How Social Networks can Help to Detect DDoS Attacks on DNS servers

R. Alonso-Rodríguez, J. Vázquez-Islas.,
L.A. Trejo-Rodríguez, R. Monroy, and E. Sánchez-Velázquez

Instituto Tecnológico y de Estudios Superiores de Monterrey,
Estado de México, México
{A01013374,A01165550,ltrejo,raulm,snora}@itesm.mx
http://homepage.cem.itesm.mx/raulm/netsec

Abstract. In this paper we propose a method for intrusion detection on DNS servers based on the social behavior of users. An interaction network is built to get user profiles with this behavior and we used mechanical methods on these profiles in order to determine if there is an attack present on the network that is affecting the performance of a recursive DNS server.

Key words: DNS, network security, graphs

1 Introduction

Domain Name System or DNS, constitutes the backbone of Internet, if someone asked us what is the function of DNS we can answer that this protocol translates names into IP addresses, but there are some cases on which the protocol also translate IP addresses into names [1], DNS is just a protocol of queries and responses to those queries. For instance, every page, web system, or service on the web has a logical identifier called IP address, then imagine a user that checks for his/her email at `http:\\mail.google.com`. `Google` is called a Domain, and `mail` a subdomain of Google, finally `.com` is a Top-Level Domain. This URL have an IP associated within, for instance 192.168.0.1, this number could be easy to remember, but an user not only checks for his/her email, but also stream videos, read the news, access an enterprise web system, etc. It becomes harder to remember all those addresses, so in 1984 was born a protocol called DNS, to map resources (computers, web pages, web services) and make it easy to access them.

DNS has become target of numerous DDoS (Distributed Denial of Service) attacks to its infrastructure, in 2002 an attack against DNS lasted about 9 hours an disable nine of the thirteen DNS root servers [2] another attack happens on February 2007 [3], this time six of the thirteen root servers were affected. The main purpose of the attack is to increment the volume of the invalid or even valid queries, in order to disable the capacity of the server to response to valid user queries. Due to the nature of DNS it is relatively simple to attack it, it's enough

to type a wrong URL on a web browser, to make a DNS server to ask for that domain to other high level DNS servers[4], and even an incorrect configuration of DNS servers can increment significantly invalid traces [5]. For instance in [6] the author presents an analysis for detecting misconfigurations at source DNS servers, by applying a statistical analysis of logs from a root DNS server. They applied techniques like PCA, LDA and K-means to cluster a data set. They validated the outcomes with DNS system operators.

There are numerous approaches to develop an effective IDS (Intrusion Detection System) on DNS. For instance, in [7] the authors present an algorithm to detect DDoS attacks under the assumption that the normal traffic presents fluctuations randomly, while the attacking traffic presents an increment trend and persistence feature. Even if the attacker generates fluctuations to mimic the attack, attack must maintain the persistent increment of the packets, until the system becomes saturated. In [8] they present a method for detecting DDoS attacks in a network by using entropy-based collaborative detection. They use the entropy and the entropy rate; to distinguish the malicious traffic from the normal one. They not only calculate the entropy to measure the uncertainty in a time slot, but also the entropy rate to probe that the malicious traffic follows almost the same entropy from the border routers through a network.

Those solutions have an approach based on volume of requests, analyzing with statistical methods the incomes requests of a DNS server or checking for other DNS features. There are few approaches for intrusion detection of DDoS attacks that cares about the content of DNS packet and there are fewer approaches that watch the domain users ask for. Our approach is about social behavior of the users, what we mean with social behavior is simple as analyzing which user asks for a particular domain, for that analysis we will use graphs to model the behavior, for convenience, that graph will be called an Interaction Network.

The paper is organized as follows. In section 2, we present the approach of social behavior profile construction based on graphs we called this graphs Interaction Networks. Section 3 explains the proposed methodology for Interaction Network analysis. In Section 4 we present the obtained (preliminary) results and discussion of these, finally in section 5 conclusions and future work is presented.

2 Interaction Networks

We purpose a model based on the social behavior that will be known as Interaction Network. The main purpose of modeling social behavior is to determine the characteristics of social profiles for instance, on a private network like an enterprise, or a university. Some of the characteristics will be, which user asks for a particular domain, how related is a user x with a user y, how many users has asked access for a particular domain, how similar are the visited domains of a user with the visited domains of another user, which domains are the less likely to be visited, etc.

In order to gather these characteristics we need firstly to relate an IP with a domain that IP asks for. So an acyclic directed graph is buit (Fig. 1).

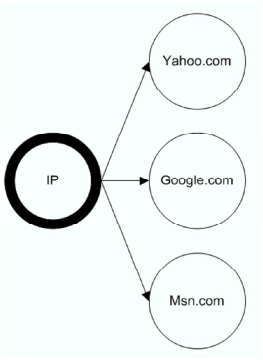


Fig. 1. Representation of a relation between an IP and a domain that IP asks for

An Interaction Network is defined as a set of relations present on a window analysis on which these relations, as previously told, are constructed when an IP query for a particular Domain. Some definitions of the model will be presented in this section, but also in subsection 2.1, 2.2, and 2.3; the definition of how is an IP and a Domain present on an interaction network, is as follows:

Let $\mathbb{W}$ be the set of all analysis windows, $\mathbb{I}$ the set of all IPs, and $\mathbb{D}$ the set of all the domains

Finally, let $qry_w(x, y)$ be a function that tell us if an IP x query for a Domain y on a window w.

Then, given $w \in \mathbb{W}$ the set of all active IPs on an analysis window will be:

$$I(w) = \{x \in \mathbb{I} | \exists y \in \mathbb{D}, qry_w(x, y)\}$$

The set of all visited Domains on a window should be defined similarly:

$$D(w) = \{y \in \mathbb{D} | \exists x \in \mathbb{I}, qry_w(x, y)\}$$

We can now deduce that a more complex structure is constructed, if we continue to relate this information, resulting in obtaining a graphical view of all the queries made to a DNS server. A social behavior could be observed in this graph because there are users visiting the same pages, or using the same web services. For instance, in a university network, students are more likely to visit amusement web pages than researchers, who in turn visit more scientific web sites than professors. This separation of user's behavior will be known as a group, and will be defined as follows:

Given

$$(w, I \subseteq I(w), D \subseteq D(w))$$

The tuple forms a group, `iff` the following condition is fulfilled:

$$\forall x \in I, y \in D \; qry_w(x, y)$$

It means, qry_w is exactly the Cartesian Product of $I \times D$ where,

- The size of the group is the cardinality of the set D, and
- The weight of the group is the cardinality of the set I

Finally we will denote $G_w(I, D)$ the group $< W, I, D >$ where n is the size of the group and t its weight. Because the analysis we will perform, a constant k is defined as the minimum size that a group must be to be considered, this means,

A group is trivial, with respect k, if its size is less than k, and none trivial in the other case.

Since we can expect more than one group of size n given a w, $\mathcal{G}_w$ is the multiset of all groups in w

Interaction Networks will also have other characteristics besides groups; these variables will be separated into two main categories, group, and global.

2.1 Local or group, variables

Local variables are symbols that represent part of the behavior of a group within an Interaction Network. For instance, there are variables, like number of domains present on a particular group that are important to study because in these case, we know that domains tends to follow a Zipf distribution [9] basically because most of the users on a network will visit more Google rather than IEEE Xplore. For that reason we will expect to see more traffic querying for particular domains. Among the variables that will be extracted from groups we have the following:

Given a group $G \in \mathcal{G}_w$

The size of a group will be determined by,

$$size(Gw) = size(G_w(I, D)) = |D|$$

The weight of the group will be defined similarly,

$$weight(Gw) = weight(G_w(I, D)) = |I|$$

Since we are interested in knowing the behavior of groups we will define the most and less active IP by the next function[1]

$$\forall Gw \in \mathcal{G}_w, \; minsize \leq size(Gw)$$

Finally the less, and the most likely to be visited Domain will be defined by a similar function,

$$\forall Gw \in \mathcal{G}_w, \; maxweight \geq weight(Gw)$$

[1] Most active IP is defined similarly so it will be omitted

2.2 Global variables

On the other hand, global variables represent the behavior of an Interaction Network. For instance, the total number of IPs on a window w will tell us that there are active users present on that window since, to be in the set of IPs, a user should asks for a domain. Since we are interested in knowing the user (IP) behavior i.e., find all the Domains the user asks for(among other variables), furthermore we are interested on which IP are querying more than other ones. We will define global variables as follows:

Given $w \in \mathbb{W}, y \in D(w)$

The set of all IPs visitors of a particular Domain is,

$$visitors_w(y) = \{x \in I(W) |\ qry_w(x,)\}$$

Moreover we are interested on the cardinality of that set, to obtain the most visited Domain on a window w,

$$pop_w(y) = |visitor_w(y)|$$

Therefore, the set of all popular Domains in w is,

$$toppopular(w) = arg_y max\ pop_w(y)$$

Finally we will get the total queries made to the most visited Domain,

$$\forall y \in D(W) maxpop(w) = max(pop_w(y))$$

The set and total visits made to the less likely to be visited Domain are defined similarly, so it will be omitted.

Since our approach tries to model user behavior we are interested in knowing the activity of a particular IP it means knowing all the characteristics that define a user, for that reason the domains a particular IP visits is defined by,

$$span_w(x) = \{y \in D(w) |\ qry_w(x,y)\}$$

The total activity, defined as the load of queries made by a single IP, $x \in I(W)$ in $w \in \mathbb{W}$ is the function,

$$activity_w(x) = |span_w(x)|$$

Then, the set of the most active IPs is given by,

$$topactive(w) = arg_x max\ activity_w(x)$$

And the maximum load of activity in w is,

$$\forall x \in I(w) maxactivity(w) = max(activityw(x))$$

3 How Interaction Networks react to DDoS attacks

We will assume that an attacker has the ability to compromise several computers on a network, or can inject DNS packets to this network, in any of those cases the attacker need to know the structure of a DNS packet to successfully launch a Distributed Denial of Service attack (DDoS). Packets on DNS are called messages, we are interested in the Question section of the message because in that section there is a field called QNAME that carries the domain a user asks for.

Attackers must fill this field even if they are asking for domains without sense like "gugul.coum", we know that this TLD don't even exist neither gugul domain. A zombie computer (in a botnet attacking a DNS sever) can ask for a set of well known domains like yahoo, google, microsoft, amazon, ebay, etc. Moreover we will assume that an attacker can delete DNS tables of a computer so every time a computer asks for www.ebay.com a DNS query will be made at least to a recursive DNS. In addition to these attack features, a DDoS will be persistent as we can note in attacks like [2][3][7].

Social networks will show if there is an abnormal behavior on the DNS queries made by users. For instance, users tend to navigate to the same sites day after day even at the same hours. So if our computer starts asking for domains that we don't query for, maybe we are compromised and if we belong to a group of behavior maybe we start to separate from this group.

Our assumption on these attacks basically tell us that there are two type of attacks, in one hand we have attacks that ask for valid domains maybe the intruder use a set of previous established domains, or maybe the attacker use a random domain generator as we can note in attacks like the Conficker A worm [10]. On the other hand we have attacks that don't follow any pattern and tends to be more random than the others (e.g abcd.con, adef.col, efasd.com), this kind of attacks are more dangerous, because it will spend more network resources, since to resolve this type of queries we need to consult other high-level instances of the DNS infrastructure. The reaction of the Interaction Networks will be noticed if we can observe a graphical representation, as we mentioned above; one of the variables affected by attacks could be the number of groups which will be larger or shorter, another affected variables could be unique IP addresses and Domains.

4 Proposed Methodology

In spite of having a graphical representation of a DDoS attack there is not enough evidence of such attack, so we need to develop a measurement to determine if there is an attack of DDoS targeting a DNS server. For the following analysis we will take logs of a BIND server from a university campus (recursive DNS server). Our preliminary analysis told us that there are behaviors of use at certain times (Fig. 2). For instance, at 7am DNS activity starts to increase slowly but from 9am to 12am the activity on DNS server quickly arises; at 4pm volume of traffic starts to decrease and at 12pm there are approximately 20 queries in 30 seconds.

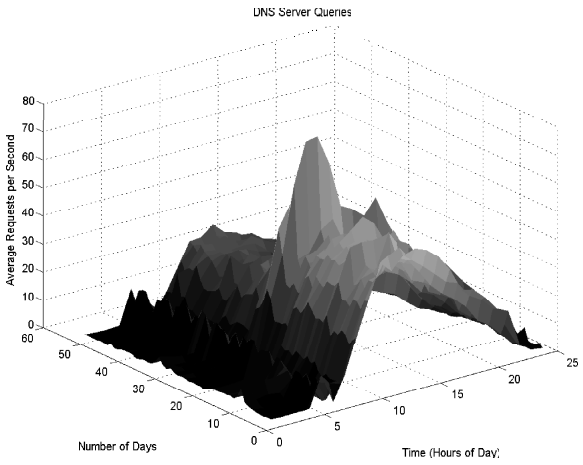


Fig. 2. Average DNS queries of 52 working days.

We can conclude that preliminary analysis told us that we need to analyze DNS traffic by hour, basically because every hour we have different population. Therefore, from this moment all window analysis will be by hour. To get a measure from our interaction network model, we will get the cardinality of some sets of data to represent the behavior in terms of values, these sets were previously defined in section 2; tuples, that are part of a matrix, are constructed with the following information:

4.1 Inter-window analysis

To extract data from global variables we will assume that we are on an analysis window w, and to construct this matrix the next values are involved:

- $numIP$ is a value that tell us the number of IPs present on w and extracted from the set $I(w)$, $numIP = |I(w)|$
- $numD$ is the number of Domains in w, obtained from $D(w)$, $numD = |D(w)|$
- $maxSizeG$ is the maximum size[2] of a group in w, $maxSG = maxsize \geq size(Gw)$
- $minSizeG$ is the minimum size of a group in w, defined similarly like maximum group size.
- $\mu SizeG$ is the mean size of the groups in w.
- $maxWeightG$ is the maximum weight of all groups, $maxWeightG = maxweight \geq weight(Gw)$
- $minWeightG$ is defined as the minimum weight of all groups. Formula omitted since can be defined similarly like max Weight

[2] If unclear see section 2.1

- $\mu WeightG$ is the mean weight of all groups.
- $numGps$ is the total number of groups given by, $numGps = |\mathcal{G}_w|$
- $maxPD$ is the maximum visits to the most queried domain, $maxPD = maxpop(w)$
- $maxAIP$ is the load of the most active IP in w, $maxAIP = maxactivity(w)$
- $entropyIP$ is the entropy of the IPs in w where the entropy
- $entropyD$ is the entropy of the Domains present in w

From this data a set of tuples will be part of an analysis matrix $\mathbb{V}$, these tuples will have the form (due to the lack of space not all the variables will be included):

$$< numIP_i, numD_i, maxSizeG_i, ..., entropyD_i >$$

Where, i corresponds to the analyzed window $w_i \in \mathbb{W}$.

Another value can be inserted in the matrix $\mathbb{V}$ since in this matrix we have tuples from different windows, we can make differentiations to determine the change on speed and acceleration in all the variables for instance, the number of IPs and Domains. We know that the initial speed and acceleration will be equal to 0.

5 Preliminary Results

A social network was built with the data of a recursive DNS server. In this case we constructed a social network from data at 07am (Fig. 3), and a behavior is still present even if there are not much users.

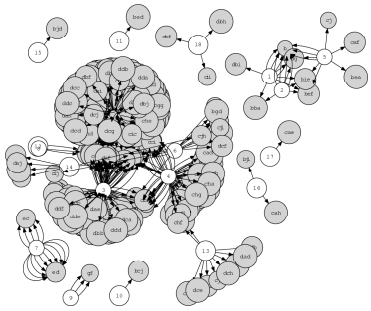
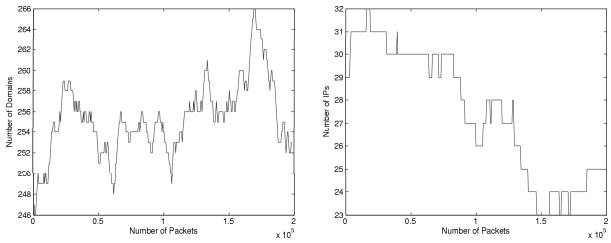


Fig. 3. Behavior of a campus at 7am.

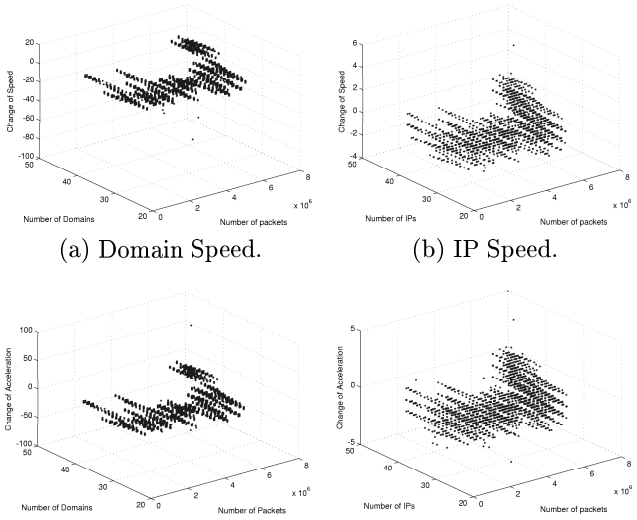
From the figure we can observe some groups, domains and IPs, since at that hour there are not much users as at 12pm. We can graphically see 2 or 3 groups. Another analysis was performed this time at 2pm, from that analysis we can note that the number of IPs and Domains does not decrease or increase suddenly (Fig. 4).



(a) Number of Domains at 2pm. (b) Number of IPs at 2pm

Fig. 4. Graphs representing the total number of Domains (a), and IPs (b)

In fact those behaviors are persistent in almost every hour in a day. Furthermore if we make differences from that behavior, a constant change of speed and acceleration will be found to show that, we performed an analysis from 8 hours of the same day, since we have logs of a DNS server without attack(we suppose) speed and acceleration are constant with respect the packets analyzed. (Fig. 5)



(a) Domain Speed. (b) IP Speed.

(c) Domain Acceleration. (d) IP Acceleration.

Fig. 5. Graphs representing the Change of Speed and Acceleration of Domains (a), (c)and IPs (b), and (d)

6 Conclusions and future work

Social networks show the interactions between users within in a network; with these relations we can construct profiles that allow identifying sudden changes in the users behavior, maybe to flag abnormal traffic or to detect DDoS attacks. We will continue studying these structures since our approach can show that in some cases under abnormal conditions the groups tend to separate each other from the universe of domains that an user usually visit.

We use graphs to model DNS traffic on a campus network, we know that graphs to model networks isn't new, but the variables we are extracting from this model is the principal contribution of the method since i.e., other approximations to characterize and detect DDoS attacks, does not care about the domain a user asks for.

Also we will perform an analysis of correlation since we are interested in discarding all those variables that provide, for instance, redundant information, this with the purpose of improving performance of the method.

We will make a "contrast analysis" were we intend to characterize behaviors between different days; we will contrast the information obtained in a particular hour in the inter-window analysis with the information of the same day and hour from another week; this with the purpose of building a profile of normality, this means to establish parameters (thresholds) to detect a DDoS attack, obtaining data like mean, standard deviation, skewness, etc.

In the nearly future, we will use information gathered by DNS-OARC and CAIDA (high-level DNS instances), who have data from the last attack dated April 7th 2009 of this worm, to test the method.

References

1. J. Klensin, Role of the Domain Name System (DNS), RFC 3467 (2003)
2. P. Vixie., G. Sneeringer, M. Schleifer.,: Events of 21-Oct-2002: ISC/UMD/Cogent: `http://c.root-servers.org/october21.txt` (2004),
3. ICANN: Factsheet, Root Server attack on 6 February 2007 (2007)
4. V. Ramasubramanian., E.G. Sirer.: Perils of Transitive Trust in the Domain Name System: Cornell University, New York (2004)
5. S. Castro., D. Wessels., M. Fomenkov., K. Claffy.: A Day at the Root of the Internet: Computer Communication Review: 38(5),41-46(2008)
6. B. Kirkpatrick., S. Lacoste., W. Xu: Analyzing Root DNS Traffic
7. Y. Huang., X. Fu., Q. Hou., Z. Yu.: The Early Detection of DDoS Based on the persistent Increment Feature of the Traffic Volume: 22nd International Conference on Advanced Information networking and Applications-Workshops.
8. S. Yu., W. Zhou.: Entropy-Based Collaborative Detection of DDoS Attacks on Community Networks: Sixth Annual IEE International Conference on Pervasive Computing and Communications (2008).
9. E. Graells., R.B.-Yaez.: Comparacin entre la Web Chilena y la Web Espanola: Revista Faz (2007)
10. P. Phillip., et al: An analysis of Conficker, SRI International (2009)