Advances in Web Development Journal
vol. 1 no. 5, pp. 43–63

Submitted: 2023-10-13 | Revised: 2023-10-13 | Accepted: 2023-10-13

Keywords: dnssec, security, dns, name server

Marek BATOR [1], Jakub PRZYSTASZ [2], Miłosz SERAFIN [2]*

SECURITY OF THE DNSSEC PROTOCOL AND ITS IMPACT ON ONLINE PRIVACY PROTECTION

Abstract

The theme of the research paper and its chapters revolve around the security of the DNSSEC protocol and its impact on online privacy. The study delves into the intricacies of the Domain Name System (DNS), exploring its fundamental workings, hierarchical architecture, and the role of the DNS root server, as well as the entities responsible for hosting the 13 DNS Root servers. The paper also examines various types of DNS attacks, including DNS Spoofing, Man-in-the-Middle attacks, DNS cache poisoning, and DNS hijacking, shedding light on the vulnerabilities within the DNS infrastructure. A significant portion of the research is dedicated to the description of DNSSEC (Domain Name System Security Extensions), emphasizing its importance and functionality within DNS zones. This includes an analysis of the mechanisms behind DNSSEC, such as RRSIG, Zone Signing Keys (ZSK), DNSKEY, and Key Signing Keys (KSK), along with discussions on trust establishment in specific DNS zones

1. University of Information Technology and Management, Poland
2. Rzeszow University of Technology, Department of Complex Systems, Poland

1.INTRODUCTION

In today's world, where more and more aspects of our lives are moving to the digital world, network security is becoming an extremely important issue. With the development of technology, there are new challenges related to user security and maintaining anonymity online. One of the key protocols that plays an important role in ensuring Internet security is DNSSEC (Domain Name System Security Extensions). It is an extension of the DNS protocol, which is responsible for converting a domain name into an IP address.

One of the main reasons for introducing the DNSSEC protocol is to enable authentication and data integrity in the domain name system. The protocol is primarily intended to protect users from cache poisoning attacks [1]. DNS servers taken over by hackers as a result of the attack, would be able to provide false information when a given DNS record is queried. This would enable criminals to redirect users to false websites. DNSSEC was developed to prevent similar situations from occurring by introducing a digital signature. It makes it possible to confirm the authenticity of the information sent by the DNS server, in order to minimize the risk of data interception and manipulation by unauthorized persons.

The impact of DNSSEC is very important in protecting online privacy, because with the rapidly increasing amount of data that is sent over the Internet every day, it is able to ensure online security for many users.

Aim of this paper

The main objective of this project will be to thoroughly investigate the security of the DNSSEC protocol and its impact on online privacy. The project will focus on the presentation, analysis and evaluation of DNSSEC security mechanisms, as well as the identification of potential threats and vulnerabilities, along with an assessment of the effectiveness of the security mechanisms used. The project will analyze two selected domains from this angle and compile the development of DNSSEC on a global scale.

2.EXPLAINING THE FUNCTIONALITY OF DNS

What is DNS?

Domain Name System is a key component of the Internet's infrastructure, converting domain names, such as "google.com," into corresponding IP addresses e.g. 216.58.215.110. This allows users to take advantage of easier-to-remember site names by typing them into a web browser to reach the appropriate servers via IP addresses and obtain the desired content [2]. The Internet Engineering Task Force has published in 1987 two documents describing the Domain Name System - RFC 1034 and RFC 1035. Eighteen years later, in 2005, IETF took a significant step forward in enhancing the security of the Domain Name System (DNS) by introducing three

documents: RFC 4033, RFC 4034, and RFC 4035. These documents collectively marked the inception of DNSSEC (DNS Security Extensions).

Why do users need a "chain" of servers in the DNS system? Wouldn't just a few servers suffice?

Using several DNS servers for the entire Internet is impossible. This is because the increase in traffic would cause DDoS attacks on the servers, making it impossible to handle real requests. In the case of DNS, critical systems on the Internet, such situations should be avoided. Another problem relates to scaling, which is solved by the hierarchical structure of DNS. If every DNS user could communicate with any random server to get a response to their query, servers would have to store a huge database of all DNS sets, which is impossible.

Key terms for DNS:

DNS Zone - a database containing records, for example *.google.com, stored on a disk at a specific location

Zone File (Zone File) - a "file" on a disk that stores a zone

Name Server (NS) - a DNS server that hosts one or more zones; it does this by storing as many zone files as there are zones

Authoritative - contains real/reference records. It is the authority for a specific domain and the only true source for a zone

Non-Authoritative/Cached - copies of records/zones are stored in a cache, such as RAM, to speed up queries.

Hierarchical DNS architecture

Thanks to the hierarchical architecture of DNS, users can divide the data stored on DNS servers and delegate the management of certain elements to specific organizations. This makes it easier to manage, and also distributes the load to DNS servers.

At the top of the DNS hierarchy are the DNS Root servers, of which there are 13 in the world, all operating from the original Root Zone. IANA manages the Root Zone and delegates its management to independent organizations, known as registries. The only special thing about the Root Zone is that it is a point that all DNS clients know and recognize. It plays a key role in the operation of DNS because it is authoritative for Top-Level Domain (TLD) domains. The Root Zone's only function is to point registries to top-level domains (TLDs). All name servers are authoritative for the domains or zones they host.

What is a DNS root server?

The administration of the Domain Name System (DNS) is organized in a hierarchical structure. It uses different managed areas called "zones," with the main zone located

at the very top of this hierarchy. Root servers are DNS name servers that operate in the main zone. These servers can directly respond to requests for records stored or held in the root zone, and can also direct other requests to the appropriate Top Level Domain (TLD) server. TLD servers are a group of DNS servers (which is a floor below the root servers in the DNS hierarchy), and are integral to resolving DNS queries [3].

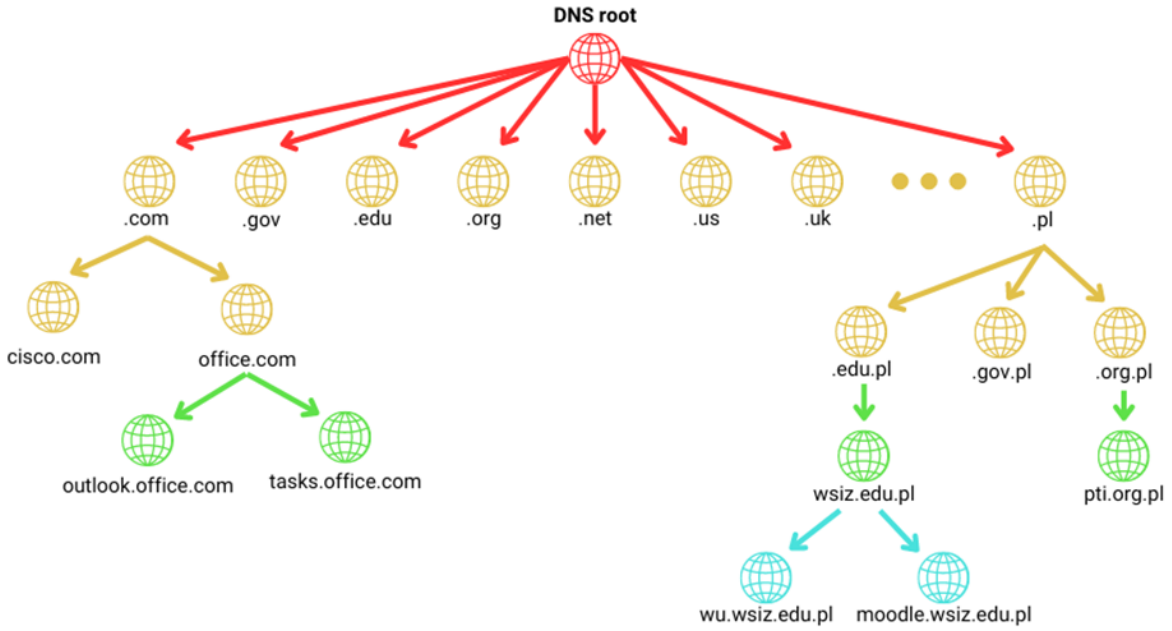

Fig. 2.1 Representation of the hierarchy in DNS

The Root zone contains relevant information about the second level of the DNS hierarchy, which includes the DNS servers of top-level domains (TLDs, Top Level Domain). There are two types of top-level domains (TLDs) in the DNS hierarchy, such as .com, and country-specific ones, such as .us or .pl. Servers at this level of the hierarchy are authoritative for their TLDs or second-level DNS zones. This means that the server at this level contains information about the TLD itself and indicates where to redirect the request to reach the third level of the hierarchy. The server for a .com TLD will be able to point to nameservers that host google.com and other .com DNS zones because they are authoritative. IANA delegates the management of these TLDs to other organizations, known as registries. For example, .pl domains are managed by the Polish unit of *NASK* [4].

List of organizations responsible for hosting 13 DNS Root servers [5]

- Verisign, Inc. (2x, United States of America)
- University of Southern California, Information Sciences Institute (ISI) (United States of America)

- Cogent Communications (United States of America)
- University of Maryland (United States of America)
- NASA Ames Research Center (United States of America)
- Internet Systems Consortium, Inc. (ISC) (United States of America)
- U.S. Department of Defense, Network Information Center (NIC) (United States of America)
- U.S. Army Research Laboratory (United States of America)
- Netnod Internet Exchange (Sweden)
- RIPE NCC (Réseaux IP Européens Network Coordination Centre) (Netherlands)
- ICANN (Internet Corporation for Assigned Names and Numbers) (United States of America)
- WIDE Project (Japan)

How does DNS work - "Walking the tree (chain)."

"Walking through the tree" shows the process of finding the right DNS zone for an authoritative answer to a DNS query [6]:

1. The person, device or service queries the hostname (eg. example.com).
   First, the local DNS cache is checked; DNS entries may be stored in a hosts file on the local device. If the hosts file does not contain an entry, some applications that process queries (such as browsers) may check their own cache.
2. The query is directed to the DNS resolver (DNS server running on the home network or ISP - Internet Service Provider).
3. DNS resolver checks its local cache for matching records.
4. If the DNS resolver does not have a result in its cache, the query is directed to the root zone through one of the root servers.
5. If the root server doesn't know the answer, it can at least help the resolver by checking its records and returning a response (IP address) that points to the name server for the .com TLD domain.
6. The resolver then asks one of the name servers for the .com TLD for www.example.com.
7. Assuming the example.com domain is registered, the .com zone contains entries (records) for example.com.
8. This moves the DNS resolver one step closer to answering - it gets the IP address of the example.com nameserver and queries the authoritative nameservers for www.example.com. This name server is authoritative for this

domain because it hosts the DNS zone and zone file for www.example.com that the .com TLD zone points to.

9. Authoritative name server for www.example.com returns the corresponding DNS record (containing IP address) to the resolver.
10. Resolver saves the result in a cache to improve performance for future queries on the same name.
11. Resolver returns the same result to the client from which the query originated.

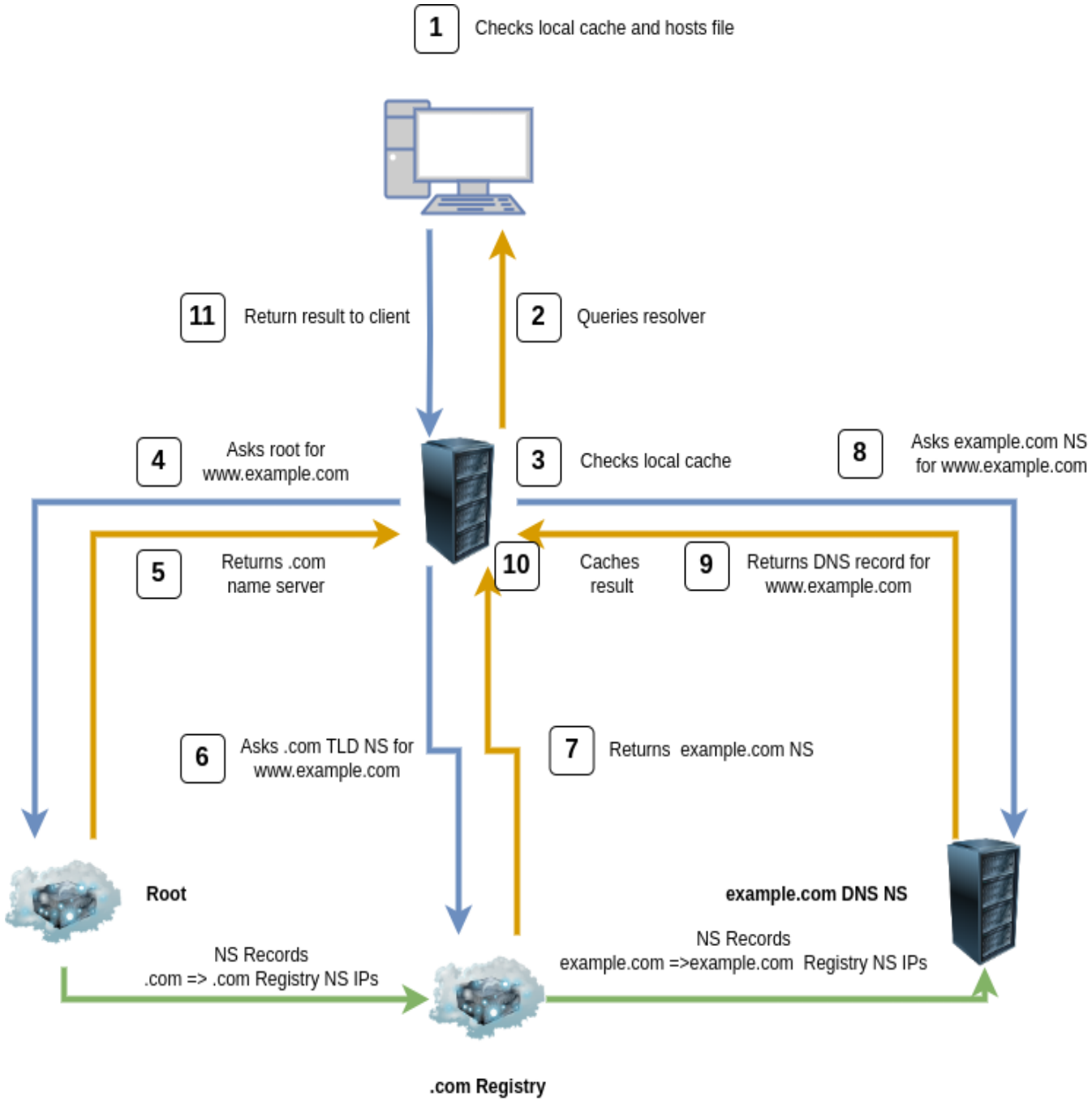

Fig. 2.2 The process of traversing the DNS tree to find the authoritative answer [6]. Source: own.

3.TYPES OF ATTACKS ON DNS

The Domain Name System (DNS) is the fundamental infrastructure of the Internet, responsible for converting human-readable domain names, such as google.com, into IP addresses that network devices can understand. However, DNS, like many other Internet systems, is vulnerable to a variety of attacks designed to disrupt its operation or use it for malicious purposes. Attacks on DNS are becoming more common and at the same time more sophisticated, posing a serious threat to network security. The purpose of such attacks may be to steal data, intercept network traffic, mislead users or prevent access to certain Internet resources. In this context, it is important to understand the different types of attacks that can target DNS.

DNS Spoofing is a technique for intercepting browser requests for a website and directing the user to another site. This can be done by changing the IP address of DNS servers or changing the IP address of the domain name server itself. A DNS spoofing attack involves an attacker impersonating a DNS server and sending responses to DNS queries that differ from those sent by a legitimate server. The attacker can send any response to the victim's query, including fake host IP addresses or other types of false information. This can be used to provide false information about services on the network, or to direct the user to a fake website designed to look like a genuine website [7].

How does DNS Spoofing work?

To demonstrate the process of DNS Spoofing, the following example scenario has been created. The scenario shows how attackers can redirect a victim to a fake website.

1. The client sends a DNS query to the DNS server, asking for the IP address for the domain name, for example google.com.
2. The attacker, being an intermediary between the client and the real DNS server, intercepts this query.
3. Instead of forwarding a query to a real DNS server, the attacker sends a fake DNS response with false information, such as providing a fake IP address for the *google.com* domain name. As a result, the user will be redirected to a fake website or other unauthorized infrastructure, where the attacker can try to steal sensitive information such as passwords or login credentials.

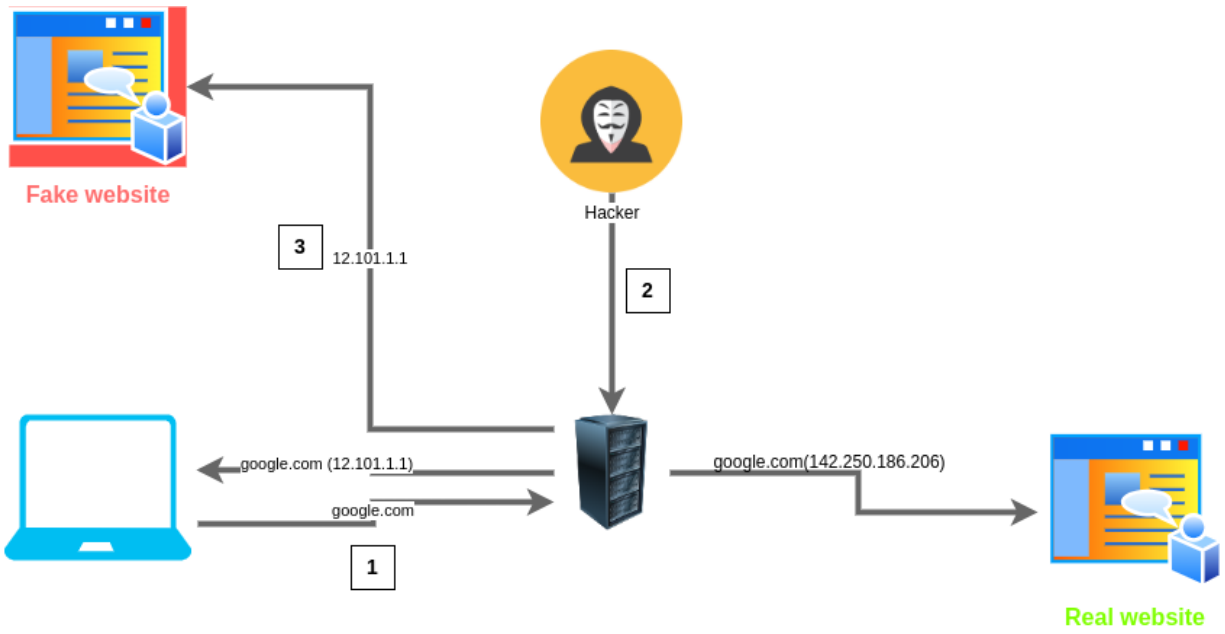

Fig. 3.1 DNS Spoofing attack process scenario. Source: own

DNS spoofing methods

Spoofing, which is one method of DNS attack, allows an attacker to trick a user's computer into using a fake DNS server. There are several ways to implement spoofing, but all are aimed at misleading the user and directing their requests to an unauthorized source.

Man-in-the-Middle attacks

One of the most common types of attacks in the DNS area is the so-called man-in-the-middle (MITM) attack. In such a scenario, an attacker, for example, intercepts communications between two SMTP servers that are used for email transmission. As a result, the attacker can intercept domain name resolution requests sent by clients. Then, the attacker routes these requests through his own network, instead of forwarding them to the real DNS server. As a result, the attacker is able to deliver fake responses pointing to any IP address, including even an address belonging to a fake phishing site [7].

DNS cache poisoning attacks

A given situation occurs when a network user tries to access a domain using a browser, the DNS resolver provides him with an IP address to locate the resources of that domain. This process, known as a DNS query, may involve the participation of more than one server. To speed up future queries, some DNS resolvers store DNS queries in a cache. This storage period is called "time-to-live" (TTL). In a DNS cache poisoning attack, the attacker inserts false IP address information into the DNS cache. This fake IP address belongs to a domain controlled by the attacker. When a network user tries to access the requested resource, they are redirected to the fake domain,

which can lead to the installation of malware. The attacker must act very quickly to time the attack before the "lifetime" of the data stored in the DNS cache expires. At the time of the attack, the client unknowingly uses this malicious data that has been deliberately inserted into its cache [8].

DNS hijacking attacks

DNS hijacking is an attack technique in which an attacker changes his IP address to make it look like an authoritative nameserver for a specific domain name. He can then send spoofed DNS responses to a client that requests information about that domain. As a result of this attack, the user is redirected to an IP address controlled by the attacker, instead of using the correct public DNS servers. This type of attack is often seen with customers who have not implemented proper security measures on their routers or firewalls. An attacker can exploit vulnerabilities in the configuration of these devices to take control of DNS traffic and manipulate redirects. This type of attack is particularly dangerous because the end user can be redirected to malicious websites that use the attack to steal personal information or install malware on the device [7].

4. HOW DNSSEC WORKS

DNSSEC (Domain Name System Security Extensions) is an extension to the Domain Name System (DNS) protocol that aims to enhance the security of the domain name system. It introduces elements of cryptography, such as asymmetric keys, to authenticate the data received when domain names are resolved to IP addresses[9]. The authentication process is based on a "chain of trust" that requires the signature of successive levels of domain zones, according to the hierarchical structure of DNS. To secure a domain name, it is necessary to sign the zone of a given domain and enter a special cryptographic hash from the public key to the parent zone where that domain is registered. The hash is signed with a private key and passed to the next parent zone. This process continues all the way to the Root zone, where there is a key called a "Trust Anchor" that is widely recognized as trusted. With DNSSEC, it is possible to verify that the data received is from the correct source and has not been altered during transmission. This provides greater security and protects users from man-in-the-middle attacks and data manipulation. [4]

DNNSEC - why is it important ?

DNSSEC (Domain Name System Security Extensions) is important for two reasons:

- allows you to authenticate the source of the data and make sure that this data is definitely coming from the zone,
- provides data integrity protection so users can be sure that this data has not been modified during transmission.

DNSSEC accomplishes its task by establishing a chain of trust between the root zone and DNS records in a cryptographically verifiable manner. DNSSEC adds to but does not replace DNS, so DNSSEC-enabled devices will receive results from both DNS and DNSSEC.

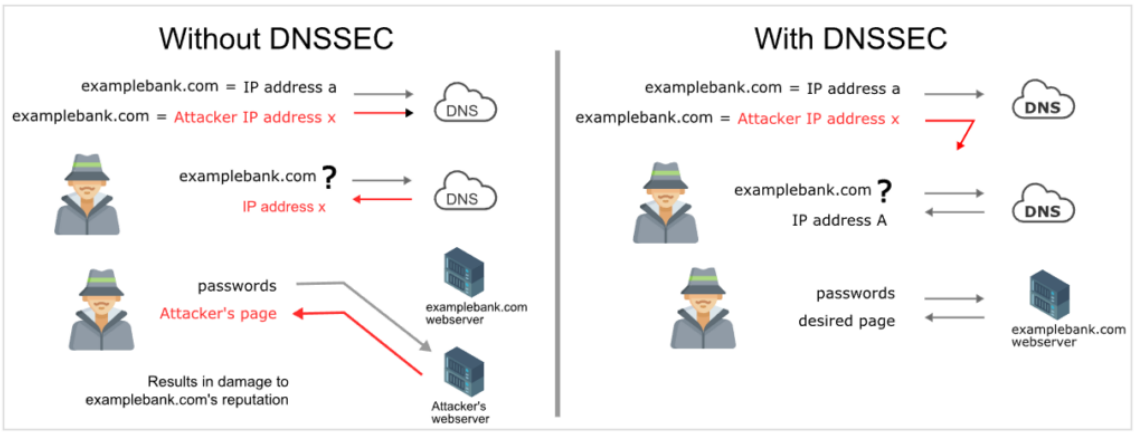

Fig. 4.1 Comparison of DNS and DNSSEC performance in a compromised environment scenario [10]

How does DNSSEC work within a zone?

RRSET (Resource Record Set) represents the grouping of all records of the same type into a record set (RRset). If we have 4 A records with the same label in the DNS zone, for example, wsiz.edu.pl, all these records will be combined into one set of A records. The same happens with AAAA and CNAME records. Now the question arises as to what A, AAAA and CNAME records are. In a nutshell, we can say that they are records used to redirect domain names. The A record is used for assigning IPv4 addresses, the AAAA record for IPv6 addresses, and the CNAME record for creating aliases for domains [11].

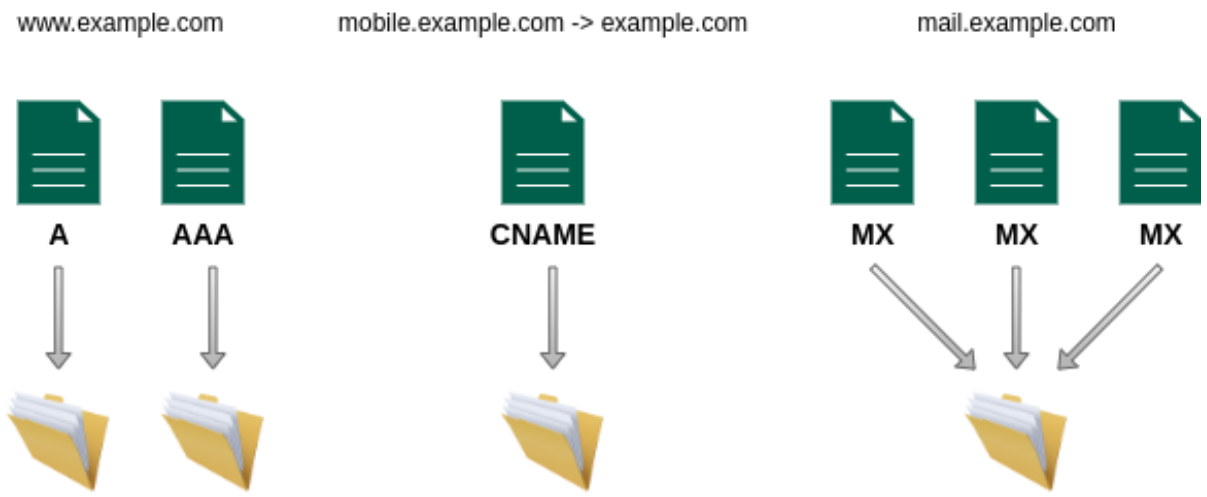

Fig. 4.2 Grouping records of the same type into a record set (RRSET). Source: own

RRSIG and zone signing keys (ZSK)

If the crook finds a way to make changes to the DNS or make the user believe the veracity of the response they give, email delivery can be redirected to the malicious server. RRSIG stores a digital signature of the record set (RRSET), created using a private and public key, a pair of cryptographic keys known as zone signing keys (ZSK). The private part of the ZSK key is protected - inaccessible to the public domain and is not stored in the zone, it is only used to create a digital signature for the record set. The signature can be stored alongside the textual record set in the zone, using the same name, but the record type is RRSIG. Any DNSSEC client will be able to see the record set (RRSET) and the corresponding RRSIG. If the record set is changed, the RRSIG must be re-generated. Any unauthorized changes without a corresponding change to the RRSIG will result in an invalid signature[12].

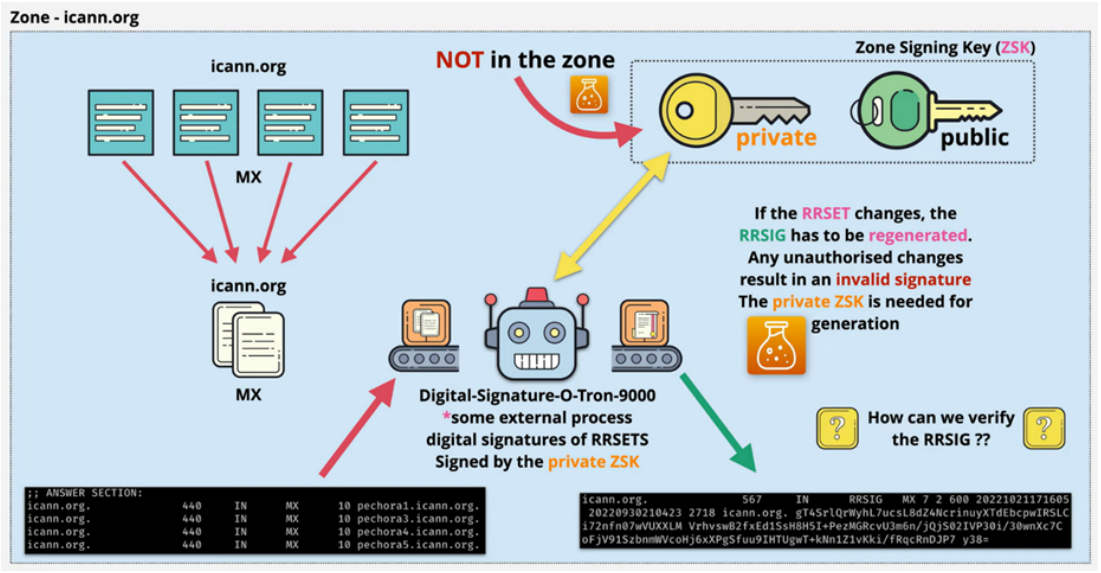

Fig. 4.3 RRSIG and zone signing keys (ZSK)[13]

DNSKEY

The purpose of a DNSKEY record is to store a cryptographic public key, which is used to validate DNSSEC digital signatures. When a DNSSEC-signed domain name is looked up, the DNS resolver uses the DNSKEY records to verify the authenticity and integrity of the DNS information it receives. In other words, DNSKEY records help in ensuring that the DNS data hasn't been tampered with during transmission. DNSKEY is stored as a separate zone record because public keys do not require the same level of protection as private keys. There is a flag value in DNSKEY that indicates the type of public key:

- The value of 256 bits indicates the Zone Signing Key (ZSK),
- The value of 257 bits indicates the Key Signing Key (KSK).

A DNSSEC resolver needs three types of records:

- RRSET,
- RRSIG,
- ZSK,

to verify data integrity. It checks the RRSIG signature to see if it matches the RRSET and if it was generated using the corresponding private part of the Zone Signing Key [14].

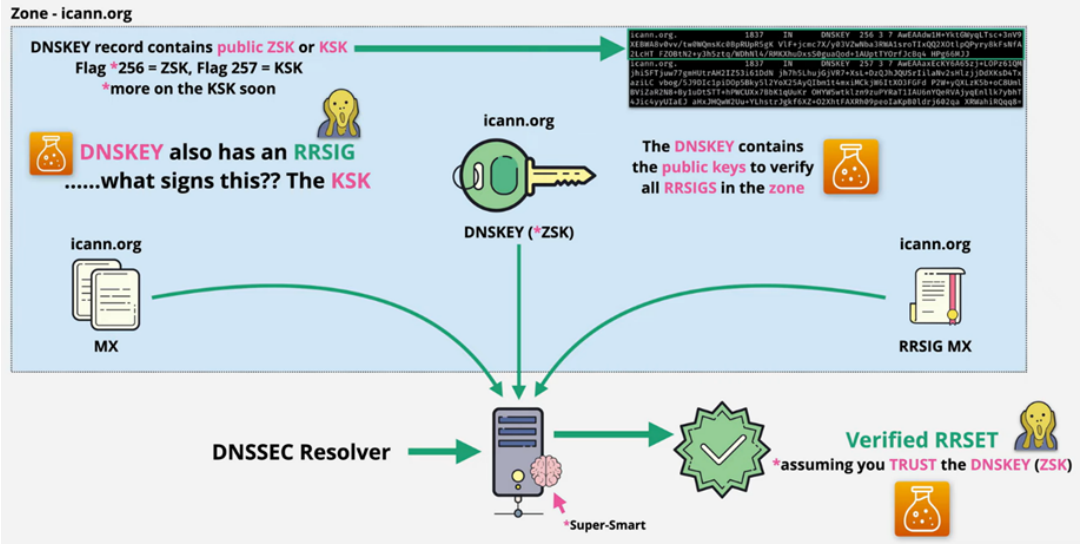

Fig. 4.4 DNSKEY - Zone signing keys (ZSK) in the data integrity verification process [14].

Key Signing Key - Where does it fit in and what mechanisms allow us to trust a zone?

Since the private ZSK key for the RRSET creates the corresponding RRSIG record, and the public part of the ZSK key is stored in the DNSKEY record (with a flag equal to 256), a potential attacker could theoretically insert a new public ZSK key, generate a fake RRSIG record using the private part of the ZSK key, change the RRSET and take control of the email for the domain. To verify that the DNSKEY record can be trusted

and has not been altered, it is necessary to have a corresponding signature, the RRSIG record. This signature is created using a key signing key (KSK). Without this chain of trust, Resolver would have to manually trust each zone in the DNS, which in turn would defeat the purpose of having a globally distributed system. To secure this, the icann.org zone is cryptographically linked to its parent zone, .org. This means that the .org zone has a way of explicitly stating that the icann.org zone can be trusted, just as in normal DNS [14].

To change the zone signing key record for the icann.org zone, it would be necessary to involve the .org parent zone. Although frequent key rotation is recommended, going up the chain for this purpose can become inefficient. This is where the KSK (Key Signing Key) comes into play. Both the KSK and the ZSK are stored inside the domain zone, but the public KSK can be referenced by the parent zone (.org in this case). This is what makes it possible to pass trust to the zone (assuming the parent zone can be trusted). In this way, a chain of trust is established between two different layers of DNS servers, and the integrity of DNS data is protected. How this inter-zone trust is established will be shown in the next section [14].

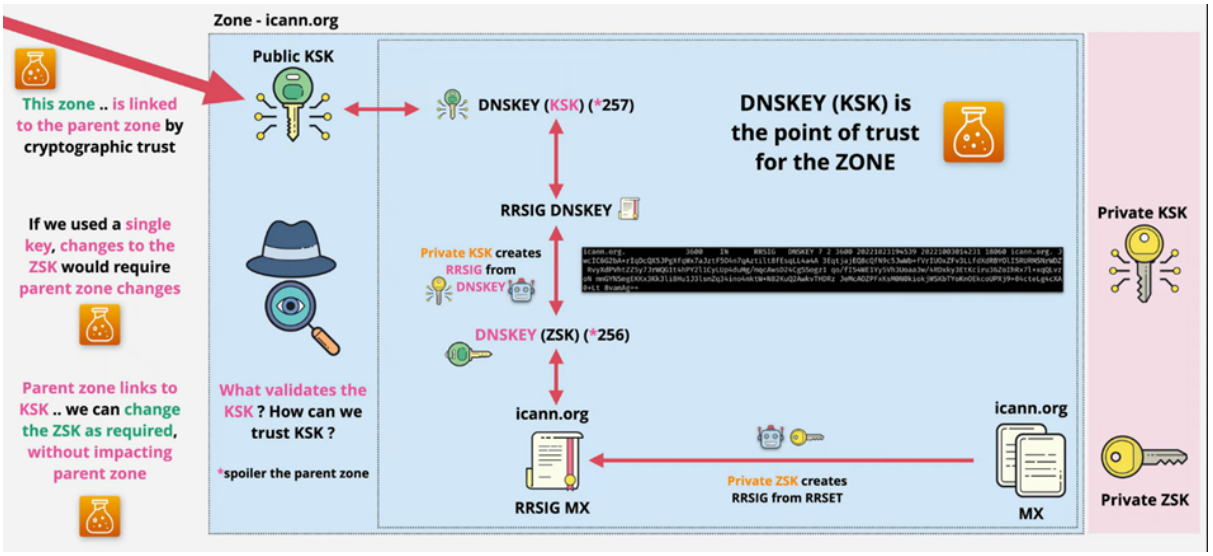

Fig. 4.5 DNSSEC mechanisms: key signing key (KSK) to provide a cross-zone chain of trust to protect the integrity of DNS data [14].

5. CONSTRUCTION OF THE DNSSEC TRUST CHAIN

In order to trust the Delegation Signer (DS) record, which connects the child zone to the parent zone, a verification process is required. The DS record is signed, just like other RRSET, which means it has a corresponding digital signature (RRSIG) in the parent zone. The verification process is an iterative process of traversing through DNS zones up to the root zone. At each step the DS Record is verified if it matches its parent's corresponding RRSIG. However, on a top level zone there is no parent node that can be used to verify records. Root's zone records are verified by keys signed by

several selected individuals from around the world during the Root Signing Ceremony [11].

Integrity of the trust chain

The ability to establish a trust relationship between a parent zone and a child zone is an integral part of DNSSEC extensions. If there is a break at any point in the chain of trust, we can no longer trust the requested records, because a potential attacker can alter the records and redirect to a fake IP address. Therefore, it is important that the entire chain of trust is intact and verified to guarantee the authenticity and integrity of the DNS data [11].

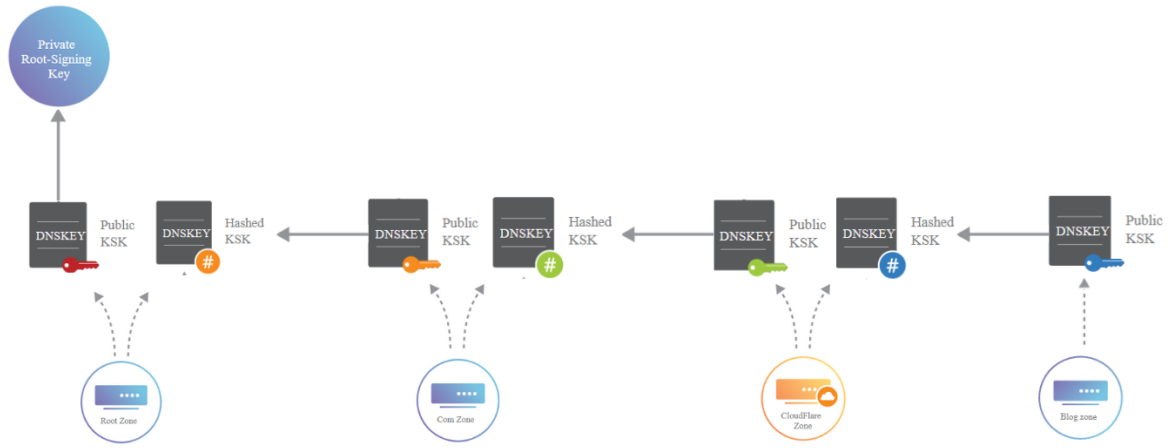

Fig. 5.1 DNSSEC Chain of trust [11]

Key Signing Ceremony

In order to solve the problem of the lack of a parent DS record, there is a special Root KSK Ceremony conducted four times annually. In this ceremony, a few selected people from different regions of the world meet and sign the root set of DNS Release Keys (DNSKEY) in a public and controlled manner. The ceremony creates an RRSIG record, which is used to verify the public KSK and ZSK of the root zone name server. In this case instead of relying on the parent's DS record, we rely on the security procedures for the private KSK. This leads to the belief that the KSK is adequately protected and that those responsible for handling it follow strict security procedures. As a result, the KSK can be considered a reliable tool used to confirm the authenticity of data in the DNSSEC system [11].

6. HOST VERIFICATION

How to verify whether a person or organization is using DNSSEC or regular DNS?

To verify whether a person or organization uses DNSSEC or plain DNS, it is necessary to check the DNS configuration for a domain. This can be done by using online tools such as *DNSSEC Analyzer*[14] to check whether a domain has DNSSEC configured. These tools analyze the domain's DNS records and tell us whether the appropriate DNSSEC records, such as RRSIG and DNSKEY, are present. In addition, with this tool we can check DS (Delegation Signer) records if we want to check whether a domain is protected by DNSSEC. DS records are stored by TLD Registries (e.g. for the ".com" domain by Verisign). DNSSEC analysis is a process that can be performed for any domain. Correct signing with cryptographic keys guarantees the authenticity, integrity and confidentiality of DNS data.

Analyzing DNSSEC problems for dnssec-deployment.org

.	✓ Found 3 DNSKEY records for . ✓ DS=20326/SHA-256 verifies DNSKEY=20326/SEP ✓ Found 1 RRSIGs over DNSKEY RRset ✓ RRSIG=20326 and DNSKEY=20326/SEP verifies the DNSKEY RRset
org	✓ Found 1 DS records for org in the . zone ✓ DS=26974/SHA-256 has algorithm RSASHA256 ✓ Found 1 RRSIGs over DS RRset ✓ RRSIG=60955 and DNSKEY=60955 verifies the DS RRset ✓ Found 3 DNSKEY records for org ✓ DS=26974/SHA-256 verifies DNSKEY=26974/SEP ✓ Found 1 RRSIGs over DNSKEY RRset ✓ RRSIG=26974 and DNSKEY=26974/SEP verifies the DNSKEY RRset
dnssec-deployment.org	✓ Found 1 DS records for dnssec-deployment.org in the org zone ✓ DS=2371/SHA-256 has algorithm ECDSAP256SHA256 ✓ Found 1 RRSIGs over DS RRset ✓ RRSIG=33369 and DNSKEY=33369 verifies the DS RRset ✓ Found 2 DNSKEY records for dnssec-deployment.org ✓ DS=2371/SHA-256 verifies DNSKEY=2371/SEP ✓ Found 1 RRSIGs over DNSKEY RRset ✓ RRSIG=2371 and DNSKEY=2371/SEP verifies the DNSKEY RRset ✓ aron.ns.cloudflare.com is authoritative for dnssec-deployment.org ✓ dnssec-deployment.org A RR has value 104.18.24.160 ✓ Found 1 RRSIGs over A RRset ✓ RRSIG=34505 and DNSKEY=34505 verifies the A RRset
dnssec-deployment.org	✓ yahir.ns.cloudflare.com is authoritative for dnssec-deployment.org ✓ dnssec-deployment.org A RR has value 104.18.24.160 ✓ Found 1 RRSIGs over A RRset ✓ RRSIG=34505 and DNSKEY=34505 verifies the A RRset

Fig. 6.1 Example result of DNSSEC signatures analysis [14].

The DNSSEC analysis for the *dnssec-deployment.org* domain (Fig. 6.1) shows that the domain has proper cryptographic key signing. By verifying DNSSEC records, it can be determined that the data in those records is protected and has not been modified during transmission.

Analyzing DNSSEC problems for onet.pl

.	✓ Found 3 DNSKEY records for . ✓ DS=20326/SHA-256 verifies DNSKEY=20326/SEP ✓ Found 1 RRSIGs over DNSKEY RRset ✓ RRSIG=20326 and DNSKEY=20326/SEP verifies the DNSKEY RRset
pl	✓ Found 1 DS records for pl in the . zone ✓ DS=59899/SHA-256 has algorithm RSASHA256 ✓ Found 1 RRSIGs over DS RRset ✓ RRSIG=60955 and DNSKEY=60955 verifies the DS RRset ✓ Found 3 DNSKEY records for pl ✓ DS=59899/SHA-256 verifies DNSKEY=59899/SEP ✓ Found 1 RRSIGs over DNSKEY RRset ✓ RRSIG=59899 and DNSKEY=59899/SEP verifies the DNSKEY RRset
onet.pl	✗ No DS records found for onet.pl in the pl zone ✗ No DNSKEY records found ✓ ns6.ringpublishing.net is authoritative for onet.pl ✓ onet.pl A RR has value 75.2.92.173 ✗ No RRSIGs found
onet.pl	✓ ns5.ringpublishing.net is authoritative for onet.pl ✓ onet.pl A RR has value 99.83.207.202 ✗ No RRSIGs found
onet.pl	✓ ns8.ringpublishing.net is authoritative for onet.pl ✓ onet.pl A RR has value 99.83.207.202 ✗ No RRSIGs found
onet.pl	✓ ns7.ringpublishing.net is authoritative for onet.pl ✓ onet.pl A RR has value 75.2.92.173 ✗ No RRSIGs found

Fig. 6.2 Example result of DNSSEC signatures analysis of domain with invalid records [14].

Performing a DNSSEC analysis for the *onet.pl* domain reveals that this particular domain does not use DNSSEC security. This is evident from the lack of DS records for the *onet.pl* domain in the .pl zone. The DS (Delegation Signer) record is what is used to establish trust between a parent zone domain (such as .pl) and a subordinate level domain (such as *onet.pl*). The DS record contains the digital signature of the subordinate domain's Key Signing Key, which is verified by the parent zone domain. The absence of DS records indicates that there is no DNSSEC implementation for that domain. This means that data transmitted by the *onet.pl* domain is not digitally signed, potentially exposing users to threats.

6. ASSESSING THE DEVELOPMENT OF DNSSEC ON A GLOBAL SCALE

The website *stats.labs.apnic.net* is a valuable tool for assessing the development of DNSSEC on a global scale. It is a statistical tool made by APNIC (Asia-Pacific Network Information Centre), which provides data related to DNS and DNSSEC. Thanks to this site, we can find out the current status of DNSSEC deployment and see how developed the technology is worldwide. On this page, the DNSSEC section presents statistics related to the implementation and adoption of DNSSEC. It shows the number of domains secured with DNSSEC, the number of DNS servers performing DNSSEC validation and other related statistics. The statistics on the map below show what percentage of the overall DNS queries in a country are resolved using DNSSEC.

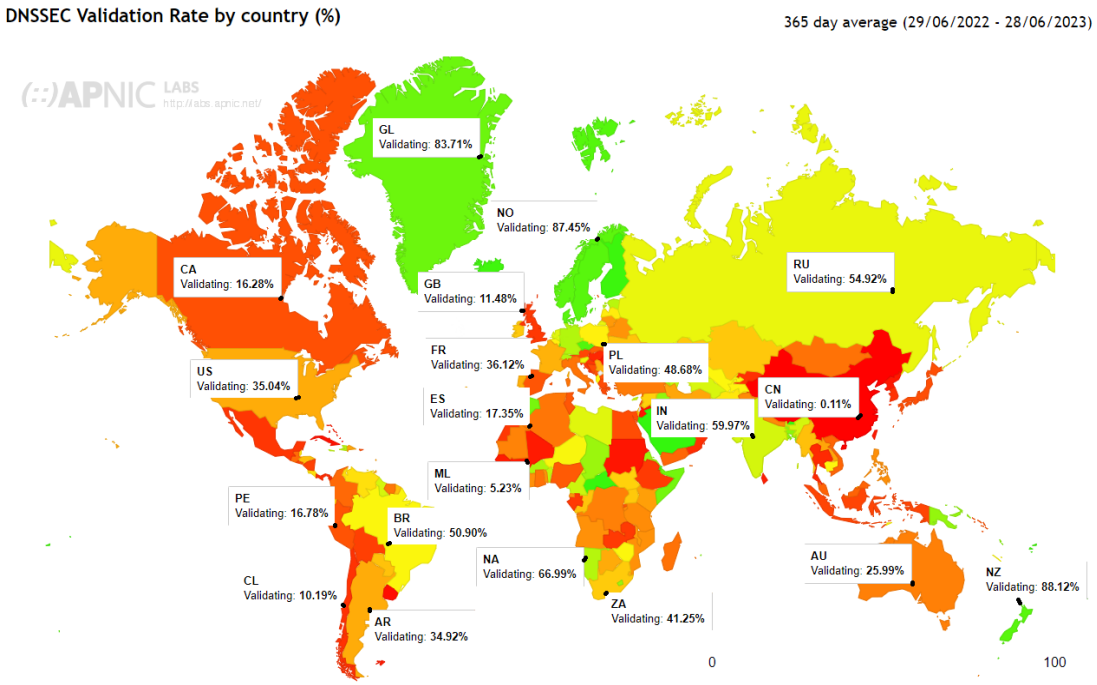

Fig. 7.1 Percentage representation of overall DNS queries per country over the last year resolved using DNSSEC[15].

Code	Region	DNSSEC Validates	Partial Validates	Total Validates	Samples	Weight	Weighted Samples
XA	World	30.72%	9.45%	40.17%	4,254,670,019	1	4,254,670,019
XE	Europe	41.37%	10.26%	51.64%	680,024,859	0.91	616,714,419
XF	Oceania	40.77%	6.31%	47.08%	38,638,864	0.82	31,518,297
XC	Americas	32.69%	7.55%	40.24%	892,412,066	0.85	760,279,749
XB	Africa	29.05%	15.88%	44.93%	289,999,173	1.5	436,129,204
XD	Asia	27.36%	8.71%	36.07%	2,353,591,436	1.02	2,409,934,446
XG	Unclassified	1.30%	0.35%	1.65%	1,270,855	0.07	92,132

Tab. 7.1 Percentage of overall DNS queries using DNSSEC in the last year 2022/23 that focus on a specific region [15].

7. CONCLUSION

The security of the DNSSEC protocol plays an important role in ensuring security on the Internet, especially in the context of protecting user privacy. DNSSEC was introduced to prevent cache poisoning attacks that could lead to users being redirected to fake websites. The protocol enables authentication and data integrity assurance in the domain name system by introducing digital signatures. DNSSEC has an important impact on online privacy, especially with the increasing amount of data being sent over the Internet. With this protocol, users can be assured that the information sent by the DNS server is authentic and has not been altered by unauthorized parties. DNSSEC provides robust security, which increases trust in the DNS infrastructure and helps combat various forms of cyber threats. Equally important is the chain of trust in DNSSEC, which plays a key role in ensuring the security and authenticity of DNS data.

The process of verifying DS records, building the chain of trust, the root signing ceremony and the integrity of the chain are all integral to DNSSEC. Through the use of cryptographic techniques and digital signatures, the integrity of the data and reliability of the DNS system is strengthened. Despite its validity, DNSSEC is not yet widely used. This may be evidenced by the fact that the number of DNSSEC validation processes on DNS servers has been relatively low in many regions around the world over the past year. This could be due to a variety of factors, such as lack of awareness or implementation complexity. However, as awareness grows and infrastructure is developed, it can be expected that DNSSEC will play an increasingly important role in ensuring security and privacy on the network. The implementation of DNSSEC is particularly valuable for websites that store sensitive data or handle financial transactions.

Author Contributions

All authors declare equal contribution to this research paper.

Conflicts of Interest

☑ *The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.*

☐ *The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:*

……

REFERENCES

[1] Shulman, H., & Waidner, M. (2014). DNSSEC for cyber forensics. EURASIP Journal on Information Security, 16(2014). https://doi.org/10.1186/s13635-014-0016-2
[2] Lutkevich, B. (n.d.). Domain Name System (DNS). TechTarget. https://www.techtarget.com/searchnetworking/definition/domain-name-system
[3] Cloudflare. (n.d.). What is a DNS Root Server? Cloudflare. https://www.cloudflare.com/learning/dns/glossary/dns-root-server/
[4] NASK. (2020). Polityka DNSSEC dla domeny .PL. [PDF]. https://www.dns.pl/formularze/DNSSEC_polityka_PL.pdf
[5] Internet Assigned Numbers Authority. (2022, October 10). Root Servers. https://www.iana.org/domains/root/servers
[6] (2005). Domain Name System (DNS). In: Understanding IPv6. Springer, Boston, MA. https://doi.org/10.1007/0-387-25614-8_9
[7] Rudra, A. (2022, May 26). What is DNS Spoofing? PowerDMARC. https://powerdmarc.com/pl/what-is-dns-spoofing/
[8] Rudra, A. (2022, August 19). What is DNS Cache Poisoning Attack? PowerDMARC. https://powerdmarc.com/pl/what-is-dns-cache-poisoning-attack/
[9] Andziński, M. (2012). Bezpieczeństwo z DNSSEC. IT w administracji, nr 5. https://www.dns.pl/formularze/Bezpieczenstwo_z_DNSSEC.pdf
[10] LearnCantrill. (2022, October 10). DNS 101 Miniseries - #5 - Why do we need DNSSEC [Video file]. https://youtu.be/thAUzOnUvP4
[11] Cloudflare. (n.d.). How DNSSEC Works. https://www.cloudflare.com/dns/dnssec/how-dnssec-works/
[12] Aitchison, R. (2011). DNSSEC. In: Pro DNS and BIND 10. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4302-3049-6_11
[13] LearnCantrill. (2022, October 10). DNS 101 Miniseries - #6 - How DNSSEC Works within a Zone [Video file]. https://youtu.be/4qlIim15xwM
[14] Verisign Labs. (n.d.). DNSSEC Debugger. https://dnssec-debugger.verisignlabs.com
[15] APNIC Labs. (n.d.). DNSSEC Statistics. https://stats.labs.apnic.net/dnssec