Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/cor (2010)

Enhancing DNS Security using Dynamic Firewalling with Network Agents

Joao Afonso

Foundation for National Scientific Computing
Lisbon, Portugal
e-mail: joao.afonso@fccn.pt

(2013)

***Abstract*—In this paper we propose a solution to strengthen the security of Domain Name System (DNS) servers associated with one or more Top Level Domains (TLD). In this way we intend to be able to reduce the security risk when using major internet services, based on DNS. The proposed solution has been developed and tested at FCCN, the TLD manager for the .PT domain. Through the implementation of network sensors that monitor the network in real-time, we are capable to dynamically prevent, detect or limit the scope of attempted intrusions or other types of occurrences to the DNS service. The platform relies heavily on cross-correlation allowing data from a particular sensor to be shared with the others. Administration tasks such as setting up alarms or performing statistical analysis are made through a web-based interface.**

***Index Terms*—DNS; risk; security; intrusion detection system; real-time;monitoring.**

I. INTRODUCTION

OBSERVING internet usage and world population statistics [1] updated on March 2011, there are 30.2% internet users – of the estimated world population of 6.8 billion. If we take a closer look to Europe this value increase to 58.3 % (with a growth rate of 353.1% between 2000 and 2011) and in North America, there are 78.3 % of internet users (growth rate of 151.7% at same period), as shown in Fig. 1.

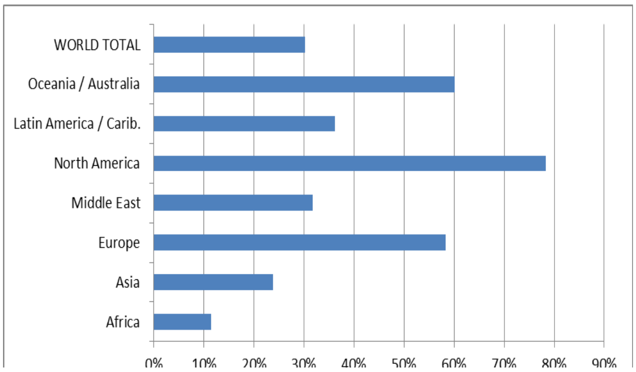

Figure1. Internet penetration (% population)

The DNS service is required to access e-mail, browse Web sites, and is needed for normal operation in all major services in the Internet (most of them use critical information, like e-banking).

Taking care of the huge number of internet users, and the risk associated with the fact that all major applications requires the DNS service, there is a security risk needed to be reduced.

DNS servers assume a pivotal role in the regular running of IP networks today and any disruption to their normal operation can have a dramatic impact on the service they provide and on the global Internet.

Although based on a small set of basic rules, stored in files, and distributed hierarchically, the DNS service has evolved into a very complex system [2].

According to other recent studies [3], there are nearly 11.7 million public DNS servers available on the Internet.

It is estimated that 52% of them allow arbitrary queries (thus allowing the risks of denial of service attacks or "poisoning" of the cache).

They are still nearly 33% of the cases where the authoritative nameservers of an area are on the same network, which facilitates the attacks of Denial of Service (DOS).

Furthermore, the type of attacks targeting the DNS are becoming more sophisticated, making them more difficult to detect and control on time.

Examples are the attacks by Fast Flux (ability to quickly move the DNS information about the domain to delay or evade detection) and its recent evolution to Double Flux [4].

A central aspect of a security system is the ability to collect statistically useful information about network traffic. This information can be used to monitor the effectiveness of the protective actions, to detect trends in the collected data that might suggest a new type of attack or simply to record important parameters to help improve the performance of the service.

The fact that the DNS is based on an autonomous database, distributed by hierarchy, means that whatever solution we use to monitor, it must respect this topology. In this paper we propose a distributed system using a network of sensors, which operate in conjunction with the DNS servers of one or more TLDs, monitoring in real-time the data that passes through them.

The ability to perform real-time analysis is crucial in the DNS area since it may be necessary to act in case of abuse, by blocking a particular access, and notifying the other sensors on the origin of the problem, since several types of attacks are directed to other DNS components

The use of a Firewall solution whose triggering rules are dynamically generated by the network sensors is a fundamental component of the system, to filter attacking systems and returning to the initial situation when the reason to filter different traffic patterns has ceased to exists, guarantees an autonomous functioning of the platform. Special care was taken to minimize the detection of false negatives and positives.

The remaining of the paper is structured as follows: Section 2 provides background information regarding related work. Section 3 introduces System Requirements. In section 4 we describe the proposed solution. Section 5 presents a case study for validation of the proposal. In Section 6 the results gathered in the case study are analyzed. Finally, Section 7 presents some conclusions and directions for further work.

II. Related Work

One of the first studies that can be witnessed in this area has the authorship of Guenter and Kolar, with a tool entitled sqldjbdns [5]. Their application uses a modified version of the traditional BIND [6] working together with a Structured Query Language (SQL) version inside a Relational database management system (RDBMS). For DNS clients, this solution is transparent and there is no difference from classic BIND.

Zdrnja presented a system for Security Monitoring of DNS traffic [7], using network sensors without interfering with the DNS servers to be monitored. This is a transparent solution that does not compromise the high availability needed for the DNS service.

Vixie proposed a DNS traffic capture utility called, DNSCap [8]. This tool is able to produce binary data using pcap format, either on standard output or in successive dump files. The application is similar to tcpdump [9] – command line tool for monitoring network traffic, and has finer grained packet recognition tailored for DNS transactions and protocol options, allowing for instance to see the full DNS message when tcpdump only shows a one-line summary.

Another tool available is DSC - DNS Statistics Collector [10]. DSC is an application for collecting and analyzing statistics from busy DNS servers. Major features include the ability to parse, summarize and search inside DNS queries detail. All data is stored in an SQL database. This tool, can work inside a DNS server or in another server that "captures" bi-directional traffic for a DNS node.

Kristoff also proposed an automated incident response system using BIND query logs [11]. This particular system, besides the common statistical analysis, also provides information regarding the kind of consultations operated. All information is available through the Web based portal. Each security incident can result in port deactivation.

III. Methodology

A. Architecture

To be able to reduce the incident risk in DNS operation, the architecture of the system that we have develop aims to improve the security, performance and efficiency of the DNS protocol, removing all unwanted traffic and reinforce the resilience of a Top Level Domain. We propose an architecture comprising an integrated protection of multiple DNS servers, working together with several network sensors that apply live rules to a dedicated firewall, acting as a traffic shaping element.

Sensors carefully located in the network monitor all the traffic going to the DNS infrastructure, identify potentially harmful traffic using a algorithm that we have developed and tested and use this information to isolate traffic that has been identified has security threats.

Several networks sensor monitor different parts of the infrastructure and exchange information related to security attacks. In this way, as shown in Fig. 2, it should also be possible to exchange critical security information between the sensors. In addition to an increase in performance, this operation should prevent an attack on a server from a source, identified by another sensor as malicious. This scenario is relevant since some kinds of attacks are directed to several components of the DNS infrastructure.

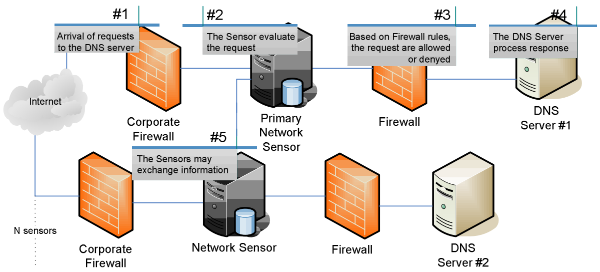

Figure2. Diagram of the desired solution

B. Heuristic

One of the crucial parts of our work is the algorithm to identify traffic harmful to the DNS. In order to implement the stated hypothesis in the architecture and keep the DNS protocol as efficient as possible, it is necessary to apply a heuristic, which in real time, evaluates all the information collected from different sources and applies convenient weights to each component and act accordingly.

The components that we have chosen to have impact in the security incidents of DNS are: the number of occurrences, analysis of type of queries been made, the amount of time between occurrences, the number of probes affected and information reported from intrusion detection systems.

Our system uses the following formula to evaluate a parameter that measures the likelihood of the occurrence of a security incident:

$$f(x) = O \cdot 0{,}2 + C \cdot 0{,}2 + G \cdot 0{,}15 + N \cdot 0{,}25 + I \cdot 0{,}20$$

Are factors considered in applying this formula:

- Occurrences (O) - Represents the number of times (instances) that have given malicious source was blocked, so that the distributed then depicted in Table I.

TABLE I – CONTRIBUTION OF THE NUMBER OF OCCURRENCES OF A SOURCE IN MALICIOUS HEURISTIC

Occurrences	*Weight*
1	25%
2	50%
3	75%
4 or more	100%

- Analysis (C) - Real-time evaluation of the deviation of the values recorded in relation to the average observed statistics, based on the criteria and weights identified below in Table II.

TABLE II – CONTRIBUTION OF EVENTS TYPIFIED A POTENTIALLY MALICIOUS SOURCE GIVEN IN HEURISTIC

Event	*Weight*
Entire zone transfer attempt (AXFR)	100%
Partial transfer zone attempt (IXFR)	50%
Incorrect query volume, 50 to 75% on average per source	75%
Incorrect query volume exceeding 75%	100%
Query volume, up 50%, the average number of access by origin	50%

Note that the estimates apply the moving average, for the determination of reference values, given the ongoing development of data collected.

- Time between occurrences (G) - time since last occurrence of a given source, distributed with the weights associated to the times below is obeisant.

TABLE III – WEIGHT OF DIFFERENT TIME BETWEEN EACH OCCURRENCE

Time	*Weight*
Less than 1 Minute	100%
Less than 1 Hour	75%
Less than 1 Day	50%
Less than 1 Week	25%

- Incidence (N) - Number of probes that report blocks in the same address source.
  For the calculation, we observed expression:

$$\frac{1}{\#Total\_Sensors - \#Sensors\_Attacked}$$

  In the above expression the factor *#Total_Sensors* represents the number of sensors running together in the infrastructure and able to exchange information among themselves.
  The other factor #Sensors_Attacked stands for the number of Sensors that are current reporting security incidents.
- Intrusion Detection Systems (I) - We considered the use of the Snort platform, being free to use, and gather a large number of notarized signatures of security incidents relating to the DNS service.

TABLE IV – INTERCONNECTION WITH TEMPORAL DATA GATHERED FROM INTRUSION DETECTION SYSTEMS

Metric: Common Vulnerability Scoring System (CVSS)	*Weight*
Low level	34%
Middle level	67%
High level	100%

For the activation of a rule in Firewall occurs will require:

1. The formula shown above take values equal to or greater than 0.25;
2. The combination of two or more criteria of the formula.
   Exception: when receiving information from all the other sensors, in which case a single criteria is sufficient;
3. It respected the existing white list in the repository, allowing considered privileged sources that are not blocked.
   In this way we avoid compromising the Internet service, considering the key role played by DNS, the White List protects key addresses from being blocked in case of false positives events.
   This list is created from a record of trusted sources, allowing all addresses listed here to be protected from being added to the Firewall rules.
   One example is the list of internal addresses, and the DNS servers of ISPs.

Instead, for the removal of a rule in the firewall will need to occur simultaneously on the following assumptions:

1. Exceeded the quarantine period, based on the parameters in use;
2. The expression of activation (heuristic) does not (still) check the referenced source.

IV. Proposed Solution

A. Diagram

As shown in Fig. 3, this solution is based on a network of sensor engines that analyze all traffic flowing into the DNS server in the form of valid or invalid queries, process the information received from other probes and issue restrictions for specific network addresses. In case an abnormal behavior is detected or there is suspicious behavior from a certain network address, it will be blocked in the firewall and the other probes notified so they can act accordingly. The system can also calculate the response time for each operation to evaluate the performance of the server.

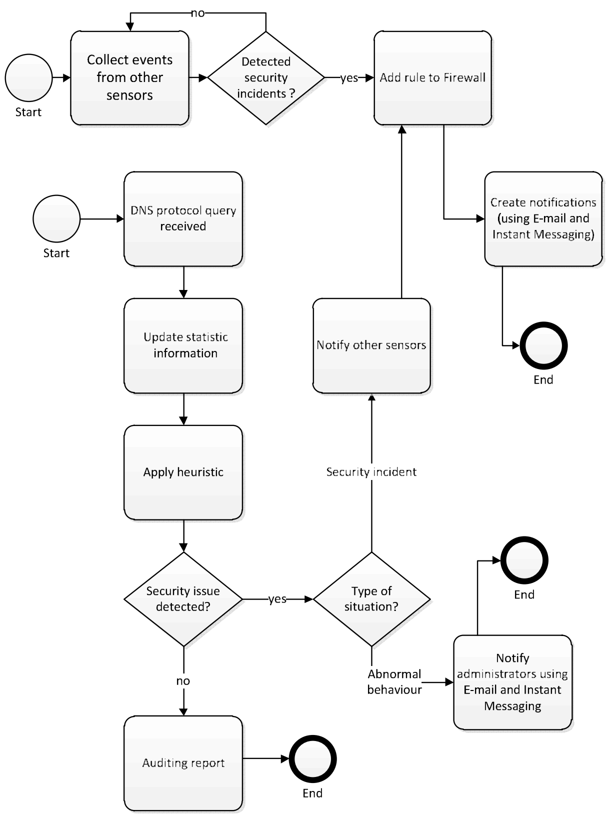

Figure 3. Block Diagram of proposed solution

For each rule inserted in the sensor firewall, there will be a period of quarantine and, at the end of this time, the sensor will evaluate the behavior of that source, to evaluate the needed to remove the rule, as shown in Fig. 4.

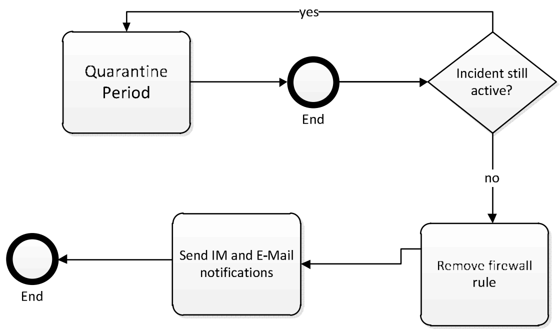

Figure 4. Quarantine procedure over the Firewall

B. Network data flow

According to our design, all data that flows through the probe heading for the DNS server is treated according to a standard set of global firewall rules, followed by specific local rules regarding to the addresses that are being blocked in real time. The queries are then delivered to the parser to be analyzed and stored in the RDBMS. At the top is the system of alarms and the Web portal (Fig. 5).

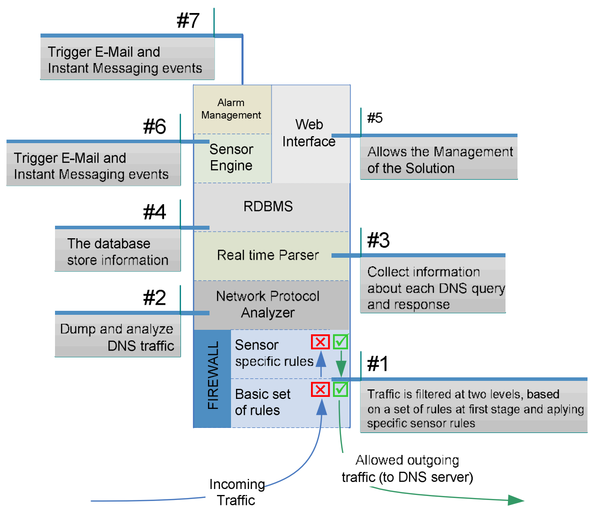

Figure 5. Network data flow

All information collected is stored in a database implemented in MySQL [12]. Taking into consideration the need to optimize the performance of the queries and to reduce the volume of information stored, the data is divided into a number of different tables.

The conversion of the IP address of source and destination (DNS server) into an integer format, has allowed for much more efficient data storage, and significant improvements in the overall performance of the solution.

The information regarding all queries made, is stored daily into a log, and kept available during the next 30 days.

Two tables containing the set of rules that are dynamically applied – add or removed, based on situations that have been triggered - control the correct operation of the firewall. For auditing purposes every action is registered.

The information required for auditing and statistical tasks never expires.

C. Statistical analysis and performance evaluation

The statistical information collected and stored in the database has a significant amount of detail. It is possible, for example, to calculate, for each sensor, the evolution of queries per unit of time (hour, day, etc) badly formatted requests, DNS queries of rare types and determine the sources that produce the larger number of consultations. It is also possible to see the standard deviation of a given measure so we can relate it to that is seen with the other hits [15].

The performance of the DNS protocol responses is permanently measured, regarding the response time per request. Data is constantly registered and an alarm is raised in case normal response times are exceeded.

V. CASE STUDY

Our proposal have been under development since September 2006 at FCCN – who has the responsibility to manage, register and maintain the domains under the .PT TLD.

At present time, there are two sensors running attached to the DNS servers (one at the primary DNS and another working together with a secondary DNS server).

The network analyzer is tshark [16], and the firewall used is IPFilter [13]. The real time parser was programmed in Java, collecting the information received from the tshark. The Web server is running Apache with PHP.

Regarding the Xmpp server [14] we choose the Jive messenger platform.

All modules are integrated together.

The entire sensor solution, as described above, as well as the web platform we developed went on-line on the 1st of January 2007, and the data from the various agents was collected from the 10th of May 2008 till now (Fig. 6).

Figure 6. Web portal

In addition to the usual operations of monitoring and collection of statistics relating to the operation of DNS service, as shown above, the solution proposed here can easily be adapt to specific situations, given the fact that it is fully configurable.

One of the applications was the event North Atlantic Treaty Organization (NATO) Summet Lisbonne 2010 at the period of 15 to 21 November 2010.

To reduce security risks in the area of the Internet involved in event, a number of areas considered most vulnerable were selected, and made a daily monitoring over them.

They were classified into four categories: law enforcement, banking, government and industry. The data collected by each sensor created a pattern of consultations for each of the categories and detect abnormal situations occurred when deviations happens from pattern (Fig. 7)

Alarmist notifications were programmed using SNMP traps.

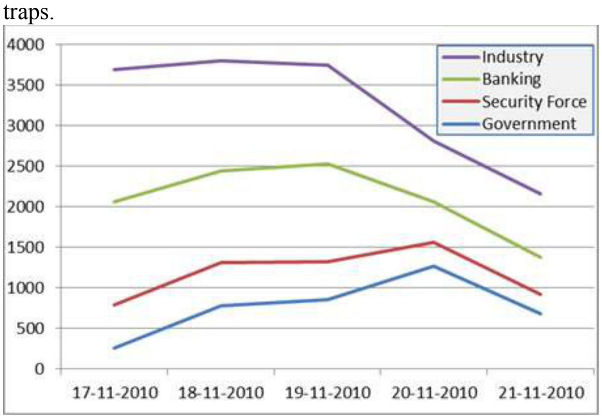

Figure 7. Monitoring DNS service at NATO event/Lisbon 2010

VI. RESULTS

We present here the results of the last 12 months of data collection (between 1st of May 2009 and 31st May 2010). The Average number of requests to the primary DNS server is up to 19,769,946 per day (228 per sec.) using last records collect on 7th August 2011.

The performance of the data analysis program is above 1240 requests processed per sec. (filtered, validated and inserted in the database).

Using the data collected by the sensors, during this time period, we were able to collect useful statistical information, e.g.:

- Daily statistics by type of DNS protocol registers accessed;
- Number of Internationalized domain name (IDN) queries;
- Number of daily queries to IPV6 AAAA DNS type (Fig. 8).

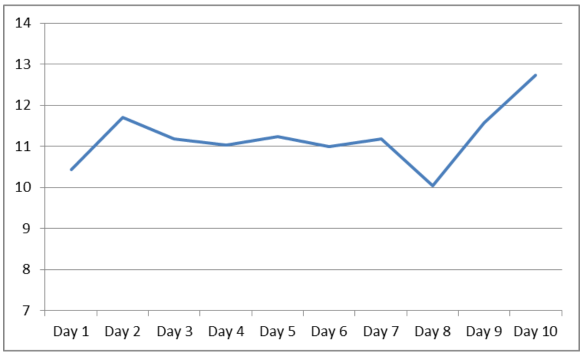

Figure 8. Statistical analysis by IPV6 records accessed (AAAA)

- Detect examples of abnormal use (that are not security incidents). For example we were able to detect that a given IP was using the primary .PT DNS server as location resolver.
  The number of queries made was excessive when compared with the average value per source, reaching values close to some Internet Service Providers that operate under the .PT domain.
- Detect situations of abuse, including denial of service attacks, with the execution of massive queries. In last 12 months of analysis there are 17 DOS attacks triggered.
  They were instantly blocked, and addresses placed in quarantine (Table V).

TABLE V. EXAMPLES WHEN THE SENSOR DETECTED SITUATIONS THAT REQUIRED THE FIREWALL RULES TO CHANGE.

Source Address	*Date / Time*	*Operation*	*Sensor*
xx.xx.200.45	2011-08-05 02:15:44	Add rule	xx.xx.44.63
xx.xx.17.122	2011-08-05 03:25:12	Remove rule	xx.xx.44.63
xx.xx.129.51	2011-08-05 04:47:14	Add rule	xx.xx.44.62
xx.xx.14.239	2011-08-05 05:27:29	Add rule	xx.xx.44.62
xx.xx.14.131	2011-08-05 08:35:38	Remove rule	xx.xx.44.63

VII. CONCLUSION AND FUTURE WORK

This article has presented a novel approach to reduce the security risk on the internet applications that use DNS service. Our solution builds upon the existing solutions that collect statistical information regarding DNS services, by adding the ability to detect and control security incidents in real time. It also adds the advantage of operating in a distributed way, allowing the exchange of information between cooperating probes, and the reinforcement of its own security, even before it is threatened.

Currently, the solution presented does not allow the processing of addresses in the IPv6 format. The technical aspects that led to this situation are linked to the need to optimize the performance of the data recorder application making it possible to store the data from all consultations. One possible option for solve this issue is to change the database engine to other solution, for instance, a commercial one. Nevertheless, all queries made to IPv6 addresses are contained in this solution (AAAA types).

We are also working on extending the data correlation capabilities of the system by adding information collected from other sources (intrusion detection systems for instance). We anticipate that this could be a valuable approach to reduce considerably the number of false positives and negatives [17].

REFERENCES

[1] Internet Usage and World Population Statistics website, [http://www.internetworldstats.com/stats.htm]. Last accessed on 7 August 2011.

[2] P. Vixie, “DNS Complexity”, ACM Queue vol. 5, no. 3, April 2007.

[3] D. Wessels, "A Recent DNS Survey", DNS-OARC, November 2007.

[4] Dave Piscitello, “Conficker Summary and Review”, ICANN, May 2010.

[5] SQLDNS website, [http://home.tiscali.cz:8080/~cz210552/ sqldns.html]. Last accessed on 7 August 2011.

[6] BIND website, [http://www.isc.org/products/BIND]. Last accessed on 7 August 2011.

[7] Bojan Zdrnja, “Security Monitoring of DNS traffic”, May 2006.

[8] Paul Vixie, D. Wessels, ”DNSCAP – DNS traffic capture utility”, CAIDA Workshop, July 2007.

[9] Duane Wessels, "Whats New with DSC", DNS-OARC, November 2007.

[10] Lawrence Berkeley National Laboratory. Tcpdump website http://www.tcpdump.org.

[11] John Kristoff, "An Automated Incident Response System Using BIND Query Logs", June 2006.

[12] MySQL website – (Open Source Database), [http://www.mysql .com]. Last accessed on 7 August 2011.

[13] IP FILTER – TCP/IP Firewall/NAT Software, [http://coombs.anu.edu.au/~avalon]. Last accessed on 7 August 2011.

[14] P. Saint-Andre, Ed., Extensible Messaging and Presence Protocol (XMPP): Core, RFC 3920, 2004.

[15] João Afonso, Edmundo Monteiro, "Development of an Integrated Solution for Intrusion Detection: A Model Based on Data Correlation", in Proc. of the IEEE ICNS'06, International Conference on Networking and Services - ICNS'06, Silicon Valley, USA, July 2006.

[16] Tshark website – The Wireshark Network Analyzer, [http://www.wireshark.org]. Last accessed on 7 August 2011.

[17] João Afonso, Pedro Veiga, “Protecting the DNS Infrastructure of a Top Level Domain: Real-Time monitoring with Network Sensors”, WSNS 2008, 4th IEEE – International Workshop on Wireless and Sensor Networks Security, Atlanta, USA, 29 September – 2 October 2008.