I.J. Wireless and Microwave Technologies, **2021, 6, 30-45**
Published Online December 2021 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2021.06.04

D-TS: A Secure and Trusted Authentication Framework for Domain Name Server

Usman Aijaz N
HKBK College of Engineering / Department of Information Science and Engineering VTU, HKBKCSERC, Bangalore-560045, India
E-mail: usman.is@hkbk.edu.in

Syed Mustafa
HKBK College of Engineering / Information Science and Engineering, Bangalore-560045, India
E-mail: mustafas.is@hkbk.edu.in

Mohammed Misbahuddin
CDAC (Centre for Development of Advanced Computing)/ACTS & BD, Bangalore- 560 100
E-mail: mdmisbahuddin@gmail.com

Received: 24 August 2021; Accepted: 26 September 2021; Published: 08 December 2021

Abstract: DNS is responsible for the hostname to IP address translation. It is an open resolver that's why vulnerable to different kinds of attacks such as cache poisoning, man-in-the-middle, DOS and DDOS, etc. DNS is responsible for the hostname to IP address translation. To protect DNS IETF added a layer of security to it known as Domain Name System Security Extensions (DNSSEC). DNSSEC is also vulnerable to phishing, spoofing, and MITM attacks. To protect DNS, along with DNSSEC we require certifying authorities to authenticate the communicating parties. DNSSEC combined with an SSL certificate issued by Certifying Authorities (CA's) can protect the DNS from various attacks. The main weakness of this system is there are too many CA's and It is not feasible to trust all of them. Any breached CA can issue a certificate for any domain name. A certificate issued from a compromised CA's is valid. In this scenario, it is necessary for the organization to limit the number of CAs and to check whether the server is signed by a trusted CA's or not. DNS Based Authentication of Named Entities (DANE) permits a domain possessor to stipulate specific CA's issue certificates for a specific resource. DANE will not allow any CA to issue certificates for any domain. It limits the number of CA's used by the client. As there were still some security issues left in it that can be resolved using a mechanism called D-TS. It is a DANE-based trusted server that acts as a third party and validates the certificates of all the entities of the network. D-TS will be a proof-of-concept for enhancing the security in communications between Internet applications by using information available in DNS. The system attempts to solve the shortcomings of DANE by establishing a trust zone between the clients and the services. By adding multiple levels of validations, it aims to provide improved authenticity of services to clients, thereby mitigating attacks like phishing, Spoofing, Dos, and man-in-the-middle attack. In this paper, we will discuss the detailed working of our proposed solution D-TS.

Index Terms: Certifying Authority (CA), DNS, DNSSEC, DANE, D-TS

1. Introduction

The Domain Name System (DNS) is an important source for communicating or message passing on the Internet. It translates the domain names to IP addresses. DNS servers eradicate the necessity for the users/customers to remember IP addresses. DNS is an overcritical part of the internet without it, one cannot imagine the internet-driven society of today. It was designed in an era where people did not envision the massive deployment of networks that the internet has become today. Hence they didn't put much thought into securing it. So, even today normal DNS queries are still sent in plain text. With the development of technology, the benefits of the internet have increased but at the same time even there are risks. The current DNS infrastructure is vulnerable to many kinds of attacks like cache poisoning, man-in-the-middle, DNS hijacking, Tampering, DOS, DDOS, and many more. So the attacker by using any one of these attacks can redirect the user to fake websites. As a result of this, an attacker can steal login credentials and other confidential information of the user. Therefore DNSSEC was introduced by IETF as a cryptographic framework configured on an open-source platform to secure the DNS system. It assures authenticity and provides integrity for the data received

from the DNS server. It makes use of Asymmetric key cryptography to prevent some of the known attacks on DNS but still, it did not make communication on the internet any more secure. DNSSEC does not protect DNS from DOS/DDOS, MITM, Phishing attacks. DNSSEC makes use of EVSSL certificates for secure communication on the internet. This certificate helps to verify the authenticity of the server to the clients. However, this solution gave rise to more problems because there were too many CA's available in the market. If any CA gets compromised then the certificate issued by that CA is valid, an attacker can steal the confidential information of the user through these compromised CA's. In the same way, an attacker can steal the login credentials of the user by using a free SSL/TLS certificate issued Let's Encrypt. To tackle this problem new protocol has emerged called DANE. This protocol provides a way for the server to specify to the users, what certificate they should expect when they connect to the website. If a web browser supporting DANE detects that it is not using the specific certificate it can warn the user that the connection is insecure. DANE removes many security threats through its secure working such as Phishing, Spoofing, identity theft, and Man in the Middle attack. Furthermore, DANE itself is not sufficient to secure DNS against DDOS attacks. There is a need for a trusted system to secure DNS. The Main Objective of this paper is to show how Trusted DANE secures DNS by providing better authenticity of services to the client. Many solutions exist to protect DNS from DDos attacks but they intern vulnerable to some other attacks. The Proposed solution gives a completely secure and trusted security framework for DNS against various kinds of attacks.

This paper is structured as follows: section 2 presents Literature review, section 3 presents DNS and its security issues, Section 4 presents DNSSEC and its security issues, Section 5 presents a brief description of SSL and EV SSL certificates, Section 6 presents the Working of DANE and its security issues, Section 7 presents the working of proposed Solution Trusted DANE, section 8 presents Proposed Methodology for Securing DNS server, section 9 presents Result and Discussion and Section 10 presents the conclusion and future work.

2. Literature Review

Many researchers identified the vulnerabilities of DNS and proposed a different solution to secure DNS. Table 1 gives the different solutions proposed by the researcher to mitigate the attacks on the Domain name server. Nobody has proposed the trusted framework to the DNS server to mitigate all possible attacks on it.

Table 1. DNS Security System

SrNo	Proposed Method	MethodologyUsed	Security Services	Attack Detection	Attack Preventiom	Type Of Attack/ Problem	Remarks
1	Kopis System[17]	Detects malware Domains at upper DNS hierarchy by Observing network traffic	Integrity	No	Yes	Malware attacks	It can detect malware Domains even Reputation information is available
2	S-DNS[18]	Reductions the success probability of DNS spoofing and cache poisoning by avoiding man-in-the-middle attacks and provides backward compatibility and simple security solution with low computation and communication overheads.	Integrity	No	Yes	Cache poisoning attack	Protocol cooperates with Identity Based Encryption server for avoiding cache poisoning attack
3	A practical approach to distinguish between valid and bogus DNS replies [19]	Uses Iptables and routine fail2ban detection	Authentication	No	Yes	DDoS attack	Focuses only onthe prevention of amplifying DDoS attacks. Other types of DNS attacks are not taken into consideration

4	T-DNS [20]	Uses TCP and Transport layer security to provide DNS security	Consistency	No	Yes	Amplifier d DDoSattacks	The use of TCP improves the privacy and security of DNS as compare toUDP
5	DNS proxy Counter measure[21]	Uses DNSproxy server(DPS) and BIND	Integrity	Yes	Yes	Cache poisoning attack	No change in DNS standards. Performance impact isminimal
6	CGA-TSIG algorithm [22]	Public key cryptography is being used toform a connection with the DNSserver. Extends RDATA field of TSIG to implement security	Authentication on	No	Yes	DNS spoofing	Developed for stateless autoconfigurationn in IPv6 asDNSSEC and TSIG protocols were not able to dealwith stateless configuration in IPv6
7	Self-feedback correction system [23]	Provides correct mapping between a domain name and IP addressfor DNS wrongresource records	Integrity and Authentication	No	Yes	Malware attacks	Uses webpage fingerprint algorithm as one parameter to provide acorrect mapping of the domain name and IPaddress that does not work for HTTPS

3. DNS: Domain Name System

DNS resolves human-readable hostnames into machine-readable binary numbers like IP addresses. IP addresses are 32-bit numbers that are very difficult to memorize. An Internet service called DNS helps to access any system on the internet through name instead of its IP address [1]. The method of DNS resolution comprises translating a hostname (such as www.example.com) into an IP address (such as 192.168.1.1). DNS is the largest distributed hierarchical database as shown in figure 1. The various types of records stored in the DNS database are Start of Authority (SOA), pointers for reverse DNS lookups (PTR), SMTP mail exchangers(MX), IP addresses (A and AAAA), name servers(NS), and domain name aliases (CNAME)[6]. Although the DNS database is not a general-purpose database. It can also store records for other types of data like human queries such as responsible person (RP) records, or automatic lookups, such as DNSSEC records. The DNS database is warehoused in a structured zone file.

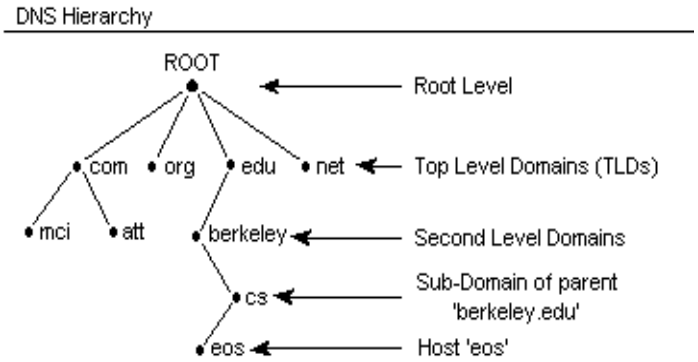

Fig.1. DNS Distributed hierarchical database [2]

3.1. Working of DNS

The client sends the requests to DNS Resolver for the IP address of the hostname www.example.com. DNS Resolver first contacts one of the root servers for the IP address of the hostname. If the root server does not have the IP address of www.example.com then it gives the Top Level Domain servers IP address. Then the Resolver contacts one of the TLD servers. The TLD server gives the IP address of an authoritative server for a particular hostname. The resolver

then contacts the authoritative server to get the IP address of the hostname. The authoritative server returns the IP address of the hostname www.example.com to DNS resolver which in turn returns to the client [8]. This Process may happen either iteratively or recursively as shown in figure 2.

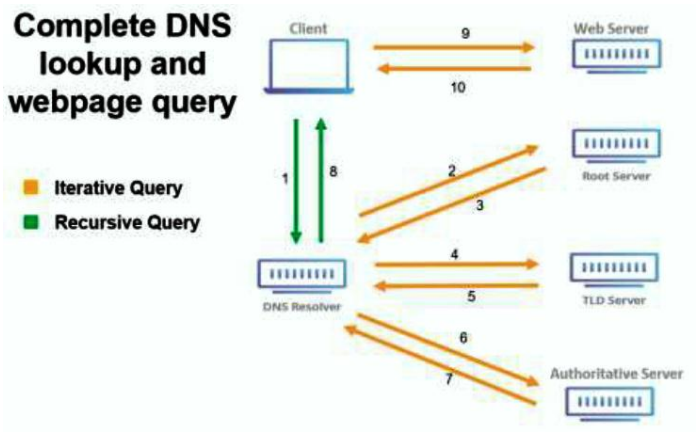

Fig. 2. Working of Domain Name Server

3.2 Attacks on DNS

DNS is available to all clients to give name resolution for their request irrespective of its domain. DNS is not configured with any administrative control, hence susceptible to different kinds of attacks as shown in figure 3. Some of these attacks such as DNS cache poisoning attacks, DOS/DDOS attacks, phishing, packet sniffing, and Man-in-the-Middle attack are explained in this section[3].

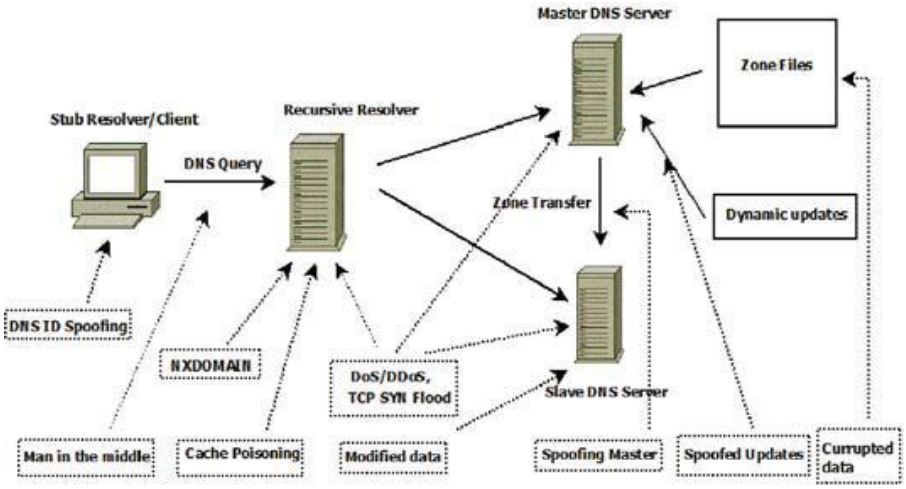

Fig. 3. Attacks on DNS infrastructure [14]

- Denial of service (DoS) attack – The attacker uses one PC and an internet connection to initiate this attack. This attack aims to flood the target DNS server with a fake DNS request so that the server or few websites are inaccessible to the legitimate user[15].
- Distributed denial of service (DDoS) attack – Attacker uses many PCs and an internet connection to initiate this attack. This attack aims to degrade the performance of the target DNS server by sending an overwhelming number of requests from hundreds or thousands of computers. This attack also aims to consume all the resources of the victim machine with unnecessary and overloading traffic[16].
- DNS spoofing (also known as DNS cache poisoning) attack – All DNS queries for name resolution are answered by the DNS server first from its cache. The attacker attempts to poison the DNS cache by storing malicious data in it. Even after entering the correct URL by the victim in a Web browser, he will be redirected to a fake website. These bogus websites are completely under the control of attackers to steal victim's sensitive data[13].
- Phishing attack- It is an illegal attempt by the attacker to steal the confidential information of the user such as login credentials and credit/debit card numbers. The attacker does this by disguising himself as a trustworthy entity and redirects the victim to a fake website whose appearance and feel are the same as a legitimate

website. Normally this kind of attack is carried out by email spoofing and text messaging.

- Packet sniffing- It is the act of capturing, gathering some or all packets of data passing through a network. Packet sniffers are used to doing this task. The data being captured can be passwords, IP addresses, protocols used in the network, and other data that will benefit the attacker to intervene in the network.
- Man-in-the-middle attack -MITM is an attack where an attacker is sitting between two communicating parties[1]. Two parties think that they are communicating directly with each other but their communication is controlled by an attacker. The aim of an attacker with an attack is to snip private data such as login credentials, debit/credit card numbers, and other details. Data acquired from this attack could be used for diverse purposes, like identity theft, illegal fund transmissions, or an illegal password change.

3.3 Drawbacks of DNS

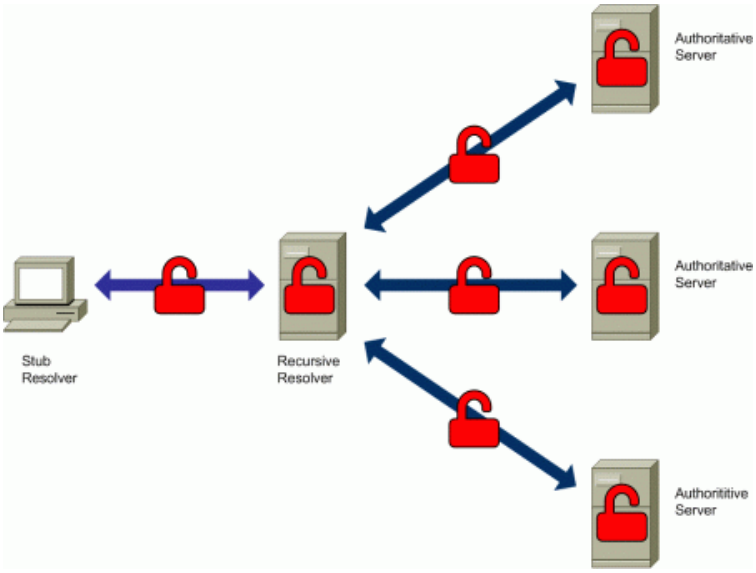

Fig. 4. Drawback of DNS

The above figure 4 shows the two main types of communications that take place as part of DNS resolution, along with the systems involved in communications:

- Communication between the stub resolver and the recursive resolver
- Communication between the recursive resolver and the authoritative DNS servers

In these two cases, the communication between the involved systems occurs in plain text. In DNS protocol no security system avoids an eavesdropper from seeing the content of DNS queries and responses, along with the addresses of the systems participating in the DNS "communications". The systems processing the queries, the recursive resolver, and the authoritative nameservers will have access to the queries themselves and possibly to other related information.

The DNS was initially developed without any kind of attention for user privacy and therefore leak information about DNS queries and responses that can be associated with specific network activity (e.g. applications employed, websites visited, people communicated with, etc.).

DNS information may leak at:

- The communications links and devices between the stub resolver and the recursive resolver;
- The recursive resolver;
- The communications links and devices between the recursive resolver and the authoritative DNS servers, and;
- the authoritative nameservers

Any entity with access to the communications links or devices between the stub resolver and the recursive resolver, or the recursive resolver and the authoritative servers, can passively collect all or part of the DNS communications. On the other hand, an attacker that does not have access to such communications links or devices can also eavesdrop on such DNS communications. This can be achieved by attacking the routing system to divert traffic to a communication link the attacker can eavesdrop.

- Its registry can only be controlled by ICANN a nonprofit making organization that has roots tied in the USA.
- DNS queries are unencrypted and can be manipulated by hackers.

- DNS servers work on the attitude of a master-slave relationship which can be broken.

4. DNSSEC: Domain Name System Security Extensions

Internet Engineering Task Force (IETF) has developed DNSSEC to discourse some DNS security issues by digitally 'signing' the data to assure that it is valid or authentic. To eliminate the vulnerability from the Internet at each stage in the DNS lookup from the root zone to the final authentic name server's zone, DNSSEC must be implemented. DNSSEC toughens authentication in DNS using digital signatures. With DNSSEC, DNS queries and responses including DNS data are cryptographically signed by the proprietor of the data [11]. DNSSEC adds two significant features to the DNS protocol:

- Data origin authentication – is a property that data has not been altered or changed within transit and the receiving party can authenticate the source of the data.
- Data integrity protection – Aims to avoid unintended modifications to data. It allows the receiver to receive the same data sent by the sender. It assures that data hasn't been altered in transit since it was initially signed by the zone owner [11].

4.1. Benefits and Drawbacks of DNSSEC

DNSSEC is of utmost effectiveness when completely accomplished starting from the root zone and then to top-level domains (TLDs) and finally to individual domain names. DNSSEC safeguards information sent by DNS servers against fraud. DNSSEC standards such as RFC4033, RFC4034, and RFC4035 are proposed to address the problem of cache poisoning in DNS [1].

Some of the benefits of DNSSEC are

- Safeguards the clients to become victims of cybercrime.
- Protect and shape the brand and name of the organization.
- Preserve clients' trustworthiness
- Attract and retain security-focused customers

DNSSEC, like many things in this world, is not without its problems. Below are a few challenges and disadvantages that it faces

- More complexity due to additional records, keys, signatures, etc. [11].
- More maintenance burden on operators.
- Very low rates of adoption around the world.
- It does not provide Confidentiality.
- It does not protect DNS from DOS/DDOS attacks

5. Secure Socket Layer and Extended Validation SSL Certificate

SSL is developed by Netscape to offer a secure channel between communicating devices on the internet. SSL is a protocol in the network protocol stack. It exists between the application and the TCP/IP protocols [12]. SSL can be used by any application-level protocol. it is mainly used for securing HTTP transactions. SSL over HTTP provides HTTPS that is the best communication security available on the web till now. SSL Certificates are digital computer files mainly used for authentication and encryption of communicating parties and their messages respectively. SSL is created on the principle of PKI and uses a digital certificate to provide secure communication in a network. SSL binds a cryptographic key to an organization's details like hostname, name of the server domain name, and organizational identity information. Nowadays many applications are on the structure of client/server or web browser/web server model. It is of utmost essential to provide the authentication of clients and servers on the web. SSL is widely used in these applications to provide authenticity and security for them using the HTTPS protocol. An organization needs to install the SSL Certificate onto its web server to achieve secure and trusted communication with a web browser [1].

Advantages of an SSL Certificate are:

- Confidentiality of data (privacy) can be achieved.
- Data Integrity (Tamper Proofing) can be achieved
- Keeps data secure between servers
- Upsurges Google Rankings
- Increases customer trust.

The EV SSL (Extended Validation SSL) certificate is the utmost form of SSL certificate existing on the market [1]. During verification of an EV SSL Certificate, the proprietor of the website undertakes a detailed and universally standardized identity verification process. In this verification process, the Domain owner has to evidence the exclusive privileges to use a domain, authorize its legal, operational, and physical existence of the domain. He also needs to prove the authorization of an entity for the issuance of the certificate. This verified identity information is comprised of the certificate. EV SSL certificate helps to resolve Website phishing, or identity theft attacks, which is a foremost threat to legitimate/authentic websites and online services.

5.1 Benefits of DNSSEC and EV SSL

DNSSEC makes use of EV SSL certificates for secure transactions over the communication links. In the process of obtaining an IP address from DNS lookup, it checks the certifications of the domains, and also each resource record in the zone file is digitally signed to authenticate and validate its presence in the file. Only properly signed and verified resource records of an authentic and certified domain are sent to the recursive resolver through an SSL connection in DNSSEEC.

EV SSL provides website visitors with extra evidence to distinguish real websites from fake ones. It weakens the success of social engineering attacks. DNSSEC uses a trusted pair of public key(s)/private key(s) to sign the digital certificate. These certificates are stored in the DNS zone files and can be used to authorize the legitimacy and reliability of the information. The attacker cannot falsify these certificates and therefore DNS is protected from a cache poisoning attack [1].

Eavesdropping is a kind of Man-in-the-Middle attack also known as a snooping attack. Using this attack the attacker accesses the data transferred over the network between communicating parties [5]. This attack is cured using an EV SSL certificate, as encrypted data is transmitted over the network which cannot be decrypted and altered by the hacker [1].

5.2 Security Issues

DNSSEC complemented with EV SSL certificates protects DNS from many attacks. DNS cannot be protected from the attacks like phishing, spoofing, and man-in-the-middle, and DDOS attacks. To secure DNS against these attacks another more secure layer named DANE is added to DNSSEC.

6. DANE: DNS Based Authentication of Named Entities

DANE is a method that allows cryptographically secured communications for online Applications. DANE has created a new type of DNS TLSA record that permits a domain to specify authorized CA's to represent it. DANE uses DNSSEC to provide a foundation of trust, and with TLSA it can serve as an origin of trust for TLS certificates. The main objective of DANE is to discourse certain weaknesses of the current PKIX system. DANE allows domains to publish information secured with DNSSEC that can add additional security to PKIX certificates used for TLS. [4] Three different methods of operation are defined in the "certificate usage" field of a TLSA record in the DANE:

1. CA Constraints: The client should accept TLS certificates issued from a specific Certification Authority else theDANE will reject the TLS certificate declaring as forged.
2. Service Certificate Constraints: The TLSA record specifies the particular TLS certificate to be used; hence the client should accept only those certificates.
3. Trust Anchor Assertion: The TLSA record specifies the trust anchor for authenticating the certificates. The clientshould use that particular domain-provided trust anchor to authenticate certificates for that domain.

Network entities in DANE must publish the list of trusted anchors and specific CA's. At the time of handshaking, there must be the inclusion of certificates to authenticate the true domains by checking trusted anchors and authorizing CA. To eliminate the need for third parties it must provide a provision through which domains can sign their certificates [9].

6.1. DANE Working

As shown in Figure 5. In step 1 the client sends the IP address request to DANE server. In step 2, the root server receives the request and returns the RRset to DANE server in step 3. In step 4 and 5 Top level domain server receives the request and returns RRset to DANE srver. In step 6 and 7 the authoritative name server receives the request from DANE server and returns RRset along with TLSA record. The client asks its local DNS server for the TLSA record of www.example.com. The DNS server performs a normal DNS lookup for www.example.com TLSA record, uses DNSSEC to validate the response that came from the example.com authoritative name servers. After receiving the validated TLSA record, the client browser computes and compares the value of the TLSA record from DNS with the certificate received from the webserver. If the two matches then only the web browser load the page, if they do not match, the web browser displays a warning and does not load the page.

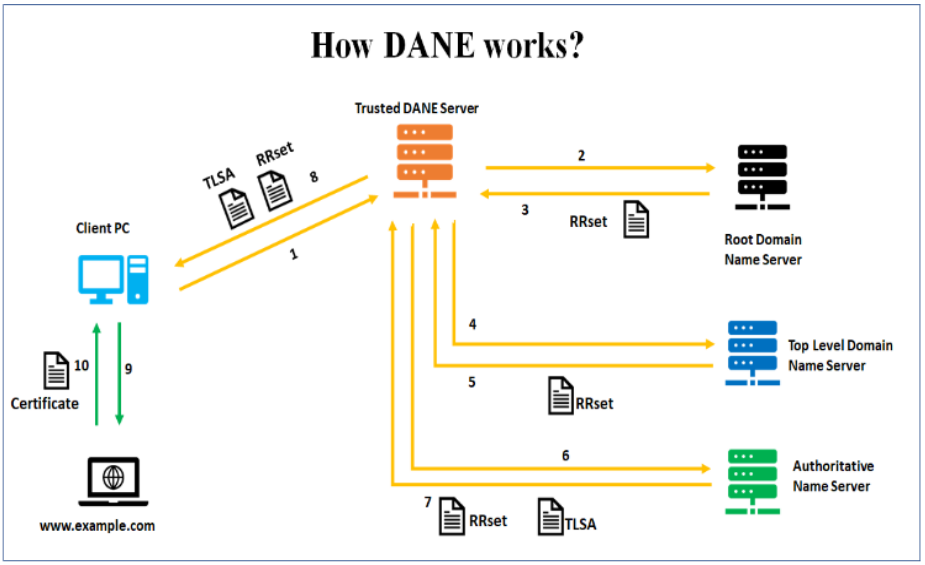

Fig.5. Working of DANE

6.2. Benefits of DANE

DANE is used to prevent phishing by using EV SSL certificates which authenticate the web pages by matching its authorized certificates with those provided by the domains and then verified. Accordingly, the domain is accepted as legal or rejected as fake. DNSSEC uses encrypted digital signatures to validate the servers. The certificates are checked (verified) and accordingly the access is granted. Only a valid certificate from the compromised Certificate Authority (CA) can spoof the data, else the spoofing is removed in DNSSEC. The compromised CA problem is further removed when we apply another more secure layer of DANE over DNSSEC [12]. The certificates from the DNS lookup and those provided by the host are matched in the case of DANE, thereby checking the authenticity of the host. The extra checking ensures that the certificates are valid and the CA has not been compromised.

Identity theft can be prevented by authenticating the user, i.e., checking whether the user browsing is authentic or not. A check on the servers should also be made so that the servers are not invalid. The certificates used by DNSSEC check the authenticity of servers while the EV SSL certificates used by DANE check the validity of the web pages, thereby, preventing identity theft.

Using the EV SSL/TLS certificates these attacks can be prevented. The highly encrypted messages can only be sniffed by the attacker but not intercepted. For reading or modifying the messages, the man-in-the-middle requires decrypting the information using the Private Key of the receiver which is with the receiver only therefore, no alteration can be made to the original text. DANE uses the EV SSL certificates; hence, this problem is solved using DANE.

6.3. Security issues

Despite protecting the network from a lot of cyber threats, there remain some trust issues in DANE. DANE does not protect in case of DDOS (Distributed Denial of Service) attacks. In the case of DDOS attacks, the online service is made unavailable by disrupting the normal traffic of the targeted server. This problem remains in the case of DANE, as DANE is only used to check the authenticity of the server. Building on the foundation of DNSSEC and improving the ideology behind the design of DANE, we propose a new solution for this problem named D-TS, DANE Based Trusted Server.

7. D-TS: DANE Based Trusted Server Proposed Solution

The main purpose of the proposed solution is to improve the security within the identity federation and also to secure the message communications between the sender and the receiver. To remove the security issues in DNS, a more secure layer of DNSSEC was added. DNSSEC validates and verifies the Resource Records stored in DNS zone files by validating, verifying, and authenticating the RRSIGs provided by the server. The signing certificates are issued by the authentic Certification Authority (CA), thereby, solving some security issues like unsigned spoofing (spoofing of domain names without valid certificates is removed), which were prevalent in DNS. In DNSSEC some issues remain that are when the authentic Certification Authority (CA) is tricked or deceived by some fraudulent user or attacker. In this case, the attacker gets valid certificates from the legal CA. The certificates issued by the compromised CA are valid. An attacker can use these certificates to redirect the user to some malicious sites [10]. This problem is solved by adding another layer of security known as DANE (DNS-based Authentication of Named Entities). As there was still some

security issues left with DANE, therefore, we have proposed our new layer of protection as D-TS (DANE-based Trusted Server) which acts as a third party and validates the certificates of all the entities of the network.

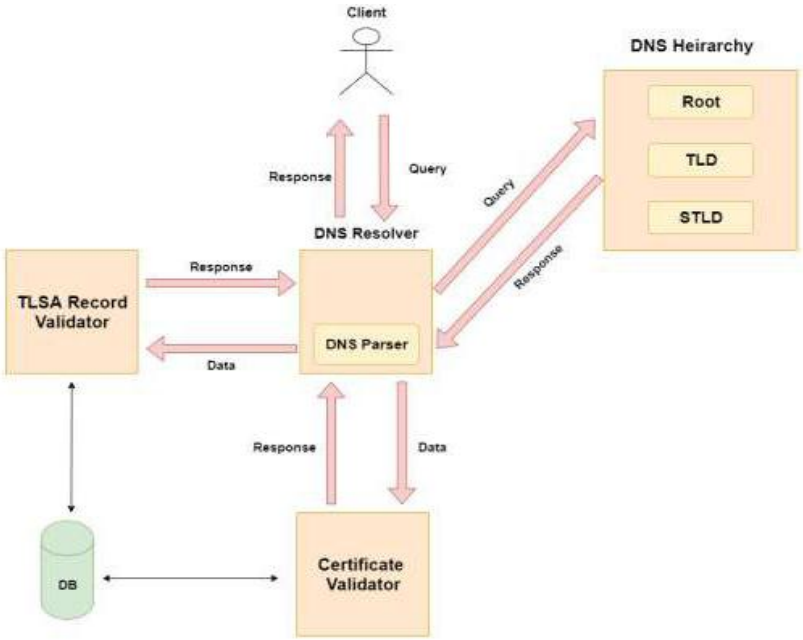

Fig.6. Proposed DANE Based Trusted Server

8. Proposed Methodology for Securing DNS server

8.1. Configuring DNS server

First, we need to install a DNS server by executing this command in the command prompt.

• Sudo apt-get installs bind9 DNS-utils

8.2 Configuring DNS Resolver

To provide DNS resolver services, we used BIND 9 – on Debian 10 with the following configuration:

• Conditional forwarding – To simulate forwarding queries to root.
• Recursion yes – to work as a recursive resolver.
• Access Control List – to allow queries from clients (for testing)

```
  GNU nano 3.2                        /etc/bind/named.conf.options
acl goodclients {

    192.168.157.0/24;
    localhost;
    localnets;

};

options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        forwarders {
                192.168.157.145;
                192.168.157.146;
                192.168.157.147;
         };
        forward only;

       recursion yes;
       allow-query { goodclients; };
```

8.3 Configuring Zone test. in

To set up multiple Authoritative name servers, we once again used BIND9 – on Debian 10 with the zone configured as follows. Configured zones are tested using both nslookup and dig utilities to verify it's working. We then digitally signed this zone for DNSSEC

```
 GNU nano 3.2                                  test.db

test.in.        3600    IN      SOA     ns.test.in.     root.test.in.   (
1234567;
1H;
3D;
1H;
2H;
)

test.in.        3600    IN      NS      ns.test.in.
ns.test.in.     3600    IN      A       192.168.157.145
www.test.in.    3600    IN      A       192.168.157.148
```

8.4 Configuring DNSEC

To enable DNSSEC on our resolver, for both sending DNSSEC queries and validating DNSSEC responses, we enabled the following options in BIND:

- Dnssec-enable yes – To enable sending DNSSEC queries.
- Dnssec-validation yes – To enable validating DNSSEC responses

```
 GNU nano 3.2                          /etc/bind/named.conf.options

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //------------------------------------------------------------------------
        dnssec-enable yes;
        dnssec-validation yes;

        listen-on-v6 { any; };

        recursion yes;
        allow-query { goodclients; };
};
```

To make our zones DNSSEC compliant, we have to digitally sign our DNS records using DNSSEC signing keys – ZSK & KSK. To generate these keys, we used the dnssec-keygen module of BIND9, as follows:

Command – dnssec-keygen -a NSEC3RSASHA1 -b 2048 -n ZONE test. In

- -a NSEC3RSASHA1 – This specifies the algorithm to be used. Use NSEC3 records, the RSA algorithm to generate keys, and use the SHA1 algorithm to generate hashes.
- -b 2048 – This specifies the size of the keys in bits.
- -n ZONE – This specifies who is the owner of the keys we are generating.

```
root@debian:/var/cache/bind# dnssec-keygen -a NSEC3RSASHA1 -b 2048 -n ZONE test.in
Generating key pair.............................+++++ ............................+++++
Ktest.in.+007+46277
root@debian:/var/cache/bind# ls
Ktest.in.+007+46277.key      managed-keys.bind      test.db
Ktest.in.+007+46277.private  managed-keys.bind.jnl
root@debian:/var/cache/bind#
```

Command – dnssec-keygen -f KSK -a NSEC3RSASHA1 -b 4096 -n ZONE test. In

- -f KSK – This is a flag to specify that keys are being generated for KSK.
- -b 4096 – Here a key of higher length is generated.


```
root@debian:/var/cache/bind# dnssec-keygen -f KSK -a NSEC3RSASHA1 -b 4096 -n ZONE test.in
Generating key pair.............................++++ ..................................++++
Ktest.in.+007+62088
root@debian:/var/cache/bind# ls
Ktest.in.+007+46277.key      Ktest.in.+007+62088.key      managed-keys.bind      test.db
Ktest.in.+007+46277.private  Ktest.in.+007+62088.private  managed-keys.bind.jnl
root@debian:/var/cache/bind#
```

8.5 Signing Zone

```
root@debian:/var/cache/bind# dnssec-keygen -f KSK -a NSEC3RSASHA1 -b 4096 -n ZONE test.in
Generating key pair.............................++++ ..................................++++
Ktest.in.+007+62088
root@debian:/var/cache/bind# ls
Ktest.in.+007+46277.key      Ktest.in.+007+62088.key      managed-keys.bind      test.db
Ktest.in.+007+46277.private  Ktest.in.+007+62088.private  managed-keys.bind.jnl
root@debian:/var/cache/bind#
```

ZSK and KSK have successfully been inserted into the zone file. Now to sign the test zone as shown below

```
root@debian:/var/cache/bind# ls
Ktest.in.+007+46277.key      Ktest.in.+007+62088.key      managed-keys.bind      test.db
Ktest.in.+007+46277.private  Ktest.in.+007+62088.private  managed-keys.bind.jnl
root@debian:/var/cache/bind# dnssec-signzone -t -g -o test.in test.db /var/cache/bind/Ktest.in.+007+
46277.private Ktest.in.+007+62088.private
dnssec-signzone: warning: test.db:16: using RFC1035 TTL semantics
Verifying the zone using the following algorithms: NSEC3RSASHA1.
Zone fully signed:
Algorithm: NSEC3RSASHA1: KSKs: 1 active, 0 stand-by, 0 revoked
                         ZSKs: 1 active, 0 stand-by, 0 revoked
test.db.signed
Signatures generated:                       9
Signatures retained:                        0
Signatures dropped:                         0
Signatures successfully verified:           0
Signatures unsuccessfully verified:         0
Signing time in seconds:                0.033
Signatures per second:                271.084
Runtime in seconds:                     0.038
root@debian:/var/cache/bind# ls
dsset-test.in.               Ktest.in.+007+62088.key      managed-keys.bind.jnl
Ktest.in.+007+46277.key      Ktest.in.+007+62088.private  test.db
Ktest.in.+007+46277.private  managed-keys.bind            test.db.signed
root@debian:/var/cache/bind# _
```

This shows that the zone was signed successfully and is now DNSSEC enabled/compliant

8.6 DANE TLSA Generation

To enable DANE for our sites, we have to first ensure that our sites are TLS enabled i.e. Apache servers configured with an X.509 certificate, generated using OpenSSL.


```
root@debian:~# openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-s
elfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt
Generating a RSA private key
...+++++
.............+++++
writing new private key to '/etc/ssl/private/apache-selfsigned.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:KA
Locality Name (eg, city) []:BAN
Organization Name (eg, company) [Internet Widgits Pty Ltd]:CDAC
Organizational Unit Name (eg, section) []:DITISS
Common Name (e.g. server FQDN or YOUR name) []:test.in
Email Address []:indercoolj123@gmail.com
root@debian:~# _
```

Now that TLS is enabled for all sites, TLSA records for these sites need to be generated and then inserted into their respective zone files. And since the zones are digitally signed, they will have to be resigned after inserting new data into it.

TLSA records were generated using the web interface. All service providers will be subjected to extensive verification processes at par with EVSSL certificate requirements. They are provided with an easy-to-use web portal as shown in fig, where they can:

- ➢ Register and Login.
- ➢ Register multiple services.
- ➢ Generate Digital Certificates if they don't have one.
- ➢ Validate their Certificates.
- ➢ Generate TLSA records for their services.
- ➢ Validate their TLSA record.
- ➢ Easily update service information or certificates if expired.

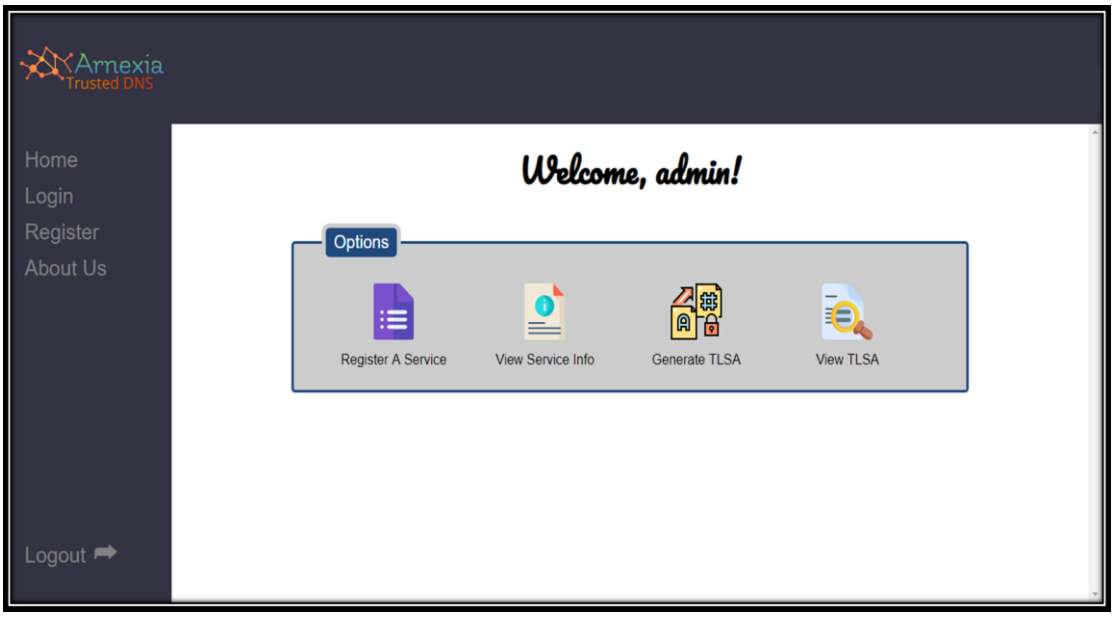

Fig.7. Web Interface for Service Provider

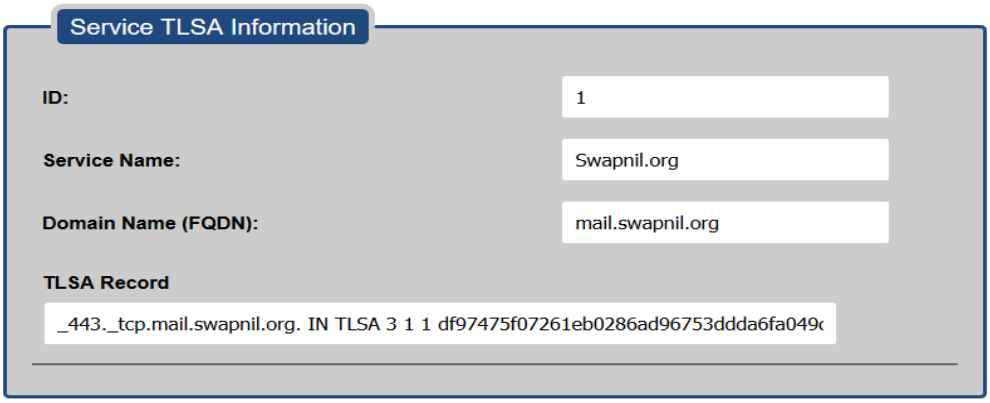

Fig.8. Web Interface for generating TLSA record.

TLSA record is inserted into the zone file and it is resigned.

9. Result and discussion:

D-TS allows checking the existence and validity of DNSSEC-signed DNS records. If a valid DNSSEC chain related to the requested domain has been found the plug-in will also check for the existence and validity of TLSA records. TLSA records store hashes of the remote server TLS/SSL certificates. The authenticity of a TLS/SSL certificate for a given domain name is verified by the DANE protocol (RFC 6698). DNSSEC and TLSA validation results are presented using various icons.

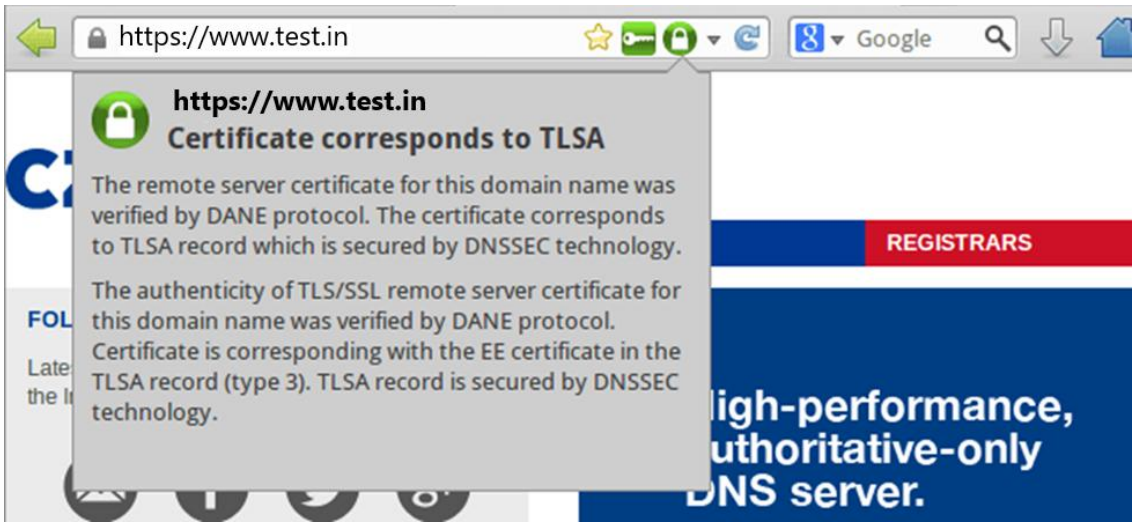

Fig.9. A domain name is correctly secured by DNSSEC and TLS/SSL certificate is verified by DANE Protocol.

For an existing domain name, this means that this domain name is correctly secured by DNSSEC as shown in Figures 9 and 10. Information about the IP address related to this domain name was validated using DNSSEC. End users are protected against domain name spoofing because this domain name is secured by DNSSEC.

For an existing domain name, this means that the authenticity of the remote server TLS/SSL certificate for this domain name was verified by DANE protocol. The certificate corresponds to the TLSA record which is secured by DNSSEC technology.

Fig 10. A domain name is correctly secured by DNSSEC

For an existing domain name, this means that the DANE protocol verification of the remote server's certificate for this domain name failed. The certificate does not correspond to the TLSA record which is secured by DNSSEC technology. This can be caused by trying to connect to an untrusted remote server or an invalid server certificate.

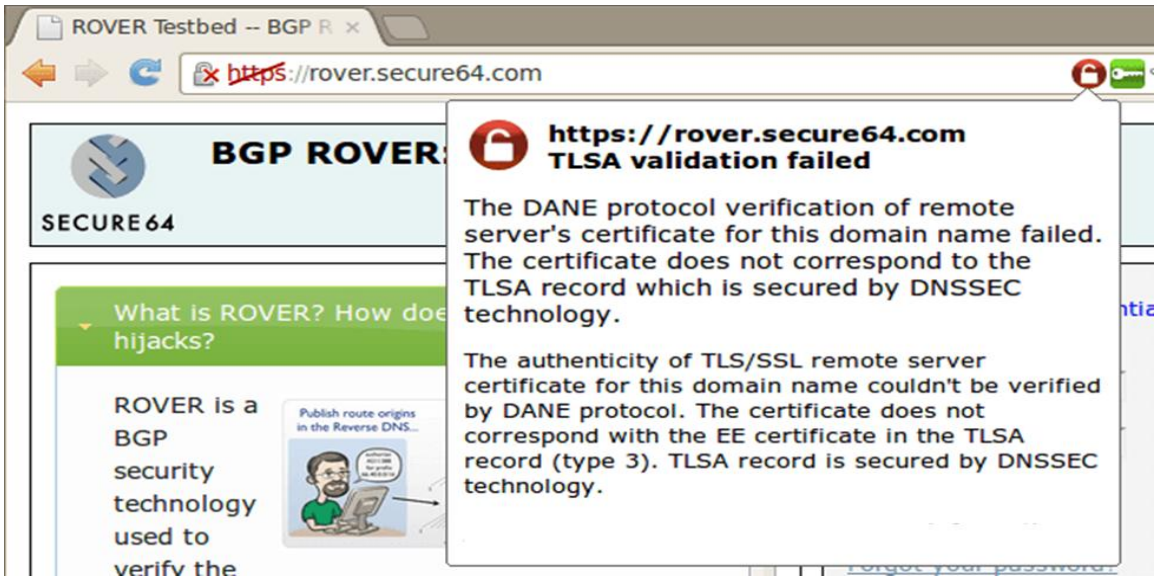

Fig 11. DANE protocol verification of remote server's certificate for this domain name failed

10. Conclusion and Future Implementation

The main aim of the proposed solution is to provide a secure and Trusted Framework for DNS. DNS is vulnerable to various attacks. IETF added DNSSEC as an additional layer of security to DNS to protect it from different attacks. DNSSEC is most effective when universally implemented but around the world, its rate of adoption is very low. DNSSEC does not solve all the security issues of DNS and it is vulnerable to Phishing, Spoofing, MITM, and DOS/DDOS attacks. The drawback of this method is compromised CA can deliver a certificate for any Domain name. DANE limits the number of CA's and it allows the owner of the domain to insist which CA's certificate is allowed for a specific resource. However, there are still some securities issues left in DANE which is cured or protected using the D-TS model. D-TS act as another security layer over DANE which provides extra security to the users if needed. D-TS using DANE is one way to go and is a valid solution for all DNS security issues. In this paper, we have fully explained the role of D-TS and its importance. D-TS verifies the Identity Provider, Service Provider, and also user. As we know that there is nothing secure in this security world and it's the trust that matters the most, therefore, if the D-TS server is itself compromised then there would be nothing more secure than this.

In the future, we can further improve this idea by implementing a few features that can complement the proposedsolution such as:

- Single-Sign-On (SSO) can be implemented within the trust zone established by the DNS server so that users can easily access services by logging in just once.
- Automation of the validation process will further enhance the performance.

References

[1] "Data Science and Security", Springer Science and Business Media LLC, 2021
[2] https://www.inetdaemon.com/tutorials/internet/dns/operation/hierarchy.shtml
[3] "17th International Conference on Information Technology–New Generations (ITNG 2020)", Springer Science and Business Media LLC, 2020.
[4] Đorđe Antić, Mladen Veinović. "Upgrading and Securing External Domain Space in the City of Niš Administration Infrastructure", Proceedings of the International Scientific Conference - Sinteza 2016
[5] Daniel M. Hein, Ronald Toegl, Stefan Kraxberger. "An autonomous attestation token to secure mobile agents in disaster response", Security and Communication Networks, 2010
[6] Zhenhua Li, Charles A. Kamhoua, Laurent L. Njilla, DaeHun Nyang. "Look-Aside at Your Own Risk: Privacy Implications of DNSSEC LookAside Validation", IEEE Transactions on Dependable and Secure Computing, 2020
[7] https://www.varonis.com/blog/what-is-saml/#:~:text= SAML%20works%20by%20passing%20 information, attempts%20to%20access%20those%20services.
[8] Sanjay, Balaji Rajendran, and Pushparaj Shetty Domain Name System (DNS) Security: Attacks Identification and Protection Methods Int'l Conf. Security and Management | SAM'18.
[9] C. Aishwarya, Raghuram M A, Sachin Hosmani, M.S. Sannidhan, Balaji Rajendran, K.Chandrasekaran, Bindhumadhava. "DANE: An inbuilt security extension", International Conference on Green Computing and Internet of Things (ICGCIoT), 2015.
[10] Jeremy Clark and Paul C. van Oorschot SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements 2013 IEEE Symposium on Security and Privacy
[11] Amir Herzberg, Haya Shulman, DNSSEC: Security and availability challenge 2013 IEEE Conference on Communications and Network Security (CNS).
[12] L. S. Huang, A. Rice, E. Ellingsen, and C. Jackson, "Analyzing Forged SSL Certificates in the Wild," *2014 IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, 2014, pp. 83-97, DOI: 10.1109/SP.2014.13.
[13] N. Usman Aijaz, Mohammed Misbahuddin, Syed Raziuddin. "Chapter 9 Survey on DNS Specific Security Issues and Solution Approaches", Springer Science and Business Media LLC, 2021
[14] Sanjay, Balaji Rajendran, and Pushparaj Shetty Domain Name System (DNS) Security: Attacks Identification and Protection Methods Int'l Conf. Security and Management | SAM'18
[15] Hariharan. M, Abhishek H. K, B. G. Prasad, "DDoS Attack Detection Using C5.0 Machine Learning Algorithm", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.9, No.1, pp. 52-59, 2019.DOI: 10.5815/ijwmt.2019.01.06
[16] Kaushik Sekaran, G.Raja Vikram, B.V. Chowdary, "Design of Effective Security Architecture for Mobile Cloud Computing to Prevent DDoS Attacks ", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.9, No.1, pp. 43-51, 2019.DOI: 10.5815/ijwmt.2019.01.05
[17] Manos Antonakakis, Roberto Perdisci, Wenke Lee and David Dagon, Detecting malware domains at the upper DNS hierarchy, August-2011, 27-27
[18] Ramzi Bassil, Roula Hobeica and Wassim Itani, Security analysis and solution for thwarting cache poisoning attacks in the domain name system, IEEE -2012, Electronic ISBN: 978-1-4673- 0747-5, Print ISBN: 978-1-4673-0745-1, Online ISBN: 978-1-4673-0746-8, DOI: 10.1109/ICTEL.2012.6221233.
[19] Muhammad Yasir Arafat, Muhammad Morshed Alam, and Feroz Ahmed, A Realistic Approach and Mitigation Techniques for Amplifying DDOS Attack on DNS, Proceedings of 10th Global Engineering, Science and Technology Conference 2-3 January 2015, BIAM Foundation, Dhaka, Bangladesh, ISBN: 978-1-922069-69-6.
[20] Liang Zhu, Zi Hu and John Heidemann, Connection-Oriented DNS to Improve Privacy and Security, IEEE-2015, DOI: 10.1109/SP.2015.18.
[21] Jonathan Trostle, Bill Van Besien and Ashish Pujari Protecting against DNS cache poisoning attacks, IEEE-2010, DOI: 10.1109/NPSEC.2010.5634454.
[22] Hosni Rafiee and Christoph Meinel, A Secure, Flexible Framework for DNS Authentication in IPv6 Autoconfiguration, IEEE-2013, DOI: 10.1109/NCA.2013.37.
[23] Caiyun Huang, Peng Zhang, Junpeng Liu, Yong Sun, Xueqiang Zou, SFCSD: A Self-Feedback Correction System for DNS Based on Active and Passive Measurement, arXiv: 1704.06569 [cs.NI].

Authors' Profiles

Mr. Usman Aijaz N did his B.E in (ISE) from Vidya Vikas College of Engineering in the year 2004, Mysore, VTU University, India. M.Tech (CSE) from Dayanand Sagar College of Engineering in the year 2010, Bangalore, VTU University, India. Currently pursuing a Ph.D. (CSE) in Cyber Security from VTU University, India.

He is having 15 years of teaching experience as an Assistant Professor. Currently, he is working as an Assistant professor at HKBK College of Engineering in Information Science and Engineering (ISE) Department, Bangalore, India. He has published two papers one in Springer and the other in the IEEE International Conference on cyber security. His research area is Cyber Security and Machine Learning.

Dr. Syed Mustafa obtained his Ph.D. in Computer Science and Engineering from Satyabhama University, Chennai, India. He is currently working as a Professor and Head of the Information Science and Engineering Department in HKBK College of Engineering under the Visvesvaraya Technological University. His area of research includes Web services, Web Mining, Social Media Data Mining, and Image Processing.

Dr. Mohammed Misbahuddin did his B.Tech (CSE) from Gulbarga University, M.Tech (S/w Engg.) from JNTU-Anantapur, and Ph.D. (CSE) in Network Security from JNTU Hyderabad. He is currently working as Joint Director (Scientist 'E') in the Centre for Development of Advanced Computing (C-DAC), E-City, and Bangalore. He is the Chief Investigator of the Cyber Security Awareness Project namely Information Security Education and Awareness (ISEA) – Phase II at C-DAC Bangalore. He is a key member of a Nationwide awareness project on Digital Signatures and PKI namely Next Generation PKI for Smart Applications. He is the Co-Investigator of a National Project named "e-Pramaan – A National e-Authentication Service along with Aadhaar". He has 17+ years of experience in Research, Training, and Project Management. He has applied for 3 patents with IPO in the area of Secure and Usable Authentication. He has been in various Programme committees of IEEE /ACM conferences and is a reviewer for two International Journals. His area of interest is Network Security & Cryptography especially Secure and Usable Authentication, Public Key Cryptography, and Risk-based Engines.

How to cite this paper: Usman Aijaz N, Syed Mustafa, Mohammed Misbahuddin, " D-TS: A Secure and Trusted Authentication Framework for Domain Name Server", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.11, No.6, pp. 30-45, 2021.DOI: 10.5815/ijwmt.2021.06.04

