate a data stream similar to Virut messages eavesdropped at the sinkhole?***

*(prediction: MICROSIMULATIONS -> RESULTS)*
***discovery*: RESULTS & DATA -> MICROSIMULATION PARAMETERS**

***Our task*: based on sinkhole data, estimate the size and structure of Virut botnet in Polish Internet**

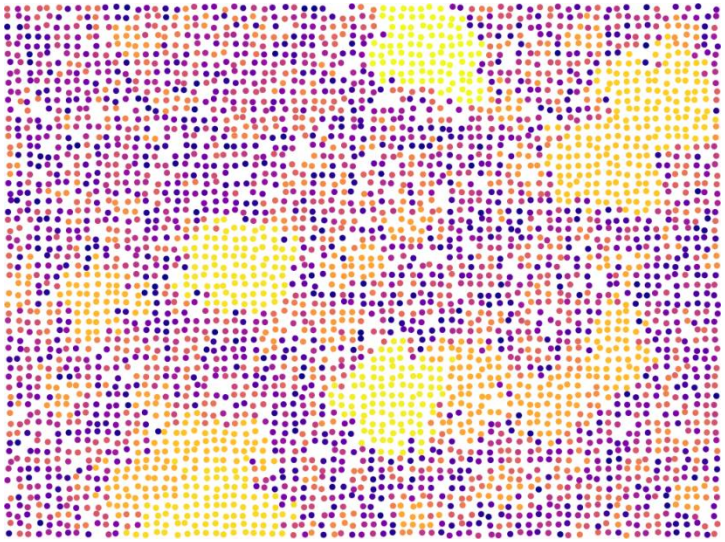
## Sinkhole data




```
/* Large (GBs) stream of data
from several years - and coming*/
```

## Polish Internet

Source: GUS

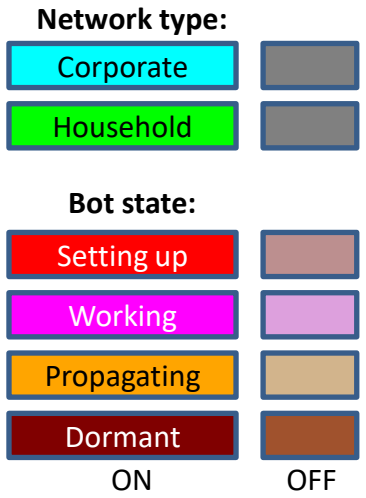


1:10

***How many hosts in Polish network are infected?***

# /* Microsimulation of a botnet infecting the Polish network */
## (vs AV and OS new releases)

Watch the movie at
https://youtu.be/yktpbrCjuy4



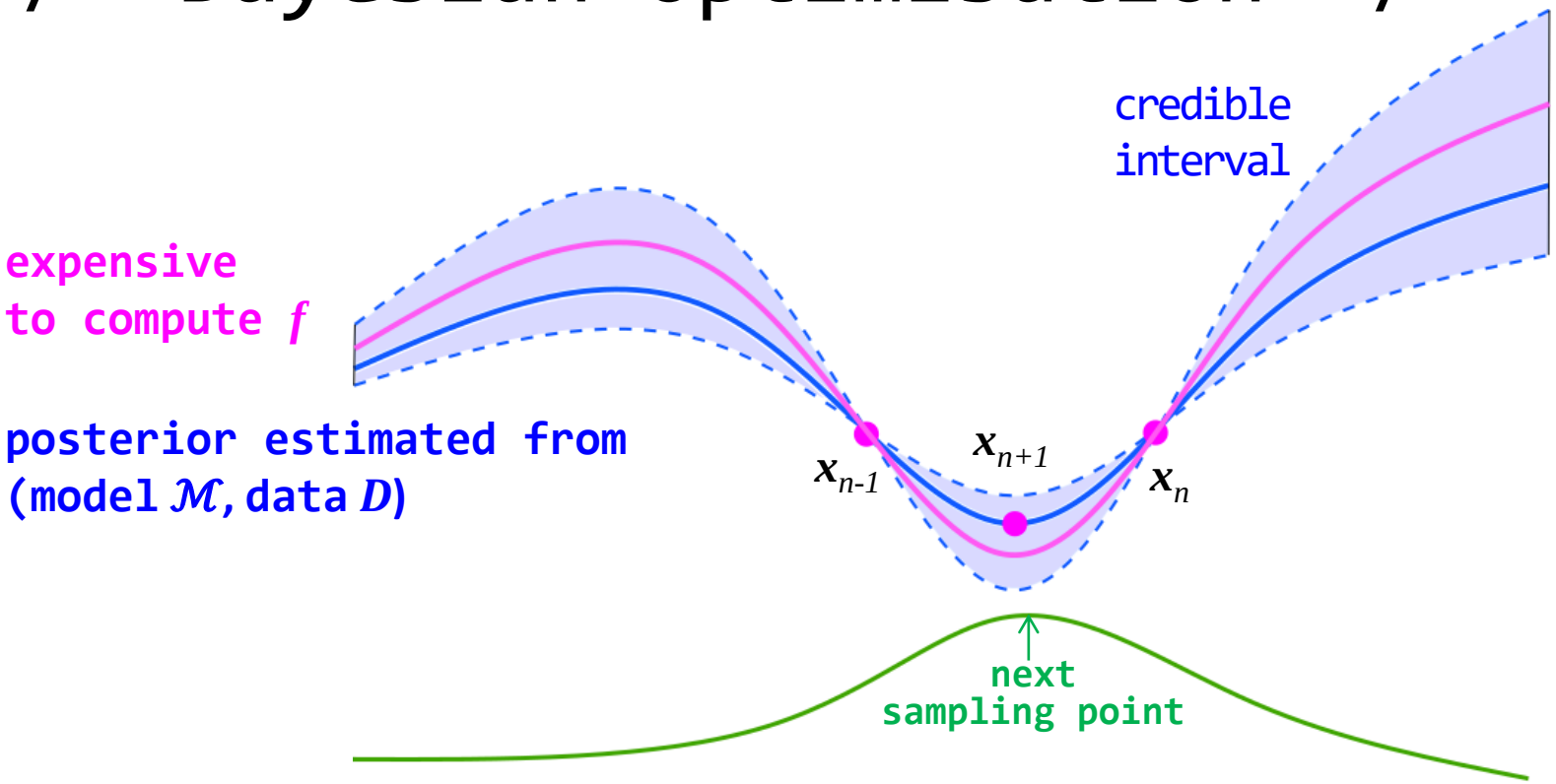
***How to guess what parameters of our microsimulation of a botnet in Polish Internet will generate a data stream similar to real-world Virut messages eavesdropped at the sinkhole?***

Solution: search the whole space of parameters and choose a set of their values which generates results similar to real-world data.

Problem: the space of those parameters is very high-dimensional and "complicated". Finding those parameters using standard methods is unfeasible given nowadays computational powers.

**That's why we will use Bayesian optimisation!**

# /* Bayesian optimisation */




• **Function $f$** (microsimulation calibration error) is computed slowly. The best we can do is to find its **statistical model $\mathcal{M}$** based on small number of samples.

• We sample single points from $f$ and fit $\mathcal{M}$ to it, until we achieve a satisfying approximation (credible interval): a heuristic called acquisition function tells us where to sample to improve the result (*exploitation*); sometimes we sample randomly to avoid getting stuck in a local minimum (*exploration*)

• We obtain a Bayesian model of the function based on observed data

# /* Bayesian optimisation */

# We want to solve

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}\in X} f(\boldsymbol{x})$$

where $f$ is the error between microsimulation results and data.

# Parameter space $X$ is high-dimensional and complicated (discrete & continuous parameters)

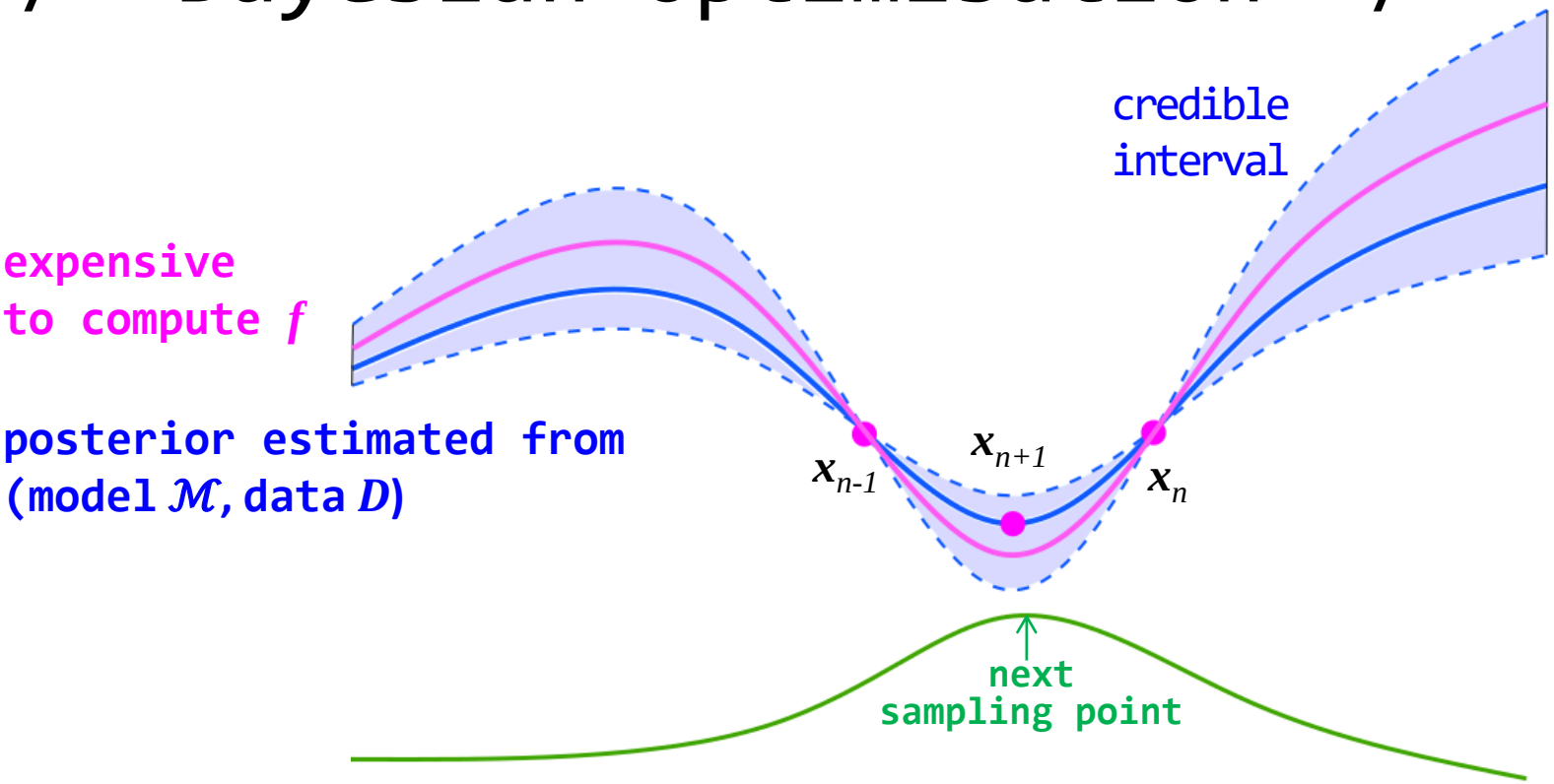
# Computing $f$ is slow

# $f$ is non-smooth -> large system simulation needed

# Traditional optimisation algorithms fail

**# Bayesian optimisation can help!**

# /* Bayesian optimisation */




# Build a statistical model $\mathcal{M}$ of $f$ based on the data $D_n=\{(\boldsymbol{x}_i, f(\boldsymbol{x}_i)), i=1, \ldots, n\}$ observed so far

# Using the acquisition function $\alpha$, choose the next sampling point $\boldsymbol{x}_{n+1}=\arg\max_{\boldsymbol{x}\in X}[\alpha(\boldsymbol{x},D_n)]$

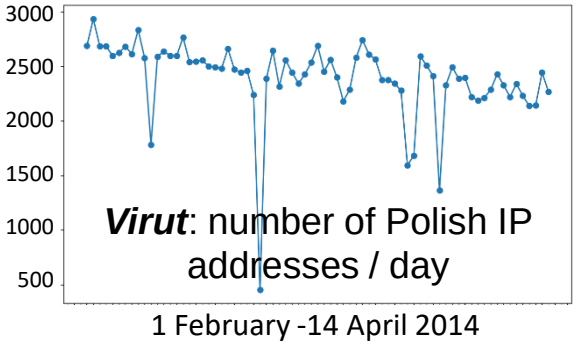
# Compute a new value $y_{n+1}=f(\boldsymbol{x}_{n+1})$ and update the data $D_{n+1}=(D_n,(\boldsymbol{x}_{n+1},y_{n+1}))$

# Update model $\mathcal{M}$ using $D_{n+1}$

# Repeat until convergence

# /* Microsimulation structure summary */

**Infection**




- infection vector: email with attached dropper
  - defence: antispam filter → antivirus heuristics → user caution
- infection: user opens the email and clicks on the attachment → payload
  - defence: antivirus signature, safe OS

(risks of opening the email and the attachment decrease in time)

- after the payload downloads and installs itself, the host joins the botnet
- botmaster can update the payload version to bypass the AV defences (signature)

**Botnet**

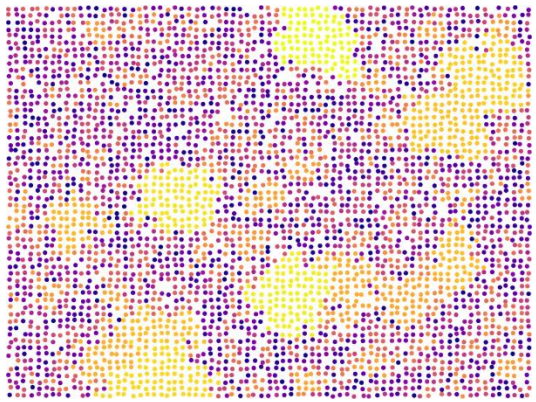
- sends spam with droppers (botnet campaign)
- periodically contacts C&C, reporting its work or requesting new instructions: PROPAGATING, WORKING, DORMANT

**Removing the infection and patching security holes**

- installing and regularly updating AV, performing regular scans of the hosts
- types, versions and releases of OS's regularly updated by system admins

# /* Microsimulation structure summary */

- One **ISP** and faultless **Internet connection**
- **Types of networks / user environments**:
  - HOME (dynamic IP)
  - OFFICE (static IP)
- **Hosts** (different risks of infection)
  - „desktop”: DESKTOP, SERVER (no risk)
  - „mobile”: LAPTOP

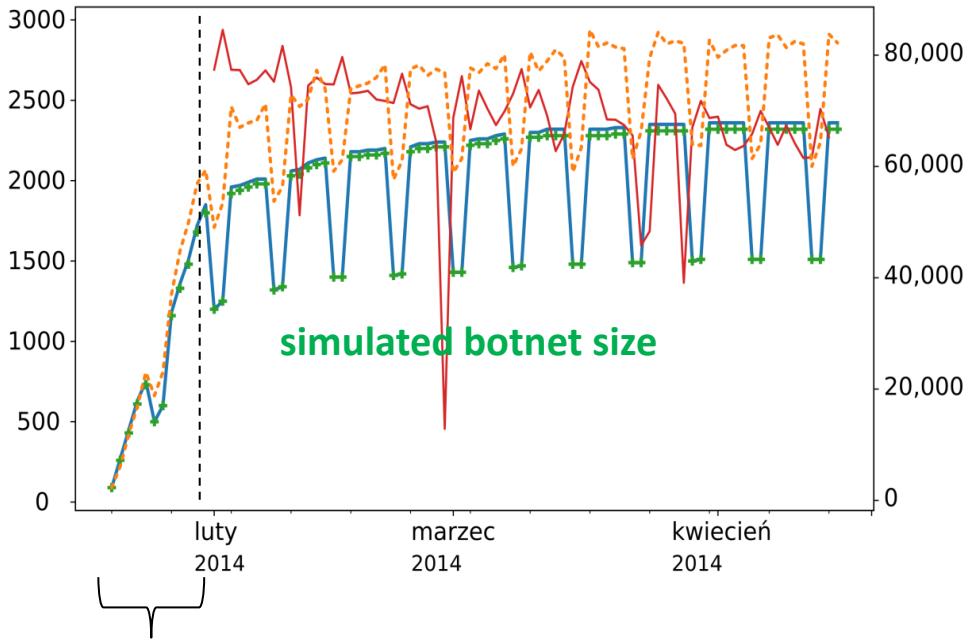


- **Users** have different schedules of using home and office computers; some computers are shared (e.g. by household members) or moved between office and home networks (laptops).
- Every network has its **administrator**. Static hosts in a network share the same system administrator. Laptop users act as their sysadmins.
- **Email accounts** have many providers (one provider in an office network). Providers can’t filter out all spam. Each user has one or two email accounts (personal and professional - if employed) .

# /* Parameters of Virut in Polish network */

## Mictosimulation and Bayesian optimisation results

Number of IPs per day:
observed≈simulated




Simulated number of messages send to sinkhole per day

- Number of emails with droppers sent by C&C to Polish Internet hosts during an 11-day botnet promotion campaign: **0.79/s**

- Estimated botnet size in April 2014: **2320**

**Mind that this result is valid only for Virut as it evolved in according to the date – drawing general conclusions about botnet size and parameters based on a single simulation result is incorrect.**

# /**** Summary ****/

# We created tools for microsimulations of realistic country-wide computer networks and their infections: https://github.com/rilwen/botnet

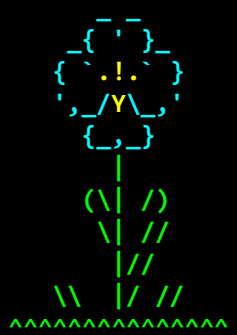
# We successfully employed Bayesian optimisation methods to discover the parameters of a botnet based on real-world data

# Using the above we can perform quantitative estimation of cybersecurity threats and design effective interventions

# Case study: discovering the size and other parameters of Virut botnet in Polish computer network based on sinkhole data – this result doesn't generalise to other instances of Virut or other botnets

# Modelling cybersecurity threats can be improved by providing more data and detailed information: behavioural studies on user practices, structure of computer networks, efficiency of defences, etc.

7h4nk u 4 yr 4773nt10n



AVERISERA
MATHEMATICAL INVESTIGATIONS OF REALITY

NASK