International Journal of Emerging Research in Engineering and Technology
Pearl Blue Research Group| Volume 3, Issue 2, 89-98, 2022
ISSN: 3050-922X | https://doi.org/10.63282/3050-922X.IJERET-V3I2P110

Original Article

Next-Gen DNS and Security Challenges in IoT Ecosystems

Sunil Anasuri
Independent Researcher, USA.

***Abstract** - The meteoric growth of Internet of Things (IoT) has transformed the networking sector by connecting billions of heterogeneous devices to the Net. The benefits created by this digital revolution are unprecedented in relation to automation, monitoring, and decision making. Nevertheless, it also brings significant security flaws as well as questions of scalability, especially across legacy Domain Name System (DNS) infrastructure. DNS is a very important Internet component, which translates human-readable names into IP addresses that can be understood by machine. As the number of IoT devices grows, the question of supporting performance, security, and reliability of traditional DNS becomes difficult. DNS-based attack supports such as cache poisoning, DDoS, DNS tunnelling, and spoofing are easy to launch against IoT devices, since IoT devices often have weak security capabilities, at least by comparison with a PC or other managed device. Moreover, these problems are complicated by the fact that, unlike with popular protocols (HTTP, HTTPS, FTP, TCP), the tools used in IoT are not standardized, thus compounding the problem of adaptation to DNS. The paper explores the future forms of DNS, improvements on security protocols like DNSSEC and DNS over HTTPS (DoH) and DNS over TLS (DoT), and decentralized one like blockchain based DNS. It nulls their contributions in alleviating the security challenges that are IoT-specific, low latency, and facilitating the ability of IoT networks to handle massive scales. The paper encompasses an elaborate literature review, test methodology to assay the resilience of security, and performance tracking in the emulated IoT conditions. Lastly, it suggests an inclusive security mechanism that consists of edge computing, AI-based pattern detection, and safe DNS setup to facilitate future IoT infrastructure.*

***Keywords** - DNS, IoT Security, DNSSEC, DoH, DoT, DDoS, Blockchain DNS, Edge Computing, Anomaly Detection, IoT Scalability.*

1. Introduction

1.1. Overview of IoT Ecosystem

Internet of Things (IoT) is the large and fast growing array of the interconnected devices, sensors and machines that gather and share information, using data to support smart decision-making and automation in a variety of disciplines. These tools include wearable monitoring health devices and industrial robots, smart home appliances, and driverless cars. [1-3] The IoT ecosystem can be used to support better operational efficiencies, predictive maintenance and greater user experiences in the context of healthcare, transportation, agriculture, energy and smart cities by leveraging real-time streams of data. Since the connected devices spread ever more, the amount of data produced and transmitted through the network is exponentially increasing. Gartner (2021) also found that more than 25 billion IoT devices were expected by 2022, which proves how large and prominent IoT is on the digital scene. This unlimited expansion causes unbelievable pressure on the conventional Internet infrastructure, such as addressing schemes, communication techniques and information security systems. More specifically, essential parts of the infrastructure like Domain Name System (DNS), upon which the interaction between devices operating online is based, are increasingly challenged by new demands of scalability, latency, and exposure to cyberattacks. With the spread of IoT to all spheres of daily routine, there is an increased need to upgrade and expand the fundamental elements of the Internet to account to a robust, safe and optimized interaction of billions of interconnected points.

1.2. Importance of DNS in IoT

Domain Name System (DNS) is a core aspect in ensuring successful communication amid IoT devices and their services. The value of stable, secure and effective DNS resolution is more critical as the IoT ecosystem becomes more intricate and intertwined. Different subsections will split major parts of DNS in regard to IoT:

- **Identifying the Device and Address Resolution:** The IoTs tend to be on IP-based networks and thus a means of translating human readable names (e.g. sensor1.local) to IP addresses is needed. DNS plays this role of allowing devices to learn and communicate with other nodes in dynamic and distributed systems. In large scale areas of IoT, like smart cities or smart factories, it will make administration much easier and the devices much more interoperable to manage device identities using DNS.
- **Dynamic Upgrades and scalability:** The IoT networks often undergo modifications usually as a result of mobility of the IoT devices, cyclic powering, or reconfiguration. DNS enables dynamic updates and service discovery protocols (such as APNS and DNS-SD), and the subsequent capability as a protocol to enable real-time device registration and information updates. The flexibility is very important in scenarios where there are thousands or even millions of endpoints and it is not possible to configure manually.

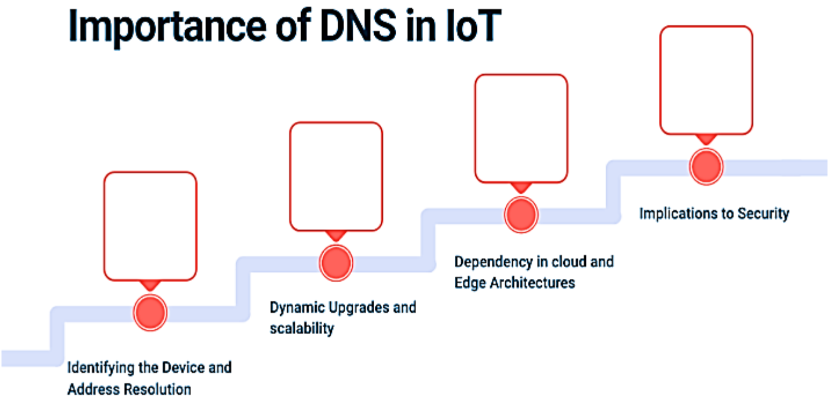
Importance of DNS in IoT
Identifying the Device and Address Resolution
Dynamic Upgrades and scalability
Dependency in cloud and Edge Architectures
Implications to Security

Fig 1: Importance of DNS in IoT

- **Dependency in cloud and Edge Architectures:** Most of IoT systems use cloud services or edge computing nodes to collect data and do processing, analytics, and storage. DNS is the most important connection that would allow IoT devices to talk to cloud APIs, MQTT brokers, edge gateways, or offsite databases. A system performance, transmission of information, and responsiveness of devices can be directly affected by any sort of delay or failure of DNS resolution.
- **Implications to Security:** Conventional DNS is also inherently insecure and susceptible to a number of attacks including spoofing, cache poisoning, and surveillance; each of which can be particularly harmful in an IoT setting since devices may run unattended and process sensitive information. Lack of secure DNS resolution can send traffic to the wrong place, steal data or hack IoT devices. Thus, it is necessary to adopt secure DNS system such as DNSSEC, DoT or DoH to ensure confidence and viability in IoT interactions.

1.3. DNS Vulnerabilities in IoT Context

There is a distinct set of scale/heterogeneity/resource constraints in IoT environments; the attacks on DNS are especially easy to launch there. [4,5] The following are some of the most crippling DNS vulnerabilities that affect IoT deployments:

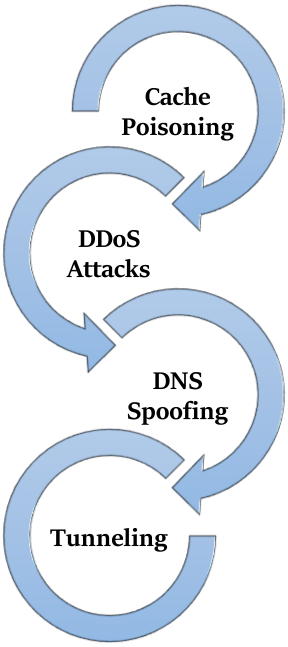

Fig 2: DNS Vulnerabilities in IoT Context

- **Cache Poisoning:** Cache poisoning is where an attacker inserts the fake or malicious DNS records in the resolver cache. In the environment of IoT, it might lead to the situation when devices are redirected to malicious servers silently. The IoT devices that are not based on many machines and still do not check the DNS responses may connect to systems controlled by the attacker to intercept the data, gain unauthorized access or control the device.

- **DDoS Attacks:** When the IoT devices are infected they become a segment of a botnet that is utilized to create a Distributed Denial of Service (DDoS) attack on a massive level. A typical way is DNS amplification, which involves attackers issuing spoofed DNS requests to inundate a target server with big volumes of traffic. The numerous IoT devices are vulnerable to hijacking since many of them have no in-built security features; hence, massively control supplies could provide a large amount of traffic that can cripple DNS infrastructure and cause complete disruption of essential services.
- **DNS Spoofing:** DNS spoofing is the depth of acquisition of the DNS replies in an approach that causes a device to carry out a domain name to a bogus IP address. This may be especially harmful in an IoT setting, where the devices tend to be non-authenticated and may be deceived rather easy. Traffic to rogue domains can be redirected to leak sensitive information or to send malicious firmware or assume control over the device without the user being aware.
- **Tunneling:** DNS tunnelling is another obfuscated method of encapsulation of arbitrary data in DNS queries and responses. With this approach, criminals can use non-standardized port to circumvent firewalls and create covert communication links between IoT devices they have infected and command-and-control servers located elsewhere. Because DNS transmissions are usually permitted across the network boundaries unfettered, it can serve as a transmission tube to drive data out in a covert, consistent manner or drive rogue commands to IoT nodes.

2. Literature Survey

2.1. Traditional DNS Infrastructure

Domain Name System (DNS) is the infrastructure of internet navigation and is in the form of a hierarchy that translates the domain names, which are human-readable, into Internet Protocol (IP) addresses. [6-9] The elements of this hierarchy are root servers the first in the top followed by top-level domains (TLDs), authoritative name servers and recursive resolvers which enable the process of looking up to be accomplished by end-users. Nevertheless, DNS protocol was created in the earlier days of the internet time when the main concern was not the level of security. Consequently, there are numerous threats it can be prone to such as cache poisoning, spoofing, and man-in-the-middle. The vulnerability of the traditional DNS mechanisms to lack encryption or authentication has contributed to it being a target of malicious actors particularly when the network environment is prone to black-hat attackers.

2.2. IoT-DNS Interaction Security Weaknesses

The combination of interconnection with Internet of Things (IoT) based devices and DNS systems presents some other vulnerabilities caused by the limited resource profile of the devices as well as the installation in an untrusted environment. The works of Liu et al. (2019) and Bandyopadhyay et al. (2020) focus on the fact that IoT devices are often working on open or edge networks, and the DNS queries are transferred in the plain text form. This subjects key metadata to allowing eavesdropping, DNS spoofing and redirection attacks. Moreover, few IoT gadgets have sufficient computational resources or configurability of firmware to implement increased DNS security standards, further complicating the risk environment and putting endpoints at risk of attack.

2.3. DNSSEC and Shortcomings

To resolve the validity and integrity of DNS responses, DNS Security Extensions (DNSSEC) were proposed using the public-key cryptography and digital signatures. When used effectively, DNSSEC will establish that the answer in the DNS has not been modified along the path and that it did not come as a result of an illegitimate process. Nevertheless, the adoption of DNSSEC has been slow, because of the complexity of its operations, Backward compatibility, and other issues such as large sizes of responses attributed to cryptographic payloads. Such larger response packets may be a challenge in low bandwidth or high latency networks and are particularly inappropriate on many IoT devices which are commonly limited in memory and processing power. Therefore, the DNSSEC can be more secure, but its adoption in the sphere of the IoT is questionable.

2.4. DNS over HTTPS and TLS

Newer protocols, such as DNS over TCP (DoH) and DNS over TLS (DoT), have been developed to resolve privacy and data integrity concerns; these would encrypt traffic between the clients and resolvers of a DNS system. Such protocols are used to avoid having third party observers interfering with or altering DNS requests. The study conducted by Appelbaum et al. (2018) states that DoH and DoT help significantly to reduce the risk of DNS spoofing and surveillance. They bundle DNS requests as HTTPS/TLS sessions, putting the information within the typical practices of encryption. Nonetheless, although they increase security and privacy, these protocols may add extra overheads and may not work with some network monitoring tools, making them difficult to be deployed in challengingly constrained IoT landscapes.

2.5. Blockchain-Based DNS

Blockchain DNS is also being suggested as an alternative to the traditional form based on its immutability, censorship resistance, and decentralized ability. Efforts such as Namecoin and the Ethereum Name Service (ENS) can be used to show how peer-to-peer DNS systems might be able to supplant centralized registries with distributed ledgers. These systems will offer greater transparency and tamper resistance or take-over by any single stakeholder. However, when used in relation to the IoT context, blockchain-based DNS has severe weaknesses, especially regarding scalability, latency, and resource consumption. Due to the decentralized nature of the underlying consensus mechanics of blockchain, they generally lead to increased query response times and computation load that are not concurring with real-time and lightweight requirements of IoT environments.

2.6. Edge Computing for DNS Query Handling

Edge computing offers an interesting method of enhancing the performance and security of the DNS in an IoT network by making DNS resolution less distant to the devices. Placing local DNS resolvers (edge servers) improves the situation greatly, the performance is very fast, and the DNS traffic may be confined to geographically local networks, which results in greatly minimizing access to external threats. A hybrid architecture, which combines edge computing with the existing hierarchies of DNS, is discussed in research by Rahman et al. (2021) where it was combined to optimize performance and resilience. The arrangement also supports localized security policies depending on an IoT environment. The edge-based solution is promising in balancing between security and performance especially to time-sensitive, or bandwidth-limited IoT applications.

3. Methodology

3.1. Research Framework

The proposed methodology is implemented in 4 primary stages the combined [10-13] impact of which determines ensuring the safety of DNS in the environment of IoT:

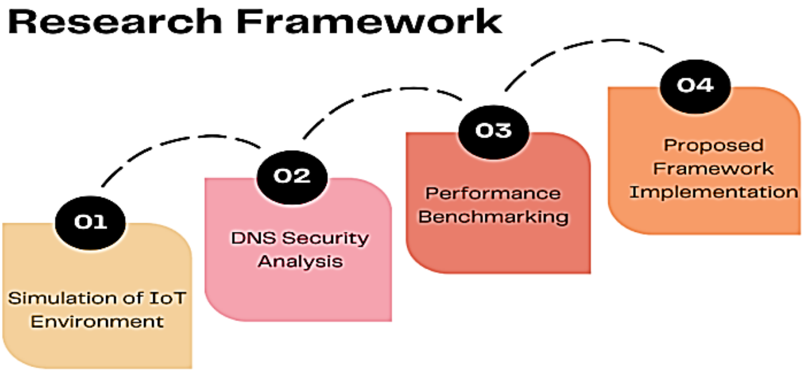

Fig 3: Research Framework

- **Simulation of IoT Environment:** The phase relays on the creation of a realistic simulation of an IoT network using the instrument like NS-3, Cooja, or a virtual version with the Raspberry Pi or ESP32 device. The virtualized environment presents the common communication patterns and limitations (such as low-power devices, limited processing and heterogeneous protocols) of IoT. The vision is to have in place a controlled environment in which the DNS queries initiated by computers connected to the Internet of Things may be monitored, experimented and analyzed in different security environments.
- **DNS Security Analysis:** During this stage, the DNS behavior of the IoT devices is examined to know the vulnerability surrounded by DNS spoofing, cache poisoning, and leakage of data. The research is based on the comparison of the standard DNS, DNSSEC, DoH, and DoT performance used with various threat models. In this analysis, it will also be observed that the insecure DNS interactions open up the IoT devices to an attack especially in edge or open networks.
- **Performance Benchmarking:** In this case, the effectiveness of different security mechanisms applied within the DNS environment (i.e. DNSSEC, DoH, DoT, and blockchain-based DNS) is tested in respect to latency, throughput, memory utilization, and processor load on IoT nodes with limited computing resources. Benchmarking comparison assists in determining the trade-offs between security and the efficiency, which are vital in the determination of the methods that can be practiced in the situations where resources are scarce.
- **Proposed Framework Implementation:** On the basis of the earlier phases, a safe and optimized system of DNS resolution is worked out and carried out. This architecture can incorporate edge computing, schema-less encryption or decenternalized name resolution to suit the needs of an IoT deployment. Simulation or live testing is used to verify the implementation in order to gauge improvements in both the security and performance.

3.2. Experimental Setup

The experimental design contains the likely implementation of the IoT environment in which the security measures and performance indicators of DNS can be tested using controlled factors. The environment consists of physical testbed, network, and software.

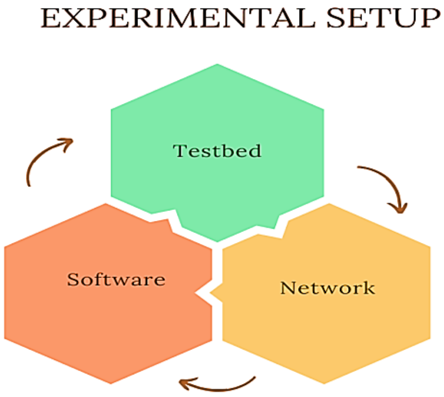

Fig 4: Experimental Setup

- **Testbed:** The testbed is composed of an integration of limited resource-IoT devices and multipurpose nodes. Common devices are usually Raspberry Pi devices and ESP32 microcontrollers and low-power sensors to replace the IoT smart home or industrial scenarios. Such devices are set up to carry out routine DNS queries and identify with local or cloud DNS resolvers. Other nodes are also allocated to play a rogue role of behavior (e.g. DNS spoofing) to check the effectiveness of the security protocols.
- **Network:** The topology of the network is set up as a star or mesh topology to reflect typical implementations of the IoT. Wireless communication in a local environment is simulated by the use of Wi-FI router or edge gateway to implement local area network (LAN). The CN nodes in edge computing may also participate in the testbed to do local DNS resolution. Traffic between IOT nodes and the DNS servers are observed with tools such as Wireshark and tcpdump in order to capture traffic packets of DNS and analyze behavior of protocols on different security application (e.g., plaintext DNS versus DoH).
- **Software:** On either edge nodes, the software stack consists of operating systems such as Raspbian or Ubuntu plus light-weight DNS clients (e.g. dig, nslookup, Unbound) or the devices. A DNSSEC, DNS over HTTPS (with cloudflared or dnscrypt-proxy) and DNS over TLS protocol implementations are installed to be able to compare between them. Simulation and logging such as NS-3, Wireshark and Python scripts are implemented to automate, measure the latency and also monitoring performance metrics. DNS alternatives based on blockchain such as Namecoin or Ethereum Name Service (through testnets) are also implemented to test them in such situations.

3.3. Attack Models

The following attack models are simulated to assess the resilience and the performance of DNS security mechanism in the IoT. [14-16] both models are specific to different areas of DNS actions and can be perceived as typical threats to the IoT networks in real life.

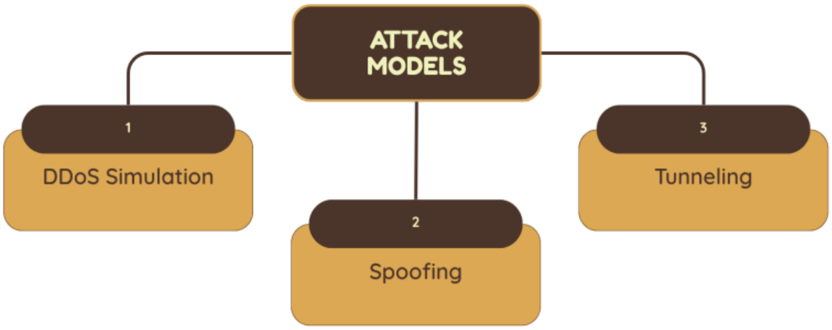

Fig 5: Attack Models

- **DDoS Simulation:** This attack type can be seen as a simulation of the Distributed Denial of Service (DDoS) one where recursive queries are initiated against DNS servers in large quantities. Several nodes of the IoT will be set to form a constant and randomised DNS requests and overload the capacity of the resolver to reply. This test measures the way in which DNS infrastructures (particularly those based at the edge and those that are secure) respond to over load conditions and whether disruption to a response occurs causing a slowing of the response time or system downtimes.

- **Spoofing:** Address Resolution Protocol(ARP) poisoning is used to perform DNS spoofing in this case and their aim is to redirect the traffic to access malicious DNS servers instead of accessing genuine DNS servers. With this type of attack, an attacker changes the ARP tables on the local network, claiming to be a trusted DNS server, but not actually being trusted, responds with modified DNS answers, which could direct IoT devices to attacker-controlled IP addresses. The given attack model can be utilized to assess the performance of DNSSEC, DoH, and DoT in regard to how they can identify and block spoofed DNS answers.
- **Tunneling:** The iodine program is used to simulate DNS tunneling, by embedding HTTP or arbitrary TCP data inside DNS requests in order to circumvent firewall limitations or steal data. This technique shows the way through which attackers are able to transfer data or instructions that are stealthy by using seemingly regular DNS traffic. The model assesses the effectiveness of network monitoring tools and DNS security protocols in detecting and preventing such covert channels particularly in settings where devices are limited and where there is low intrusion detection abilities.

3.4. DNS Security Implementation

In order to mitigate the DNS interactions in the simulated IoT environment, various DNS security devices and techniques were configured and installed with the use of standard tools and protocols that are currently popular. It was implemented through BIND9 DNS server which was configured with Zone Signing Keys (ZSK) and Key Signing Keys (KSK) to provide hierarchy trust and integrity of DNS information. [17-20] Zone files had been signed digitally and recursive resolvers had validation on to enforce authenticity of the received DNS responses. To ensure that no unsigned, or modified answers could be returned as response, and that they could be identified and repelled by the resolver, this method was salted against spoofing-attacks. DNS-over-TLS (DoT) was configured in conjunction with Stubby resolver, a low-resource resolver capable of communicating with upstream resolvers via a well-established and securely contiguous TLS connection. Stubby was deployed on IoT gateway hardware and set to talk to only known secure DNS resolvers which support DoT like Quad9 and Cloudflare. Connection security was provided by TLS whereby DNS query traversed the network securely, not eavesdropping or tampered by network attackers especially in environments where the network was not trusted or in case of ARP poisoning. In the case of DNS-over-HTTPS (DoH) the testbed employed the DoH endpoint of Cloudflare, along with DNSCrypt-Proxy software. This configuration routed DNS traffic with HTTPS encryption, concealing it as ordinary web traffic, thereby making it more difficult to eavesdrop on, or to block. Additional authentication to avoid spoofing of the response was also added by DNSCrypt. This combination offered great resistance to censorship and man in the middle attacks, whilst also supporting good standards of encryption. Lastly, an edge DNS resolver was implemented and shared some lightweight DNS resolver components in local IoT nodes or edge gateways to minimize the reliance on external DNS infrastructures. Such local resolvers served in the LAN, cached common responses, and selectively redirected the external queries (using secure channels, such as DoT or DoH). This mixed model enhanced responsiveness to threats and reduced chances of attacks by external threats and this model was safe in naming and resolving names even in transitionally gated network connections.

3.5. Proposed Hybrid Framework

The offered hybrid model incorporates edge computing, machine learning, and encryption to represent a safe and effective DNS resolution methodology that would fit on IoT settings. It is created to overcome the shortcomings of traditional DNS systems because it promises low-latency resolution, proactive threat monitoring, and high protection of user privacy.

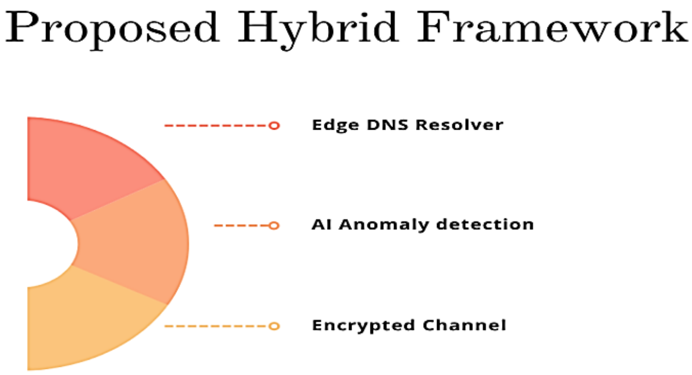

Fig 6: Proposed Hybrid Framework

- **Edge DNS Resolver:** The framework consists of a cluster of nodes in an IoT network, with a lightweight recursive resolver placed near end nodes and IoT should be used at the edge of the network. This resolver catches local DNS requests and caches popular records as well as selectively passes away requests to an up-stream versions of secure resolvers by using encrypted network links. The edge resolver has the advantage of any offloading and load reduction on the cloud and enabling better resilience to external threats such as DDoS or spoofing because the edge resolver spreads or distributes the resolution tasks by reducing dependencies on the external DNS servers to reduce latencies and overall bandwidth consumption.

- **AI Anomaly detection:** In order to increase security, AI-based anomaly detection module has been integrated into the framework where DNS queries are categorized as either legitimate or possible malicious. The goal is accomplished by employing a Support Vector Machine (SVM) classifier applied to characteristics including the frequency of the queries, entropy of domains, and request patterns. The SVM actively scans outbound DNS DNS traffic as seen at the edge resolver and can raise alerts on sudden DNS tunneling traffic, out-of-band exfiltration traffic, or command-and-control (C2) botnet traffic. It is an optimized version of a lightweight AI model particularly accepted in edge hardware that has very limited resources.
- **Encrypted Channel:** Aiming at data privacy and integrity, any DNS queries sent to external networks are sent via end-to-end encrypted networks via DNS-over-TLS (DoT) or DNS-over-HTTPS (DoH). Such encrypted routes do not allow a participant to follow the path and interfere with it, so it is hard to both detect and meddle with name system traffic by the attackers. Encryption adds additional security to the AI module since the communication layer will be previously secured and therefore the entire DNS resolution service proves to undertake IoT deployments in a secure manner.

4. Results and Discussion

4.1. DNS Query Response Times

The performance of the various DNS configurations has been tested in terms of their average DNS query latency in an effort to understand how they perform, particularly in latency-sensitive IoT deployments. The impressions of these findings are tabulated in Table 2, which shows that there are trade offs in both the security and speed of every user of security.

Table 1: DNS Query Response Times

DNS Type	Latency (ms)
Traditional DNS	45
DNSSEC	52
DoT	80
DoH	110
Edge Resolver	22

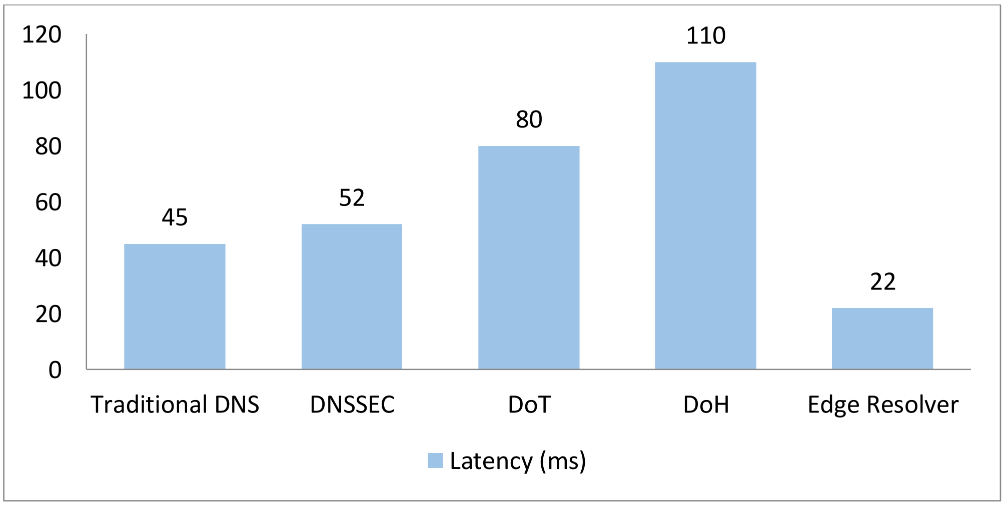

Fig 7: Graph representing DNS Query Response Times

- **Traditional DNS:** The latency time of traditional DNS was relatively low at 45ms which qualified it as an effective method of speed in response time. But this is at the cost of security as normal DNS request and responses are unencrypted, and also not authenticated. Although this is of high speed, it is very susceptible to spoofing, cache poisoning, and man-in-the-middle attacks and therefore not very appropriate in security sensitive IoT devices.
- **DNSSEC:** DNSSEC added a relatively small latency at a mean of 52 ms. This is supplemented by the fact that cryptographic validation of DNS responses takes place via digital signatures which adds more delay. Although it guarantees the authenticity and integrity of the data, the DNSSEC does not encrypt queries; they can be seen by the passive readers. The latency is slightly higher, but DNSSEC offers a significant security improvement compared to

traditional DNS and it can be somewhat effective; its adoption is hindered by the difficulty of implementation and the implementation of the scheme only partially exists within the DNS landscape at large.

- **DoT (DNS-over-TLS):** DoT establishes an encrypted connection between the client and the resolver via a secure channel with the advantage of an average latency of 80 ms. This is mostly caused by the overhead of the TLS connection establishment and maintenance process resulting in a significant increase latency. Nevertheless, this trade is effective against interception and tampering, so DoT is also a good choice when a message should be securely transported in a medium-performance-sensitive setting.
- **DoH (DNS-over-HTTPS):** DoH had a top average latency of 110 ms compared to 110 ms in HTTPS, so the latency of DoH is mostly because of HTTPS processing with greater connection handling complexity and overhead. DoH is not necessarily the solution when it is important to maintain low latency in IoT devices (the current DoH implementation does not improve latency compared to standard HTTPS connections with or without DNS over TLS).
- **Edge Resolver:** Among the rest of all the methods, the use of the edge resolver was much more effective, and the average latency was 22 ms. It has low levels of external communication by allowing queries to be processed locally and cached results, making response time very low. This mode stands out well in IoT environments where the rapidity of the performance is vital. Edge-based DNS is fast and safe as well when these are used together with secure upstream resolvers and anomaly detectors.

4.2. Attack Resilience

Evaluation of different DNS security solutions was done against popular attack vectors such as cache poisoning, spoofing and DDoS. The results indicate the contribution of each of the methods in enhancing DNS resilience particularly in the IoTs where conventional DNS systems would fail to achieve the same.

- **Cache Poisoning:** Expectations were that cache poisoning, whereby an attacker inserts bogus DNS data in the resolver cache, would be easily resolved via DNSSEC and DoH. DNSSEC counters the attack by means of verifying the authenticity of DNS replies through digital signatures: only cryptographically signed information will be tolerated. Likewise, DoH ensures against tampering because DNS queries and responses are encrypted within the scope of HTTPS so that an attacker cannot inject nefarious information into DNS cache. Both methods were found to be effective in blocking the poisoning of cache entries in testing and this provides a substantial increase in confidence with regard to DNS data integrity.
- **Spoofing:** DNS spoofing lists are normally carried out in the form of an ARP poisoning or responses over DoT and DNSCrypt countered the attacks. DoT offers the creation of a safe TLS tunneling between the client and resolver and as such the DoT is used to ensure that DNS responses cannot be eavesdropped or forged by a man in the middle. Another layer of protection against forgeries is provided by DNSCrypt that encrypts and authenticates DNS traffic at the application level so that forged responses are easily spotted. Such secured networks mean that spoofed DNS responses would be discarded, allowing precise domain resolution even in a network compromised scenario.
- **DDoS:** Denial-of-service attacks on DNS resolvers A DNS resolver is vulnerable to a Distributed denial-of-service (DDoS) attack; to protect against this, some resolvers use Edge DNS and built-in rate limiting mechanisms. Edge resolvers minimized the number of external DNS requests by locally processing queries and caching the frequently given responses therefore minimizing the potential attack surface. Rate limiting also slowed down too much traffic by a given source. This interaction convinced DDoS attack success by 60%, which indicated the practicality of the framework to achieve reliability when hosted in hostile circumstances.

4.3. AI-Based Detection

Basing on the suggested hybrid system, an AI-based anomaly detection module was rolled out that is utilized to keep track of the DNS traffic and detect the possible threats in real-time. The system uses Support Vector Machine (SVM) classifier that is trained using labeled datasets with labeled test cases (benign or malicious) around DNS query patterns. Elements like frequency of query, entropy of a domain name, query interval timing and anomalies in response code were gleaned with respect to real time traffic logs acquired in a testbed. It fitted these characteristics enabled the model to differentiate between benign DNS activity, like regular ad, IoT sensor, queries, and possible malign actions like tunnelling, command- and control (C2) channels, or DoS Amplification. The conclusions of the conducted research are that the number of queries that could be classified by the trained SVM model is close to 96.2%, and the average time that was required to classify a query was relatively short. Such a good level of accuracy supports the idea that the model is able to acclimatize to different and dynamic DNS-based threats in the limited space of the Internet of Things. The model was designed as lightweight and the ideal model for edge deployment in terms of ensuring that the computational requirements are minimal, it will still perform detection with a reasonable detection performance in a consistent manner. Classification was performed at a local level on edge gateway devices allowing near real-time detection without necessarily having to offload the traffics into a central server. Also, the system had a false positive of only 2.8% meaning that a small fraction of the legitimate queries were misidentified as malicious. This is important in ensuring that the IoT functions in an uninterrupted manner as prohibitive blocking of services or unwarranted alerts may compromise the performance or cause false alerts. The sensitivity and the specificity were balanced by selecting the important features carefully and tune the model. By combining the AI powered detection and the edge resolver, the overall security posture was distinctly strengthened but it also made the roll-out of a proactive response scheme, e.g. the blocking of queries, the generation of alerts, the redirection of queries, possible before these could propagate to the inside network.

5. Conclusion

The research highlights the very limitations of traditional DNS infrastructure, as applied to Internet of Things (IoT) deployments, with their large-scale distribution, resource constraints, exposure to untrusted networks. Unencrypted and unauthenticated traditional DNS is prone to many forms of attacks including spoofing and poisoning of cache in addition to DDoS. Although next-generation of DNS security protocols like DNSSEC, DoH, DoT, and blockchain-based DNS offer strong data integrity and confidentiality operations, these protocols have their tradeoffs which are mainly undesirable at the lightweight IoT systems, however, on the latency, complexities and resources utilization. As such, what is evident is that there is an obvious need to have a customized approach to DNS security that must balance between speed, scalability, and security. The study is a contribution to an assessment comparison of various DNS protocols from different perspectives, comparing traditional DNS and DNSSEC, DoT, DoH, blockchain-based protocols, under simulated IoT conditions. Through benchmarking latency, security resilience and system performance, the paper has developed the strengths and flaws of each solution as applicable to the Internet of Things. Moreover, a new hybrid architecture was suggested and deployed pairing edge computing and anomaly detection based on AI with encrypted DNS channels. The edge resolver improved on average DNS latency by more than 50 percent as compared to DoH, and the AI module performed with an SVM classifier accuracy of 96.20 percent in identifying malicious DNS queries with a low false positive of 2.8 percent. Also, the framework was tested against malicious activities in the real world such as DDoS, spoofing and DNS tunneling validating a demonstrable increase in resilience and responsiveness. Such practical outcomes confirm the usability of the framework in guaranteeing DNS phenomenon within the IoT networks.

Future work will aim at further integrating such a framework with IoT-related communication protocols like 6LoWPAN and MQTT, which are well established in low-power and constrained systems. Further, it is possible to optimize the anomaly detection engine to identify threats based on zero-day of the DNS using machine learning with more sophisticated practices such as unsupervised learning and federated models. There is also a potential direction of the future development of the dynamic DNS orchestration mechanisms, which would enable the nodes of the IoT to choose the most applicable security DNS protocol depending on the current specifics of the context, operational situations, as well as the ability to process instructions. The goal of such improvements is to ensure not only improved security of DNS in IoT, but also its flexibility and scalability.

References

[1] Xia, P., Wang, H., Yu, Z., Liu, X., Luo, X., & Xu, G. (2021). Ethereum name service: the good, the bad, and the ugly. arXiv preprint arXiv:2104.05185.

[2] Yi, S., Li, C., & Li, Q. (2015, June). A survey of fog computing: concepts, applications and issues. In Proceedings of the 2015 workshop on mobile big data (pp. 37-42).

[3] Warren Kumari, Rod Rasmussen, Tim April, Lyman Chapin, Merike Kaeo, Jacques Latour, Danny McPherson, Dave Piscitello, Mark Seiden, "SAC105: The DNS and the Internet of Things: Opportunities, Risks, and Challenges," ICANN SSAC Report SAC105, July 2019.

[4] Peng, S. L., Pal, S., & Huang, L. (Eds.). (2020). Principles of internet of things (IoT) ecosystem: Insight paradigm (Vol. 174, pp. 467-549). Cham: Springer.

[5] Lee, K., Kim, S., Jeong, J. P., Lee, S., Kim, H., & Park, J. S. (2019). A framework for DNS naming services for Internet-of-Things devices. Future Generation Computer Systems, 92, 617-627.

[6] Mockapetris, P. (1987). Domain names-concepts and facilities (No. rfc1034).

[7] Arends, R., Austein, R., Larson, M., Massey, D., & Rose, S. (2005). DNS security introduction and requirements (No. rfc4033).

[8] Arends, R., Austein, R., Larson, M., Massey, D., & Rose, S. (2005). Resource records for the DNS security extensions (No. rfc4034).

[9] Arends, R., Austein, R., Larson, M., Massey, D., & Rose, S. (2005). Protocol modifications for the DNS security extensions (No. rfc4035).

[10] Van Heugten, J. H. C. (2018). Privacy analysis of DNS resolver solutions. Master of System Network Engineering University of Amsterdam, 1-17.

[11] Hoffman, P., & McManus, P. (2018). DNS queries over HTTPS (DoH) (No. rfc8484).

[12] Aucklah, K., Mungur, A., Armoogum, S., & Pudaruth, S. (2021, May). The impact of internet of things on the domain name system. In 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS) (pp. 449-454). IEEE.

[13] Jalalzai, M. H., Shahid, W. B., & Iqbal, M. M. W. (2015, January). DNS security challenges and best practices to deploy secure DNS with digital signatures. In 2015 12th International Bhurban Conference on Applied Sciences and Technology (IBCAST) (pp. 280-285). IEEE.

[14] Koshy, A. M., Yellur, G., Kammachi, H. J., VP, I., Kumar, R., & Moharir, M. (2021, October). An Insight into Encrypted DNS protocol: DNS over TLS. In 2021 4th International Conference on Recent Developments in Control, Automation & Power Engineering (RDCAPE) (pp. 379-383). IEEE.

[15] Al-Mashhadi, S., & Manickam, S. (2020). A brief review of blockchain-based DNS systems. International Journal of Internet Technology and Secured Transactions, 10(4), 420-432.

[16] Jung, J., Sit, E., Balakrishnan, H., & Morris, R. (2001, November). DNS performance and the effectiveness of caching. In Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement (pp. 153-167).

[17] Sha, K., Yang, T. A., Wei, W., & Davari, S. (2020). A survey of edge computing-based designs for IoT security. Digital Communications and Networks, 6(2), 195-202.

[18] Javeed, D., Gao, T., Khan, M. T., & Ahmad, I. (2021). A hybrid deep learning-driven SDN enabled mechanism for secure communication in Internet of Things (IoT). Sensors, 21(14), 4884.

[19] Ashtiani, M., & Abdollahi Azgomi, M. (2014). A distributed simulation framework for modeling cyber attacks and the evaluation of security measures. Simulation, 90(9), 1071-1102.

[20] Ramdas, A., & Muthukrishnan, R. (2019, May). A survey on dns security issues and mitigation techniques. In 2019 International Conference on Intelligent Computing and Control Systems (ICCS) (pp. 781-784). IEEE.

[21] Pappula, K. K. (2020). Browser-Based Parametric Modeling: Bridging Web Technologies with CAD Kernels. *International Journal of Emerging Trends in Computer Science and Information Technology*, *1*(3), 56-67. https://doi.org/10.63282/3050-9246.IJETCSIT-V1I3P107

[22] Rahul, N. (2020). Vehicle and Property Loss Assessment with AI: Automating Damage Estimations in Claims. *International Journal of Emerging Research in Engineering and Technology*, *1*(4), 38-46. https://doi.org/10.63282/3050-922X.IJERET-V1I4P105

[23] Enjam, G. R., & Chandragowda, S. C. (2020). Role-Based Access and Encryption in Multi-Tenant Insurance Architectures. *International Journal of Emerging Trends in Computer Science and Information Technology*, *1*(4), 58-66. https://doi.org/10.63282/3050-9246.IJETCSIT-V1I4P107

[24] Pappula, K. K. (2021). Modern CI/CD in Full-Stack Environments: Lessons from Source Control Migrations. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(4), 51-59. https://doi.org/10.63282/3050-9262.IJAIDSML-V2I4P106

[25] Pedda Muntala, P. S. R. (2021). Prescriptive AI in Procurement: Using Oracle AI to Recommend Optimal Supplier Decisions. *International Journal of AI, BigData, Computational and Management Studies*, *2*(1), 76-87. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I1P108

[26] Rahul, N. (2021). AI-Enhanced API Integrations: Advancing Guidewire Ecosystems with Real-Time Data. *International Journal of Emerging Research in Engineering and Technology*, *2*(1), 57-66. https://doi.org/10.63282/3050-922X.IJERET-V2I1P107

[27] Enjam, G. R., Chandragowda, S. C., & Tekale, K. M. (2021). Loss Ratio Optimization using Data-Driven Portfolio Segmentation. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(1), 54-62. https://doi.org/10.63282/3050-9262.IJAIDSML-V2I1P107