This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

DDoS Attacks with Randomized Traffic Innovation: Botnet Identification Challenges and Strategies

Vincenzo Matta, Mario Di Mauro, and Maurizio Longo

***Abstract*—Distributed Denial-of-Service (DDoS) attacks are usually launched through the *botnet*, an “army” of compromised nodes hidden in the network. Inferential tools for DDoS mitigation should accordingly enable an early and reliable discrimination of the normal users from the compromised ones. Unfortunately, the recent emergence of attacks performed at the application layer has multiplied the number of possibilities that a botnet can exploit to conceal its malicious activities. New challenges arise, which cannot be addressed by simply borrowing the tools that have been successfully applied so far to earlier DDoS paradigms. In this work, we offer basically three contributions: i) we introduce an abstract model for the aforementioned class of attacks, where the botnet emulates normal traffic by continually learning admissible patterns from the environment; ii) we devise an inference algorithm that is shown to provide a consistent (i.e., converging to the true solution as time elapses) estimate of the botnet possibly hidden in the network; and iii) we verify the validity of the proposed inferential strategy on a testbed environment. Our tests show that, for several scenarios of implementation, the proposed botnet identification algorithm needs an observation time in the order of (or even less than) one minute to identify correctly almost all bots, without affecting the normal users’ activity.**

***Index Terms*—Distributed Denial-of-Service, DDoS, Cybersecurity, Signal Processing for Network Security.**

I. Introduction and Motivation

CYBER-SECURITY ranks among the biggest challenges of modern times. Whether we are talking of phishing, website sabotages, or even of terrorist attacks, protecting our digital lives is an issue of paramount importance. Networks, and especially the Internet, became the natural attackers’ habitat to hide a broad variety of threats. For instance, a dangerous attack to a powerful target site (e.g., a big e-commerce portal) is often launched through a series of apparently innocuous attacks to some powerless, but most vulnerable, sites (e.g., some personal computers).

One of the most popular threats is the Denial-of-Service (DoS) attack, which can be broadly categorized as a *volumetric* attack, where the target destination is overwhelmed by a huge number of requests, eventually leading to the impossibility of serving any of the users. In particular, with a *Distributed* DoS (DDoS) attack, such a huge number of requests is produced in parallel by a net of robots (the *botnet*). According to one of the classical DDoS representations, a relatively large ensemble of machines (the *bots* or *zombie* “army”), acts cooperatively under the supervision of one or more coordinators (the *botmasters*). The bots may be either themselves malicious users acting consciously, or they may be legitimate users that have been preliminarily infected, (e.g., by worms and/or Trojans).

The existence itself of an anomalous request rate is essentially uncovered, and, hence, its detection is not a big deal. The main challenge is instead ascertaining whether the anomaly is caused by a DDoS attack, and, if so, performing a correct/early identification of the botnet hidden in the network. These operations are crucial to achieve successful DDoS mitigation, since discriminating legitimate from malicious users would allow the destination to ban the latter, without denying the service to the former. Providing inference solutions to botnet discovery and identification is the main subject of this work.

A. Related Work

The earliest DoS paradigms (see, e.g., TCP SYN flooding), relied on specific protocols’ vulnerabilities, and were characterized by the repetition of one (or few) requests with a huge rate [1]. In this situation, the single source of the attack can be identified by computing its unusually large request rate.

The *distributed* variants of such attacks exploit basically the same kind of vulnerabilities and repetition schemes, but for the fact that the *large* request rate is now obtained by aggregating *many small* individual bot rates. Nevertheless, in such attacks, the bots can be still identified at a single-user level. Indeed, normal traffic patterns are typically characterized by a certain degree of innovation, while the repetition scheme implicitly emphasizes the bot character. In fact, several useful inferential strategies have been proposed for such kind of DDoS attacks.

The literature about DDoS attacks is rich. With no pretence of completeness, we introduce briefly some recent works on the subject, and we refer the Reader to the survey in [2] for a more comprehensive summary. In [3], statistical methods to identify DDoS attacks are proposed, relying on computing entropy and frequency-sorted distributions of selected packet attributes. The DDoS identification is then based on the detection of anomalies in the characteristics of the packet attributes. In [4], the Authors propose a hierarchical method based on macroscopic-level network monitoring to capture shifts in spatial-temporal traffic patterns, which are then used to inform a detection system about where and when a DDoS flooding attack possibly arises in a source network. The work presented in [5] relies on the application of an entropy detection method, where the key to identify the DDoS attack is the randomness of some attributes in the packets’ headers. In [6], two new information metrics, the generalized entropy metric and the information distance metric, are employed to detect low-rate DDoS attacks, by evaluating the dissimilarity between legitimate and attack traffic. A mathematical model to examine shrew DDoS attacks (where TCP flows are constrained to a small fraction of their ideal rate at low attack costs) is

The authors are with DIEM, University of Salerno, via Giovanni Paolo II, I-84084, Fisciano (SA), Italy (e-mail: {vmatta, mdimauro, longo}@unisa.it).

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

introduced in [7]. The Authors propose a methodology aimed at capturing the adjustment behaviors of TCP congestion window at the victim's side, in order to evaluate the interplay between attack patterns and network environment.

More closely connected to this work is the new class of *application-layer* DDoS attacks, which is recently emerging as one of the most powerful threats [8]–[11]. In such attacks, the malicious traffic patterns are disguised as normal ones by leveraging the many possibilities offered at the application layer (for instance, when surfing through a website, more and more web-pages are likely to be explored as time elapses). By assigning a sufficient degree of variability to each individual bot's pattern, identification strategies based on single-user inspection become harmless. Building on such new possibilities, in this work we shall introduce a formal model for DDoS attacks where the botnet gets at its disposal a certain *emulation dictionary* to build the traffic patterns.

A number of intriguing questions arise. Despite the strong power given to the attacker, is it still possible to consistently unveiling the presence of a botnet? If so, which are the pertinent inferential strategies, and which the performance limits? Which is the fundamental trade-off between the botnet learning ability and the inference performance?

B. Relevant Inferential Tools and Methods

The inferential strategies available in the literature are not conceived to manage the class of DDoS attacks with increasing emulation dictionary [2]. While in principle it is possible to generalize and take inspiration from some of these strategies, plug-in solutions to our problem are currently unavailable. Therefore, new inferential solutions must be conceived.

Classical parametric methods (e.g., maximum likelihood, Neyman-Pearson tests) typically offer a high degree of tractability, analytical results and performance guarantees, but they are suited to those situations where a detailed knowledge of the models is available [12]–[15], a condition that is far from being verified in our setting. As a result, the aforementioned benefits are often paid in the coin of scarce robustness, sensible performance loss, low degree of versatility.

In contrast, fully data-driven techniques (e.g., machine learning) do not require a detailed knowledge of the underlying models, and provide a high degree of versatility, often at the price of hard physical interpretation of the metrics, heavy algorithm-tuning when parameters change, lack of analytical results and performance guarantees.

In order to partly circumvent the limitations of both approaches, as well as to retain some advantages thereof, in the present work we follow some emerging trends in signal processing for network cyber-security applications, which lie somehow in-between parametric and fully data-driven techniques. As notable examples, we mention: sparsity-aware algorithms for unveiling traffic volume anomalies [16]–[18]; universal algorithms for tracing information flows across the network [19]–[24]; hypothesis testing cast in the framework of adversarial signal processing [25], for the case where the adversaries can corrupt the data and the statistical hypotheses are specified only through training data [26]. Inspired by a common underlying philosophy, such works suggest to pursue the following principled approach: i) focus on minimal-and-realistic physical assumptions; ii) envisage physically-meaningful descriptive indicators arising from the modeling assumptions; iii) devise consequently an inference strategy.

The DDoS class considered in this work builds upon and generalizes some dangerous threats that have been recently documented in the literature. To the best of our knowledge, this is the first attempt to provide a systematic analysis and to devise suitable countermeasures for such kind of attacks.

A short and limited version of this work appears in the conference publication [27]. The main novelties introduced in this work include: complete proofs of all theorems; discussion and examples aimed at illustrating the physical interpretation and the relevant implications of the theoretical results; a comprehensive and formal illustration of a botnet identification condition and of the corresponding identification algorithm; an extended campaign of experiments on a testbed environment.

C. Main Result

This work deals with the design and analysis of inference strategies aimed at identifying a botnet in the context of *distributed* denial-of-service attacks. In our setting: i) the network analyst collects traffic patterns from across the network, and has access to the message content; ii) the meaning of the messages produced by an individual user provides no special information about its nature, legitimate, or malicious; and iii) no specific assumptions are made about the characterization of the traffic patterns of a normal user. In this respect, the inference strategies proposed in this work are non-parametric.

Starting from the attacks documented in the literature, we introduce a formal model for randomized DDoS attacks with increasing emulation dictionary, which is defined by the following main features: i) the botnet emulates the normal traffic patterns by gleaning admissible messages from an emulation dictionary; and ii) the botnet is given the strong power of learning an emulation dictionary that becomes richer and richer as time elapses, so as to guarantee a sufficient variability across messages. In order to quantify the botnet *learning ability*, in this work we introduce the Emulation Dictionary Rate (EDR), namely, the increase of dictionary cardinality per unit time.

Notably, the considered class of DDoS attacks is more general and powerful than many attacks documented in the literature. The assumption of such great power in the attacker's hands might perhaps look overly pessimistic. At the same time, a worst-case analysis is perfectly suited to security applications, and allows getting important insights as regards the botnet identifiability under challenging operational conditions.

The fundamental descriptive indicator employed in this work to ascertain the nature of network users is the Message Innovation Rate (MIR), namely, the number of distinct messages per unit time, transmitted by a given group of users. The relevance of the MIR for botnet identification purposes arises since, in view of the coordination in the DDoS attack, the users belonging to a botnet are expected to exhibit a smaller degree of innovation than normal users, which act by their own nature independently one each other.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

Our first contribution determines the MIR for a botnet $\mathcal{B}$, with either deterministic or Poisson transmission scheduling. Denoting by $\lambda_{\mathcal{B}}$ the transmission rate corresponding to the overall transmission activity in $\mathcal{B}$, and by α the EDR, we show that the MIR converges in probability to the following innovation rate (Theorem 1):

$$\boxed{\mathscr{R}(\alpha,\lambda_{\mathcal{B}})=\frac{\alpha\,\lambda_{\mathcal{B}}}{\alpha+\lambda_{\mathcal{B}}}} \tag{1}$$

Our second contribution consists of devising an algorithm that, under a suitable Botnet Identification Condition (BIC), guarantees that the botnet hidden in the network is correctly identified as time elapses (Theorem 2).

Finally, as a third contribution, all of the aforementioned theoretical results are tested and validated on a testbed environment; the experimental outcomes are definitely encouraging.

Notation. $\mathbb{P}[\cdot]$ and $\mathbb{E}[\cdot]$ denote the probability and the expectation operators, respectively. Given an ensemble of random variables X_t (with either continuous or discrete index t), the notation $X_t \xrightarrow{\text{p}} X$ means that X_t converges in probability to X as $t\to\infty$ [28].

II. Network Activity Indicators

We start by introducing the basic quantities that will be used to describe the network activity. The first quantity relates to the *transmission* activity of the network users. Each user employs a certain scheduling, which is identified by the transmission epochs of its own messages. More in general, for any given subnet $\mathcal{S}$ of the network, we can define the *aggregate* pattern that comprises all (ordered) transmission epochs of the users belonging to $\mathcal{S}$, formally: $T_{\mathcal{S}}(1), T_{\mathcal{S}}(2),\dots,$ where $T_{\mathcal{S}}(i)$ is the i-th (random) transmission epoch of users belonging to $\mathcal{S}$. Likewise, the pattern of an individual user u becomes: $T_u(1), T_u(2),\dots,$ where, with a slight abuse of notation (which will be used throughout the work), we have written u in lieu of $\{u\}$. The total number of transmissions occurred in $\mathcal{S}$, up to a given (deterministic) time t is denoted by $N_{\mathcal{S}}(t)\triangleq|\{i : T_{\mathcal{S}}(i)\leq t\}|$.

As an indicator of the *transmission* activity, we introduce the *empirical* transmission rate at time t, namely,

$$\boxed{\hat{\lambda}_{\mathcal{S}}(t)\triangleq\frac{N_{\mathcal{S}}(t)}{t}} \tag{2}$$

Whenever a limiting rate (as t goes to infinity) is meaningfully defined, it will be denoted by $\lambda_{\mathcal{S}}$, which will be simply referred to as the transmission rate of subnet $\mathcal{S}$.

Two examples of transmission schedulings which are relevant for our DDoS application, and which admit a limiting rate, are the synchronous, constant-rate transmission scheduling, and the independent Poisson scheduling. In the former case, all users transmit synchronously, and the (constant) interval between two transmissions has duration $1/\lambda$. The empirical transmission rate clearly obeys: $\hat{\lambda}_{\mathcal{S}}(t)\to\lambda\,|\mathcal{S}|$ as $t\to\infty$. In the latter case, the transmission pattern of user u is a Poisson process with rate λ_u, and the processes are mutually independent. Since the aggregate of independent Poisson processes is still a Poisson process, as a straightforward application of the (weak) law of large numbers, we have [29]: $\hat{\lambda}_{\mathcal{S}}(t)\xrightarrow{\text{p}}\sum_{u\in\mathcal{S}}\lambda_u$.

As a second indicator of the network activity, we define a quantity that relates to the *content* of the messages sent by network users. We are interested in the *new* messages that are incrementally produced by the users during their activities, namely, in a Message Innovation Rate (MIR). In order to obtain a formal definition of the MIR, let $\mathscr{D}_{\mathcal{S}}(t)$ denote the empirical dictionary composed by the *distinct* messages sent, up to time t, by users within $\mathcal{S}$. For the sake of clarity, we remark that, if the same message is sent, e.g., twice, from users belonging to $\mathcal{S}$, it appears only once in the dictionary $\mathscr{D}_{\mathcal{S}}(t)$. The *empirical* Message Innovation Rate (MIR) is:

$$\boxed{\hat{\rho}_{\mathcal{S}}(t)\triangleq\frac{|\mathscr{D}_{\mathcal{S}}(t)|}{t}} \tag{3}$$

In particular, if $\hat{\rho}_{\mathcal{S}}(t)\xrightarrow{\text{p}}\rho_{\mathcal{S}}$, the limiting value $\rho_{\mathcal{S}}$ will be simply referred to as the MIR of subnet $\mathcal{S}$.

III. Randomized DDoS with Emulation Dictionary

A botnet $\mathcal{B}_{\text{tot}}$, composed by B_{tot} malicious nodes, sends messages to the destination under attack in order to saturate its resources. The botnet mimics normal patterns by picking messages from an emulation dictionary, which is learned *continually* (i.e., its cardinality increases with time), in order to ensure that a reasonable innovation rate can be sustained. Such a dictionary construction can occur in many different ways. For instance, by means of one or more powerful botmasters, the botnet might be able to perform an on-line monitoring of normal activities from across the network. From such a monitoring, sequences of messages corresponding to normal patterns of activity are collected, allowing the construction of a dictionary of admissible messages.

Let $\mathscr{E}(t)$ be the (common) dictionary available at time t to all botnet members. We assume that the number of messages available for emulation grows, asymptotically, in a linear fashion. Therefore, it makes sense to introduce the Emulation Dictionary Rate (EDR) as:

$$\boxed{\alpha\triangleq\lim_{t\to\infty}\frac{|\mathscr{E}(t)|}{t}} \tag{4}$$

Given the emulation dictionary, the botnet has clearly many ways to build the traffic patterns. At one extreme, the botmaster disseminates B_{tot} *disjoint* (say, equal-size) portions of $\mathscr{E}(t)$ through the botnet. Then, each bot builds its traffic pattern by scanning, in a sequential fashion, its portion of the emulation dictionary. Such a scheme would clearly maximize the independence among the bots. With this policy, the problem would become equivalent to the case that each bot owns a distinct emulation dictionary with EDR equal to α/B_{tot}. However, since B_{tot} must be large, it is unrealistic to assume that a botmaster can learn so many patterns to build B_{tot} distinct dictionaries that are in turn so rich to guarantee a credible emulation. Therefore, in the case of disjoint dictionaries, the number of distinct messages available to a single bot would be typically small, implying a suspiciously high degree of

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

replication, which would make the bots easily identifiable by single-user inspection.

At the other extreme, each bot might simply use *all* messages contained in $\mathscr{E}(t)$. Clearly, such scheme maximizes the innovation of each individual bot, but also maximizes the dependence inside the botnet. By inspection of the messages sequentially sent by two or more bots, a traffic analyst would recognize an anomalous behavior.

We hence assume that the attacker has devised some intermediate strategy to circumvent the aforementioned issues. We introduce a class of *randomized* DDoS attacks, where a bot that intends to transmit at time t picks a message from the available emulation dictionary $\mathscr{E}(t)$, and sends such a message to the destination. The message is chosen uniformly at random, so that the probability of a particular message is simply $1/|\mathscr{E}(t)|$.

The corresponding evolution of the empirical dictionaries, for any subnet $\mathcal{B}$ of $\mathcal{B}_{\text{tot}}$, is easily obtained as follows. Given the empirical dictionary $\mathscr{D}_{\mathcal{B}}(t)$, the empirical dictionary $\mathscr{D}_{\mathcal{B}}(t+\tau)$ is obtained by adding the *distinct* messages not contained in $\mathscr{D}_{\mathcal{B}}(t)$, which have been selected during the interval τ by the bots belonging to $\mathcal{B}$.

A. Characterization of the Botnet Message Innovation Rate

Let us preliminarily introduce the following function:

$$\boxed{\mathscr{R}(\alpha,\lambda) \triangleq \frac{\alpha\,\lambda}{\alpha+\lambda}} \tag{5}$$

Our first result provides a closed-form expression for the MIR of a botnet.

Theorem 1 (Botnet MIR): Consider a botnet $\mathcal{B}_{\text{tot}}$ launching a DDoS attack, where the node transmission policies are either synchronous with constant transmission rate, or independent Poisson processes, with rates λ_u, for $u \in \mathcal{B}_{\text{tot}}$. Consider a subset of the botnet $\mathcal{B} \subseteq \mathcal{B}_{\text{tot}}$. Let $\mathscr{E}(t)$ be the emulation dictionary available to the botnet, with emulation dictionary rate α, and let $\mathscr{D}_{\mathcal{B}}(t)$ be the empirical dictionary of the subnet $\mathcal{B}$ at time t. Then, the message innovation rate of $\mathcal{B}$ is:

$$\boxed{\frac{|\mathscr{D}_{\mathcal{B}}(t)|}{t} \xrightarrow{\text{p}} \rho_{\mathcal{B}} = \mathscr{R}(\alpha,\lambda_{\mathcal{B}})} \tag{6}$$

where $\lambda_{\mathcal{B}} = \sum_{u\in\mathcal{B}} \lambda_u$ is the aggregate transmission rate of the considered botnet subset.

Proof: See Appendix A. ■

REMARK I. From (5) and (6) we see that increasing the EDR α and/or the transmission rate λ corresponds to increasing the MIR. Besides, the MIR is always smaller[1] than $\min(\alpha,\lambda)$, which makes sense, since the number of new messages can exceed neither the number of messages in the emulation dictionary ($\mathscr{R}(\alpha,\lambda) \le \alpha$), nor the overall number of transmitted messages ($\mathscr{R}(\alpha,\lambda) \le \lambda$). Notably, the quantity $\min(\alpha,\lambda)$ is the MIR corresponding to a practical scheme where the patterns are obtained by taking sequentially (in a deterministic way) the messages of the emulation dictionary. With such a scheme, if $\alpha > \lambda$, a new message can be always found in $\mathscr{E}(t)$, and the maximum rate of distinct messages is λ. Likewise, if $\lambda > \alpha$, all messages in $\mathscr{E}(t)$ can be selected, along with some unavoidable repetitions, and the maximum rate of distinct messages is α.

REMARK II. As α goes to infinity, the MIR converges to λ. In fact, as the number of messages in the emulation dictionary goes to infinity, each transmission would correspond with high probability to a new message, and the MIR will eventually reach the maximum allowable value λ. Likewise, as λ goes to infinity, we see that the MIR converges to α. In fact, as the number of sent messages goes to infinity, the emulation dictionary is completely spanned, and the MIR will eventually saturate to its maximum allowable value α.

REMARK III. The MIR is symmetric in α and λ, implying that both quantities, even if they have a completely different practical meaning, play the same role as regards their effect on the MIR. In particular, we can write $\mathscr{R}(\alpha,\lambda) = (1/\alpha + 1/\lambda)^{-1}$, which reveals that the rate $\mathscr{R}(\alpha,\lambda)$ can be represented as the inverse of a time interval given by the sum of the average time between two messages available in the emulation dictionary, $1/\alpha$, and the average time between two transmissions, $1/\lambda$.

REMARK IV. For strictly positive α and λ we have:

$$\boxed{\mathscr{R}(\alpha,\lambda_1) + \mathscr{R}(\alpha,\lambda_2) > \mathscr{R}(\alpha,\lambda_1+\lambda_2)} \tag{7}$$

The latter inequality can be straightforwardly checked by exploiting the definition of $\mathscr{R}(\alpha,\lambda)$ in (5). More interestingly, such inequality can be explained in the light of the physical interpretation of Theorem 1. In fact, the LHS in (7) corresponds to the MIR of a botnet made of two subnets: i) featuring transmission rates λ_1 and λ_2, respectively, and ii) picking messages from two *disjoint* dictionaries, each one with EDR equal to α. In contrast, the RHS corresponds to the MIR of a botnet made of two subnets, still featuring transmission rates λ_1 and λ_2, but picking messages from a *common* dictionary with EDR α. Hence, the lower bound follows.

REMARK V. Our focus is on *genuinely-distributed* DoS attacks where the number of bots is large, *and* the transmission rate of each bot is not anomalous. For comparison purposes, let us consider another DDoS strategy, where B_{tot} *disjoint* (say, equal-size) portions of the emulation dictionary are disseminated through the botnet. Assuming for simplicity that all bots have unitary transmission rates, the MIR of user u, and the MIR of the whole botnet will be, respectively,

$$\rho_u = \frac{\alpha}{\alpha + B_{\text{tot}}}, \qquad \rho_{\mathcal{B}_{\text{tot}}} = \sum_{u\in\mathcal{B}_{\text{tot}}} \rho_u = \frac{\alpha\,B_{\text{tot}}}{\alpha + B_{\text{tot}}}, \tag{8}$$

where the first relationship follows from Theorem 1, while the second relationship follows from disjointness of the emulation (and, hence, of the empirical) dictionaries.

On the other hand, for our *coordinated* DDoS with *common* emulation dictionary, Theorem 1 gives:

$$\rho_u = \frac{\alpha}{\alpha+1}, \qquad \rho_{\mathcal{B}_{\text{tot}}} = \frac{\alpha\,B_{\text{tot}}}{\alpha + B_{\text{tot}}}. \tag{9}$$

Notably, the rightmost formulas in (8) and (9) reveal that the MIR for the case of disjoint dictionaries *is the same as the MIR of a botnet using a common emulation dictionary*. On the other hand, the leftmost formulas in (8) and (9) reveal that the MIR of a single bot for the case of disjoint

[1] For $x > 0$ and $y > 0$, one has $x/(x+y) \le 1$.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

dictionaries *is approximately* B_{tot} *times smaller than the MIR of a single bot for the case of a common emulation dictionary*. Such a reduced degree of innovation matches the observations reported below (4), concerning the flaws of deterministic DDoS attacks based on disjoint emulation dictionaries.

REMARK VI. Assume that the traffic analyst must estimate α based on the patterns collected from a certain subnet $\mathcal{S}$. From (5) and (6), we have $\alpha = \lambda_{\mathcal{S}}\,\rho_{\mathcal{S}}/(\lambda_{\mathcal{S}} - \rho_{\mathcal{S}})$. Accordingly, a reasonable estimator of α can be obtained by replacing ρ and λ with their empirical counterparts, yielding:

$$\boxed{\hat{\alpha}_{\mathcal{S}}(t) \triangleq \frac{\hat{\lambda}_{\mathcal{S}}(t)\,\hat{\rho}_{\mathcal{S}}(t)}{\hat{\lambda}_{\mathcal{S}}(t) - \hat{\rho}_{\mathcal{S}}(t)}} \tag{10}$$

In view of Theorem 1, such estimator converges in probability to α as t goes to infinity, for any $\mathcal{S} \subseteq \mathcal{B}_{\text{tot}}$.

In contrast, when dealing with normal users, such an interpretation fails in general, since: i) a limiting value α does not necessarily exist, and ii) the generative mechanism of normal patterns is not necessarily interpreted in terms of random picking from an emulation dictionary. Nevertheless, the quantity $\hat{\alpha}_{\mathcal{S}}(t)$ can be meaningfully defined also for arbitrary subnets (i.e., composed also, or even exclusively, by normal users), since it represents the ratio between the empirical rate of "distinct" messages $\hat{\rho}_{\mathcal{S}}(t)$, and the empirical rate of "repeated" messages $\hat{\lambda}_{\mathcal{S}}(t) - \hat{\rho}_{\mathcal{S}}(t)$, scaled[2] by the empirical transmission rate $\hat{\lambda}_{\mathcal{S}}(t)$. Such an interpretation is useful since it is now independent from the particular model adopted (transmission scheduling, botnet or normal behavior, etc.). In the following, even when dealing with arbitrary subnets, we shall loosely refer to $\hat{\alpha}_{\mathcal{S}}(t)$ as the *empirical*, or *estimated* EDR.

Finally, exploiting (5) and (10), the empirical MIR $\hat{\rho}_{\mathcal{S}}(t)$, for an *arbitrary* subnet $\mathcal{S}$, can be expressed as:

$$\boxed{\hat{\rho}_{\mathcal{S}}(t) = \mathscr{R}(\hat{\alpha}_{\mathcal{S}}(t), \hat{\lambda}_{\mathcal{S}}(t))} \tag{11}$$

IV. Botnet Identification Condition

The coordination implied in the *distributed* DoS attack introduces some correlation between the empirical dictionaries of the bots, due to the common emulation dictionary where messages are selected. In contrast, the empirical dictionaries of two normal users are expected to be weakly correlated, due to independence among their activities. Likewise, the empirical dictionaries of a bot and of a normal user are expected to be weakly correlated, since the network employed by the botmaster to acquire the emulation dictionary is usually not part of the network monitored by the traffic analyst.

On the other hand, even in the presence of normal (thus, *independent*) users, it is realistic to assume a certain degree of *physiological* correlation among the users' activities. Distinct users can reasonably share parts of their dictionaries, e.g., their surfing activities might partly overlap, due to common interests, popular web-pages, peculiar structure of the destination of interest, etc. Similar considerations apply when dealing with a subset of the botnet and a subnet made only of normal users.

Given the very limited amount of information and assumptions we made, and according to the above discussion, any meaningful strategy to discriminate a normal from a malicious behavior, cannot but be based on the degree of dependence among the users. In our setting, a convenient way to measure the degree of dependence is provided by the empirical message innovation rate in (3). However, the mere availability of a good network indicator does not provide a *quantitative* way to discriminate normal users from bots. In order to design an algorithm for botnet identification, we need to define a proper *identification threshold*. To this aim, we can use as reference case for a malicious behavior, the MIR corresponding to the activity performed by a botnet. In order to understand how such operation can be implemented, let us start by considering the case that we must decide whether users 1 and 2 belong to a botnet. Assume for now that the empirical EDRs of the two users obtained through (10) are comparable (the explicit dependence on t being suppressed, for ease of notation, here and in the forthcoming discussion):

$$\hat{\alpha}_1 \approx \hat{\alpha}_2 \approx \hat{\alpha}. \tag{12}$$

When both users belong to a botnet, in view of Theorem 1, for t large enough we can write:

$$\hat{\rho}_{\{1,2\}} \approx \mathscr{R}(\hat{\alpha}, \hat{\lambda}_1 + \hat{\lambda}_2) \triangleq \hat{\rho}_{\text{bot}}. \tag{13}$$

Moreover, *irrespectively of the users' nature*, the empirical MIR of the aggregate subnet $\{1, 2\}$ can be upper bounded by the MIR corresponding to disjoint dictionaries, namely,

$$\begin{aligned} \hat{\rho}_{\{1,2\}} &\leq \hat{\rho}_1 + \hat{\rho}_2 = \mathscr{R}(\hat{\alpha}_1, \hat{\lambda}_1) + \mathscr{R}(\hat{\alpha}_2, \hat{\lambda}_2) \triangleq \hat{\rho}_{\text{sum}} \\ &\approx \mathscr{R}(\hat{\alpha}, \hat{\lambda}_1) + \mathscr{R}(\hat{\alpha}, \hat{\lambda}_2), \end{aligned} \tag{14}$$

where the second equality follows from (11), while the approximate equality follows from (12). Since from (7) we know that $\hat{\rho}_{\text{bot}} < \hat{\rho}_{\text{sum}}$, it makes sense to introduce a threshold lying between the two points $\hat{\rho}_{\text{bot}}$ and $\hat{\rho}_{\text{sum}}$, formally, for $\epsilon \in (0, 1)$:

$$\hat{\rho}_{\text{bot}} < \gamma = \hat{\rho}_{\text{bot}} + \epsilon(\hat{\rho}_{\text{sum}} - \hat{\rho}_{\text{bot}}) < \hat{\rho}_{\text{sum}}. \tag{15}$$

When the two users belong to a botnet, from (13) we see that, for large t, the empirical MIR $\hat{\rho}_{\{1,2\}}$ converges to the value $\hat{\rho}_{\text{bot}}$. On the other hand, using Theorem 1, it is easy to verify that $\hat{\rho}_{\text{sum}} - \hat{\rho}_{\text{bot}}$ converges in probability to a positive quantity, which implies that, for *any* $\epsilon > 0$, as time elapses, the empirical MIR will stay sooner (higher ϵ) or later (lower ϵ) *below* the threshold γ, yielding:

$$\boxed{1 \text{ AND } 2 \text{ are bots} \Rightarrow \hat{\rho}_{\{1,2\}} < \gamma} \tag{16}$$

Consider now the case that at least one user is normal. Were the dictionaries of the two users perfectly disjoint, we would clearly observe, for any $\epsilon \in (0, 1)$, that $\hat{\rho}_{\{1,2\}} \approx \hat{\rho}_{\text{sum}} > \gamma$. However, we already noticed that some correlation is expected to exist even among normal users, or among normal users and bots. It is also natural to assume that such a correlation is weaker than the correlation exhibited by groups of bots, since the latter are choosing their messages from one and the same underlying dictionary.[3] Accordingly, we might expect

[2] The scaling simply corresponds to expressing the result on a per-time-unit basis, rather than on a per-transmission basis.

[3] In making such assumption, we imply that the specific mechanism used to build normal patterns has a minor influence.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

that, when at least one user is normal, for sufficiently small ϵ, the empirical MIR still stays above the threshold, namely:

$$\boxed{1 \text{ OR } 2 \text{ are normal} \Rightarrow \hat{\rho}_{\{1,2\}} > \gamma} \tag{17}$$

In summary, if the empirical MIR stays below γ, we can declare that the two users form a botnet, otherwise, we can declare that at least one user is normal.

Two main points emerge. First, the essential feature enabling a successful discrimination is the assumption in (17), which accordingly plays the role of a Botnet Identification Condition (BIC). Second, the determination of the threshold γ relies on a tuning parameter ϵ, which is in principle related to the intrinsic (and unknown) properties of the normal traffic patterns. Remarkably, the experimental study conducted in the forthcoming Sec. VI will show clearly that: i) the BIC can be safely used, and ii) the choice of ϵ is by no means critical, even in the non-parametric scenario where no prior information about the normal users' behavior is available.

Unfortunately, all that glitters is not gold. There is an important complication that has been deliberately overlooked so far. According to the above explanation, we need to compare the empirical MIR to the MIR of a *reference* botnet. However, a botnet is characterized by a *common* underlying EDR α, while in practice we shall typically have $\hat{\alpha}_1 \neq \hat{\alpha}_2$ (especially when at least one user is normal), implying that the approximation in (12) is unsupported. One approach could be that of discarding *ab initio* the botnet hypothesis whenever $\hat{\alpha}_1$ and $\hat{\alpha}_2$ are too dissimilar. The qualification of being "too dissimilar" translates into the appearance of some extra tuning parameter, possibly depending on time, which we want definitely to avoid.

Another possibility is clearly that of choosing as reference EDR some intermediate value comprised between $\hat{\alpha}_1$ and $\hat{\alpha}_2$. In this connection, we remark that the naïve choice of the arithmetic average does not work for the following reason. It can be simply verified that, in general, there exist values of $\lambda_1, \lambda_2, \alpha_1, \alpha_2 \in \mathbb{R}^+$ for which $\mathscr{R}(\lambda_1, \alpha_1) + \mathscr{R}(\lambda_2, \alpha_2) < \mathscr{R}(\lambda_1 + \lambda_2, 1/2(\alpha_1 + \alpha_2))$, implying that the empirical MIR, *even for the case of disjoint dictionaries*, is not necessarily greater than the MIR of a botnet with reference EDR given by the arithmetic average of $\hat{\alpha}_1$ and $\hat{\alpha}_2$. A systematic way to select a proper intermediate value is substantially more involved, and is the object of the forthcoming section.

A. Reference EDR by Replacement and Reassignment

Let us consider two (disjoint) subnets $\mathcal{S}_1$ and $\mathcal{S}_2$, with focus on the case that at least one of them is composed only by normal users, with $\hat{\alpha}_{\mathcal{S}_1} \neq \hat{\alpha}_{\mathcal{S}_2}$. Recall that we are considering a fixed time t, and that the explicit dependence of all quantities upon t is suppressed for ease of notation.

Since a botnet has *common* underlying EDR, and since we want to compare the behavior of $\mathcal{S}_1 \cup \mathcal{S}_2$ to that of a botnet, it would be useful to envisage a new pair of traffic patterns for $\mathcal{S}_1$ and $\mathcal{S}_2$ possessing the following characteristics:

i) The individual EDRs of $\mathcal{S}_1$ and $\mathcal{S}_2$ are equal, namely (superscript $'$ refers to the "new" patterns),

$$\hat{\alpha}'_{\mathcal{S}_1} = \hat{\alpha}'_{\mathcal{S}_2} = \hat{\alpha}'. \tag{18}$$

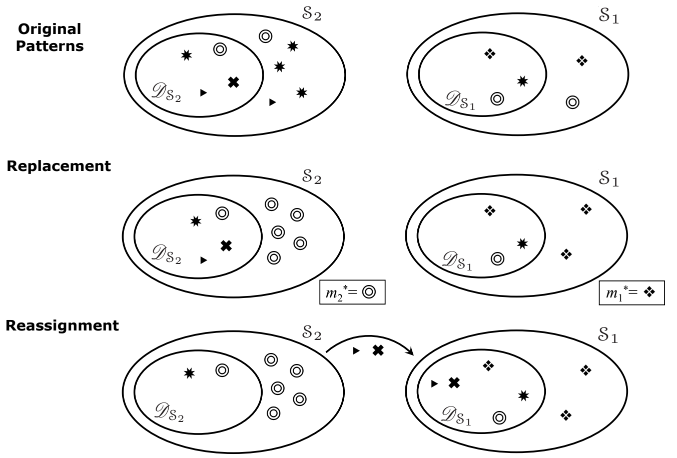

Original Patterns
$\mathcal{S}_2$
$\mathcal{S}_1$
$\mathscr{D}_{\mathcal{S}_2}$
$\mathscr{D}_{\mathcal{S}_1}$
Replacement
$\mathcal{S}_2$
$\mathcal{S}_1$
$\mathscr{D}_{\mathcal{S}_2}$
$\mathscr{D}_{\mathcal{S}_1}$
m_2^*= ◎
m_1^*= ❖
Reassignment
$\mathcal{S}_2$
$\mathcal{S}_1$
$\mathscr{D}_{\mathcal{S}_2}$
$\mathscr{D}_{\mathcal{S}_1}$

Fig. 1. The RR procedure, pictorial exemplification.

ii) The transmission rate and the MIR of the network $\mathcal{S}_1 \cup \mathcal{S}_2$ coincide with those of the original traffic patterns.

We now illustrate a Replacement and Reassignment (RR) procedure, which finds such a new pair starting from the original pattern configuration. Such a procedure relies on the intuitive consideration that, if some messages are reassigned from the subnet with highest EDR to the other subnet, the resulting EDRs tend to keep closer each other. In order to avoid misunderstandings, we remark that *the RR procedure does not correspond to any real/physical operations made on the traffic patterns. The RR procedure is a conceptual experiment used to demonstrate that it is possible to construct two patterns possessing the aforementioned requirements i) and ii).*

The RR procedure goes as follows — see Fig. 1 for a pictorial illustration.

1. *Replacement of repeated messages*. The traffic pattern of a subnet $\mathcal{S}$ contains $|\mathscr{D}_{\mathcal{S}}|$ distinct messages, the remaining $N_{\mathcal{S}} - |\mathscr{D}_{\mathcal{S}}|$ ones being repetitions of messages contained in $\mathscr{D}_{\mathcal{S}}$. The first step of the procedure amounts to replacing such $N_{\mathcal{S}} - |\mathscr{D}_{\mathcal{S}}|$ messages by one and the same message, say it m^*, contained in $\mathscr{D}_{\mathcal{S}}$. The replacement is applied to both subnets $\mathcal{S}_1$ and $\mathcal{S}_2$, with the corresponding replacing messages being m_1^* and m_2^*. Since replacement acts only on the message content, the transmission rates do not change. Moreover, since replacement leaves unaltered the number of distinct messages within each subnet, the MIR of the subnets, and the MIR of $\mathcal{S}_1 \cup \mathcal{S}_2$, are unaltered.[4]

2. *Reassignment of messages*. Some messages will be reassigned from one subnet to the other subnet (only in one direction, namely, either from $\mathcal{S}_2$ to $\mathcal{S}_1$ or from $\mathcal{S}_1$ to $\mathcal{S}_2$). For the sake of clarity, assume that $\mathcal{S}_2$ is "passing" some of its messages to $\mathcal{S}_1$, with the prescription that the replacing message m_2^* is never passed. Since, after replacement, all messages different from m_2^* appear only once in the pattern of $\mathcal{S}_2$, we see that *all* messages passed to $\mathcal{S}_1$ are necessarily distinct. The rate of messages (number of messages normalized to the current time t) that are reassigned from $\mathcal{S}_2$ to $\mathcal{S}_1$ is denoted by Δ. Accordingly, a negative Δ will correspond to

[4] The MIR is determined only by the content of the empirical dictionaries.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

the converse situation where $\mathcal{S}_1$ passes some of its messages to $\mathcal{S}_2$. As a result, the transmission rates of the pattern configuration after reassignment are:

$$(\hat{\lambda}'_{\mathcal{S}_1}, \hat{\lambda}'_{\mathcal{S}_2}) = (\hat{\lambda}_{\mathcal{S}_1} + \Delta, \hat{\lambda}_{\mathcal{S}_2} - \Delta). \tag{19}$$

Moreover, since the correlation between the two patterns is weak (recall that one of the subnets is composed only by normal users), we assume that it is always possible to reassign messages that do not belong to the intersection of the two empirical dictionaries. Such assumption, along with the fact that all passed messages are distinct, implies that, in terms of individual MIRs, what is lost by a subnet is exactly gained by the other subnet. Formally:

$$(\hat{\rho}'_{\mathcal{S}_1}, \hat{\rho}'_{\mathcal{S}_2}) = (\hat{\rho}_{\mathcal{S}_1} + \Delta, \hat{\rho}_{\mathcal{S}_2} - \Delta). \tag{20}$$

Note that not all values of Δ are admissible. For instance, if messages from $\mathcal{S}_2$ are reassigned to $\mathcal{S}_1$, the rate of reassigned messages cannot exceed the rate of distinct messages owned by $\mathcal{S}_2$, namely, $\Delta \leq \hat{\rho}_{\mathcal{S}_2}$. Likewise, in the converse case, $-\Delta \leq \hat{\rho}_{\mathcal{S}_1}$, finally yielding:[5]

$$-\hat{\rho}_{\mathcal{S}_1} \leq \Delta \leq \hat{\rho}_{\mathcal{S}_2}. \tag{21}$$

Moreover, since the reassignment changes only the "owner" of a given message, the MIR of the *aggregate* network $\mathcal{S}_1 \cup \mathcal{S}_2$ is left unaltered, namely, $\hat{\rho}'_{\mathcal{S}_1 \cup \mathcal{S}_2} = \hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2}$.

3. Choice of Δ for the equilibrium condition. At the end of the reassignment procedure, the new EDRs corresponding to $\mathcal{S}_1$ and $\mathcal{S}_2$ become, respectively, $\hat{\alpha}'_{\mathcal{S}_1} = \hat{\lambda}'_{\mathcal{S}_1}\hat{\rho}'_{\mathcal{S}_1}(\hat{\lambda}'_{\mathcal{S}_1} - \hat{\rho}'_{\mathcal{S}_1})$, and $\hat{\alpha}'_{\mathcal{S}_2} = \hat{\lambda}'_{\mathcal{S}_2}\hat{\rho}'_{\mathcal{S}_2}/(\hat{\lambda}'_{\mathcal{S}_2} - \hat{\rho}'_{\mathcal{S}_2})$, where we have exploited (10). In order to get a common reference EDR $\hat{\alpha}'$, we enforce the condition in (18), which, using (19) and (20) into the latter two equations, amounts to seek a value $\Delta^\star$ such that:

$$\hat{\alpha}' = \frac{(\hat{\lambda}_{\mathcal{S}_1} + \Delta^\star)(\hat{\rho}_{\mathcal{S}_1} + \Delta^\star)}{\hat{\lambda}_{\mathcal{S}_1} - \hat{\rho}_{\mathcal{S}_1}} = \frac{(\hat{\lambda}_{\mathcal{S}_2} - \Delta^\star)(\hat{\rho}_{\mathcal{S}_2} - \Delta^\star)}{\hat{\lambda}_{\mathcal{S}_2} - \hat{\rho}_{\mathcal{S}_2}}, \tag{22}$$

with the additional prescription that condition (21) is met. Therefore, the explicit formula for $\Delta^\star$ is found by solving a quadratic equation, and by simple algebra it can be verified that the solution fulfilling (21) is:

$$\begin{aligned} \Delta^\star \quad = \quad & \frac{\hat{\lambda}_{\mathcal{S}_1}\hat{\lambda}_{\mathcal{S}_2} - \hat{\rho}_{\mathcal{S}_1}\hat{\rho}_{\mathcal{S}_2}}{(\hat{\lambda}_{\mathcal{S}_1} - \hat{\rho}_{\mathcal{S}_1}) - (\hat{\lambda}_{\mathcal{S}_2} - \hat{\rho}_{\mathcal{S}_2})} \\ - \quad & \frac{\sqrt{(\hat{\lambda}_{\mathcal{S}_1} - \hat{\rho}_{\mathcal{S}_1})(\hat{\lambda}_{\mathcal{S}_2} - \hat{\rho}_{\mathcal{S}_2})(\hat{\lambda}_{\mathcal{S}_1} + \hat{\rho}_{\mathcal{S}_2})(\hat{\lambda}_{\mathcal{S}_2} + \hat{\rho}_{\mathcal{S}_1})}}{(\hat{\lambda}_{\mathcal{S}_1} - \hat{\rho}_{\mathcal{S}_1}) - (\hat{\lambda}_{\mathcal{S}_2} - \hat{\rho}_{\mathcal{S}_2})}. \end{aligned} \tag{23}$$

From (22), it is easily verified that a positive $\Delta^\star$ corresponds to $\hat{\alpha}_{\mathcal{S}_1} < \hat{\alpha}' < \hat{\alpha}_{\mathcal{S}_2}$ (while the latter two inequalities are reversed when $\Delta^\star < 0$), implying that the subnet with the highest EDR "passes" a fraction of its messages to the other subnet. In summary, we conclude that:

$$\boxed{\min(\hat{\alpha}_{\mathcal{S}_1}, \hat{\alpha}_{\mathcal{S}_2}) \leq \hat{\alpha}' \leq \max(\hat{\alpha}_{\mathcal{S}_1}, \hat{\alpha}_{\mathcal{S}_2})} \tag{24}$$

According to the above explanation, when at least one of the subnets is composed only by normal users, we can write:

$$\begin{aligned} \hat{\rho}_{\text{sum}}(\mathcal{S}_1, \mathcal{S}_2) \;&\triangleq\; \hat{\rho}_{\mathcal{S}_1} + \hat{\rho}_{\mathcal{S}_2} \overset{\text{(a)}}{=} \hat{\rho}'_{\mathcal{S}_1} + \hat{\rho}'_{\mathcal{S}_2} \\ &\overset{\text{(b)}}{=}\; \mathscr{R}(\hat{\alpha}', \hat{\lambda}'_{\mathcal{S}_1}) + \mathscr{R}(\hat{\alpha}', \hat{\lambda}'_{\mathcal{S}_2}) \\ &\overset{\text{(c)}}{>}\; \mathscr{R}(\hat{\alpha}', \hat{\lambda}'_{\mathcal{S}_1} + \hat{\lambda}'_{\mathcal{S}_2}) \\ &\overset{\text{(d)}}{=}\; \mathscr{R}(\hat{\alpha}', \hat{\lambda}_{\mathcal{S}_1} + \hat{\lambda}_{\mathcal{S}_2}) \triangleq \hat{\rho}_{\text{bot}}(\mathcal{S}_1, \mathcal{S}_2), \end{aligned} \tag{25}$$

where (a) follows from (20); (b) follows from (11); (c) follows from (7); and (d) follows from (19). On the other hand, when $\mathcal{S}_1$ and $\mathcal{S}_2$ form a botnet, Theorem 1 implies that, for t large enough, $\hat{\alpha}_{\mathcal{S}_1} \approx \hat{\alpha}_{\mathcal{S}_2} \approx \alpha$, which in turn implies that $\hat{\alpha}' \approx \alpha$ in view of (24). Therefore, in this case the inequality $\hat{\rho}_{\text{sum}}(\mathcal{S}_1, \mathcal{S}_2) > \hat{\rho}_{\text{bot}}(\mathcal{S}_1, \mathcal{S}_2)$ is justified by the approximations: $\hat{\rho}_{\text{sum}}(\mathcal{S}_1, \mathcal{S}_2) \approx \mathscr{R}(\alpha, \lambda_{\mathcal{S}_1}) + \mathscr{R}(\alpha, \lambda_{\mathcal{S}_2})$ and $\hat{\rho}_{\text{bot}}(\mathcal{S}_1, \mathcal{S}_2) \approx \mathscr{R}(\alpha, \lambda_{\mathcal{S}_1} + \lambda_{\mathcal{S}_2})$.

We have in fact shown that, for *arbitrary transmission schedulings and message-picking policies*, the empirical MIR of a botnet with reference EDR value (22) does *always* provide a lower bound to the sum of individual MIRs.[6]

B. Threshold Setting

Let us introduce an intermediate threshold lying between the lower bound and the upper bound in (25), namely, for $\epsilon \in (0, 1)$,

$$\boxed{\gamma(\mathcal{S}_1, \mathcal{S}_2) = \hat{\rho}_{\text{bot}}(\mathcal{S}_1, \mathcal{S}_2) + \epsilon\,[\hat{\rho}_{\text{sum}}(\mathcal{S}_1, \mathcal{S}_2) - \hat{\rho}_{\text{bot}}(\mathcal{S}_1, \mathcal{S}_2)]} \tag{26}$$

When $\mathcal{S}_1$ and $\mathcal{S}_2$ form a botnet, from Theorem 1 it is immediately seen (recall that $\hat{\alpha}'$ will converge to the true α) that $\hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2} < \gamma(\mathcal{S}_1, \mathcal{S}_2)$ as $t \to \infty$.

When at least one of the subnets is made of normal users, the degree of dependence among their patterns is low. Since $i)$ we have shown that there exist two patterns, *with common EDR*, $\hat{\alpha}'$, and with the same *joint* properties (overall transmission rate and MIR) of the original patterns; and $ii)$ the RR procedure only replaces and/or reassigns messages, it is expected that the joint MIR of a botnet with EDR $\hat{\alpha}'$ is lower than $\hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2}$. Otherwise stated, it is reasonable to assume that $\hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2}$, even if not coinciding with the upper bound $\hat{\rho}_{\text{sum}}(\mathcal{S}_1, \mathcal{S}_2)$ in (25), is still sufficiently far from the lower bound $\hat{\rho}_{\text{bot}}(\mathcal{S}_1, \mathcal{S}_2)$. These considerations, for small ϵ, implicitly define the following identification condition.

Botnet Identification Condition (BIC)

Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be two subnets with $\mathcal{S}_1 \bigcap \mathcal{S}_2 = \emptyset$. *If at least one of the subnets is composed only by normal users*:

$$\boxed{\hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2} \geq \gamma(\mathcal{S}_1, \mathcal{S}_2)} \tag{27}$$

We remark that the case of $\mathcal{S}_1$ arbitrary vs. $\mathcal{S}_2$ arbitrary is not dealt with. This is not unintentional, since, as it will be clear from Theorem 2, the two situations discussed are sufficient to devise a *consistent* botnet identification algorithm.

[5] Actually, since we exclude the replacing messages m_1^* or m_2^* from the reassignment procedure, a subnet cannot pass all its distinct messages. However, for large t the contribution of a *single* message becomes irrelevant.

[6] We remark that the aforementioned result does not relate in any way to the deterministic or Poisson scheduling and to the random message picking that characterize the class of DDoS attacks considered in the present work.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

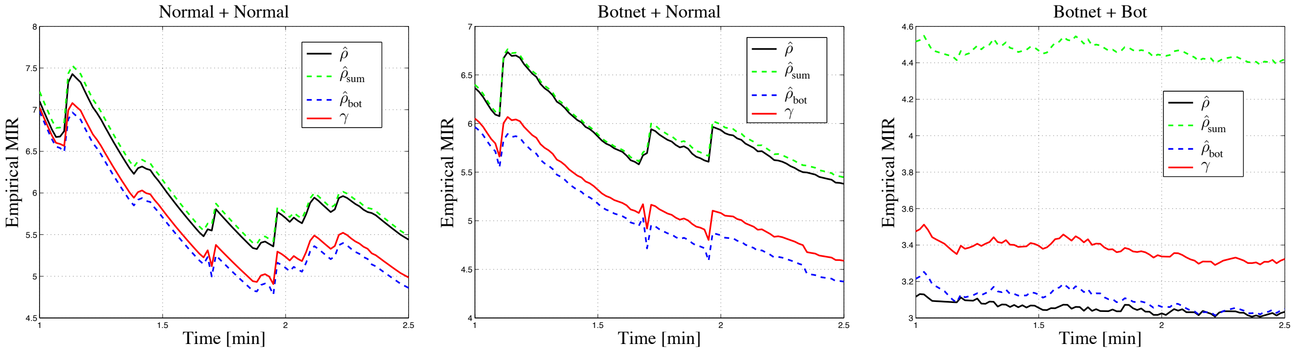


Fig. 2. Time evolution of the empirical message innovation rate $\hat{\rho}$ (solid, black), compared to the identification threshold γ (solid, red). For comparison purposes, the upper bound corresponding to the case of disjoint dictionaries, $\hat{\rho}_{\text{sum}}$ (dashed, green), and the lower bound corresponding to the botnet case, $\hat{\rho}_{\text{bot}}$ (dashed, magenta) are displayed. Moving from left to right, the different panels refer to i) the union of two normal users; ii) the union of a botnet of size 10 and a normal user; and iii) the union of a botnet of size 10 and a bot.

In summary, we end up with the following recipe:

$$\mathcal{S}_1 \text{ AND } \mathcal{S}_2 \text{ contain only bots} \quad \Rightarrow \quad \hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2} < \gamma(\mathcal{S}_1, \mathcal{S}_1), \tag{28}$$

$$\mathcal{S}_1 \text{ OR } \mathcal{S}_2 \text{ contain only normal users} \quad \Rightarrow \quad \hat{\rho}_{\mathcal{S}_1 \cup \mathcal{S}_2} \geq \gamma(\mathcal{S}_1, \mathcal{S}_1). \tag{29}$$

In Fig. 2, we illustrate the significance of the BIC. The normal users' activity refers to a monitoring campaign conducted in a testbed environment. The bots' activity has been generated according to the model described in Sec. III. The details of such a campaign will be given in the forthcoming section. In all the three panels we display: the empirical MIR, the threshold γ in (26), along with its upper ($\hat{\rho}_{\text{sum}}$) and lower ($\hat{\rho}_{\text{bot}}$) bounds. An observation window of 2.5 min is considered. All the relevant quantities are updated each 1 s, and both quantities are displayed as functions of time, in the interval between 1 and 2.5 min.

In the leftmost panel, we address the case of a pair of normal users. We see that the MIR stays (slightly) below the upper bound, meaning that a certain degree of correlation exists. However, *the MIR stands clear above the threshold, as prescribed by (29), and confirming the validity of the BIC*.

In the middle panel, the two subnets under test, $\mathcal{S}_1$ and $\mathcal{S}_2$, are a botnet of size 10, and a normal user, respectively. Conclusions similar to those pertaining to a normal-normal pairing can be drawn, substantiating again the BIC. We further see that, at the beginning of the observation window, the activities of the two subnets are almost independent, i.e., the MIR essentially matches the upper bound. As time elapses, a certain degree of correlation appears, *but the MIR still stays above the threshold*.

Finally, in the rightmost panel, the case of a botnet/bot interaction is addressed. We see that the empirical MIR: i) approaches, as time elapses, the quantity $\hat{\rho}_{\text{bot}}$, in perfect agreement with Theorem 1, and ii) stands clear below the threshold, in perfect agreement with (28).

In summary, the picture obtained from the above analysis reveals that the theoretical findings of Theorem 1, as well as the conjectured behavior of the normal users implied by the BIC, are confirmed *over the experimental network traces*.[7]

Before dwelling on the detailed description of the botnet identification algorithm, it is worth commenting on a possible limitation of the proposed approach. There might be particular situations where the BIC is violated because some normal users, even if acting in uncoordinated manner, exhibit a certain degree of correlation. Since the identification algorithm is presumed to discover dependencies among traffic patterns, it could erroneously declare the (honest) machines as bots. One example of the aforementioned situations relates to the influence of traffic patterns of distinct machines that perform the same automated update mechanisms of programs and/or operating systems, i.e., during daily/weekly update cycle.

V. The BotBuster Algorithm

We now focus on the derivation of the inference algorithm aimed at disclosing a botnet possibly hidden in the network. The *BotBuster* algorithm is described by the pseudo-code reported in the right column above, and basically exploits the fact that, given two disjoint subnets, the BIC allows to discriminate the situation where both subnets are part of a botnet, from the situation where at least one of them is made of normal users. We shall show that the proposed algorithm possesses the fundamental requirement of *consistency*, namely, *the guarantee that the botnet is correctly identified as t grows*.

Let us examine how the algorithm works. First, note that a botnet made of one user, besides making little sense in practice, is by definition non-identifiable, since we assumed that the characteristics of the messages at a single-user level do not reveal any special information. Now, at the beginning of the algorithm, user 1 is initially declared as a bot, namely, $\hat{\mathcal{B}} = \{1\}$. Then, it is checked whether users 1 and 2 form a botnet. If so, $\hat{\mathcal{B}} = \{1, 2\}$ is taken as the current botnet estimate. If not, $\hat{\mathcal{B}} = \{1\}$ is retained. Then, it is checked whether the currently estimated botnet $\hat{\mathcal{B}}$ forms a bot with user 3, and so on. At the end of the inner loop, the algorithm ends up with

[7] Our experiments have been repeated for many pairs of normal users. For illustrative purposes, in Fig. 2 we report one sample of such experiments, which is representative of the observed behavior. The quantitative analysis addressing the average behavior across users is deferred to Sec. VI.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

```
Algorithm: $\hat{\mathcal{B}}_{\text{new}}$=BotBuster
$\mathcal{N} = \{1, 2, \dots, N\}$; $\hat{\mathcal{B}}_{\text{new}} = \emptyset$;
for $b_0 \in \mathcal{N}$ do
    $\hat{\mathcal{B}} = \{b_0\}$;
    for $j \in \mathcal{N} \setminus \{b_0\}$ do
        if $\hat{\rho}(\hat{\mathcal{B}} \cup \{j\}) < \gamma(\hat{\mathcal{B}}, \{j\})$ then
            $\hat{\mathcal{B}} = \hat{\mathcal{B}} \bigcup \{j\}$;
        end
    end
    if $|\hat{\mathcal{B}}| > \max(1, |\hat{\mathcal{B}}_{\text{new}}|)$ then
        $\hat{\mathcal{B}}_{\text{new}} = \hat{\mathcal{B}}$;
    end
end
```

an estimate $\hat{\mathcal{B}}$. If the cardinality of the estimated set is greater than one, it is taken as a current estimate.

The procedure is then restarted by choosing user 2 as initial pivot, and sequentially checking the remaining users as explained before. At the end of the inner loop, the algorithm ends up with another estimate $\hat{\mathcal{B}}$. If the cardinality of the estimated set is greater than one *and* greater than the cardinality of the previously estimated set[8], then it is taken as a current estimate. Otherwise, the previous estimate is retained. The procedure ends when all users have been scanned as pivots.

We see that, under the BIC, all checks performed by the algorithm will give eventually the right answer, with probability tending to 1 as $t \to \infty$. BotBuster is accordingly expected to provide a *consistent* botnet estimator, as will be stated and proved in the forthcoming Theorem 2. The algorithm complexity is $\mathcal{O}(N^2)$ (only pairwise checks are performed), which is definitely tolerable, since we are seeking, within a network of size N, a subset of unknown size that matches some prescribed conditions. Finally, the looping structure of the algorithm makes it naturally open to parallelization, which is especially important for large networks.

In order to quantify the algorithm performance, we need to choose some meaningful indicators. With reference to a network $\mathcal{N} = \{1, 2, \dots, N\}$, containing a botnet $\mathcal{B}$, and letting $\hat{\mathcal{B}}(t)$ be the botnet estimated at time t by BotBuster, we introduce the following performance indices:

$$\eta_{\text{bot}}(t) = \frac{\mathbb{E}[|\hat{\mathcal{B}}(t) \cap \mathcal{B}|]}{|\mathcal{B}|}, \quad \eta_{\text{nor}}(t) = \frac{\mathbb{E}[|\hat{\mathcal{B}}(t) \cap (\mathcal{N} \setminus \mathcal{B})|]}{|\mathcal{N} \setminus \mathcal{B}|}, \tag{30}$$

namely, the expected fraction of *correctly banned users* (i.e., discovered bots), and the expected fraction of incorrectly-banned users (i.e., normal users erroneously declared as bots). Clearly, $\eta_{\text{bot}}(t)$ (resp., $\eta_{\text{nor}}(t)$) is not defined when $\mathcal{B} = \emptyset$ (resp., when $\mathcal{B} = \mathcal{N}$). We would like to see $\eta_{\text{bot}}(t) \to 1$, and $\eta_{\text{nor}}(t) \to 0$ as t goes to infinity. Under the ideal assumption that the BIC is always verified, such requirement is in fact fulfilled, as stated in the following theorem.

Theorem 2 (Consistency of BotBuster): Consider a network $\mathcal{N} = \{1, 2, \dots, N\}$, containing a botnet $\mathcal{B}$, with $|\mathcal{B}| \neq 1$, launching a randomized DDoS attack. The bots' transmission policies are either synchronous with constant transmission rate, or independent Poisson processes, while the normal users' transmission policies are arbitrary. Then, for any finite emulation dictionary rate α, the algorithm BotBuster is consistent, namely,

$$\boxed{\lim_{t\to\infty} \eta_{\text{bot}}(t) = 1, \qquad \lim_{t\to\infty} \eta_{\text{nor}}(t) = 0} \tag{31}$$

The claim for the case $\mathcal{B} = \emptyset$ (resp., $\mathcal{B} = \mathcal{N}$) is intended to hold with reference solely to $\eta_{\text{nor}}(t)$ (resp., to $\eta_{\text{bot}}(t)$).

Proof: See Appendix B. ■

Theorem 2 reveals that the botnet estimated by BotBuster converges to the *true* one as time elapses. The fundamental requirement enabling such result is the BIC validity. On the other hand, in real-world applications, the assumption that the BIC is verified *for all* normal/normal and botnet/normal interactions, as well as *for all* time epochs, is surely an *over-idealized* one. It cannot be excluded that, occasionally, two independent users feature an unusual degree of superposition between their empirical dictionaries, giving rise to spurious clusters of normal users that might be erroneously included in the estimated botnet. What is expected to be true even in real-world applications, is that such cases are rare and that the clusters' cardinality is small. Now, since the algorithm selects the estimate $\hat{\mathcal{B}}$ with the *highest* cardinality, and since *distributed* DoS attacks with small botnet sizes make little sense, estimated botnets of unreasonably small cardinality should be easily ruled out by BotBuster. As a result, the final estimate is likely to contain the true botnet, plus (possibly) a small fraction of normal users. Thus, *even under non-ideal operation conditions*, it is expected that $\eta_{\text{bot}}(t) \to 1$ as $t \to \infty$, whereas $\eta_{\text{nor}}(t)$ possibly takes on some small value.

VI. Experimental Results

A. Network Traces Collection and DDoS Attack Generation

As regards the measuring stage that precedes the botnet identification algorithm, we adopt the following pipeline. Packets are preliminarily filtered by using a popular software package for packet capturing and network protocol analysis. At the output of such preliminary filtering stage: $i)$ only the traffic directed to the destination that is being monitored is retained; $ii)$ among the surviving packets, only the application-layer traffic is retained; $iii)$ the resulting packets are divided on the basis of their source IP address, and are finally fed to the botnet identification algorithm.

A popular e-commerce website has been selected as target destination of the attack. Clearly, the normal users have no attacking intent, they perform ordinary surfing activity. About 20 min of (application-layer) traffic have been collected, from 10 independent users, which were students and researchers working in our laboratory, and carrying on their surfing activity almost independently. In order to help understanding the nature and significance of the dataset, we report that the total number of TCP flows is about 26800, the median of flows across users

[8] When t is large and the BIC is *perfectly* verified, the inner loop ends with either an empty set or the true botnet. Thus, selecting the estimate with the highest cardinality might appear redundant. Such operation is instead useful when operating under non-ideal conditions, as we shall explain soon.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

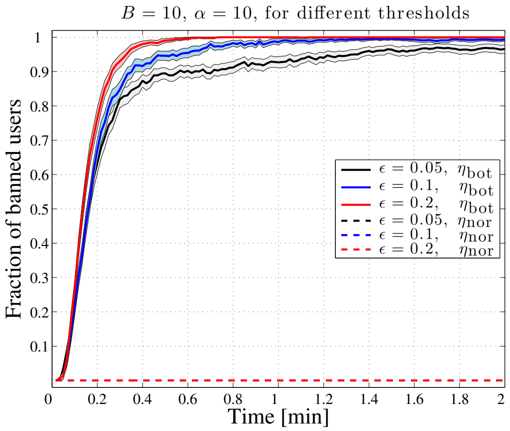


Fig. 3. Fraction of banned users as a function of time, for different values of the threshold parameter ϵ. The monitored network is composed of 10 normal users, and contains $B = 10$ bots. Solid curves refer to correctly banned bots, whereas dashed curves refer to erroneously banned normal users. The depicted curves are computed over 100 Monte Carlo trials. Per each trial, 2-min chunks of each user are randomly selected among the available chunks. Shaded areas correspond to 95% confidence intervals.

is 2846, the minimum number of flows is 1042, the maximum number of flows is 3925, and the average packet size is 776 bytes. Supported by these numbers, and by a trace-by-trace inspection, we conclude that the activity of the users during the monitored period is reasonably sustained, and compatible with typical traffic, meaning that the patterns are neither trivial (users effectively send requests) nor anomalous (users do not overload the destination with huge rates).

The collected streams have been partitioned into chunks of 2 min. In the forthcoming analysis we take two perspectives. In one scenario, the number of normal users is 10, each user has multiple 2-min chunks, and, per each trial, we choose randomly one trace per user. In the other scenario, 2-min chunks belonging to the *same* user have been treated as if they were coming from *distinct* users. In this way, we multiplied (fictitiously) the number of normal users. This is clearly an approximation, since, e.g., fictitious users stemming from the same user might feature an additional-and-spurious degree of dependence. On the other hand, this (possible) increase of dependence goes in the direction of (possibly) increasing the fraction of normal users mistakenly marked as bots. Therefore, the simulations performed in the "multiplied" scenario are expected to provide a conservative performance assessment.

The DDoS attack has been generated so as to fall into the class described in Sec. III. Given the dictionary of messages obtained from the *whole* activity recorded in the laboratory, it is assumed that, at epoch t, only the first $\lfloor e_0 + \alpha t \rfloor$ messages of such a dictionary are available to the botnet, giving rise to the emulation dictionary $\mathscr{E}(t)$, for fixed parameters e_0 (size of the dictionary at $t = 0$) and α. Independently at each bot, a Poisson time-scheduling is randomly generated, and, per each transmission epoch t, each bot picks messages at random from the currently available $\mathscr{E}(t)$. In the forthcoming analysis, we shall generally set $e_0 = 100$. We shall remark explicitly when a different choice is adopted.

B. Setting the Threshold

We recall that our algorithm is non-parametric, namely, that it does assume knowledge neither of the transmission rates, nor of the parameters of the botnet emulation dictionary (e_0 and α). In contrast, the size of the network is obviously known. The only input parameter is the factor ϵ appearing into (26). In Fig. 3 we consider a network comprising 10 normal users plus 10 bots. The botnet EDR is $\alpha = 10$. We remark that such a value is compatible with some of the empirical values $\hat{\alpha}$ estimated over the normal users' traces. The BotBuster algorithm has been implemented for three values of the threshold parameter $\epsilon \in (0, 1)$, namely, $0.05, 0.1$, and 0.2, and the estimates obtained for the fraction of banned users have been averaged over 100 Monte Carlo trials. The observation window lasts 2 min, and the simulation points refer to the output of the algorithm taken each 1 s. We see that the dashed curves are in practice invisible, revealing that the estimated η_{nor} is almost zero for all the considered values of ϵ. This behavior should be contrasted to what will be observed later on in Fig. 7, where, in the absence of a botnet, the BIC was occasionally violated. However, as discussed at the end of Sec. V, the spurious-and-small estimated clusters containing normal users can be efficiently ruled out by the fact that the algorithm selects, as a final estimate, only the cluster with maximum size, which is expected to contain only bots.

With regard to the fraction of correctly identified bots, we see that η_{bot} increases as ϵ increases from 0.05 to 0.2. In fact, increasing ϵ makes it easier staying *below* the threshold, which facilitates the inclusion of a node in the estimated botnet.

The analysis summarized in Fig. 3 reveals that the choice of the threshold is not critical, and the algorithm offers excellent performance for a relatively large range of ϵ. Indeed, recall that $\epsilon \in (0, 1)$, and that ϵ must be "small", so that $\epsilon = 0.05$ up to 0.2 can be definitely considered a "large", flexible range.

C. Role of the Emulation Dictionary Rate

In Fig. 4, the different curves refer to three EDR values (which, we recall, is *not known* to the algorithm). The threshold parameter ϵ was set to 0.2. Let us start by examining the behavior of η_{nor}. We see that, irrespectively of the EDR value, η_{nor} stays approximately constant at 0, which matches our previous evidences and observations.

Let us switch to the analysis of η_{bot}. The lowermost curve corresponds to the highest EDR value considered in the figure, namely, to $\alpha = 50$. Compared to what we have observed in the network traces collected in our testbed environment, such an EDR is a kind of relatively high value. We see that the average percentage of correctly identified bots is relatively large ($>$ 80%), even at the beginning of the monitoring activity. Then, the estimated η_{bot} increases, approaching unity as time elapses, *in perfect accordance with the theoretical results of Theorem* 2.

Next we examine the incidence of the EDR on the algorithm performance. We see that the curves corresponding to η_{bot} move upward as α decreases. This sounds perfectly reasonable, since α quantifies the learning ability (i.e., the power) of the botnet. On the other hand, for each value of α, the performance must eventually reach the limiting value of

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

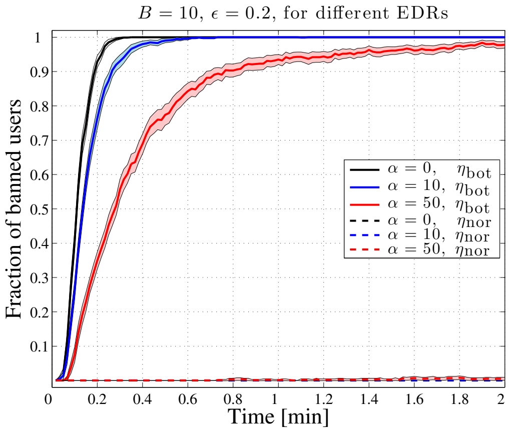


Fig. 4. Fraction of banned users as a function of time, for different values of the EDR α. The monitored network is composed of 10 normal users, and contains $B = 10$ bots. Solid curves refer to correctly banned bots, whereas dashed curves refer to erroneously banned normal users. The depicted curves are computed over 100 Monte Carlo trials. Per each trial, 2-min chunks of each user are randomly selected among the available chunks. Shaded areas correspond to 95% confidence intervals.

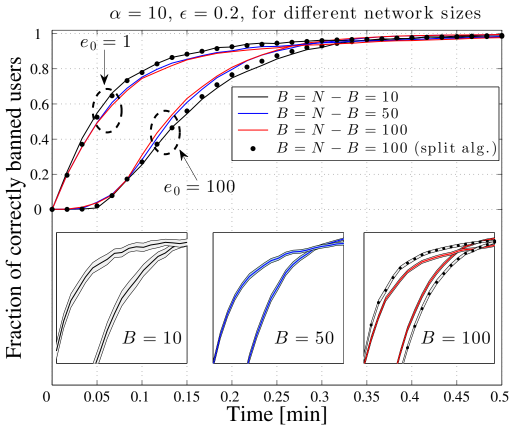


Fig. 5. Fraction of correctly banned users as a function of time, for different network sizes N ("multiplied" scenario — see main text). The two groups of curves refer to different sizes of the initial dictionary. Markers refer to a parallel implementation of the BotBuster algorithm over 10 disjoint and equal-size subsets of the entire network, for the case $B = N - B = 100$. The depicted curves are computed over 100 Monte Carlo trials. In the insets, a close-up of the curves in the range $\eta_{\text{bot}} \geq 0.5$ is displayed, reporting also the 95% confidence intervals (shaded areas).

unity after a sufficiently long time. In particular, the uppermost curve corresponds to the degenerate case $\alpha = 0$, namely, to the classical and well-documented case where the botnet uses repeatedly the same patterns. As such, the case $\alpha = 0$ could be addressed by other (simpler) tools, since a normal user will seldom feature such a small innovation rate. In summary, the conducted analysis emphasizes that the performance decreases with the botnet learning ability α.

D. Scaling with Network Size

In order to ascertain the feasibility of the proposed methodology, it is crucial to capture how the performance scales as the network size is increased.[9] In Fig. 5 we display the fraction of correctly banned users for networks of increasing size, featuring a balanced proportion of bots and normal users. The rightmost group of curves corresponds to a size of the initial dictionary chosen as done in the previous numerical experiments ($e_0 = 100$), while the leftmost group corresponds to the limiting case of an almost empty initial dictionary ($e_0 = 1$). As to the rightmost family of curves, the performance increases slowly when the size is varied from 20 to 100, while it stays almost constant when the size is varied from 100 to 200. The opposite behavior is observed for the leftmost group. Joint inspection of the two cases implies the following observations: i) no monotonic behavior emerges with respect to N; ii) the variation in performance when N is varied up to an order of magnitude is modest. The latter clues are crucial to support feasibility of the proposed algorithm.

In addition, in Fig. 5 we have reported the performance corresponding to a "partitioned" application of BotBuster (markers), where the largest network ($N = 200$) is examined by splitting the nodes into 10 sub-groups, and BotBuster is then launched in parallel over each sub-group. As it should be expected, such a parallel application delivers approximately the performance corresponding to an individual sub-group.[10] Otherwise stated, a parallel implementation over subnets, say, of $1/10$ of the total network size, keeps at least the promises of the performance corresponding to a network 10 times smaller, with additional computational and/or time savings. This is a further important element that supports feasibility of the proposed method. Moreover, it should be noticed that the implementation of BotBuster over the whole network (i.e., without splitting) offers the additional feature of performing joint checks among the sub-groups, which would be skipped by a parallel implementation.

To complete the analysis, we report that η_{nor} (not displayed to enhance readability) is practically zero in all the scenarios of Fig. 5, irrespectively of the network size. Finally, we stress that simulations were carried for networks up to 200 nodes, and the algorithm was able to guarantee the real-time requirement, yet on a standard laptop, without memory/code optimization, with such issues being beyond the scope of the work.

E. More Bots and/or Spoofed Addresses

The setting considered in this work encompasses naturally the relevant scenario of spoofed source IP addresses, which is becoming rather common in DDoS attacks. In such scenario, each bot can change its source IP address by (randomly) choosing from a collection of spoofed addresses. In the randomized DDoS attack considered in this work, the bot traffic streams are constructed by picking subsequent messages independently from an emulation dictionary that is shared among all the bots. Accordingly, a botnet of B nodes employing a set of A randomly spoofed addresses (with $A > B$), is equivalent to a botnet of A nodes performing the attack. Since the goal of the network analyst is banning the machines that launch the

[9] We shall consider the aforementioned scenario where normal users are fictitiously multiplied by treating chunks of the same user as distinct users.

[10] A rough performance prediction can be easily made assuming an approximate equipartition of bots among the sub-groups.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

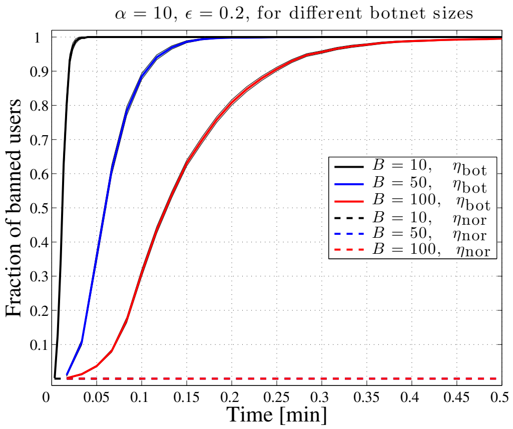


Fig. 6. Fraction of banned users as a function of time, for different botnet sizes B, in the constant attack-rate regime. The monitored network contains 100 normal users ("multiplied" scenario — see main text). Solid curves refer to correctly banned bots, whereas dashed curves refer to erroneously banned normal users. The depicted curves are computed over 100 Monte Carlo trials. Shaded areas correspond to 95% confidence intervals.

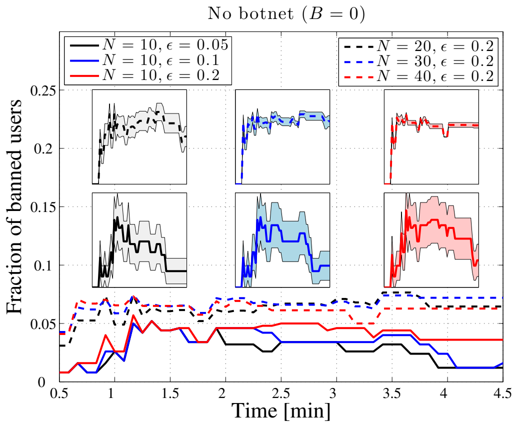


Fig. 7. Fraction of banned users as a function of time, for different values of the threshold parameter ϵ, and for different network sizes. The monitored network contains no bots. The depicted curves are computed over 100 Monte Carlo trials. Per each trial, 5-min chunks of each user are randomly selected among the available chunks. In the insets, the pertinent curves are displayed along with the 95% confidence intervals (shaded areas).

attack (not associating a physical machine to its IP address), we conclude that the performed analysis applies directly to the case of spoofed IP addresses, provided that the number of bots is replaced by the number of IP addresses globally employed by the botnet. For the sake of brevity, such "effective" number will be still denoted by B.

There are at least two meaningful regimes to examine the case of increasing number of bots and/or spoofed addresses: $i)$ the regime where B increases, while the individual bots' transmission rate, λ_{bot}, is constant, implying a growth of the total DDoS attacking rate $B\lambda_{\text{bot}}$; $ii)$ the regime where B increases while keeping the attacking rate constant. As regards the former regime, differently from the analysis of the previous section, varying B corresponds to varying the relative proportion of bots and normal users. This notwithstanding, the evidences arising from the simulation pertaining to such scenario are very similar to those observed in Fig. 5, and are accordingly not reported. In summary, in this regime the dependence of η_{bot} upon B is not obvious (no monotonic behavior emerges with respect to B, which is partly explained by noting that increasing B should augment the botnet "visibility", but also the number of possible algorithm mistakes), and the performance is little sensitive to variations of B.

Let us now move on to examine the second regime of operation. It is expected that, for a given total attacking rate, the botnet has more convenience in distributing its requests over more bots and/or spoofed addresses, which corresponds to increasing B while proportionally reducing λ_{bot}. Such scenario is illustrated in Fig. 6. Since in this case increasing B implies a reduction of the transmission rate, it is expected that the time to reach convergence increases, and that the mutual dependencies are disseminated over a larger number of bots, resulting into a reduced botnet identifiability. Such behavior is reflected by the shifting of the curves in Fig. 6.

From a practical perspective, there are two important evidences arising from Fig. 6. First, increasing B by even one order of magnitude does not jeopardize botnet identifiability ($\eta_{\text{bot}} \approx 0.99$ before half a minute). Such a mild dependence is supported by the following observation: convergence of the algorithm should roughly depend upon the average number of transmissions $\lambda_{\text{bot}} t$. Accordingly, the curves are expected to undergo a delay shift that scales approximately as $1/\lambda_{\text{bot}}$, i.e., almost linearly with B because the product $B\lambda_{\text{bot}}$ is constant.

More remarkably, we see that a relatively strong botnet power is required in order to impair substantially the algorithm performance. Let us now see why. The rate of requests of the "normal" part of the network scales as $(N - B)\lambda_{\text{nor}}$. Assume that the request rate necessary to saturate the attacked website resources must be κ times larger than the overall rate of normal users. This implies the condition $\lambda_{\text{bot}}/\lambda_{\text{nor}} = \kappa(N - B)/B$. Now, the rightmost curve (the more advantageous for the botnet) corresponds to the case that $\kappa = 1$ and $\lambda_{\text{bot}} = \lambda_{\text{nor}}$, namely, to the optimistic (from the botnet perspective) assumptions that a relatively low attack rate is sufficient to impair the target site ($\kappa = 1$), and that the available number of bots and/or spoofed addresses equals the number of normal users. In contrast, the leftmost curve corresponds to the (perhaps more realistic) case that the botnet is one order of magnitude smaller than the "normal" part of the network. In order to increase substantially the time needed to reach a satisfying accuracy, the botnet should use a number of bots that exceeds the number of normal users by at least one order of magnitude.

F. Absence of Botnet ($B = 0$)

One might question that, in an Intrusion Detection System (IDS) pipeline, a botnet identification algorithm is usually triggered by (or coupled with) a detection stage hunting for anomalous rises in the request rate. Thus, in the absence of such anomaly, identifying a dependence among small group of users does not imply (nor legitimate) a banning action. Nevertheless, we find it useful to verify that the algorithm works properly even with $B = 0$. Such analysis is also an indirect validation of the BIC, since it focuses specifically on the dependencies among normal users.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

In Fig. 7, we display the fraction of erroneously banned users, η_{nor}, for several scenarios, namely: a network with 10 normal users, for three values of the threshold parameter ϵ; networks of increasing size, examined for the same value of ϵ. Differently from the previous analysis, here we focus on a longer observation window, in order to track possible (undesirable) increasing trends of η_{nor}.

Let us start by examining the dependence upon the threshold parameter ϵ. Were the BIC exactly verified for any subset of normal users, and for any time epoch, the fraction of banned users should be always zero. As already discussed, in practice the BIC is expected to be verified approximately. This notwithstanding, in Fig. 7 we see that the percentage of erroneously banned users is very small for all the thresholds in the considered range, never exceeding 6%. Notably, such behavior suggests that a BIC violation is unlikely to occur, and that, in any case, it involves small groups of users. In summary, we conclude that, as happens for the case $B > 0$, the dependence of performance upon the threshold is not critical.

We move on to examine the variation in performance as the number of users increases. We see that the performance approximately increases (see curves corresponding to $N = 10, 20, 30$), but then approximately decreases ($N = 40$). This behavior suggests that, as N grows, the cardinality of spurious clusters is not significantly influenced by the latter growth, which in turn implies a reduction in the relative *fraction* of erroneously banned users. In summary, we conclude that, as already shown for the case $B > 0$, the scaling of performance with the network size does not emerge as a critical issue.

VII. Concluding Remarks

We considered Distributed Denial of Service (DDoS) attacks launched by bots that are capable to learn the application layer interaction possibilities, so as to avoid repeating one simple operation many times. Such enhanced capability of the attacker makes it impossible to identify one of those many bots relying only on its individual activity patterns.

The main contributions of this work are as follows: i) we introduced a formal model for the class of randomized DDoS attacks with increasing emulation dictionary; ii) we proposed an inference algorithm aimed at identifying the botnets executing such advanced DDoS attacks, and we ascertained consistency of the algorithm, namely, the property of revealing the true botnet as time elapses; iii) we evaluated the proposed methodologies on a testbed environment.

To give a snapshot of the performance delivered by the BotBuster algorithm: for a network with 100 normal users and 100 bots, 90% of the bots are correctly guessed in about a quarter of minute, while the fraction of normal users that are incorrectly banned is in practice zero.

There are many questions that remain open, and that might deserve further investigations. To mention a few: testing the algorithm over *more datasets*, in order to examine the impact on performance of the nature of the site under attack, and or the different behaviors of users surfing on the web; conducting a refined convergence analysis in order to characterize, from an analytical viewpoint, the *time needed to reach a prescribed accuracy*, and the dependence of such time upon the network/botnet size and other relevant system parameters; examining the problem from an *adversarial* perspective where the botnet-identification strategy and the kind of DDoS attack are jointly optimized by looking for equilibrium solutions that manage the attacker's and defender's conflicting requirements; generalizing the theoretical analysis and tools to *multi-clustered* DDoS attacks, where several botnets (using different emulation dictionaries) launch simultaneously their attack.

Appendix A

In the following, the symbol $o(g_n)$ will denote a function such that $o(g_n)/g_n \to 0$ as $n \to \infty$. Also, when convenient for notational reasons, the expectation of X is denoted by $\bar{X}$.

Proposition 1 (Useful recursion): Let $a, c > 0$, $b \in \mathbb{R}$, $n \in \mathbb{N}$, $\eta_n = 1 - 1/(c + an)$, and $f_n = \eta_n \, f_{n-1} + b$. We have:

$$f_n = f_0 \prod_{\ell=1}^{n} \eta_\ell + b \left(1 + \sum_{k=2}^{n} \prod_{\ell=k}^{n} \eta_\ell \right), \tag{32}$$

or:

$$f_n = f_0 \prod_{\ell=1}^{n} \eta_\ell + \frac{ab}{1+a} \left[n + \left(1 + \frac{c}{a}\right) \left(1 - \prod_{\ell=1}^{n} \eta_\ell \right) \right], \tag{33}$$

and the following limit holds:

$$\boxed{\lim_{n\to\infty} \frac{f_n}{n} = \frac{ab}{1+a}} \tag{34}$$

Proof: First, observe that:

$$f_1 = f_0 \eta_1 + b, \qquad f_2 = f_0 \eta_1 \eta_2 + b(1 + \eta_2), \ldots \tag{35}$$

which yields (32) by recursion. Let now $\hat{f}_n$ denote the RHS in (33). By the induction principle, the claim in (33) will be proved if we show that $f_1 = \hat{f}_1$, and that

$$\boxed{f_n = \hat{f}_n \Rightarrow f_{n+1} = \hat{f}_{n+1}} \tag{36}$$

Making explicit the definition of η_1 where needed, we have:

$$\hat{f}_1 = \eta_1 f_0 + \frac{ab}{1+a}(1 + 1/a) = \eta_1 f_0 + b = f_1. \tag{37}$$

Assuming now that $f_n = \hat{f}_n$, we can write

$$f_{n+1} = \eta_{n+1} \, \hat{f}_n + b = f_0 \prod_{\ell=1}^{n+1} \eta_\ell - \frac{ab}{1+a} \left(1 + \frac{c}{a}\right) \prod_{\ell=1}^{n+1} \eta_\ell$$
$$+ \underbrace{\frac{ab}{1+a} \left(n + 1 + \frac{c}{a}\right) \left(1 - \frac{1}{c + a(n+1)}\right) + b}_{= \frac{ab}{1+a}\left[n+1+\left(1+\frac{c}{a}\right)\right]} = \hat{f}_{n+1}. \tag{38}$$

Finally, the claim in (34) follows by observing that the term $\prod_{\ell=1}^{n} \eta_\ell$ in (33), vanishes as $n \to \infty$. ■

Corollary 1 (Small perturbations): Let $a, b > 0$, $n \in \mathbb{N}$, and let f_n be a nonnegative sequence such that:

$$f_n \le f_{n-1} \left(1 - \frac{1}{an + o(n)} \right) + b + o(1). \tag{39}$$

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

Then:

$$\limsup_{n\to\infty} \frac{f_n}{n} \le \frac{ab}{1+a}. \tag{40}$$

If the inequality in (39) is reversed, the constant b can be relaxed to be an arbitrary real number, and:

$$\liminf_{n\to\infty} \frac{f_n}{n} \ge \frac{ab}{1+a}. \tag{41}$$

Proof: Clearly, it suffices to prove (40). In the following, $\epsilon > 0$ is an arbitrarily small constant. For n large enough, and for all $c \in \mathbb{R}$, we have:

$$0 < 1 - \left(\frac{1}{an + o(n)}\right) \le 1 - \frac{1-\epsilon}{c + an}. \tag{42}$$

Moreover, we have $b + o(1) \le b + \epsilon$. Since f_n is nonnegative by assumption, a certain n_0 exists, such that, for all $n > n_0$:

$$f_n \le f_{n-1}\left(1 - \frac{1-\epsilon}{c+an}\right) + b + \epsilon. \tag{43}$$

Introducing, for $m = 1, 2, \dots,$ the definition

$$\eta_m = 1 - \frac{1-\epsilon}{c + a(n_0 + m)} = 1 - \frac{1}{\frac{c+an_0}{1-\epsilon} + \frac{a}{1-\epsilon} m}. \tag{44}$$

from (43) we get, by recursion:

$$f_{n_0+m} \le f_{n_0} \prod_{\ell=1}^{m} \eta_\ell + (b+\epsilon)\left(1 + \sum_{k=2}^{m}\prod_{\ell=k}^{m} \eta_\ell\right). \tag{45}$$

In view of (44), Proposition 1 allows to conclude that:

$$\limsup_{n\to\infty} \frac{f_n}{n} \le \frac{\frac{a}{1-\epsilon}(b+\epsilon)}{1 + \frac{a}{1-\epsilon}}, \tag{46}$$

and, hence, the claim in (40) follows from arbitrariness of ϵ. ■

Proof of Theorem 1: First, we prove the claim for the synchronous scheduling, where all bots transmit regularly at intervals of constant duration $\tau = 1/\lambda$. Accordingly, we consider a *slotted* system with discrete time index $n \ge 0$, and introduce the quantities:

$$\mathscr{D}_n \triangleq \mathscr{D}_{\mathcal{B}}(n\tau),\ M_n \triangleq |\mathscr{D}_n|,\ \mathscr{E}_n \triangleq \mathscr{E}(n\tau),\ e_n \triangleq |\mathscr{E}_n|, \tag{47}$$

where we further observe that:

$$\lim_{n\to\infty} \frac{e_n}{n\tau} = \alpha \Rightarrow e_n = \alpha\tau\, n + o(n). \tag{48}$$

Now, for the synchronous case, it suffices to show that:

$$\frac{M_n}{n\tau} \xrightarrow{\text{p}} \frac{\alpha B\lambda}{\alpha + B\lambda} \Leftrightarrow \frac{M_n}{n} \xrightarrow{\text{p}} \frac{\alpha\tau\, B}{\alpha\tau + B} \triangleq \rho, \tag{49}$$

where B is the cardinality of subnet $\mathcal{B}$. Observe preliminarily that, by the orthogonality principle, we can write:

$$\mathbb{E}\left[\left(\frac{M_n}{n} - \rho\right)^2\right] = \mathbb{E}\left[\left(\frac{M_n - \bar{M}_n}{n}\right)^2\right] + \left(\frac{\bar{M}_n}{n} - \rho\right)^2, \tag{50}$$

and, since mean-square convergence implies convergence in probability [28], it suffices to show that, as $n \to \infty$, both terms on the RHS in (50) vanish.[11] We start by showing that $\bar{M}_n/n \to \rho$. At time n, the probability that k bots out of B pick a message outside $\mathscr{D}_{n-1}$ is (conditionally on M_{n-1}):

$$\binom{B}{k}\left(1 - \frac{M_{n-1}}{e_n}\right)^k \left(\frac{M_{n-1}}{e_n}\right)^{B-k}. \tag{51}$$

Let us introduce the binomial random variable $\hat{X}_n$, with probability mass function given by (51), whose (conditional) expectation and variance are:

$$\mathbb{E}[\hat{X}_n | M_{n-1}] = B\left(1 - \frac{M_{n-1}}{e_n}\right), \tag{52}$$

and

$$\text{VAR}[\hat{X}_n | M_{n-1}] = B\left(1 - \frac{M_{n-1}}{e_n}\right)\frac{M_{n-1}}{e_n}. \tag{53}$$

In order to build $\mathscr{D}_n$, we must select all the *distinct* messages among the k available ones. Ignoring repetitions, we can write:

$$M_n \le M_{n-1} + \hat{X}_n, \tag{54}$$

and, taking expectations:

$$\bar{M}_n \le \bar{M}_{n-1}\left(1 - \frac{1}{\alpha\tau n/B + o(n)}\right) + B, \tag{55}$$

having used (52) and the expression of e_n appearing on the RHS in (48). Direct application of Corollary 1 now yields:

$$\limsup_{n\to\infty} \frac{\bar{M}_n}{n} \le \frac{\alpha\tau\, B}{\alpha\tau + B}. \tag{56}$$

Let us now prove the above (reversed) inequality for the $\liminf$. To this aim, we split $\mathscr{E}_n$ into C non-overlapping cells:

$$\mathscr{E}_n = \bigcup_{c=1}^{C} \mathscr{E}_{c,n}, \quad \left\lfloor \frac{|\mathscr{E}_n|}{C} \right\rfloor \le |\mathscr{E}_{c,n}| \le \left\lfloor \frac{|\mathscr{E}_n|}{C} \right\rfloor + 1, \tag{57}$$

where C is an arbitrary integer. Since we focus on the regime where $n \to \infty$, it can be safely assumed that the initial number of words in the emulation dictionary obeys: $e_0 \ge C$. Let now:

$$\mathscr{D}_n = \bigcup_{c=1}^{C} \mathscr{D}_{c,n}, \quad M_{c,n} \triangleq |\mathscr{D}_{c,n}|, \quad M_n = \sum_{c=1}^{C} M_{c,n}, \tag{58}$$

and the events, for $j = 1, 2, \dots, B$, and $c = 1, 2, \dots, C$:

$$\mathcal{A}_{j,c} \triangleq \{\text{bot } j \text{ picks a message belonging to } \mathscr{E}_{c,n} \setminus \mathscr{D}_{c,n-1}\}. \tag{59}$$

Then we have, for any j:

$$\mathbb{P}[\mathcal{A}_{j,c} | M_{c,n-1}] = \frac{|\mathscr{E}_{c,n}| - M_{c,n-1}}{|\mathscr{E}_n|} \triangleq p_{c,n}, \tag{60}$$

with the dependence of $p_{c,n}$ upon $M_{c,n-1}$ being suppressed for ease of notation. From (57), we have:

$$\frac{1}{C} - \frac{1}{e_n} - \frac{M_{c,n-1}}{e_n} \le p_{c,n} \le \frac{1}{C} + \frac{1}{e_n} - \frac{M_{c,n-1}}{e_n}. \tag{61}$$

Now, $M_{c,n-1}$ increases by *at least* 1 whenever *at least* one bot picks a new message belonging to the c-th cell. This implies:

$$\mathbb{E}[M_{c,n} | M_{c,n-1}] \ge M_{c,n-1} + Bp_{c,n} - (Bp_{c,n})^2, \tag{62}$$

[11] In fact, we prove a stronger result in terms of *mean-square* convergence.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

$$\hat{\mathcal{B}}_1=\{1\},\quad \mathcal{E}_2=\left\{\hat{\rho}_{\hat{\mathcal{B}}_1\cup\{2\}}<\gamma(\hat{\mathcal{B}}_1,\{2\})\right\},\quad \mathcal{E}_3=\left\{\hat{\rho}_{\hat{\mathcal{B}}_2\cup\{3\}}<\gamma(\hat{\mathcal{B}}_2,\{3\})\right\},\ \dots\ \mathcal{E}_B=\left\{\hat{\rho}_{\hat{\mathcal{B}}_{B-1}\cup\{B\}}<\gamma(\hat{\mathcal{B}}_{B-1},\{B\})\right\},$$
$$\mathcal{E}_{B+1}=\left\{\hat{\rho}_{\hat{\mathcal{B}}_B\cap\{B+1\}}\geq\gamma(\hat{\mathcal{B}}_B,\{B+1\})\right\},\ \dots\ \mathcal{E}_N=\left\{\hat{\rho}_{\hat{\mathcal{B}}_B\cap\{N\}}\geq\gamma(\hat{\mathcal{B}}_B,\{N\})\right\}. \tag{73}$$

where we used the inequality $(1-p)^B\leq 1-Bp+(Bp)^2$. On the other hand, for large n and small $\epsilon>0$, from (61), we get $p_{c,n}^2\leq(1/C+1/e_n)^2\leq C^{-2}+\epsilon$, and, hence, from (62):

$$\mathbb{E}[M_{c,n}|M_{c,n-1}]\geq M_{c,n-1}+Bp_{c,n}-\left(\frac{B}{C}\right)^2-\epsilon', \tag{63}$$

for a certain small ϵ'. Conversely, using the lower bound in (61), and averaging over $M_{c,n-1}$, for large n we get:

$$\bar{M}_{c,n}\geq\bar{M}_{c,n-1}\left(1-\frac{B}{e_n}\right)+\frac{B}{C}\left(1-\frac{B}{C}\right)-\epsilon'', \tag{64}$$

for a certain small ϵ''. Summing over c, we get:

$$\begin{aligned}\bar{M}_n &\geq \bar{M}_{n-1}\left(1-\frac{B}{e_n}\right)+\underbrace{B\left(1-\frac{B}{C}\right)-C\,\epsilon''}_{b}\\ &= \bar{M}_{n-1}\left(1-\frac{1}{\alpha\tau/B+o(n)}\right)+b,\end{aligned} \tag{65}$$

having used e_n in (48). Invoking now Corollary 1, we obtain:

$$\liminf_{n\to\infty}\frac{\bar{M}_n}{n}\geq\frac{\alpha\tau\,b}{\alpha\tau+B}\geq\frac{\alpha\tau\,B}{\alpha\tau+B}, \tag{66}$$

where the latter inequality follows from the definition of b, since C and ϵ are arbitrary. Equation (66), along with (56), yields that the second term on the RHS in (50) vanishes. Let us switch to the first term in (50). In view of the ascertained convergence of expectations, the variance will be proved to vanish if we show that: $\mathbb{E}[M_n^2]/n^2\to\rho^2$. Now, in the light of (54), we can write: $\mathbb{E}[M_n^2|M_{n-1}]\leq M_{n-1}^2+\mathbb{E}[\hat{X}_n^2|M_{n-1}]+2\,M_{n-1}\mathbb{E}[\hat{X}_n|M_{n-1}]$, which, using (52) and (53), yields:

$$\begin{aligned}v_n &\leq v_{n-1}\frac{n-1}{n}\left[1-\frac{2B}{e_n}+\frac{B(B-1)}{e_n^2}\right]\\ &+ B\frac{\bar{M}_{n-1}}{n}\left(2-\frac{2B-1}{e_n}\right)+\frac{B^2}{n},\end{aligned} \tag{67}$$

having also introduced the definition $v_n\triangleq\mathbb{E}[M_n^2]/n$. Now, the first term appearing on the RHS can be represented as

$$v_{n-1}\left(1-\frac{1}{\frac{\alpha\tau}{\alpha\tau+2B}n+o(n)}\right). \tag{68}$$

Likewise, the second term appearing on the RHS in (67) can be written as $2B\rho+o(1)$. Applying Corollary 1, we get:

$$\limsup_{n\to\infty}\frac{\mathbb{E}[M_n^2]}{n^2}=\limsup_{n\to\infty}\frac{v_n}{n}\leq 2B\rho\,\frac{\frac{\alpha\tau}{\alpha\tau+2B}}{1+\frac{\alpha\tau}{\alpha\tau+2B}}=\rho^2. \tag{69}$$

Now, subadditivity of limit superior implies:

$$\begin{aligned}&\limsup_{n\to\infty}\mathbb{E}\left[\left(\frac{M_n-\bar{M}_n}{n}\right)^2\right]\\ &\leq \limsup_{n\to\infty}\frac{\mathbb{E}[M_n^2]}{n^2}+\limsup_{n\to\infty}\left(-\frac{\bar{M}_n^2}{n^2}\right)\leq 0,\end{aligned} \tag{70}$$

with the latter inequality coming from (69), and from $\bar{M}_n/n\to\rho$. The claim for the synchronous case is so proved.

As regards the Poisson case, we consider again the slotted system in (47), but for the fact that τ is now an arbitrarily *small* interval. Let A denote the number of transmission attempts in a single slot, that is, a Poisson random variable with expectation $\bar{A}=\sum_{u\in\mathcal{B}}\lambda_u\tau=\lambda_{\mathcal{B}}\tau$. Since the A transmissions correspond to A independent choices of messages from the emulation dictionary, for small τ the system behaves as if we had A synchronous bots, where A is now *random*. Thus, the proof for the Poisson case boils down to modify slightly the previous proof in order to take into account such additional randomness. Specifically, Eq. (55) should be modified by considering a random number of bots A, and then taking expectations, yielding:[12] $\bar{M}_n\leq\bar{M}_{n-1}(1-\bar{A}/e_n)+\bar{A}$. Likewise, Eq. (62) becomes: $\mathbb{E}[M_{c,n}|M_{c,n-1}]\geq M_{c,n-1}+1-\mathbb{E}[(1-p_{c,n})^A|M_{n-1}]$. Since, for the Poisson random variable A, it is easy to show that $\mathbb{E}[(1-p)^A]=e^{-\bar{A}p}\leq 1-\bar{A}p+(\bar{A}p)^2$, the conclusion in (49) still holds true, with B simply replaced by $\bar{A}$. Finally, the inequality in (67) becomes:

$$\begin{aligned}v_n &\leq v_{n-1}\frac{n-1}{n}\left[1-\frac{2\bar{A}}{e_n}+\frac{\overline{A(A-1)}}{e_n^2}\right]\\ &+ \frac{\bar{M}_{n-1}}{n}\left(2\bar{A}-\frac{\overline{A(2A-1)}}{e_n}\right)+\overline{A^2}.\end{aligned} \tag{71}$$

Having shown that all the equations used to prove the pertinent convergence hold true with B replaced by $\bar{A}$, we conclude that: $\frac{M_n}{n}\xrightarrow{\text{p}}\frac{\alpha\tau\,\bar{A}}{\alpha\tau+\bar{A}}=\frac{\alpha\lambda_{\mathcal{B}}}{\alpha+\lambda_{\mathcal{B}}}$. ■

Appendix B

Proof of Theorem 2: Let us focus on a single step of the BotBuster loop, i.e., the algorithm behavior for a fixed b_0. Consider first the case that b_0 is a normal user, and introduce, for $j\in\mathcal{N}\backslash\{b_0\}$, the events: $\mathcal{E}_j=\{\hat{\rho}_{\{b_0\}\cup\{j\}}\geq\gamma(\{b_0\},\{j\})\}$. Eq. (29) reveals that, for any j, $\mathbb{P}[\mathcal{E}_j]\to 1$ as $t\to\infty$. But we also have that, for b_0 normal,

$$\mathbb{P}[\text{inner loop outputs }\hat{\mathcal{B}}=\{b_0\}]=\mathbb{P}[\cap_{j\in\mathcal{N}\backslash\{b_0\}}\mathcal{E}_j]\to 1, \tag{72}$$

where the convergence follows by the fact that each of the events has probability converging to one as $t\to\infty$.

In contrast, if b_0 is a bot, we distinguish two cases: $i)$ if j is normal, from (29) we conclude that $\hat{\rho}_{\{b_0\}\cup\{j\}}\geq\gamma(\{b_0\},\{j\})$ with probability converging to one as $t\to\infty$, while $ii)$ if j is a bot, from (28) we conclude that $\hat{\rho}_{\{b_0\}\cup\{j\}}<\gamma(\{b_0\},\{j\})$ with probability converging to one as $t\to\infty$. Assume now, without loss of generality, that the first B users are bots, that

[12] We implicitly use: $i)$ the independence between scheduling policy and message picking, and $ii)$ the memoryless property of the Poisson process.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TIFS.2017.2692685

$b_0 = 1$, and that the remaining users are normal. In (73), we introduce the events corresponding to the inner loop over index j, as well as the associated botnet estimates at step j, denoted by $\hat{\mathcal{B}}_j$. After noticing that, in the definition of these events, the inequality signs in the threshold comparisons are different for $j \leq B$ and for $j > B$, it is seen that the event $\hat{\mathcal{B}} = \{1, 2, \dots, B\}$ corresponds to the event $\cap_{j=2}^{N} \mathcal{E}_j$. Since, in view of the above points $i)$ and $ii)$, we have $\mathbb{P}[\mathcal{E}_j] \to 1$, we conclude that (if $b_0 = 1$ is a bot):

$$\mathbb{P}[\text{inner loop outputs } \hat{\mathcal{B}} = \{1, 2, \dots, B\}] = \mathbb{P}[\cap_{j=2}^{N} \mathcal{E}_j] \to 1, \tag{74}$$

which implies the validity of (31). ■

References

[1] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 6th ed., Pearson, 2013.

[2] N. Hoque, D. Bhattacharyya, and J. Kalita, “Botnet in DDoS attacks: trends and challenges,” *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2242–2270, fourth quarter 2015.

[3] L. Feinstein, D. Schnackenberg, R. Balupari, and D. Kindred, “Statistical approaches to DDoS attack detection and response,” in *Proc. DARPA Information Survivability Conference and Exposition*, Washington, DC, USA, Apr. 2003, pp. 303–314.

[4] J. Yuan and K. Mills, “Monitoring the macroscopic effect of DDoS flooding attacks,” *IEEE Trans. Depend. Secure Comput.*, vol. 2, no. 4, pp. 324–335, Oct. 2005.

[5] L. Li, J. Zhou, and N. Xiao, “DDoS attack detection algorithms based on entropy computing,” in *Proc. ICICS 2007*, Zhengzhou, China, Dec. 2007, pp. 452–466.

[6] Y. Xiang, K. Li, and W. Zhou, “Low-rate DDoS attacks detection and traceback by using new information metrics,” *IEEE Trans. Inf. Forensics and Security*, vol. 6, no. 2, pp. 426–437, Jun. 2011.

[7] J. Luo, X. Yang, J. Wang, J. Xu, J. Sun, and K. Long, “On a mathematical model for low-rate shrew DDoS,” *IEEE Trans. Inf. Forensics and Security*, vol. 9, no. 7, pp. 1069–1083, Jul. 2014.

[8] “Layer 7 DDoS.” http://blog.sucuri.net/2014/02/layer-7-ddos-blocking-http-flood-attacks.html.

[9] “Taxonomy of DDoS attacks.” http://www.riorey.com/types-of-ddos-attacks/#attack-15.

[10] “Global DDoS threat landscape.” https://www.incapsula.com/blog/ddos-global-threat-landscape-report-q2-2015.html.

[11] S. Ferretti and V. Ghini, “Mitigation of random query string DoS via gossip,” *Commun. in Comput. and Inf. Sci.*, vol. 285, pp. 124–134, 2012.

[12] S. Marano, V. Matta, and L. Tong, “Distributed detection in the presence of Byzantine attacks,” *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 16–29, Jan. 2009.

[13] S. Marano, V. Matta, and P. Willett, “Distributed detection with censoring sensors under physical layer secrecy,” *IEEE Trans. Signal Process.*, vol. 57, no. 5, pp. 1976–1986, May 2009.

[14] M. Barni and B. Tondi, “The source identification game: an information theoretic perspective,” *IEEE Trans. Inf. Forensics and Security*, vol. 8, no. 3, pp. 450–463, Mar. 2013.

[15] B. Kailkhura, S. Brahma, B. Dulek, Y. S Han, and P. Varshney, “Distributed detection in tree networks: Byzantines and mitigation techniques,” *IEEE Trans. Inf. Forensics and Security*, vol. 10, no. 7, pp. 1499–1512, Jul. 2015.

[16] M. Mardani, G. Mateos, and G. B. Giannakis, “Dynamic anomalography: tracking network anomalies via sparsity and low rank,” *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 1, pp. 50–66, Feb. 2013.

[17] M. Mardani, G. Mateos, and G. B. Giannakis, “Recovery of low-rank plus compressed sparse matrices with application to unveiling traffic anomalies,” *IEEE Trans. Inf. Theory*, vol. 59, no. 8, pp. 5186–5205, Aug. 2013.

[18] M. Mardani and G. B. Giannakis, “Estimating traffic and anomaly maps via network tomography,” *IEEE/ACM Trans. Networking*, DOI: 10.1109/TNET.2015.2417809, date of publication, Apr. 2015.

[19] P. Venkitasubramaniam, T. He, and L. Tong, “Anonymous networking amidst eavesdroppers,” *IEEE Trans. Inf. Theory*, vol. 54, no. 6, Jun. 2008.

[20] T. He and L. Tong, “Distributed detection of information flows,” *IEEE Trans. Inf. Forensics and Security*, vol. 3, no. 3, pp. 390–403, Sep. 2008.

[21] T. He and L. Tong, “Detection of information flows,” *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 4925–4945, Nov. 2008.

[22] T. He, A. Agaskar, and L. Tong, “Distributed detection of multi-hop information flows with fusion capacity constraints,” *IEEE Trans. Signal Processing*, vol. 58, no. 6, pp. 3373–3383, Jun. 2010.

[23] J. Kim and L. Tong, “Unsupervised and nonparametric detection of information flows,” *Signal Processing*, vol. 92, no. 11, pp. 2577–2593, Nov. 2012.

[24] S. Marano, V. Matta, T. He, and L. Tong, “The embedding capacity of information flows under renewal traffic,” *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1724–1739, Mar. 2013.

[25] M. Barni and F. Pérez González, “Coping with the enemy: advances in adversary-aware signal processing,” in *Proc. IEEE ICASSP*, Vancouver, Canada, May 2013, pp. 8682–8686.

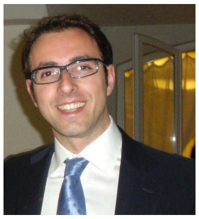
[26] M. Barni and B. Tondi, “Binary hypothesis testing game with training data,” *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 4848–4866, Aug. 2014.

[27] V. Matta, M. Di Mauro, and M. Longo, “Botnet identification in randomized DDoS attacks,” in *Proc. EUSIPCO*, Budapest, Hungary, Aug./Sep. 2016, pp. 2260–2264.

[28] H. Shao, *Mathematical Statistics*, 2nd ed., Springer, 2003.

[29] S. Ross, *Stochastic Processes*, 2nd ed. John Wiley & Sons, Inc., 1996.

Vincenzo Matta received the Laurea degree (cum laude) in Electronic Engineering and the PhD degree in Information Engineering from the University of Salerno, Italy, in 2001 and 2005, respectively. Currently, he is an Associate Professor at the Department of Information & Electrical Engineering and Applied Mathematics of the University of Salerno. His research interests cover the wide area of statistical signal processing and information theory, with current emphasis on: adaptation and learning over networks; the interplay between inference, communications and security in distributed systems; multi-object/multi-sensor tracking and data fusion; detection of gravitational waves. He has published about 100 articles, on international journals, and proceedings of international conferences. Vincenzo Matta serves as an Associate Editor for the IEEE Transactions on Signal and Information Processing over Networks,for the IEEE Signal Processing Letters, and for the IEEE Transactions on Aerospace and Electronic Systems.



Mario Di Mauro received the Laurea degree in Electronic Engineering from the University of Salerno in 2005 and the MS Degree in Networking from the University of L’Aquila jointly with Telecom Italia Centre in 2006. He worked as Research Engineer at CoRiTel (Research Consortium on Telecommunications, led by Ericsson Lab Italy) and then as Research Fellow at Salerno University. His main fields of interest include: network security, data analysis for telecommunication infrastructures, and reliability of complex systems. He is the author of several scientific papers, and of a patent on a telecommunication aid for impaired people. He is currently enrolled in a PhD program at the Graduate School of Information Engineering at Salerno University.

Maurizio Longo (MSEE, Stanford, 1977; Laurea in Electronic Eng., 1972, University of Napoli) is Full Professor of Telecommunications at the University of Salerno (Italy), where he also serves as the Director of the CoRiTel (Research Consortium on Telecommunications) Lab. and as the Chairman of the Graduate School of Information Engineering. He also held academic positions at the Universities “Federico II” and “Parthenope” in Napoli, the University of Lecce and the Aeronautical Academy. In 1986-87 and in 1990 he was on leave at Stanford University (California), as a Formez Fellow and as a NATO-CNR Senior Fellow. He is author of over 150 papers in international journals and conference proceedings, mainly in the in the fields of telecommunication networks and statistical signal processing.

Copyright (c) 2017 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.