DNSSEC Application in Terms of Cyber Security and Social Engineering: Sedres

by

Kaan Tekiner, Yeditepe University, Istanbul, February 2021, Turkey
Email: kaantekiner96@gmail.com, Org. Language: English

Thesis Project Report

Dr. Lecturer Çağla ÖZEN

Faculty of Commerce

Department of Management Information Systems

Istanbul, 2021

Table of Contents

List of Figure

Outline

Title: DNSSEC Application in Terms of Cyber Security and Social Engineering: Sedres.

Thesis: DNS systems are technologies that make websites easily accessible and are a structure that almost every person with an internet connection uses in daily work over and over. While DNS security is critical among cyber security operations, it is especially necessary for end users to use their internet networks safely and peacefully. The DNS security literature should be fed with security applications that are easy to integrate and compact.

I. Introduction

A) Domain Name System architecture.
B) The DNS security literature should be fed with security applications that are easy to integrate and compact.

II. DNSSEC & Literature Review

A) Studies in the literature on the security of DNS systems.
B) Descriptions and quotations for weak points of DNS systems.

III. The Gap

A) Deficiencies and reservations of IDS and IPS systems on DNS security.
    1) The consequences of IPS systems' fear of weight and slowness in DNS security.

IV. Data

A) Situations caused by data being the most valuable raw material in the 21st century.

V. Information & Cyber Security

A) Obligation to protect information.
B) Linking of information and cybersecurity.

VI. Infrastructure & Problems

A) The ‘ease of use impairing security’ concept.
B) The CIA (Confidentiality, Integrity and Availability) triangle.

VII. Hacking People: Social Engineering

A) The human factor in social engineering

VIII. The DNS Scenario

A) Simplified, Domain Name System’s purpose.

IX. DNS Security (DNSSEC) & Vulnerability Pronities

Introduction

The Domain Name System (DNS) is used to associate a domain name with an Internet Ptotocol (IP) address, although it is an indispensable element of internet communication and infrastructure. Thanks to its intended use, it is one of the most used protocols on internet structures. DNS is used in our daily lives by users during domain name resolution and web surfing with it. In this period when every device in the world is interconnected and network structures are constantly growing, the DNS protocol has started to play an even more important role.

When the Internet and network structures were discovered and first used, it can be seen that security has not been an issue for a long time. As is known, these effects continue today. Based on this situation, it can be predicted that DNS structures used in today's systems are also prone to some security problems.

Within the cyber security sector, many open source applications, commercial softwares, servers and firewalls were produced to solve these DNSSEC problems, awareness training was given to internal and individual users and tracking systems were developed. Proxy-based controlling certificates used on firewalls, HTTPS structure, encryption algorithms, central DNS servers, many harmful DNS disclosure sources that can be found on the internet and many other structures can be given as examples of the measures taken up to this day.

However, none of these measures were sufficient to completely solve the problems. From this point of view, it can be observed that today's technology systems need more security applications and structures.

The Sedres application, which takes its name from the "Secure DNS Resolution" mold, is an open source software produced to support the literature and DNS Security technologies. It aims to eliminate the security threats to be mentioned and to ensure the security of end users. The contributions it plans to provide to the DNSSEC world, usage purposes and algorithmic defense methods will be explained in this thesis.

If enough work is not carried out, this issue is not taken into consideration as much as necessary and as a result these security problems persist; sure, cybercriminals can cause critical problems and manipulations in DNS systems thanks to some technical attacks and manipulations they perform. What these problems are or might be, how they are technically handled, what actions are taken for their solutions and their evaluations will be mentioned in the continuation.

To summarize, this paper and project tends to evaluate DNS systems under the framework of cyber security and to examine the threats that end users performing transactions in both the institution and home environment are exposed to in these aspects.

DNSSEC & Literature Review

With the strengthening of the place of structures such as the Internet, intranet, and extranet in human history, the emergence of cyber security and recognition by information organizations, studies on DNS Security have also been initiated and products have been developed on this subject. Among these studies, not only the ones carried out on DNS Security, but also the related issue has been mentioned within different security studies.

Articles written and studies conducted on the subject reveal that this system is currently an unsafe, sensitive [7] protocol, and this provides support for perturbations and development expectations.

‘DNS is a highly sensitive part of every ICT system. If attackers can take control of DNS it would give them unlimited possibilities to abuse the organization at different aspects.’

Davidowicz has clearly stated on the document he prepared that DNS systems have general structures, critical importance and they are faced with many threats [4].

‘The DNS plays a critical role in supporting the Internet infrastructure by providing a distributed and fairly robust mechanism that resolves Internet host names into IP addresses and IP addresses back into host names. The DNS also supports other Internet directory-like lookup capabilities to retrieve information pertaining to DNS Name Servers, Canonical Names, Mail Exchangers, etc. Unfortunately many security weaknesses surround IP and the protocols carried by IP. The DNS is not immune to these security weaknesses. The accuracy of the information contained within the DNS is vital to many aspects of IP based communications.

The threats that surround the DNS are due in part to the lack of authenticity and integrity checking of the data held within the DNS and in part to other protocols that use host names as an access control mechanism. In response to this, the IETF formed a working group to add DNS Security (DNSSEC) extensions to the existing DNS protocol (Domain Name System (DNS) Security, Diane Davidowicz, 1999).'

Apart from this, many studies describe and support the problems experienced in the security phase of DNS systems. S. Rose has a consistent discourse (RFC 4033 [8]) on this subject.

'Due to a deliberate design choice, DNSSEC does not provide confidentiality.'

'It is known the fact that DNS is weak in several places. Using the Domain Name System we face the problem of trusting the information that came from a non authenticated authority, the name-based authentication process, and the problem of accepting additional information that was not requested and that may be incorrect (DNS Security - Antonio Lioy, Fabio Maino, Marius Marian, Daniele Mazzocchi, Dipartimento di Automatica e Informatica, Politecnico di Torino, Terena Networking Conference, May 22-25, 2000).'

As mentioned [1], the authorization and authorization control mechanisms used within DNS security have weaknesses and manipulable algorithms.

If to talk about another quote[1] on the same paper, the effects of security deficiencies in DNS systems are not only this. Some security flaws in DNS systems can also lead to deceiving systems and users.

'Suppose the following scenario: a user wants to connect to host A by means of a telnet client. The telnet client' asks through a resolver the local name server to resolve the name A into an IP address, it receives a faked answer, and then initiates a TCP connection to the telnet server on the machine A (so it thinks). The user sends his login and password to the fake address.'

As also mentioned in RFC 4033 [2] [DNS Security Introduction and Requirements - R. Arends, R. Austein, M. Larson, D. Massey, S. Rose, March 2005] like;

'Accordingly, DNSSEC is not designed to provide confidentiality, access control lists, or other means of differentiating between inquirers. DNSSEC provides no protection against denial of service attacks. Security-aware resolvers and security-aware name servers are vulnerable to an additional class of denial of service attacks based on cryptographic operations. The DNS security extensions provide data and origin authentication for DNS data. The mechanisms outlined above are not designed to protect operations such as zone transfers and dynamic update.'

In the referenced[6] document, there are expressions supporting this situation.

'Attacks & Implications To support these claims, we show two concrete attacks that leverage absolute time to attack DNS resolver caches. First, we present a cache expiration attack: when time is shifted forwards, the DNS cached responses expire sooner than expected, effectively flushing the cache.

Second, we present a cache sticking attack: when time is shifted backwards, the cached responses stick in the cache for longer than intended. We show how these attacks can be used to harm DNS performance (causing latency into DNS responses) and DNS availability (increasing the risk of denial of service). We also discuss how they can be used to aid for fast-fluxing, cache poisoning and other well-known threats to the DNS. We also present two similar attacks on DNSSEC signature validation; by shifting the time on the validating resolver forwards or backwards, we can force a valid signature to be deemed invalid, causing DNSSEC denial-of-service and several other problems (The Impact of Time on DNS Security, Aanchal Malhotra, Willem Toorop, Benno Overeinder, Ralph Dolmans and Sharon Goldberg, Boston University, NLnet Labs, Amsterdam, July 5, 2019).'

It can be easily seen that DNS systems are prone to denial of service and encryption-derived operational weaknesses. Considering that these and similar systems work on the virtual port number 53 and accept UDP-derived connections by default; the mentioned approach can be considered appropriate and correct.

Considering the fact that DNS systems process network packets over 512 Bytes as TCP instead of UDP, it is a must to consider the denial of service issue, and it also confirms the interpretation of the quote.

As noticed earlier in the document, with the emergence of these and similar problems and causing anxiety, secure DNS application software has also been produced. These products, which aim to support reliability and secure transaction stability on DNS systems, are generally offered as open source code. Lecture Notes in Computer Science, vol 6307 [3] has some descriptions on this subject.

'Several secure DNS protocols have been proposed, including DNSSEC and DNSCurve. DNSCurve provide link-level security while DNSSEC provide objectbased security of DNS messages using cryptographic means. However, the deployment of DNSSEC has proven slow, and many hosts have on-path hardware that interferes with DNSSEC's larger packet sizes.'

At the same time, such applications should not only be evaluated for security and reliability. Stability, longevity, speed, ease of use and installation are also values that should be taken into account.

If it should be mentioned as an example that programs and software running in the background consist of algorithms developed to serve a certain purpose or a certain purpose. The complexity and density of the path between inputs and outputs depends on how much control and processing mechanisms the application has. The addition of security layers to a software causes the related algorithms to be more complex and gingerly. This puts the application under greater strain, affecting availability and speed factors.

When evaluated in terms of strength, stability, and usability, security layers have some such disadvantages as well as benefits. This also applies to DNSSEC Applications.

On the other hand, many of such security applications work by combining many modules that serve different purposes and focus on different vulnerability vectors. Although these are easier to develop and are fragmented, they cannot prevent all possible security vulnerabilities unless they are produced specifically to solve a problem. Attacks that are called DNS Poisoning and aim to manipulate users can be defined as one of them. On the other hand, this attack is generally not noticed by security systems.

Lecture Notes in Computer Science, vol 6307 [3] has some comment on this subject like; *'The delay in deploying secure DNS motivates the need for local networks to protect their recursive DNS resolution infrastructure. Traditional solutions such as IDS and packet-inspection tools provide limited protections against some classes of attacks, but do not detect DNS poisonings.'*

As stated by Reference [1] regarding the problems in DNS security, ownership authorization and controls on DNS systems can be insufficient in some cases. This situation causes system control ownerships and indirectly information security to be in danger.

'For the top-level domain zones and for the root zone the problem of securing the private keys must be discussed in a different way. An attacker who could get the private key of a top-level zone would become authoritative for all the subdomains below. In the same way anyone who could obtain the root zone private key would be in control over the entire DNS space of all the resolvers configured to use the public key of the root zone, excepting those that are configured with the public key of a subdomain they belong to.'

Even aside from all that has been described so far, there is another simple issue that seriously endangers information security: encryption and visual blockages. Bortzmeyer's explanation on this subject is as follows[5].

'Today, almost all DNS queries are sent over UDP [thomas-ditl-tcp]. This has practical consequences when considering encryption of the traffic as a possible privacy technique. Some encryption solutions are only designed for TCP, not UDP.'

The Gap

IDS systems are applications that are integrated with different setup and hosting techniques and monitor abnormal behavior and activity from cyber security perspectives. They can be run on end user machines, service servers, application servers or centralized network traffic sensor servers. While these operations are performed, predetermined rule sets are generally used. Network packets in source traffic are subjected to these rules sequentially, and packets that are called malicious are listed for some other operations.

Due to the nature of IDS systems; they do not operate on live, streaming and processed network traffic. Instead, they take a copy of the relevant source traffic and process on it. This situation is generally preferred in order not to intercept the live stream, so to avoid possible weight and slowness. The structures that do this work with the opposite mechanism and apply the rule sets on the flowing real traffic can be described as IPS.

It is a fact that detecting and testing the reliability of DNS queries takes time and preliminary work. This approach is not a problem for IDS systems and is a procedure more prone to them. However, DNS security is an issue that needs to be addressed in real time, as in IPS systems. This situation indicates that an issue that can be handled more easily by IDS systems should be solved and applied on IPS systems. Even if such approaches and analyzes are made on IPS systems, some problems show themselves.

To summarize, it can be said that the decision as to whether the responses to DNS queries are safe may not always be reliable; as IDS systems analyze the activities that are already occurring, and IPS systems handle some security checks very lightly to affect the speed factor as little as possible.

Especially, a security product that works on DNS query security phases and can act like IPS can help to solve such situations, and to make the security configurations in the workflows healthier.

Data

All kinds of outputs, results and interpretable that show how a job is done or what is achieved as a result of the work, is data. Humans, animals, machines, systems, and infrastructures are constantly generating data about themselves and others.

Data is considered one of the most precious values of our age because it can be understood, classified, categorized, and these outputs can generate serious awareness and intelligence. Many institutions and organizations doing business in the IT sector, makes large-scale investments in data collection and for their systems. Nowadays, big part of them build their fields of work; on data.

Information & Cyber Security

Intangible value called ‘information’, is that the general name given to meaningful and processed form of data collected from people, environment, systems and all similar points.

Information value may be general, useless, worthless and insignificant, also it can be special, valuable or precious. In the age we live in; classified, correctly interpreted and categorized information is incredibly valuable, due to the awareness and intelligence it can easily generate.

Since it is such a special, valuable and targeted value; information is obliged to be protected. The mechanism that undertakes this task is; information security. The purpose of all physical and cyber systems and structures operating within the concept of information security (primarily personally and publicly); is to protect information, to limit it where it belongs, to prevent leakage and to ensure that it is processed in accordance with the law.

The concept of information security is a broad that includes all the security measures taken within cyber space, as well as all physical.

Cyber security is a sub-branch in the world of information security. As mentioned, cyberspace is the environment that contains almost all of the information produced when the age we are in is considered. All institutions, individuals, software products and the like that serve the security goal in cyber space; It is within the subject of cyber security.

Also, security applications running on proprietary systems and devices, network security, application security, system security, infrastructure security, software security and all similar products within institutions, cyber security companies, personnel and systems of them, cyber security departments and teams, etc. It is all included in the concept of cyber security.

Infrastructure & Problems

Information technology infrastructures are machine and software communities produced to serve people. These topologies that receive, transfer, process, store and serve information are critical for the execution and continuity of a business.

It should be known that these systems are designed and built to work in very long terms and to provide continuous service. Considering the technical usability, the most important concepts about these systems are; it is convenience, continuity and accessibility. In other words, the more easily the system can be used by users, the higher the efficiency obtained. In terms of cyber security, problems arise right here. In order for a system to work continuously and to be accessible in all situations, the structure must be redundant and large.

When examined within cyber security concepts, the ease of use of a system or application means that its security is less efficient on the side of the CIA (Confidentiality, Integrity and Availability) triangle. That is, the easier a system is to use, the less reliable on security it is. On the other hand, the fact that the system grows in virtual volume means that the attack surface increases.

The insecurity of the system due to this surface area and the weakening of control mechanisms for usability; causing cyber attackers to target the structure, it also significantly triggers cyber-attack activities to be done.

Hacking People: Social Engineering

End of the day, structures and machines within information systems are used for serving people. As it is the uses, human beings are also an important factor from the cyber security perspective. The operations carried out within the scope of cyber-attacks are not limited to exploiting the security vulnerabilities of applications, information systems, machines and network elements. On the other side, the weakest link of all these systems is the human factor.

The hacks of attackers which is by deceiving people with their campaigns, images, texts and procedures, both virtually and physically; called social engineering attacks.

When a person clicks on a fake link sent to them, downloads a harmful file for being deceived, or gives their information to an attacker who is not an authority; he falls victim to a social engineering attack.

The DNS Scenario

Within information systems, machines communicate through each other with a location identifier called Internet Protocol (IP) address, which indicates where the machine is currently lands on the network. A user can access the website he wants to view with the help of the relevant IP address, but it is very difficult to keep in mind these IP addresses, which are only numbers and can be considered complex.

Because of this, the protocol called DNS created to make people's lives easier. When the user wants to reach a website, he does this by using the domain name he has in mind. The browser takes the domain address the user entered, and asks for a DNS server in institution. The DNS server returns the response; containing IP address of website server, which is mapped to that domain name. So, the browser accesses the website the user wants to view via this IP address.

To summarize; The user enters the domain name in the browser, the browser asks for this domain name to the DNS server, the DNS server gives the IP address of the related website, the browser goes to this IP address and displays the website to the user.

DNS Security (DNSSEC) & Vulnerability Pronities

This structure, which has been used for a long time around the world, is prone to both human and application-based security vulnerabilities, if the correct cyber security measures are not taken. For example, an attacker who took over the specified DNS server and changed the database records; can redirect users' browsers to malicious IP addresses, thereby allowing them to log into a fake website that will steal their information. Other examples may be like the following scenarios;

- The attacker purchases a domain name that looks very similar to the original domain, and builds a trustworthy-looking website; where he can get the registered users information. The attacker promotes and advertises this website to people through social media and bulk e-mails. The user, who is directed to a fake website by clicking on these attractive ads, enters their credit card information and falls victim to a social engineering attack. The attacker puts thousands of credit card information on the internet for sale and gains profit by harming users.

- The attacker purchases a domain name very similar to that of the organization and sends an e-mail to one of the administrators; with viruses, using the fake domain named e-mail account. Administrator does not understand that this email and domain name is fake, download the attached file to his computer and run it. The attacker gains remote access thanks to the malware, and begins to take over the corporates information system. After successful exploitation, attacker executes a ransomware on network and lock down the entire organization for ransom.

With these scenarios, it can be predicted how powerful a cyber-attack social engineering can be. But, if the internal DNS server could understand that domain names that users are trying to access during DNS resolutions could be fake or harmful, it would severely restrict the attacker's range of action and prevent cyberattacks from continuing.

Sedres is a secure DNS server application that was created to prevent such scenarios and was developed on the exact solution method mentioned above.

Sedres: The DNSSEC Security Layer Application

As mentioned before, people exhibit weaker than average decision mechanisms within emotional states such as fear, curiosity, excitement, enthusiasm and happiness. The attackers, who realize these situations and want to use them for their own benefit, organize campaigns to deceive people in cyberspace and obtain their personal information and sometimes their material assets.

Sedres is a software developed to eliminate such human-induced security problems in DNS systems and to prevent people from falling victim to social engineering attacks.

Mission & Vision

The curiosity, excitement and fear of the human factor make them the weakest link of security philosophies. According to Sedres's vision, the users in this situation should be prevented from being threatened. Preventing people and machines from being deceived and stabilizing information security will preserve motivation and peace.

Sedres has made it a mission that users who want to make DNS resolutions within the LAN and WAN; do not become victims of social engineering attacks.

Workflow & Usage

Sedres software is written in Python3 language and should be installed on a server that can obtain and run both this language and the libraries it needs. On the other hand, the application comes with some database, configuration and log files of its own. It is important that this file hierarchy is not broken during installation. Otherwise, the relevant file path changes must be specified in the source code.

Sedres is a DNS Server running within a LAN or WAN, depending the organizations network infrastructure configurations. Its use and efficiency can be explained simply;

The application is positioned as a DNS Server to be used by clients and / or servers within the relevant network infrastructure. The IP Address set as the DNS Service of the application is advertised on the clients' machines or servers.

After this phase, when the relevant machines want to do a DNS resolution, they will send these packets with domain names to Sedres's DNS Service. The application examines the relevant DNS packet received and passes certain security checks before responding. Meanwhile, a blacklist system is used.

As mentioned, the application has a black-list that can be added and changed by the operator. This list is updated periodically, with a delay period determined by the system administrator. The application uses suspicious DNS records published by the "Ulusal Siber Olaylara Müdahale Merkezi (USOM, TR-CERT/Turkey)" organization for this update process. Any other DNS recors can also be added to the list by the operator. In this way, addresses that are considered harmful by the institution or organization are introduced to the application.

If the requested domain name fails to pass the security checks, no response will be given by the application to the resolver. In this case, the user or the machine will not be able to learn the IP address of this domain name that may harm him and he will be protected from a possible cyber attack.

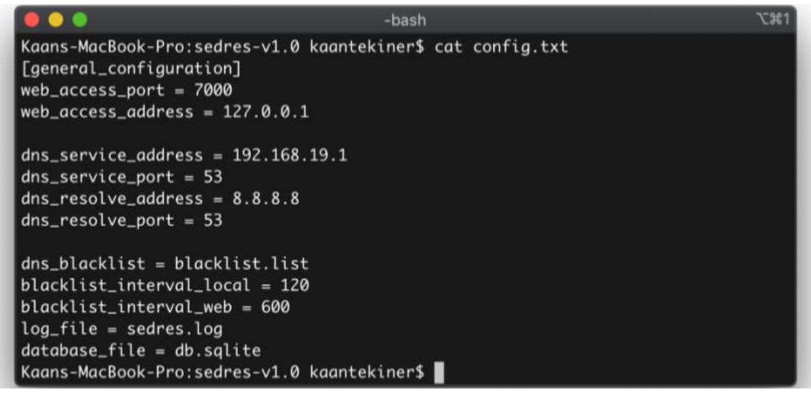
If the requested domain name is not included in the blacklist, the application decides that it should not be blocked and redirects the corresponding package to another DNS server specified by the system administrator. Application can use any valid IP Address for execute DNS resolves. After this action, it returns the IP Address response directly to the client.

Since Sedres does not change the originality of the DNS package, the system, process and algorithm can work stably.

Disclosure: It is important that the application is the only DNS Server of this machine in the client role. Otherwise, the client can make a request to a different server again for a domain name that the application indicates that it is unsafe and does not respond because of this. This can cause the client to learn the IP Address corresponding to the domain name. As a result, the client can become a victim of a cyberattack.

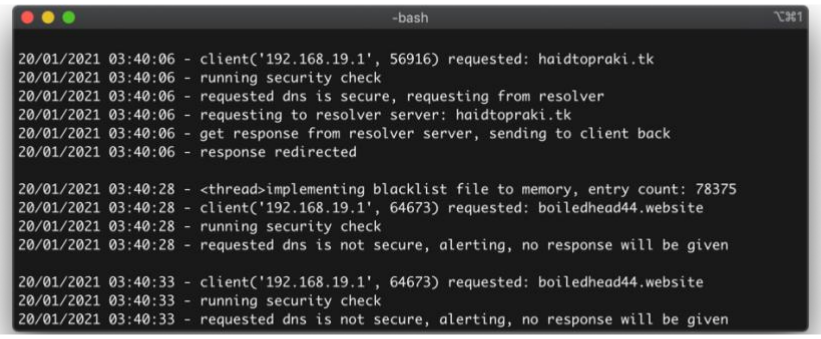
Disclosure: Both DNS Service and Web Service scripts must be executed for an efficient use. It can also be run without activating the web application.

Sedres performs logging and recording operations in real time. At the same time, it shows the DNS resolutions considered as malicious in an alarm list. Images of usage and runtime can be found at POC below.



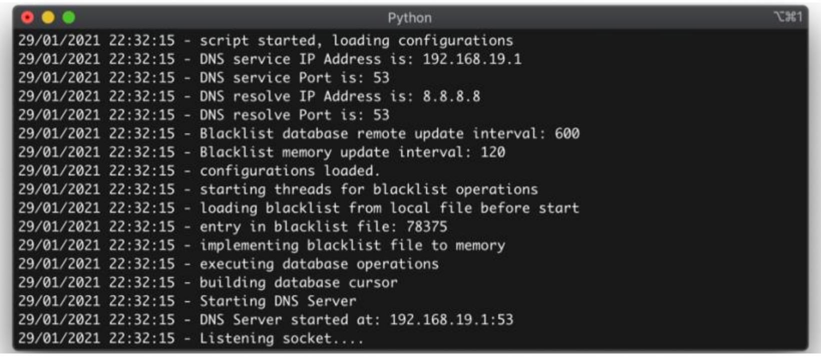
```
Kaans-MacBook-Pro:sedres-v1.0 kaantekiner$ cat config.txt
[general_configuration]
web_access_port = 7000
web_access_address = 127.0.0.1

dns_service_address = 192.168.19.1
dns_service_port = 53
dns_resolve_address = 8.8.8.8
dns_resolve_port = 53

dns_blacklist = blacklist.list
blacklist_interval_local = 120
blacklist_interval_web = 600
log_file = sedres.log
database_file = db.sqlite
Kaans-MacBook-Pro:sedres-v1.0 kaantekiner$ 
```

Figure 1 - Configuration options of Sedres application.



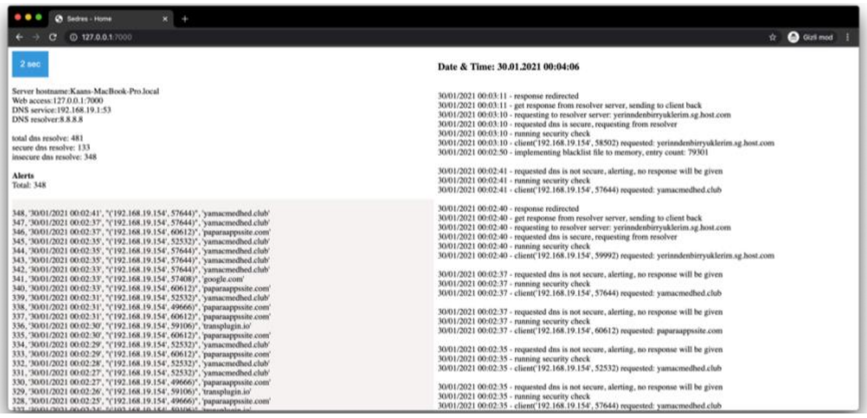
```
20/01/2021 03:40:06 - client('192.168.19.1', 56916) requested: haidtopraki.tk
20/01/2021 03:40:06 - running security check
20/01/2021 03:40:06 - requested dns is secure, requesting from resolver
20/01/2021 03:40:06 - requesting to resolver server: haidtopraki.tk
20/01/2021 03:40:06 - get response from resolver server, sending to client back
20/01/2021 03:40:06 - response redirected

20/01/2021 03:40:28 - <thread>implementing blacklist file to memory, entry count: 78375
20/01/2021 03:40:28 - client('192.168.19.1', 64673) requested: boiledhead44.website
20/01/2021 03:40:28 - running security check
20/01/2021 03:40:28 - requested dns is not secure, alerting, no response will be given

20/01/2021 03:40:33 - client('192.168.19.1', 64673) requested: boiledhead44.website
20/01/2021 03:40:33 - running security check
20/01/2021 03:40:33 - requested dns is not secure, alerting, no response will be given
```

Figure 2 - Application logs related to DNS resolutions and other operations.



```
29/01/2021 22:32:15 - script started, loading configurations
29/01/2021 22:32:15 - DNS service IP Address is: 192.168.19.1
29/01/2021 22:32:15 - DNS service Port is: 53
29/01/2021 22:32:15 - DNS resolve IP Address is: 8.8.8.8
29/01/2021 22:32:15 - DNS resolve Port is: 53
29/01/2021 22:32:15 - Blacklist database remote update interval: 600
29/01/2021 22:32:15 - Blacklist memory update interval: 120
29/01/2021 22:32:15 - configurations loaded.
29/01/2021 22:32:15 - starting threads for blacklist operations
29/01/2021 22:32:15 - loading blacklist from local file before start
29/01/2021 22:32:15 - entry in blacklist file: 78375
29/01/2021 22:32:15 - implementing blacklist file to memory
29/01/2021 22:32:15 - executing database operations
29/01/2021 22:32:15 - building database cursor
29/01/2021 22:32:15 - Starting DNS Server
29/01/2021 22:32:15 - DNS Server started at: 192.168.19.1:53
29/01/2021 22:32:15 - Listening socket....
```

*Figure 3 - DNS Server is listening for resolve client DNS requests, using **8.8.8.8** for redirect **secure ones.***



Server hostname:Kaans-MacBook-Pro.local
Web access:127.0.0.1:7000
DNS service:192.168.19.1:53
DNS resolver:8.8.8.8

total dns resolve: 481
secure dns resolve: 133
insecure dns resolve: 348

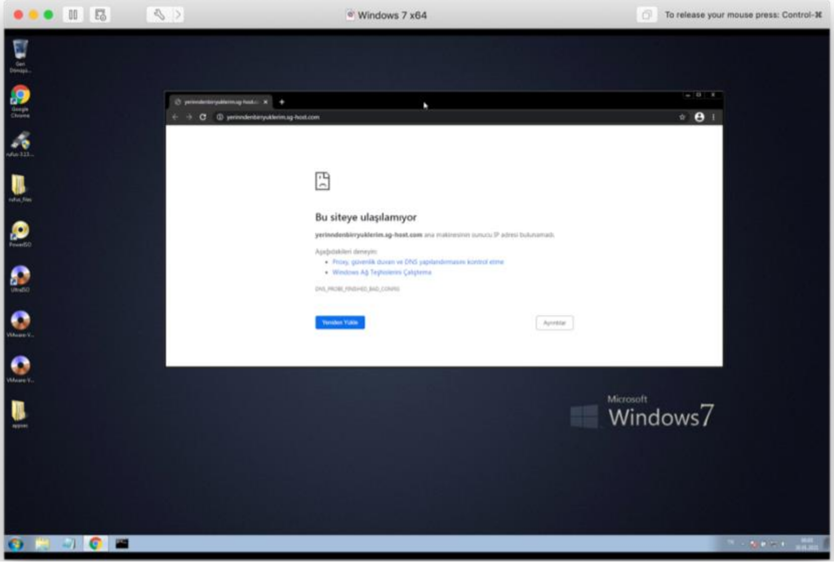
Alerts
Total: 348

```
348, '30/01/2021 00:02:41', "('192.168.19.154', 57644)", 'yamacmedhed.club'
347, '30/01/2021 00:02:37', "('192.168.19.154', 57644)", 'yamacmedhed.club'
346, '30/01/2021 00:02:37', "('192.168.19.154', 60612)", 'paparaappssite.com'
345, '30/01/2021 00:02:35', "('192.168.19.154', 52532)", 'yamacmedhed.club'
344, '30/01/2021 00:02:35', "('192.168.19.154', 57644)", 'yamacmedhed.club'
343, '30/01/2021 00:02:35', "('192.168.19.154', 57644)", 'yamacmedhed.club'
342, '30/01/2021 00:02:33', "('192.168.19.154', 57644)", 'yamacmedhed.club'
341, '30/01/2021 00:02:33', "('192.168.19.154', 57408)", 'google.com'
340, '30/01/2021 00:02:33', "('192.168.19.154', 60612)", 'paparaappssite.com'
339, '30/01/2021 00:02:31', "('192.168.19.154', 52532)", 'yamacmedhed.club'
338, '30/01/2021 00:02:31', "('192.168.19.154', 49666)", 'paparaappssite.com'
337, '30/01/2021 00:02:31', "('192.168.19.154', 60612)", 'paparaappssite.com'
336, '30/01/2021 00:02:30', "('192.168.19.154', 59106)", 'transplugin.io'
335, '30/01/2021 00:02:30', "('192.168.19.154', 60612)", 'paparaappssite.com'
334, '30/01/2021 00:02:29', "('192.168.19.154', 52532)", 'yamacmedhed.club'
333, '30/01/2021 00:02:29', "('192.168.19.154', 60612)", 'paparaappssite.com'
332, '30/01/2021 00:02:28', "('192.168.19.154', 52532)", 'yamacmedhed.club'
331, '30/01/2021 00:02:27', "('192.168.19.154', 52532)", 'yamacmedhed.club'
330, '30/01/2021 00:02:27', "('192.168.19.154', 49666)", 'paparaappssite.com'
329, '30/01/2021 00:02:26', "('192.168.19.154', 59106)", 'transplugin.io'
328, '30/01/2021 00:02:25', "('192.168.19.154', 49666)", 'paparaappssite.com'
```

Date & Time: 30.01.2021 00:04:06

```
30/01/2021 00:03:11 - response redirected
30/01/2021 00:03:11 - get response from resolver server, sending to client back
30/01/2021 00:03:10 - requesting to resolver server: yerinndenbirryuklerim.sg.host.com
30/01/2021 00:03:10 - requested dns is secure, requesting from resolver
30/01/2021 00:03:10 - running security check
30/01/2021 00:03:10 - client('192.168.19.154', 58502) requested: yerinndenbirryuklerim.sg.host.com
30/01/2021 00:02:50 - implementing blacklist file to memory, entry count: 79301

30/01/2021 00:02:41 - requested dns is not secure, alerting, no response will be given
30/01/2021 00:02:41 - running security check
30/01/2021 00:02:41 - client('192.168.19.154', 57644) requested: yamacmedhed.club

30/01/2021 00:02:40 - response redirected
30/01/2021 00:02:40 - get response from resolver server, sending to client back
30/01/2021 00:02:40 - requesting to resolver server: yerinndenbirryuklerim.sg.host.com
30/01/2021 00:02:40 - requested dns is secure, requesting from resolver
30/01/2021 00:02:40 - running security check
30/01/2021 00:02:40 - client('192.168.19.154', 59992) requested: yerinndenbirryuklerim.sg.host.com

30/01/2021 00:02:37 - requested dns is not secure, alerting, no response will be given
30/01/2021 00:02:37 - running security check
30/01/2021 00:02:37 - client('192.168.19.154', 57644) requested: yamacmedhed.club

30/01/2021 00:02:37 - requested dns is not secure, alerting, no response will be given
30/01/2021 00:02:37 - running security check
30/01/2021 00:02:37 - client('192.168.19.154', 60612) requested: paparaappssite.com

30/01/2021 00:02:35 - requested dns is not secure, alerting, no response will be given
30/01/2021 00:02:35 - running security check
30/01/2021 00:02:35 - client('192.168.19.154', 52532) requested: yamacmedhed.club

30/01/2021 00:02:35 - requested dns is not secure, alerting, no response will be given
30/01/2021 00:02:35 - running security check
30/01/2021 00:02:35 - client('192.168.19.154', 57644) requested: yamacmedhed.club
```

Figure 4 - Sedres is running with web application panel.



Windows 7 x64
To release your mouse press: Control-⌘
yerinndenbirryuklerim.sg-host.com
Yeniden Yükle
Ayrıntılar
Microsoft
Windows7

Bu siteye ulaşılamıyor

yerinndenbirryuklerim.sg-host.com ana makinesinin sunucu IP adresi bulunamadı.

Aşağıdakileri deneyin:

- Proxy, güvenlik duvarı ve DNS yapılandırmasını kontrol etme
- Windows Ağ Teşhislerini Çalıştırma

DNS_PROBE_FINISHED_BAD_CONFIG

*Figure 5 - When a client asks for **insecure** resolution, get timeout **by Sedres.***

In the POC study supported by visual citations; It can be seen that the application runs stable and performance, blocks all DNS requests that it decides to be unsafe, while allowing those that it sees as safe, and executing sessions/connections successfully.

Usage Areas

Sedres can be used in any structure that performs a DNS resolution. Considering that there are more clients requesting IP Addresses of different domain names within LAN structures, it would be more appropriate to use them in these structures. The most suitable points where the Sedres server can be located can be listed simply;

Active Directory Areas

When using Active Directory systems within LAN structures, DNS resolution methods are usually performed through this server. Clients make a request to port 53 of the AD machine, and in return they obtain their IP Address. Sedres can be configured as the DNS server machine that AD itself uses. In this logic, the AD machine will process DNS operations first, and it will ask Sedres for all domain names it deems appropriate. Sedres will provide access to the internet through the Firewall and a secure process will be processed at the end of the day. The first scenario specifies a secure DNS resolution.

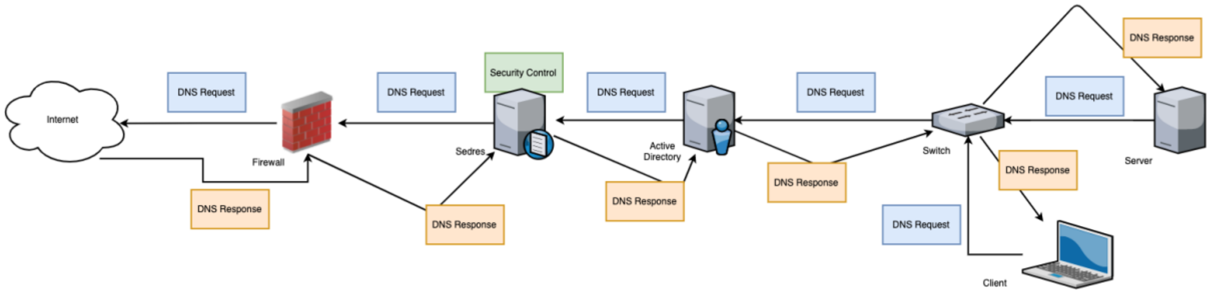


Figure 6 – Secure DNS Resolution in Active Directory Topology

If Sedres decides that the request is unsafe, it will time out the request and activate its alarm mechanisms.

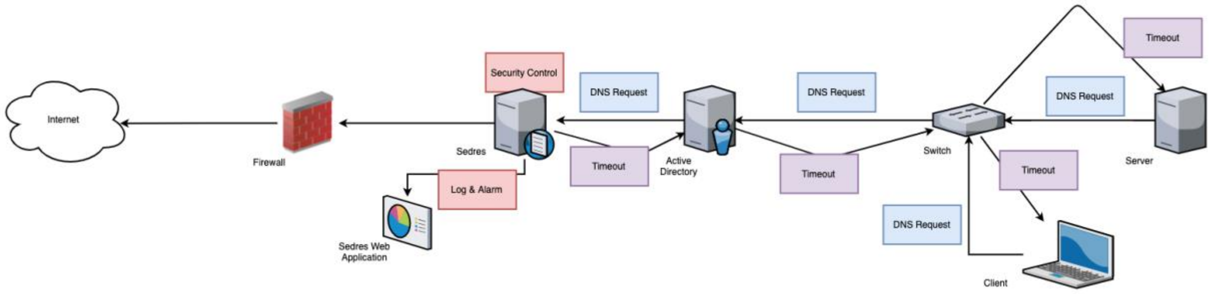


Figure 7 – Insecure DNS Resolution in Active Directory Topology

Firewall Endpoints

When Active Directory-like structures are not used, clients and servers can try to do DNS resolution directly over Firewalls and the internet. This is a method that is much more suitable for threats. If Sedres is positioned between the Firewall and Switch devices in these structures, it can provide a secure resolution. The first scenario specifies a secure DNS resolution.

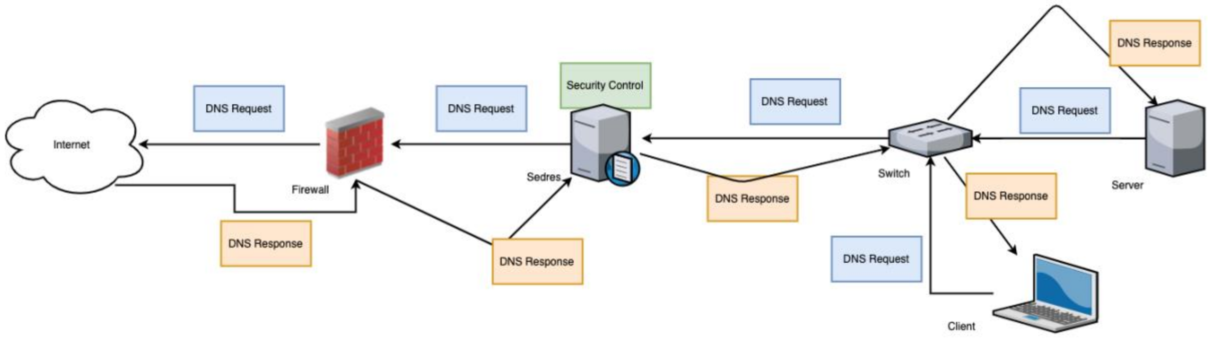


Figure 8 – Secure DNS Resolution in Firewall Endpoint Topology

If Sedres decides that the request is unsafe, it will time out the request and activate its alarm mechanisms.

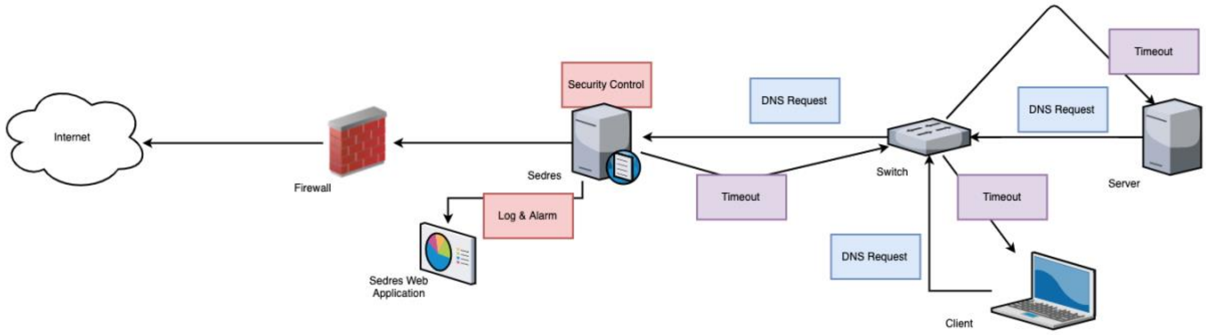


Figure 9 – Secure DNS Resolution in Firewall Endpoint Topology

DMZ Server Architecture

DMZ networks are an area that just web servers and machines which accessible over the internet is deployed. They are protected by special Firewall systems called WAF. If an attacker hijacks the relevant machines or tricks them into resolving a malicious DNS Address, Sedres can block it. The first scenario specifies a secure DNS resolution.

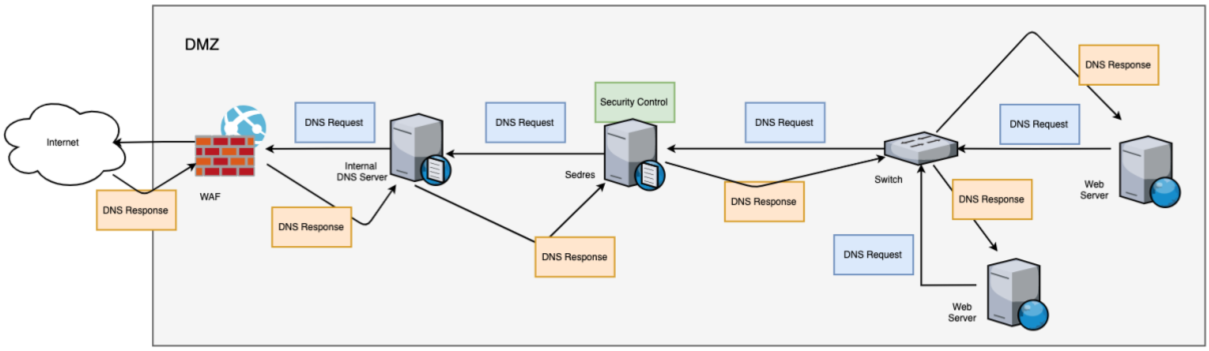


Figure 10 – Secure DNS Resolution in DMZ Server Topology

If Sedres decides that the request is unsafe, it will time out the request and activate its alarm mechanisms.

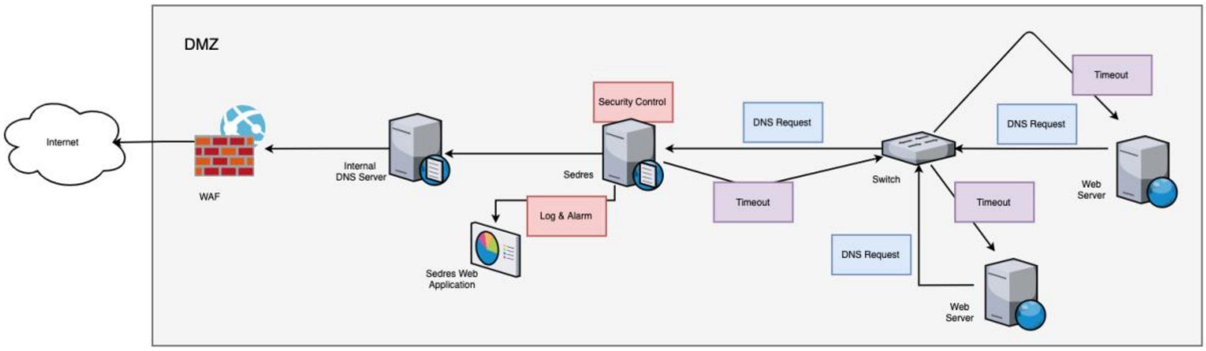


Figure 11 – Insecure DNS Resolution in DMZ Server Topology

Terminology, Technical Overview and Mechanisms

Reference [9] [10] [11], the technical explanations and descriptions within it, clearly describe the technical terms and terminologies within the scope of the Sedres application.

Intrusion Detection System: *'A computer-implemented intrusion detection system and method that monitors a computer in real-time for activity indicative of attempted or actual access by unauthorized persons or computers. The system detects unauthorized users attempting to enter into a computer system by comparing user behavior to a user profile, detects events that indicate an unauthorized entry into the computer system, notifies a control function about the unauthorized users and events that indicate unauthorized entry into the computer and has a control function that automatically takes action in response to the event.'*[9]

Intrusion Prevention System: *'An IPS can be defined as an in-line product that focuses on identifying and blocking malicious network activity in real time. In general, there are two categories: rate-based products; and content-based (also referred to as signatureand anomaly-based). The devices often look like firewalls and often have some basic firewall functionality. But firewalls block all traffic except that for which they have a reason to pass, whereas IPS pass all traffic except that for which they have a reason to block.'*[10]

Proxy: *'A local proxy that is located between a client device and a host system may be used to identify, or provide information about, a client device or identity using a client device that accesses a host system. The local proxy may append parental control information (such as a parental control level) to communications sent by the client device. The host system may provide, or restrict, access to information or features based on the information appended to communications sent by the client device.'*[11]

From a technical point of view, Sedres is a simple Proxy structure. Thanks to the "intervening" and “intercepting” features provided by this structure and the system, it can act as an "Intrusion Prevention System" and an "Intrusion Detection System" simultaneously and in real time.

If the IP Address of the machine that clients recognize as the DNS Server is set as Sedres' DNS Service IP Address, all DNS requests made by the clients will forward to the relevant listener port of the Sedres application.

Sedres uses the Python socket library to listen for requests from the relevant port. The request loads that come with a mixture of bytes and strings are first made meaningful by using the necessary and appropriate parsing methods; and then transmitted to the algorithm for security checks.

Sedres uses a blacklist as a security solution. This list, determined by introducing the values read from a file in the machine to the system as a string, is a database of malicious domain names decided by USOM or the operator. Sedres periodically sends requests to update this list by USOM. Therefore, the Sedres server must have internet access. At the same time, the application maintains a database for the logging operation. In this phase, the SqLite3 database and the corresponding Python library are used.

The follow-up of the works, configurations and records can be done directly on the system, but this situation reduces the efficiency of the ease of use. Sedres has its own web application as a solution to this situation. To access the panel, the IP Address determined during the installation can be used. The web application is run using the Python Flask library.

Application version is currently **v1.0.**

Pros

- Sedres does not accept network packets larger than 512 bytes in size in order to avoid denial of service attacks. This can prevent possible TCP phase attacks. At the same time, the relevant listener port is configured to only listen to UDP packets.

- Since the application has an IPS-based structure and is set up in Proxy scenario, it can perform instant and real-time controllers, as well as real-time logging.

- It uses a simple web panel for easy monitoring, which can controlled by the related operator. Due to the structure of HTTP systems, the relevant panel can also be remotely controlled when the relevant server is exposed to the Internet network with NAT operations.

- Due to its nature, the application can be positioned between any two DNS endpoints. In structures where Active Directory systems exist, if the application is hosted between AD and Firewall structures, it can provide maximum efficiency.

Cons

- Even if the application has a suitable infrastructure to integrate load distribution systems, it does not have these features on the basis of the relevant version. This causes the application to experience performance losses under heavy requests or load and because of that, responds late to the DNS queries offered.

- No special Exeption structures are used for error handling. This situation should be corrected and every error that occurs should have an accurate return, that can be followed.

- The application is run on the basis of the scripting system of the Flask library. This situation is not suitable for "best practice" structures. A web service to be implemented will increase performance and stability.

- Byte based Header information of the supplied packets is not parsed using stable methods. To correct this situation, an appropriate parsing algorithm should be used.

- Sedres only displays the alarms it generates in a list on the web panel. E-mail, SMS and mobile application support should also be provided to strengthen notification and information delivery mechanisms.

Conclusion

In our age, data and information have turned into the most valuable raw materials. When the digitalized world is examined, it is obvious that these raw materials can now be obtained, transported and seized very easily.

On the other hand, Domain Name Systems has always facilitated the lives of people and machines, and has become one of the most necessary and popular mechanisms of internet, intranet and extranet infrastructures.

The link between these two issues is that DNS systems are the first of many mechanisms that people use to advertise their information. Within cyber security operations, in cyber space; Some malicious fictions and social engineering campaigns which targeting DNS systems; causes personal informations to be obtained unauthorizedly, by very natural and simple methods.

Sedres is a security application product developed by Kaan Tekiner, in order for individuals and end users to use the internet and local network structures securely, not to be affected by attacks caused by manipulations in DNS systems, and to protect their personal information from attackers.

References

[1] *(DNS Security - Antonio Lioy, Fabio Maino, Marius Marian, Daniele Mazzocchi, Dipartimento di Automatica e Informatica, Politecnico di Torino, Terena Networking Conference, May 22-25, 2000*

[2] RFC 4033[DNS Security Introduction and Requirements - R. Arends, R. Austein, M. Larson, D. Massey, S. Rose, March 2005]

[3] Antonakakis M., Dagon D., Luo X., Perdisci R., Lee W., Bellmor J. (2010) A Centralized Monitoring Infrastructure for Improving DNS Security. In: Jha S., Sommer R., Kreibich C. (eds) Recent Advances in Intrusion Detection. RAID 2010. Lecture Notes in Computer Science, vol 6307. Springer, Berlin, Heidelberg.

[4] Domain Name System (DNS) Security, Diane Davidowicz, 1999

[5] DNS Privacy Considerations - S. Bortzmeyer, August, 2015

[6] The Impact of Time on DNS Security, Aanchal Malhotra, Willem Toorop, Benno Overeinder, Ralph Dolmans and Sharon Goldberg, Boston University, NLnet Labs, Amsterdam, July 5, 2019

[7] The Impacts of DNS Protocol Security Weaknesses, Willian A. Dimitrov and Galina S. Panayotova, University of Librarian Studies and Information Technologies, Sofia, 1784, Bulgaria

[8] DNS Security Introduction and Requirements, Scott Rose, Matt Larson, Dan Massey, Rob Austein, Roy Arends, Colorado State University, NIST, March 2005

[9] Intrusion Detection System, Craig H. Rowland, United States Patent, Mar 12, 1999, Austin, TX (US)

[10] Intrusion Detection Systems and Intrusion Prevention Systems, Andreas Fuchsberger, Information Security Group, Royal Holloway, University of London, Egham, Surrey TW20 0EX, United Kingdom

[11] Local Proxy Server for Establishing Device Controls, Patrick Meenan, Donald P. Sengpiehl, Rich Thornberg, United States Patent, Jul. 31, 2002, Gainesville, VA (US)