ICASSP2021
TORONTO
Canada
June 6-11, 2021
Metro Toronto Convention Centre

.DIEM

Application-Layer DDoS Attacks with Multiple Emulation Dictionaries

- Michele Cirillo
- Mario Di Mauro
- Vincenzo Matta
- Marco Tambasco

Department of Information and Electrical Engineering and Applied Mathematics (DIEM)
University of Salerno (Italy)

Problem and Motivation (1/2)

Distributed DoS attack (**DDoS**): huge number of apparently innocuous requests produced in parallel by a net of robots (*botnet*) in order to saturate the resources of a target site

Classic DDoS attacks: typically implemented at the **transport layer** of the TCP/IP protocol stack, where some flags are maliciously manipulated (e.g., SYN flood attacks [1])

Application-layer DDoS (or L7-DDoS): a particularly dangerous DDoS variant where the botnet floods a target by emulating ordinary web surfing (e.g., HTTP requests)

Useful taxonomy of L7-DDoS attacks [2]:

- **Asymmetric Workload attacks:** an attacker sends few requests that require computationally expensive responses from the server (e.g. SQL-based queries)
- **Request Flooding attacks:** the attack flows exhibit a request rate higher than legitimate traffic

[1] C. C. Zhou, N. Duffield, D. Towsley, and W. Gong, "Adaptive defense against various network attacks,'' IEEE Journal on Selected Areas in Communications, vol. 24, no.10, pp. 1877-1888, Oct. 2006.

[2] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, and E. Knightly, "DDoS-Shield: DDoS-resilient scheduling to counter application layer attacks,'' IEEE/ACM Transactions on Networking, vol. 17, no.1, pp. 26-39, Jul. 2008.

Problem and Motivation (1/2)

Distributed DoS attack (**DDoS**): huge number of apparently innocuous requests produced in parallel by a net of robots (*botnet*) in order to saturate the resources of a target site

Classic DDoS atta TCP/IP protocol stack, where some [1])

Application-layer nt where the botnet floods a tar)

Useful taxonomy

- **Asymmetric** computationally expensive re
- **Request Floo** han legitimate traffic

«The HTTP attack included in Mirai is a highly flexible attack with several customizations that could prove difficult to defend against without the right tools»

R. Winward, "IoT Attack Handbook," https://www.radware.com/iot-attack-ebook/

[1] C. C. Zhou, N. Duffield, D. Towsley, and W. Gong, "Adaptive defense against various network attacks,'' IEEE Journal on Selected Areas in Communications, vol. 24, no.10, pp. 1877-1888, Oct. 2006.

[2] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, and E. Knightly, "DDoS-Shield: DDoS-resilient scheduling to counter application layer attacks,'' IEEE/ACM Transactions on Networking, vol. 17, no.1, pp. 26-39, Jul. 2008.

Problem and Motivation (2/2)

Starting point: a sophisticated variant of *Request Flooding Attacks* introduced in [3]

- To evade detection, the botnet emulates regular traffic patterns by gleaning admissible messages from an **emulation dictionary**
- The emulation dictionary is learnt by the botnet continually (i.e. the dictionary cardinality increases with time) to sustain a reasonable **degree of innovation**

Current approaches

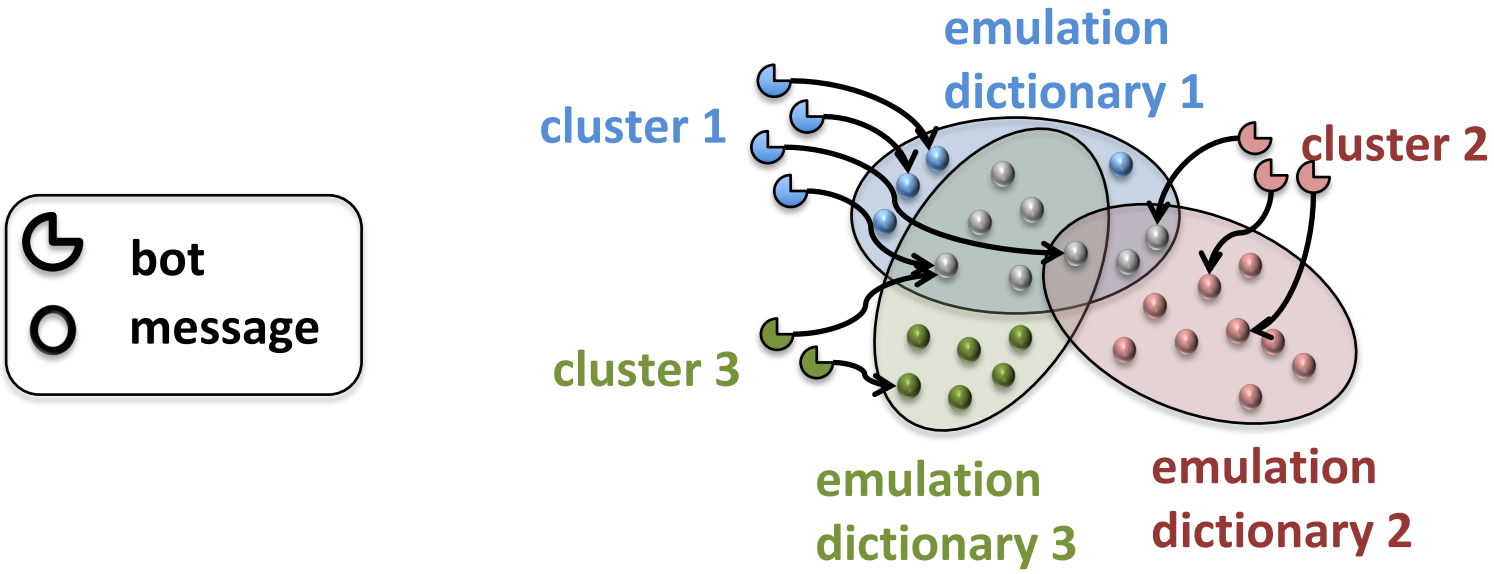
- Classic entropy-based approaches [4] are ineffective to contrast such attacks, since the **individual** traffic activity is not suspicious
- A solution that exploits dependencies across nodes is proposed in [3], under the assumption that all bots use the **same** emulation dictionary

[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

[4] Y. Xiang, K. Li, and W. Zhou, "Low-rate DDoS attacks detection and traceback by using new information metrics,'' IEEE Transactions on Information Forensics and Security, vol. 6, no.2, pp. 426-437, Jun. 2011.

Main Novelty of This Work

- Bots are often organized in **clusters** (governed by Command & Control servers [5])
- Different clusters use different emulation dictionaries exhibiting a certain diversity, but also some commonalities → multiple overlapped emulation dictionaries



[5] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," Computer, vol. 50, no.7, pp. 80-84, Jul. 2017.

Useful Network Indicators (1/2)

At time t, bots belonging to cluster i pick legitimate messages from the emulation dictionary $\mathscr{E}_i(t)$

Emulation Dictionary Rate (EDR) $\alpha \triangleq \lim_{t\to\infty} \frac{|\mathscr{E}_i(t)|}{t}, \quad i = 1, 2, \ldots, C$

Activity carried by a generic subnet $\mathcal{S}$

message content immaterial

Empirical Transmission Rate $\widehat{\lambda}_{\mathcal{S}}(t) \triangleq \frac{N_{\mathcal{S}}(t)}{t}$

Empirical Message Innovation Rate (MIR) $\widehat{\rho}_{\mathcal{S}}(t) \triangleq \frac{|\mathscr{D}_{\mathcal{S}}(t)|}{t}$

$N_{\mathcal{S}}(t) \triangleq$ no. of transmissions measured up to time t in a given subnet $\mathcal{S}$

$\mathscr{D}_{\mathcal{S}}(t) \triangleq$ empirical dictionary of distinct messages sent up to t by users within $\mathcal{S}$

Mario Di Mauro (mdimauro@unisa.it)

Useful Network Indicators (2/2)

Intuition: the message innovation rate of a botnet is smaller than the message innovation rate of normal users

Empirical Message Innovation Rate (MIR) ⇢ $\widehat{\rho}_S(t) \triangleq \frac{|\mathscr{D}_S(t)|}{t}$

Mario Di Mauro (mdimauro@unisa.it)

.DIEM

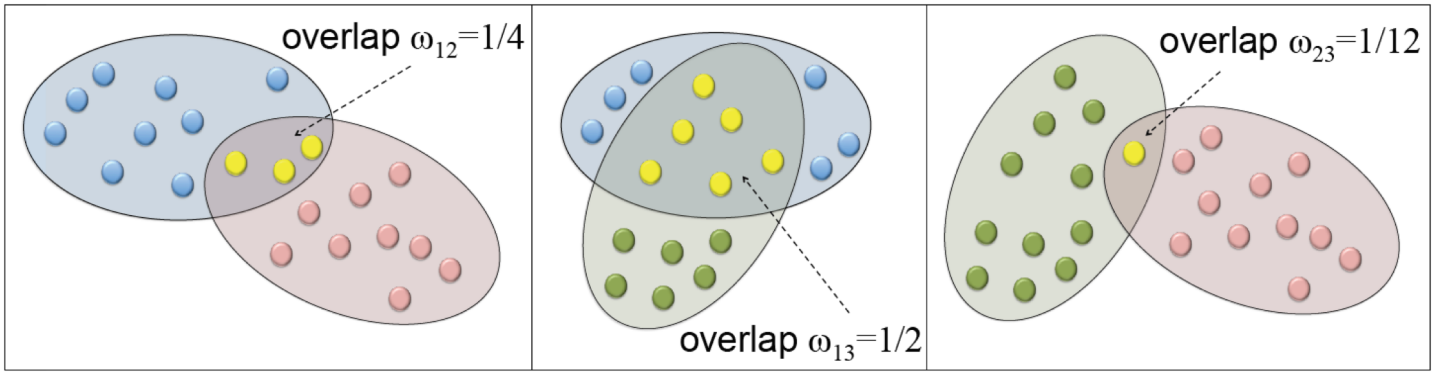
Cluster Interaction (1/2)

Intersection of the emulation dictionaries of clusters i and j

$$\mathscr{E}_{ij}(t) \triangleq \mathscr{E}_i(t) \cap \mathscr{E}_j(t)$$

Overlap degree of the emulation dictionaries of clusters i and j

$$\lim_{t\to\infty} \frac{|\mathscr{E}_{ij}(t)|}{|\mathscr{E}_i(t)|} = \lim_{t\to\infty} \frac{|\mathscr{E}_{ij}(t)|}{|\mathscr{E}_j(t)|} = \omega_{ij} = \omega_{ji} \in (0,1)$$



Cluster Interaction (2/2)

MIR of a botnet using one and the same emulation dictionary:

$$\mathscr{R}(\alpha,\lambda) \triangleq \frac{\alpha\lambda}{\alpha+\lambda}$$

MIR of the joint subnet $\mathcal{B}_i \cup \mathcal{B}_j$:

$$\hat{\rho}_{\mathcal{B}_i\cup\mathcal{B}_j}(t) \xrightarrow{\text{m.s.}} \rho_{\mathcal{B}_i\cup\mathcal{B}_j} = \omega_{ij}\mathscr{R}(\alpha,\lambda_{\mathcal{B}_i}+\lambda_{\mathcal{B}_j}) + (1-\omega_{ij})[\mathscr{R}(\alpha,\lambda_{\mathcal{B}_i})+\mathscr{R}(\alpha,\lambda_{\mathcal{B}_j})]$$

more innovative than

less innovative than

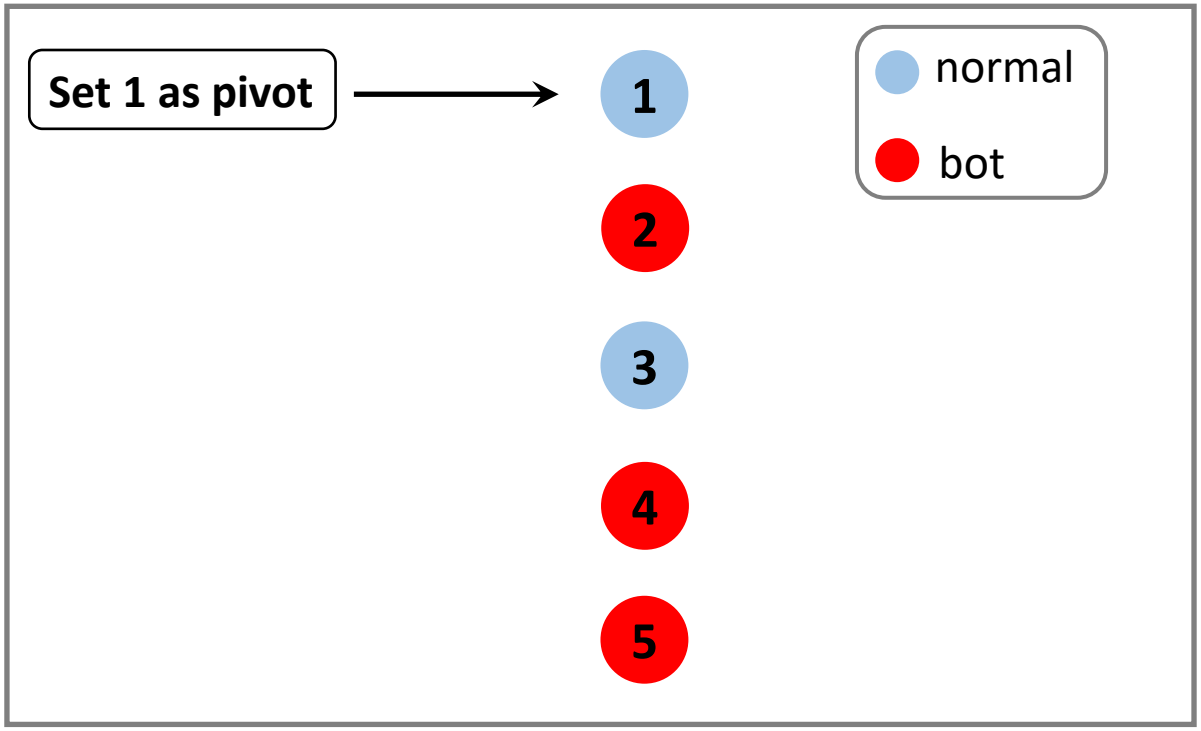
$$\rho_{\text{tot}} = \mathscr{R}(\alpha,\lambda_{\mathcal{B}_i}+\lambda_{\mathcal{B}_j}) \leq \rho_{\mathcal{B}_i\cup\mathcal{B}_j} \leq \mathscr{R}(\alpha,\lambda_{\mathcal{B}_i})+\mathscr{R}(\alpha,\lambda_{\mathcal{B}_j}) = \rho_{\text{sum}}$$

total overlap ($\omega_{ij}=1$)

no overlap ($\omega_{ij}=0$)

Botnet Identification Algorithm

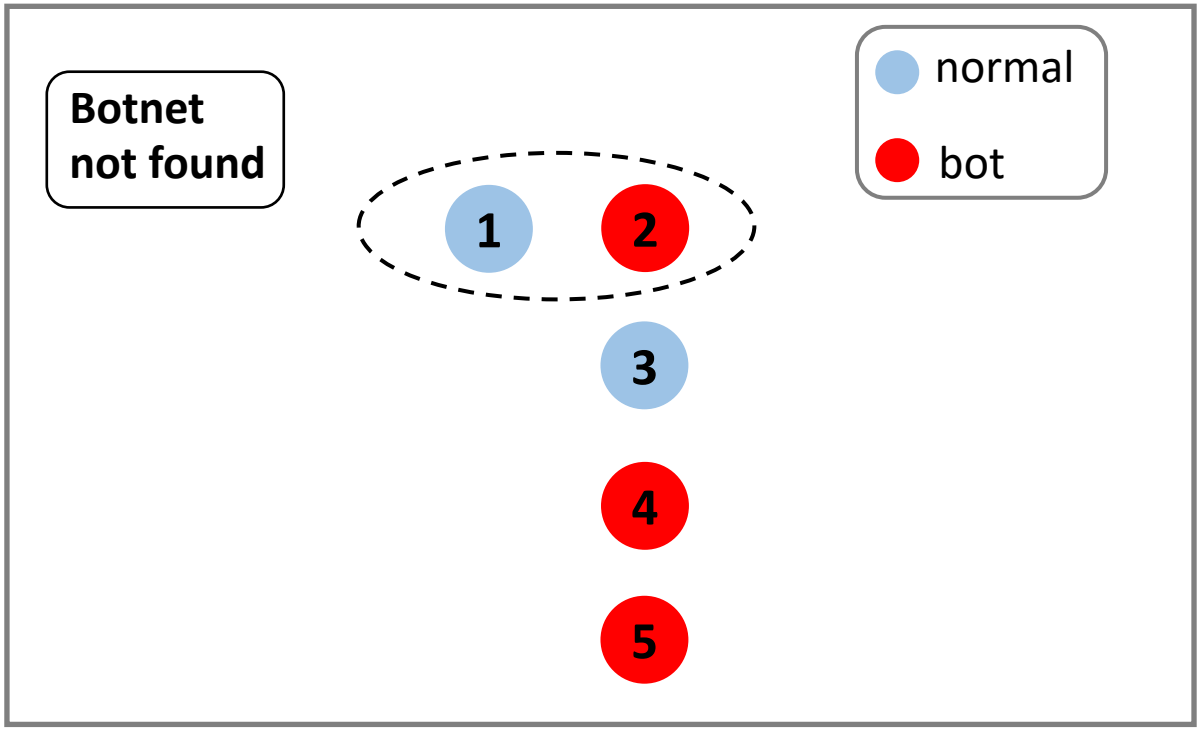
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

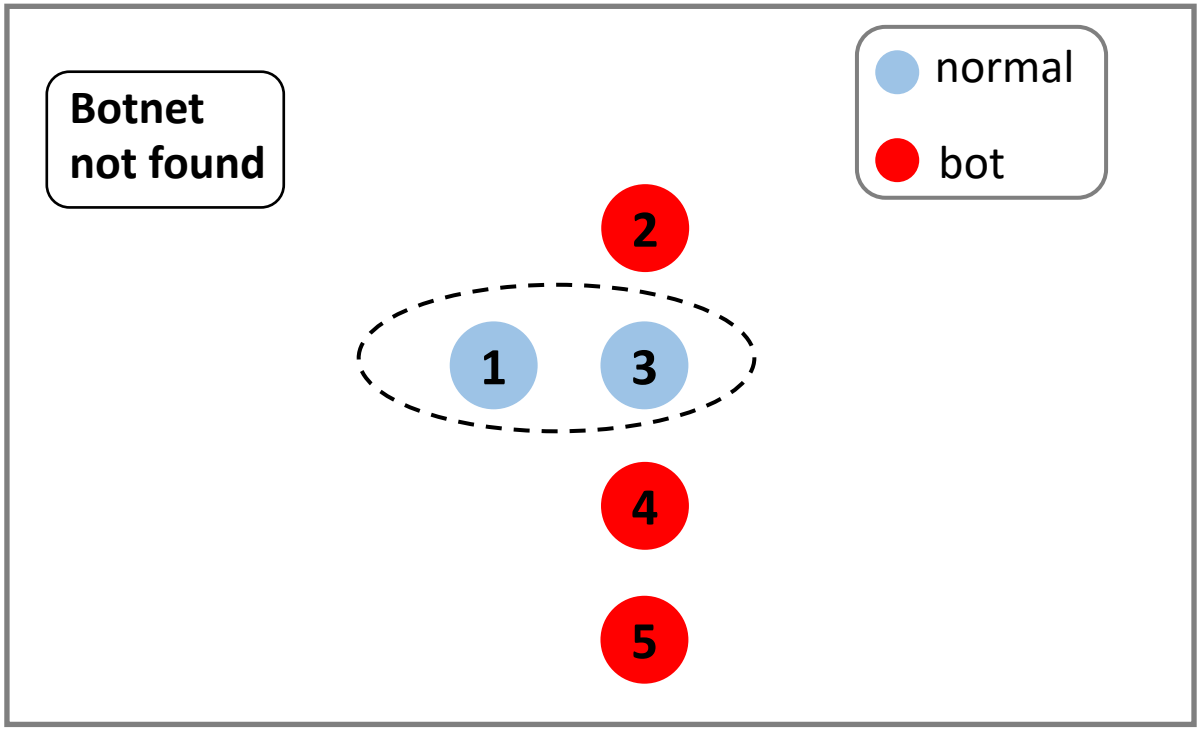
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

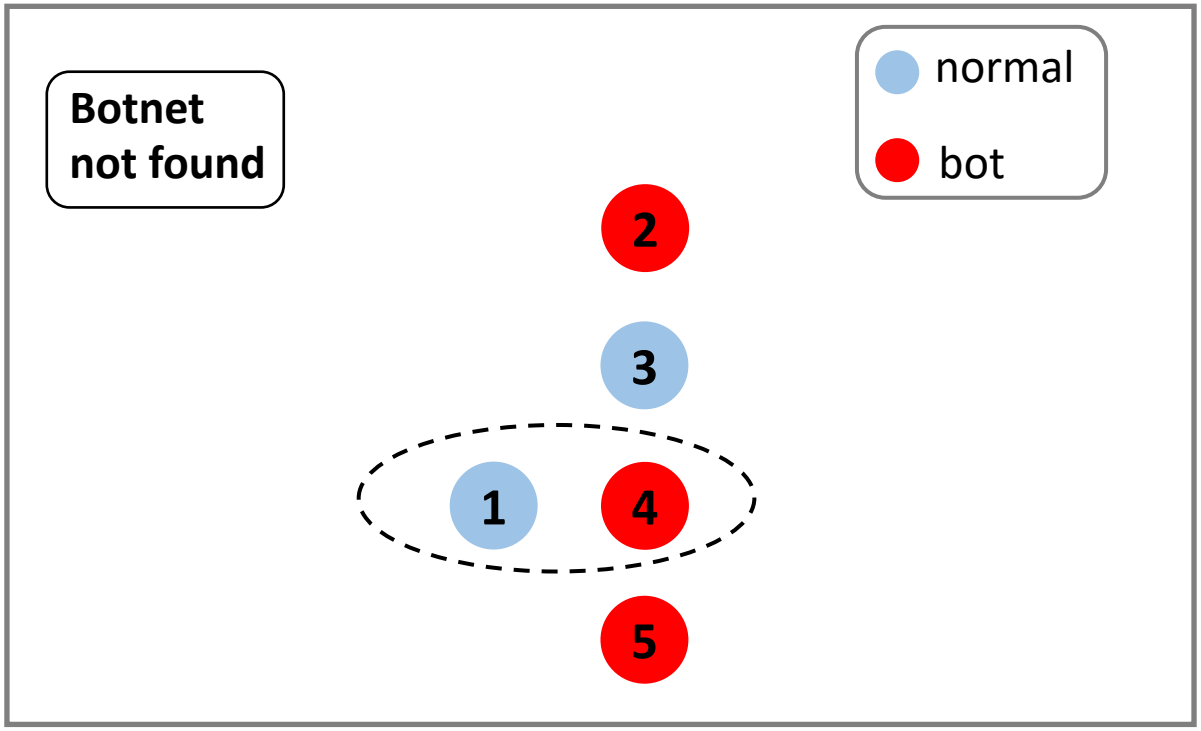
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

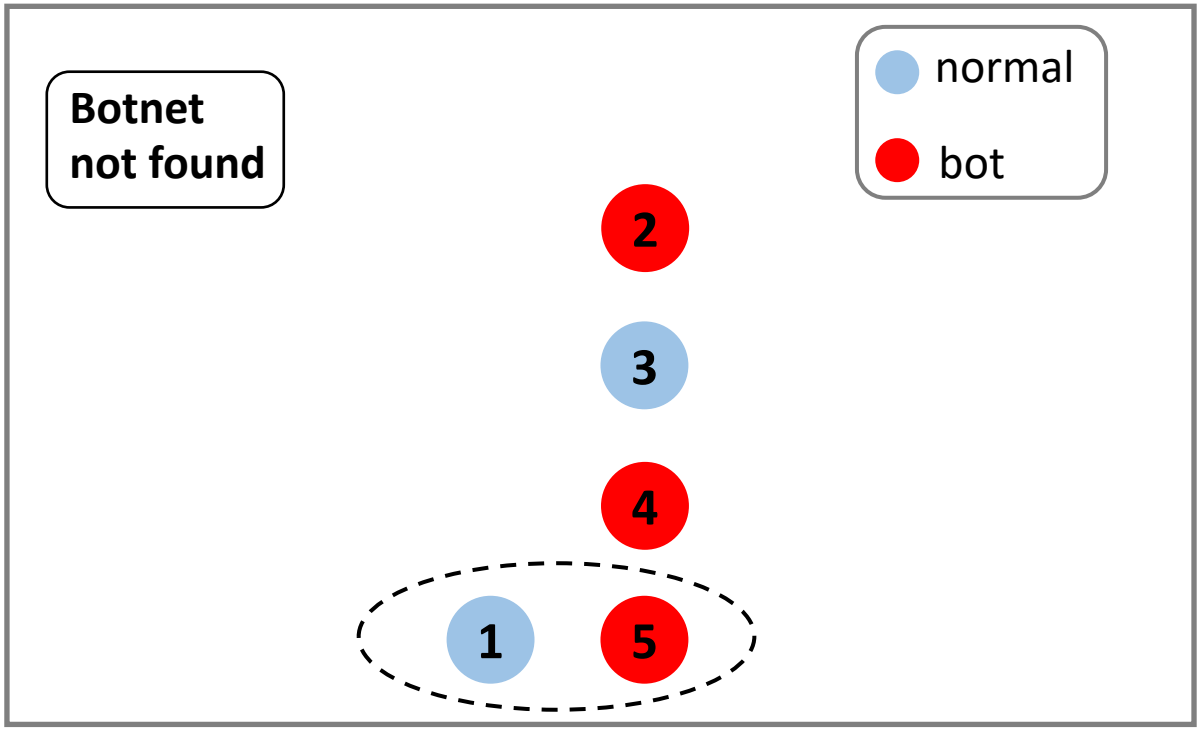
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

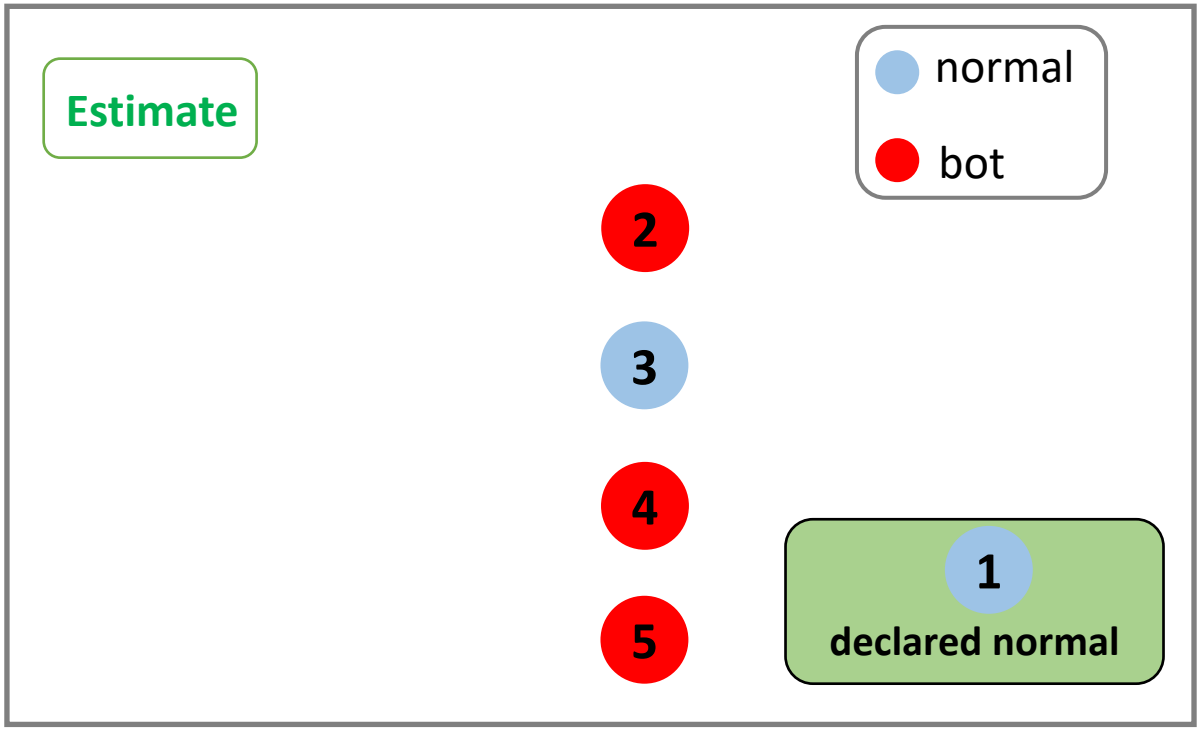
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

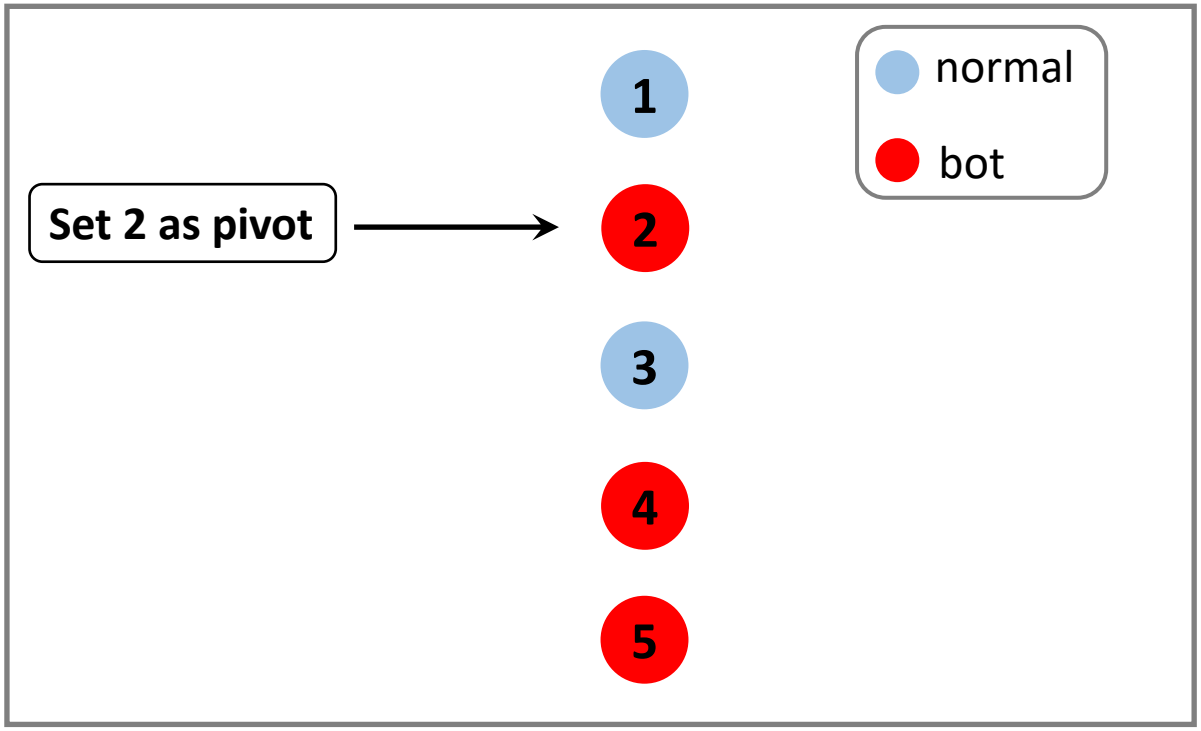
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

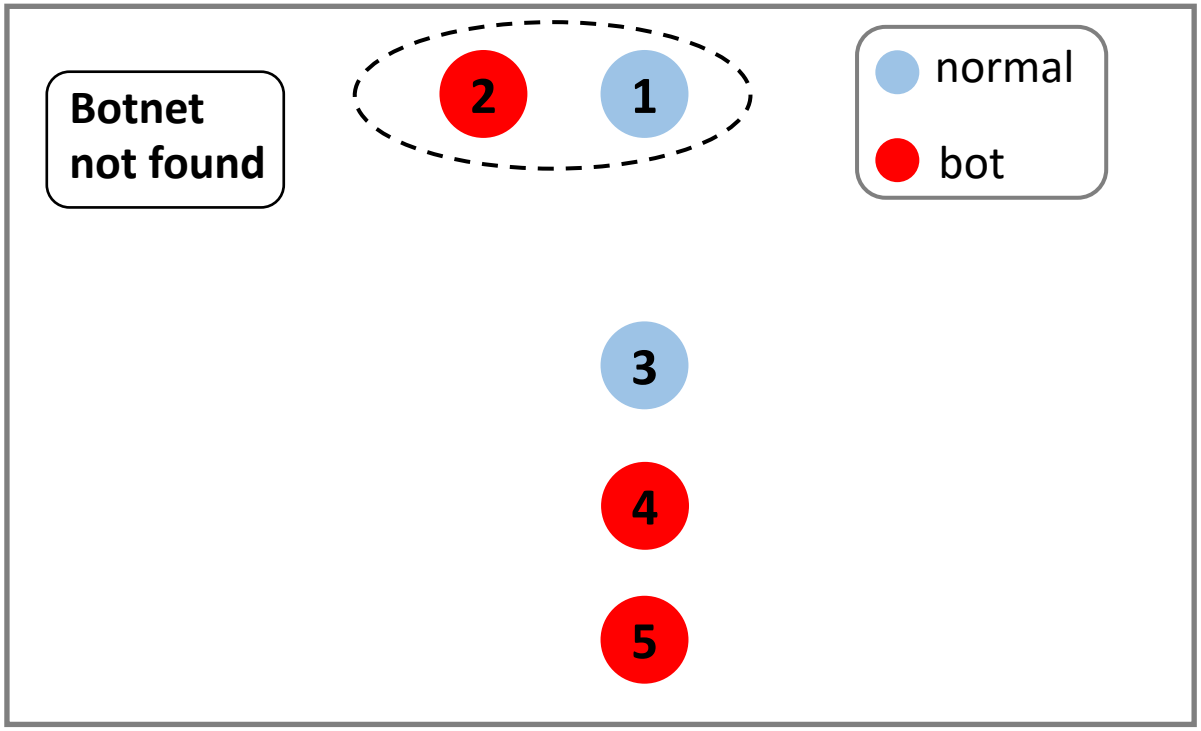
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

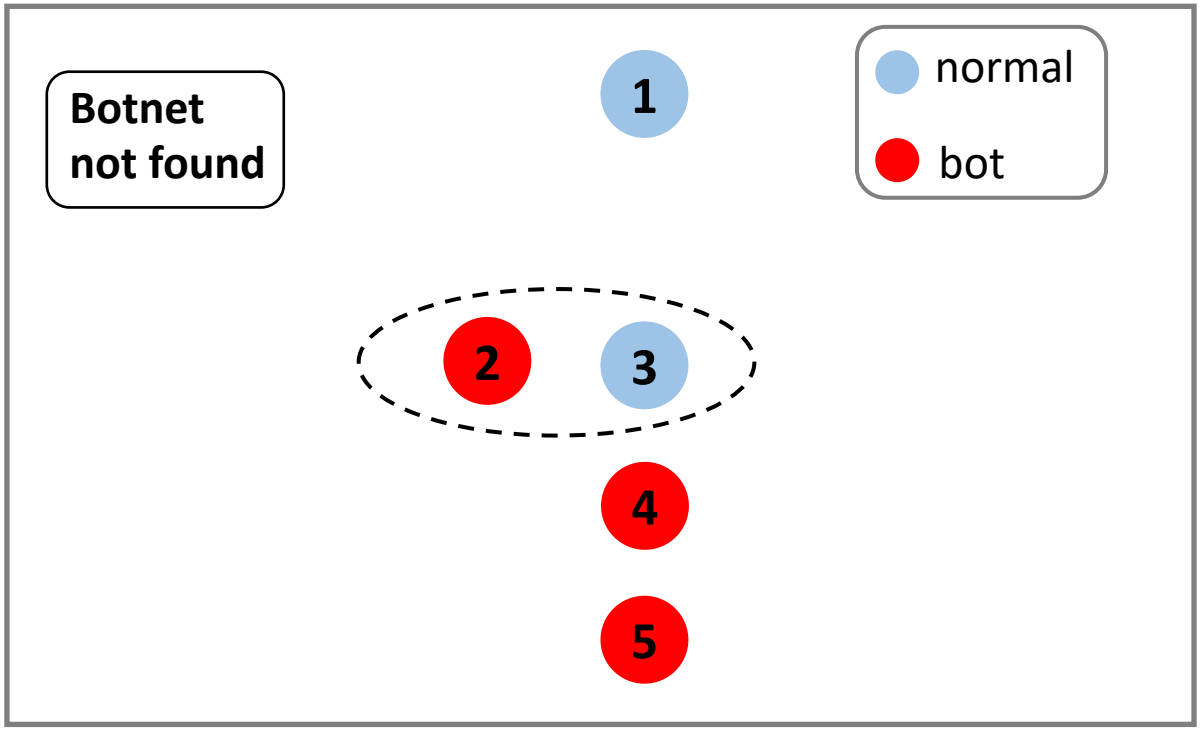
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

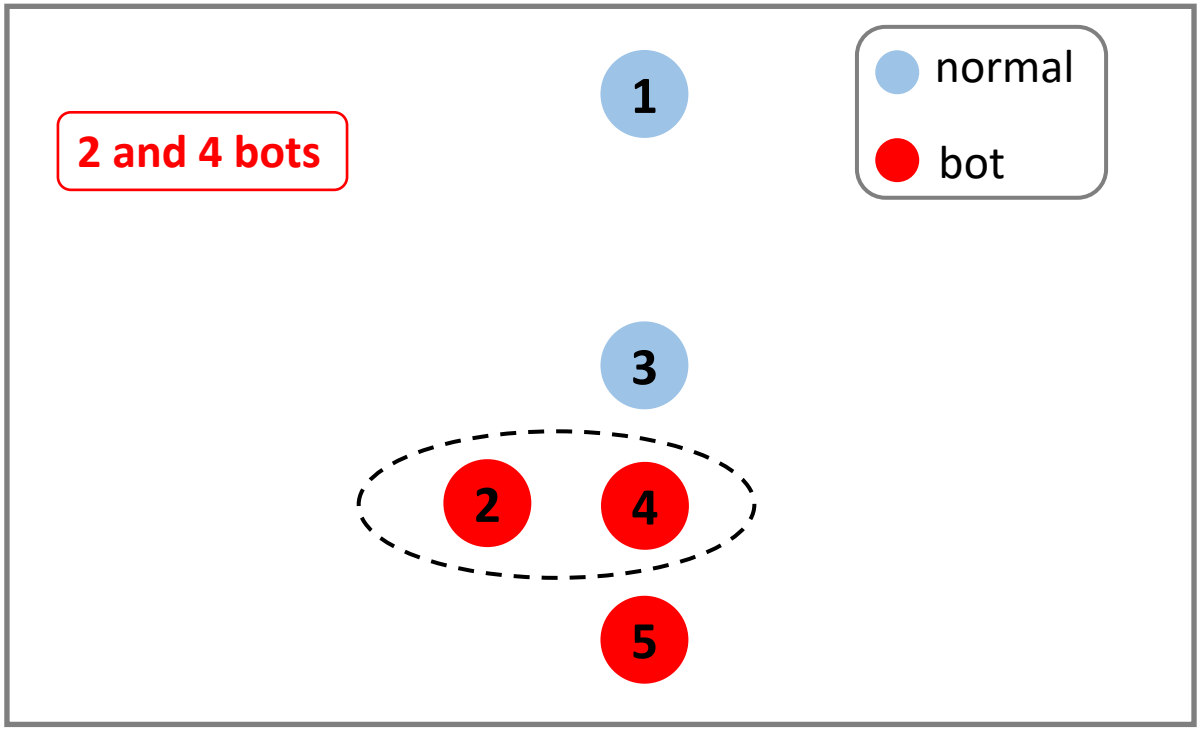
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

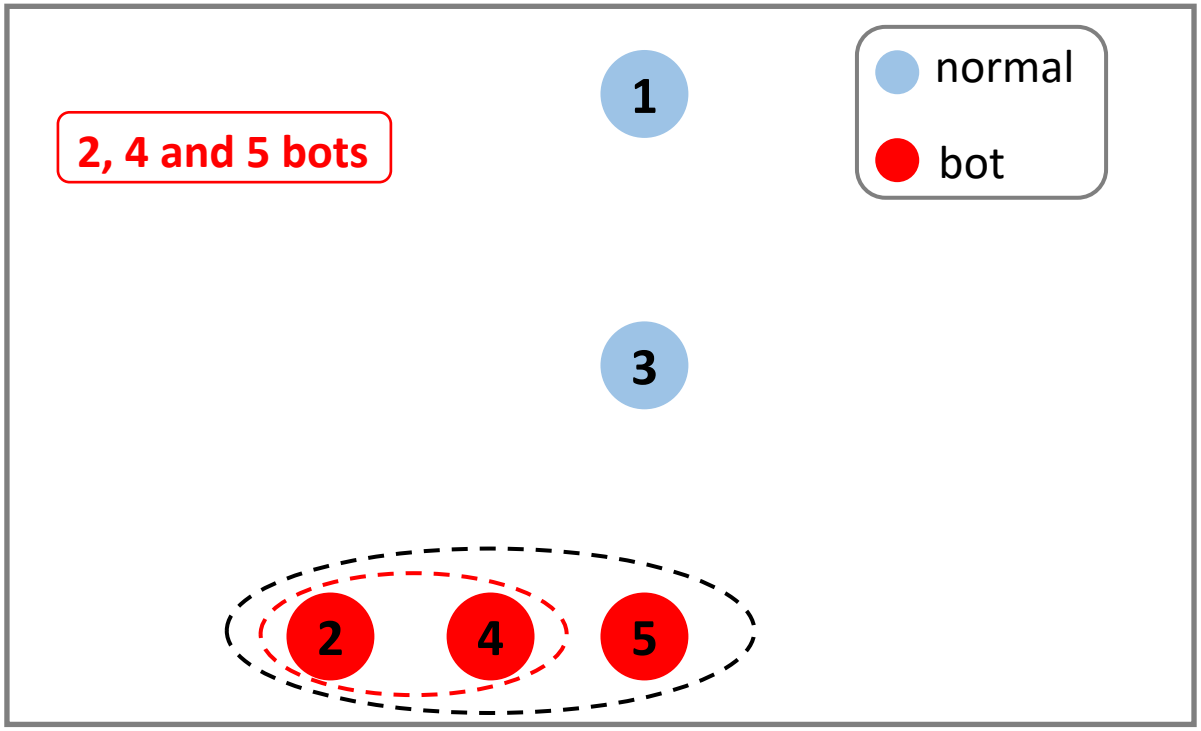
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

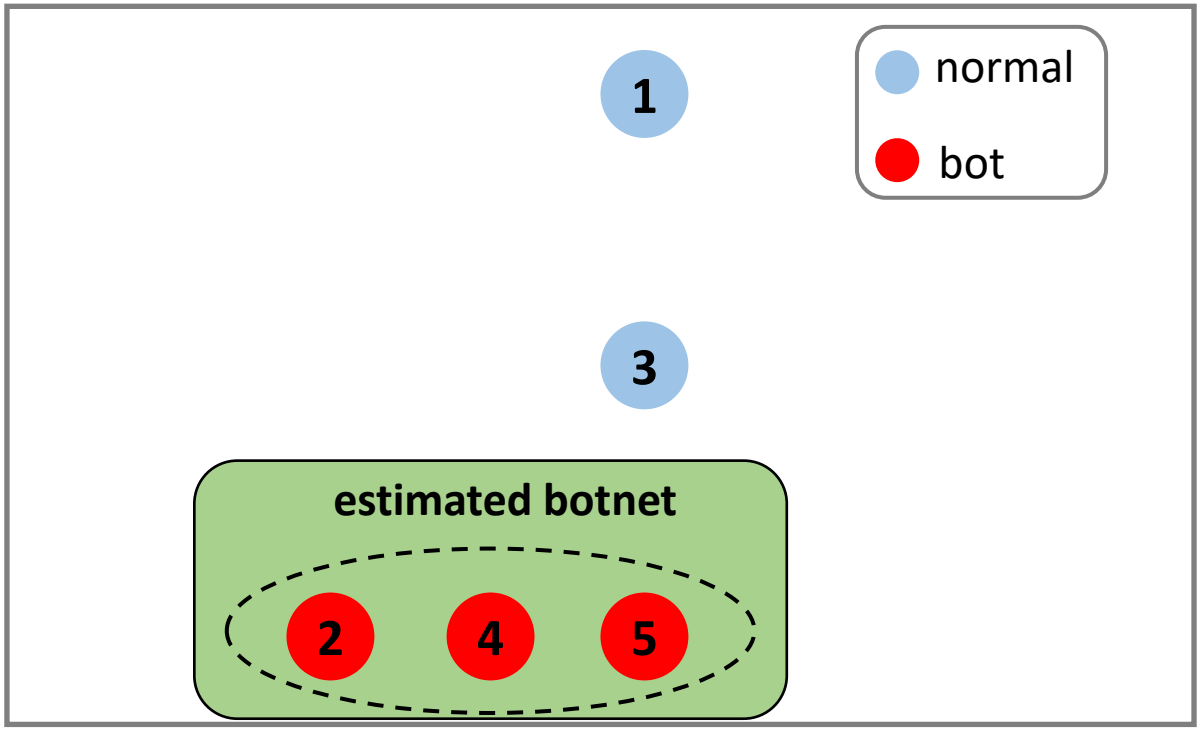
- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

Botnet Identification Algorithm

- The algorithm **BotClusterBuster** devised in this work inherits the pairwise check mechanism adopted in the BotBuster algorithm [3]



[3] V. Matta, M. Di Mauro, and M. Longo, "DDoS attacks with randomized traffic innovation: Botnet identification challenges and strategies,'' IEEE Transactions on Information Forensics and Security, vol. 12, no.8, pp. 1844-1859, Apr. 2017.

BotClusterBuster

In the new multi-cluster scenario, the pairwise checks involve nodes that can belong to:

- the same cluster
- different clusters

Main Question

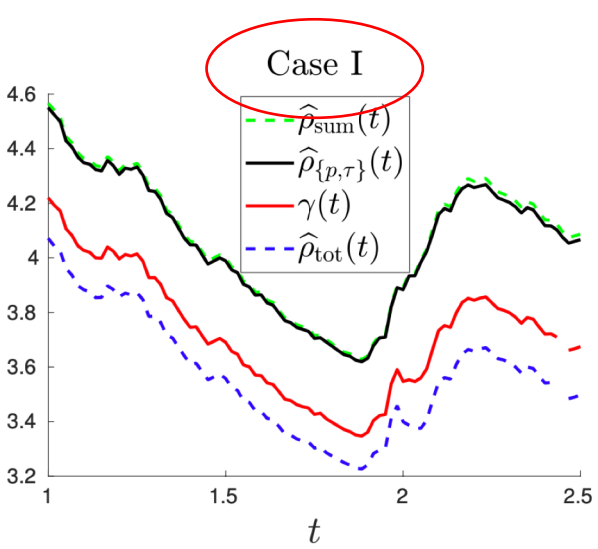
How this affects botnet identification?

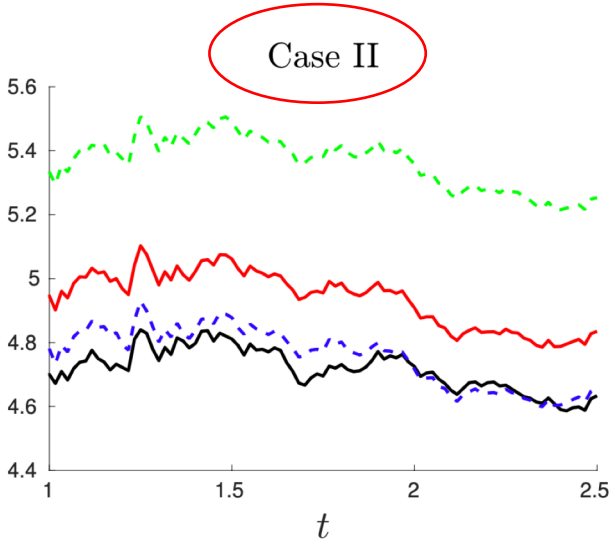
MIR Evolution Over Time

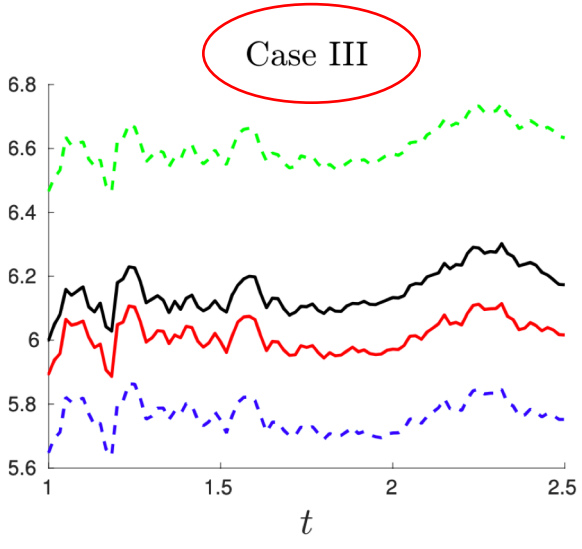
$$\gamma(t) = \theta\,\widehat{\rho}_{\text{tot}}(t) + (1-\theta)\,\widehat{\rho}_{\text{sum}}(t), \quad \theta \in (0,1)$$

$$\widehat{\rho}_{\{p,\tau\}}(t) \leq \gamma(t) \Rightarrow \text{estimated botnet } \{p,\tau\}$$

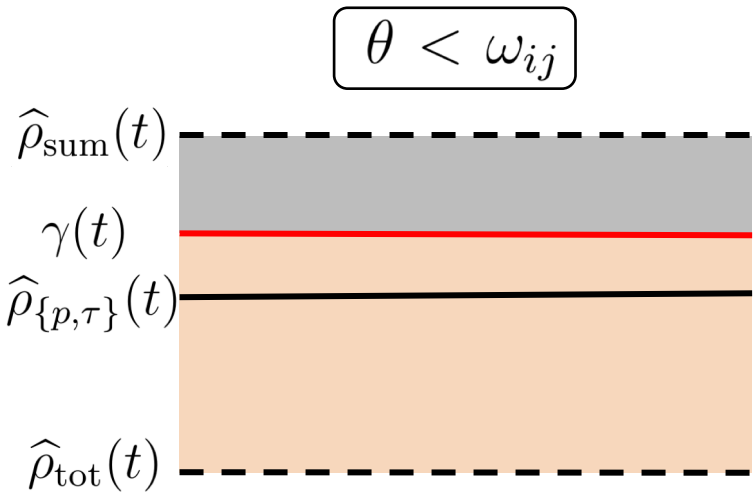
Nodes p,τ	$\widehat{\rho}_{\{p,\tau\}}(t)$
I. at least one normal	$\approx \widehat{\rho}_{\text{sum}}(t)$
II. bots from the same cluster	$\approx \widehat{\rho}_{\text{tot}}(t)$
III. bots from clusters i and j	$\approx \omega_{ij}\widehat{\rho}_{\text{tot}}(t) + (1-\omega_{ij})\widehat{\rho}_{\text{sum}}(t)$







Botnet vs. Cluster Identifiability

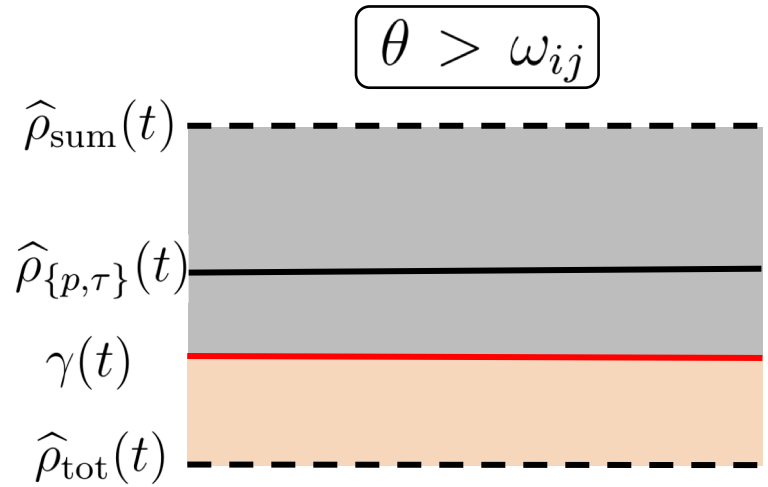


Declare normal

Bots from different clusters are **aggregated**

Declare bot

Botnet Identification



Declare normal

Bots from different clusters are **separated**

Declare bot

Cluster Identification

Mario Di Mauro (mdimauro@unisa.it)

BotClusterBuster Pseudo-Code

Algorithm: $\widehat{\mathcal{B}}(t)$=BotClusterBuster(traffic patterns until time t, θ, κ, ξ)

$\mathcal{N} = \{1, 2, \ldots, N\}$
for $p \in \mathcal{N}$ **do**
  $\widehat{\mathcal{B}}(p;t) = \{p\}$
  for $\tau \in \mathcal{N} \setminus \{p\}$ **do**
    $\gamma(t) = \theta\widehat{\rho}_{\text{tot}}(t) + (1-\theta)\widehat{\rho}_{\text{sum}}(t)$
    if $\widehat{\rho}_{\widehat{\mathcal{B}}(p;t)\cup\{\tau\}}(t) \leq \gamma(t)$ **then** $\widehat{\mathcal{B}}(p;t) = \widehat{\mathcal{B}}(p;t) \bigcup \{\tau\}$
  end
  if $|\widehat{\mathcal{B}}(p;t)| = 1$ **then** $\widehat{\mathcal{B}}(p;t) = \emptyset$
  % begin cluster expurgation
  if $\widehat{\lambda}_{\widehat{\mathcal{B}}(p;t)}(t) \leq \frac{\xi\kappa}{1+\kappa}\widehat{\lambda}_{\mathcal{N}}(t)$ **then** $\widehat{\mathcal{B}}(p;t) = \emptyset$
  % end cluster expurgation
end
$\widehat{\mathcal{B}}(t) = \bigcup_{p=1}^{N} \widehat{\mathcal{B}}(p;t)$

Comparing the MIR against the threshold

Cluster “expurgation”

Union of survived clusters

It is possible to introduce three strategies to identify the clusters

- **Max rule**: retain only the maximum-size emerged cluster
- **Union rule**: retain all clusters
- **Cluster Expurgation rule**: discard "spurious" clusters by exploiting the transmission activity carried by the candidate clusters

Performance Indices

Consider a network $\mathcal{N}$ containing a botnet $\mathcal{B}$, and let $\hat{\mathcal{B}}(t)$ be the botnet estimated at time t

$$\eta_{\text{bot}}(t) = \frac{\mathbb{E}[|\hat{\mathcal{B}}(t) \cap \mathcal{B}|]}{|\mathcal{B}|}$$

Expected fraction of **correctly banned users**
We want $\eta_{\text{bot}}(t) \to 1$ as t goes to infinity

$$\eta_{\text{nor}}(t) = \frac{\mathbb{E}[|\hat{\mathcal{B}}(t) \cap (\mathcal{N} \setminus \mathcal{B})|]}{|\mathcal{N} \setminus \mathcal{B}|}$$

Expected fraction of **incorrectly banned users**
We want $\eta_{\text{nor}}(t) \to 0$ as t goes to infinity

Experimental Setting

- We constructed 2 datasets of **real-world traffic**
  - **Lab dataset:** web-surfing activities, on an e-commerce site, of researchers and students in the Co.Ri.TeL Laboratory @ the University of Salerno
  - **Campus dataset:** web-surfing activities, on an auction portal, of people around the Campus of the University of Salerno
- Simulated botnet activity, with emulation dictionaries built by using legitimate patterns taken from the above datasets
- Overall network made of 100 bots and 100 normal users

Results (1/2)

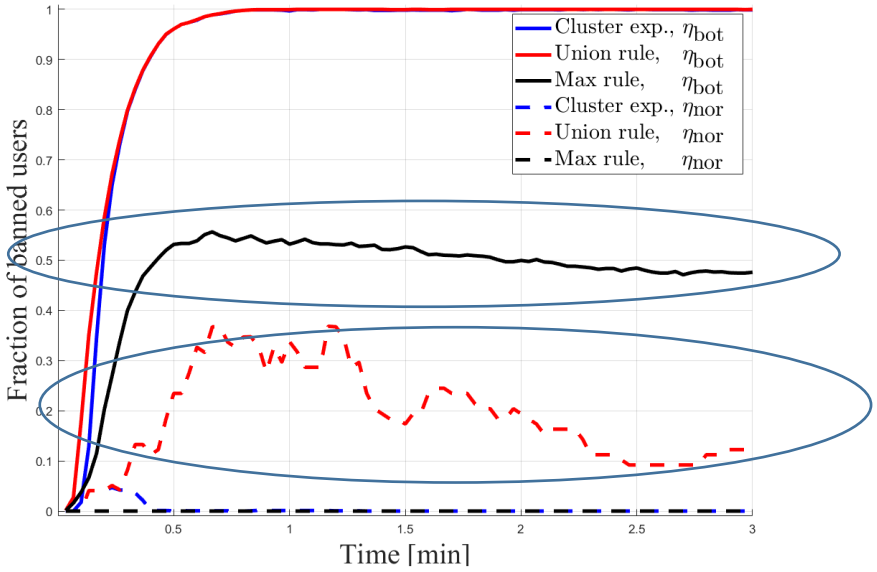
3 clusters of: 50, 30, 20 bots

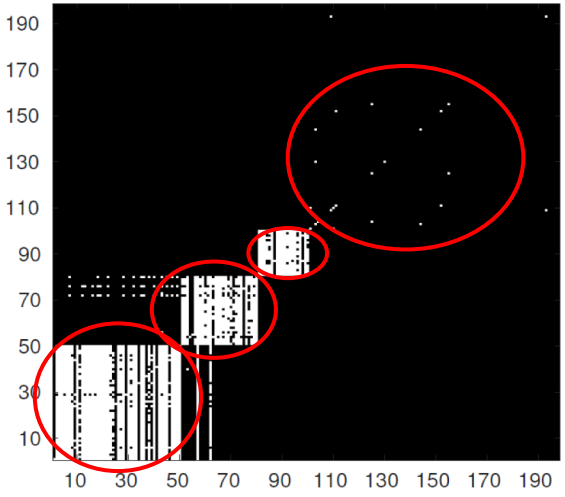
$$\alpha_1 = \alpha_2 = \alpha_3 = 10$$

$$\omega_{12} = \frac{3}{4}$$

$$\omega_{13} = \omega_{23} = \frac{1}{2}$$

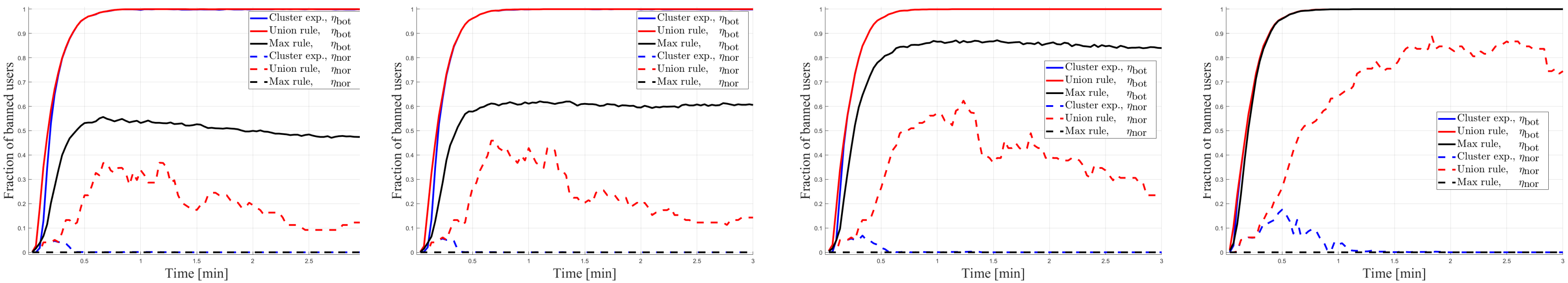
$$\theta = 0.95$$

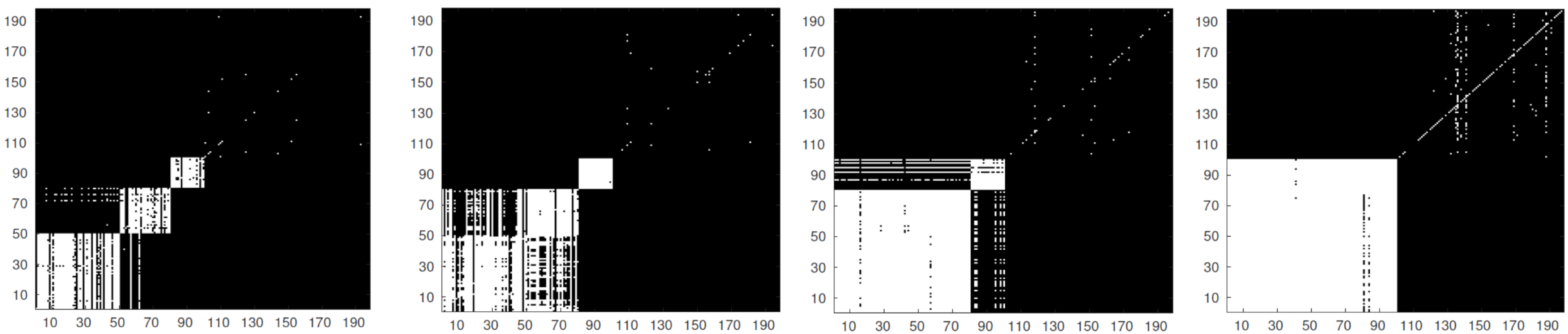




Mario Di Mauro (mdimauro@unisa.it)

Results (2/2)





$\theta = 0.95$ $\theta = 0.9$ $\theta = 0.75$ $\theta = 0.5$

Mario Di Mauro
(mdimauro@unisa.it)

.DIEM

Concluding Remarks

- We considered L7-DDoS attacks launched by a botnet whose members, organized in clusters, can access **overlapped emulation dictionaries**
- Under this demanding setting, we designed and characterized a novel algorithm, BotClusterBuster, which was shown to provide faithful botnet identification

Open Questions

- Enrich the experimental part (legitimate traffic patterns came from real users, while malicious patterns were artificially generated)
- Can we smartly drive successive nodes' comparisons along specific paths (perhaps leveraging side information) to reduce the computational complexity of the algorithm?

Thanks for your attention!

Mario Di Mauro
(mdimauro@unisa.it)