Dr. Jagdish W. Bakal, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

AUDS: Design and Development of Agentless Unified Defense System

Dr. Jagdish W. Bakal, Paras Thakur, Darin Joy Peringalloor, Vaishnavi Pawar
Department of Computer Engineering Pillai HOC College of Engineering and Technology, Rasayani

Abstract- Today most of the internet users faces different types of cyber threats like phishing websites, DNS spoofing attacks, malicious redirects, and also unwanted tracking activities. However users make the use of blacklists and signature based detection in order to block the websites but this method is not effective for newly created malicious websites. So to solve this issue, we decided to design and developed the Agentless Unified Defense System (AUDS). This system gives real time protection without any need of extra software. It works directly inside our browser. When the site looks unsafe, then it blocks the access. Also when the site appears to be suspicious, it opens in a separate Disposal Window to keep the browser safe. In this way, AUDS provides simple web security.

Keywords: Cybersecurity, Browser Extension, Phishing Detection, DNS Monitoring, Real-Time Protection, Machine Learning, Malicious Ads, Threat Prevention, URL Analysis, User Safety.

I. INTRODUCTION

People uses the internet for communication, online shopping, banking purpose, cloud services and also for remote work. Hence this makes the life more convenient but it also increases the risk of cyber attacks. Attacks like phishing, DNS spoofing, malicious advertisements, tracking scripts, and hidden redirects are more common now. This attacks are mainly use to steal the sensitive and important information like passwords, bank details, and personal data without the users consent.[1]

Many traditional system such as antivirus and browser protection tools only rely on static blacklist format. This methods can block the known threats but they are not effective for new attacks. Now a days attackers uses techniques like domain obfuscation, AI generated attacks to avoid the detection. Hence the traditional security methods are not enough.[2]

Phishing contributes around 60% of cyber incidents. Exploitation of system vulnerabilities is about 21.3%. Hence this shows that social engineering techniques are the common entry point for modern cyber attacks.

Another major issue with cybersecurity solution is that they do not provide real time protection while browsing. There are tools which requires installation and configuration to work inside browsers.[4] This is difficult for non-technical users. Also users only react when the attack happens. To overcome this limitation there is a need of real time, lightweight system which works in browsers.

The Agentless Unified Defense System (AUDS) is a browser based cybersecurity system which protects the users from phishing attacks, DNS spoofing, malicious redirects. It doesn't need any external agents or complicated installations [7].

AUDS mainly checks the DNS, TLS certificates, and domain reputation to detect suspicious activity during browsing. Before you can use a website, the system checks to make sure whether the website is real. It also filters out potentially dangerous domains in real time and uses a safe Disposal Window to keep bad websites away from the rest of the system. This makes it less likely that the system will be hacked.

II. LITERATURE SURVEY

Recent researches in web security usually focuses on protecting the users from attacks like phishing, malicious redirects, DNS spoofing. As the cyber attacks have become more advanced with the AI generation techniques, researchers have developed many new detection methods like certificate verification, URL pattern analysis, DNS checking, and machine learning models. However there is lot of

improvement in cybersecurity but still many system focus on only one security layer. Even many studies has detected the phishing by checking SSL/TLS certificates and monitoring login forms on websites. Here they mainly check for certificate validity, issuer details, and the presence of input fields to identify suspicious sites. This works well in traditional phishing websites, but it is not effective for modern attacks which uses AI. Also this method struggle with multi- stage attacks because there is no DNS level monitoring and real-time analysis.

Researchers have also studied on the open redirect vulnerabilities, which are commonly used to spread malware and perform phishing attacks. Most detection tools collects the URLs from web pages and then add test payloads to observe how the redirection happens. This method can detect simple redirect patterns. But it mainly requires complex backend processing and high computational resources. Hence these systems are mainly designed for testing environments and not for real-time browsing. Modern attacks use dynamic scripts, multiple redirect chains, and hidden URLs,. This can easily bypass the static detection methods. As a result, many redirect analysis tools do not work effectively within the browser environment. DNS-based threats such as spoofing and poisoning have also help researchers to improve domain name resolution security. While these methods improve DNS security and privacy, but they operate at the network or resolver level and require complex router or system configurations. Hence this makes them difficult for non- technical users to set up. Also the cryptographic processes introduce delays. Because of this they did not offer real time protection during active browsing.

Earlier the browser security extensions mainly focuses on enforcing HTTPS connections and blocking spam links. As these methods improved the basic security, they lacked advanced detection. They cannot handle modern threats such as AI-based phishing pages, DNS spoofing, malicious advertisements, and hidden redirect chains. Static blacklist systems also faces limitations because attackers now a days create short lived domains and change their infrastructure in order to avoid detection. As a result the single signature based detection is not sufficient in today's threat environment. Overall the research shows the progress in the areas like phishing, DNS security, URL classification and redirects analysis. But most of the existing tool detects only one type of threat at a time instead of providing a unified, multi-layered security approach.

Today's cyber attacks usually uses many tricks together. For example, attackers may misuse SSL certificates, change DNS settings, redirect users to a fake websites, and even show misleading content in the same attack. Because of this, we need a browser-based security system that can check websites in real time before they open it, use different detection methods, and provide protection without complicated setup or extra software.

III. MOTIVATION

The digital world has change the way people now communicate, make payments and even access online services everyday. As more number of services move online so more users will get exposed to advanced cyber threats. Earlier the attacks usually consist of spam emails or simple phishing websites, but now attackers use more advanced techniques. It mainly contains Ai generated phishing pages, fake domain names, fast-changing DNS setups, malicious ads, hidden tracking tools, and encrypted redirect chains. Also modern attacks are designed in a way which closely look like real websites both in design and function. This makes it difficult for normal users to identify the attack. Because of this many users shares sensitive information like domain names, fast-changing DNS setups, malicious ads, hidden tracking tools, and encrypted redirect chains.

Even there are many cyber security tools present today, but problem still exist. Most browser security system rely on static blacklists, signature based detection, or centralized threat databases. These method can block known threats, but they are not effective against newly created phishing websites or short- term zero-day attacks that disappear quickly. Many security solution also work outside the browser and require manual setup, complex

installation, or changes in network settings. Hence this make them difficult for regular users to use it. Because of this, gaps remain in the security system that attackers can take advantage of.

Another major problem is that many browsers do not perform the real time validation before loading a webpage. The malicious scripts, hidden redirects and even fake login forms often run in the same environment as trusted websites. Hence this increases the risk of session hijacking, cookie theft, and unauthorized access to user data. The warning messages alone are not sufficient because many users ignore them or they do not understand the technical details properly. These issues mainly highlights the need of lightweight and browser-integrated security system which works automatically without disturbing the users experience. To address this challenge the Agentless Unified Defense System (AUDS) was developed as a simple and effective browser based protection mechanism. The main goal was to create a system or extension that does not require the external security tool, or a complex router configurations, or large installations. The AUDS directly operates within the browser just by intercepting the HTTPS request and validating them before the webpage get loads. Hence this early validation mainly helps to prevent the malicious scripts and fake content from executing in the main browsing session.

AUDS is based on a multi-layered security approach. It combines the DNS monitoring, SSL/TLS certification, domain reputation checking. DNS monitoring helps to detect the suspicious domain resolution patterns while certificate validation ensures that the communication between users and the websites is secure. Domain reputation analysis also adds more another layer of trust checking. The Disposal Window feature provides browser-level isolation by opening the suspicious websites in to a restricted environment. This is done with limited resources and permissions .This prevents harmful websites from accessing the user credentials, cookies, or local system data. The overall aim of AUDS is to provide strong security while keeping the system simple and user friendly.

IV. EXISTING SYSTEM

Most existing browser security tools reply on the old blacklists or single-layer detection methods. However they do not provide the unified, real time and also adaptive protection directly inside the browser. Also the traditional system depends on static databases, centralized monitoring or a external tools which makes it difficult to detect the fast changing threat . Threats like phishing attacks, DNS spoofing, and malicious redirects are difficult to detect.

A. An Open Redirect Analysis Tool (IEEE 2024)

This tool pulled links from web pages and added test payloads to find URLs that were likely to redirect. It used an exhaustive list to mark suspicious URLs and looked at how they redirected.

The system can find described redirect patterns, but it doesn't cover all payloads and isn't automated for live browsing sessions. It also puts a lot of stress on big web pages. To fix this problem, we made and put into place a real-time URL security checking system (as shown in our proposed system diagram). This system mainly compares the user request with a database of a know website. This is done before allowing the access. If a URL is identified as unsafe then is immediately blocks it. If it is verified as a safe then the users continues to browse normally. Compared to static redirects analysis tools, this approach mainly provides a more practical solution.

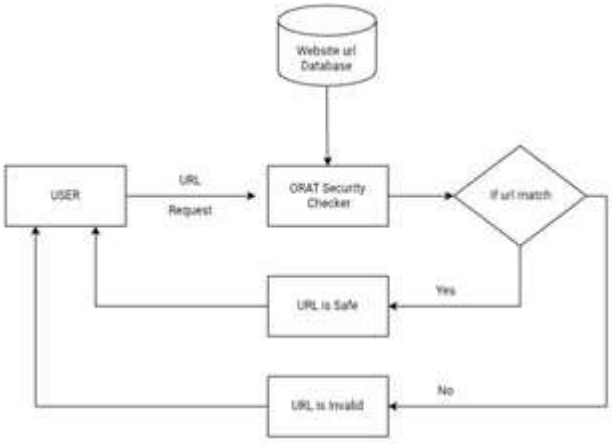

Figure 1: Existing System Architecture

B. Improving DNS with DNSSEC and DoT (IEEE 2024)

This system used DNSSEC for DNS validation and DNS over TLS (DoT) to allow encrypted DNS queries. This method mainly improves the DNS security and also help to prevent certain spoofing attacks. However the setup process is complex for the regular users and cryptographic operation also slow down the DNS resolution. It also requires configuration at network or router level.

In contrast to our approach performs the DNS monitoring and intrusion detection directly at the browser level . It does not require any router settings. This makes the system easier to use.

V. PROBLEM STATEMENT

As the services continuous to grow the users are more exposed to advanced cyber threat. Today the attackers uses more than just a simple phishing emails. The bypass security checks it using techniques like DNS manipulation, misuse of certificates, redirect chains, and hidden tracking methods. Many malicious websites look almost identical to legitimate ones. This makes it difficult for users to identify fake pages. Also most browser protection system react only when threat is detected. They rely more on centralized blacklists format. Hence newly created or short-lived phishing domains often go unnoticed. In addition, many tools either completely block access or simply show warning messages. Another major issue is the lack of built-in isolation in standard browsers. When the users open any trusted website it runs on same environment as a trusted sessions. Hence this increases the risk of password theft, session hijacking, and misuse of system resources. Without proper resource control, network filtering, or sandbox execution, even a single malicious interaction can compromise the users security.

Cyber threats today are constantly changing and becoming more advanced. Attackers can bypass traditional blacklist- based systems by using short-lived domains, AI-generated phishing websites, fast-flux DNS methods, and encrypted malicious traffic. Also many of these domain get disappear before they are officially reported. This makes the static detection method inactive. Another issue is that current security solutions are not unified. Many tools usually focuses on only one layer of protection, such as URL filtering, DNS security, or certificate verification. However, modern attacks often target multiple layers at the same time. This creates security gaps that attackers can exploit. Usability is also a concern. Many users ignore browser warnings or do not fully understand SSL indicators and certificate details. Traditional tools either block websites completely or display general alerts without offering a safe way to inspect suspicious sites. In addition, most browsers run all web content in the same environment, which increases the risk of credential theft and session hijacking when a malicious website is opened.

Hence there is a clear need of a browser based security that does not require any external agents. Before even allowing the users to access the website the system check the DNS integrity and SSL/TLS certificates in real time. It should also provide a separate and restricted space for opening a suspicious websites. Also at the same time the solution must be secure and fast. It should not affect the normal browsing experience.

VI. METHODOLOGY

1. REQUIREMENT ANALYSIS

We were able to figure out what the Agentless Unified Defense System (AUDS) needed by looking at present web- based cyber threats, research publications, and the flaws in current browser security solutions. The research focused on important topics like.

The research focused on important topics like:

- More phishing attacks that employ phoney login pages
- Sending users to the wrong domain by spoofing DNS
- Using or faking SSL/TLS certificates in a way that isn't right
- No checking in real time before going to the website

- Websites that look suspicious don't get isolated at the browser level.

The major goal was to build a light security framework that works with browsers and makes sure that websites are real before letting visitors access. It also protects against suspicious websites without the need for outside agents or big system installs.

2. Module Design

The system has the following functional parts:

- URL Request Monitoring Module: This module watches HTTPS requests performed by users in real time through the browser extension.
- SSL/TLS Certificate Validation Module: This module makes sure that the certificate is real and valid before letting visitors visit a website.
- DNS Monitoring & Reputation Module: This module monitors the reputation of a domain and the integrity of DNS to discover domains that are false or malicious.
- Security Decision Engine: Looks at the validation results and makes a choice about whether the URL is safe or not.
- Disposal Window Module: This module opens websites that look dubious in a limited, private browsing environment to keep the system safe.
- User Alert & Control Module: This module sends users clear security notifications and helps them decide who can access their data without interrupting their work.

Each module can work on its own, yet they all work together in the same browser-based framework.

3. Real-Time Pre-Validation

One of the main idea of AUDS is to validate the websites before they are fully loaded. Unlike the traditional system that reacts only after the page content appears, the AUDS mainly intercepts HTTPS requests at the browser extension level. It checks them before the page is displayed. Hence this early validation mainly helps to prevent the malicious scripts , hidden redirects, and fake login forms from running in the main browser environment. Just by checking the website at an early stage the suspicious domain can be detected before the user interacts with them.

4. Disposal Window Mechanism

In order to handle the suspicious websites, AUDS uses a controlled mechanism called the Disposal Window. When the website is marked as suspicious but not fully confirmed as malicious, then it is opened in a restricted environment instead of a browser. This environment runs inside a lightweight Docker-based sandbox. It makes the use of Chromium container with limited CPU, memory, file access, and network permissions. Extra firewall rules and WebRTC restrictions are applied. This is done to stop unauthorized communication between the sandbox and the main system. Even if a malicious script runs inside this isolated space, it cannot access user sessions, cookies, credentials, or local system resources. Hence this approach mainly helps to keep the main browsing environment safe.

5. Security Decision Framework

The Security Decision Engine combines the results from certificate verification and DNS monitoring. This is done to decide whether a website is safe or not. Instead of showing complex technical warnings, the system displays simple status messages like Safe, Suspicious, or Blocked. Hence this makes the user easy to understand the risk and reduces the chances of ignoring warnings. Also the users are allowed to interact with the website only when the risk level is considered acceptable.

6. Deployment Easy

While developing AUDS, maintaining system efficiency was an important focus. The architecture is designed in such a way that uses minimal CPU and memory so that browsing speed is not affected. Also as the system is agentless it does not require any external security software or network-level configuration. This makes easier for non-technical users to set up. Optimization techniques were also applied to certificate validation, DNS checks, and sandbox initialization for a smooth user experience.

7. Security Adaptation

AUDS was designed to be expandable in the future. Its modular structure allows new features such as machine learning based URL classification, behavior analysis. If the cyber threats evolve, additional validation layers and detection techniques can be

integrated without affecting the existing system. Hence this ensure that the framework remains useful and adaptable over time. Also the architecture of AUDS is modular and flexible. These allows us to respond to new types of cyber attacks.It performs real-time browser monitoring along with DNS and certificate validation to detect threats before a webpage loads. Even the detection logic and isolation rules can be improved based on security findings. Because of its modular design the new security components can be added when required while keeping the system lightweight and user-friendly.

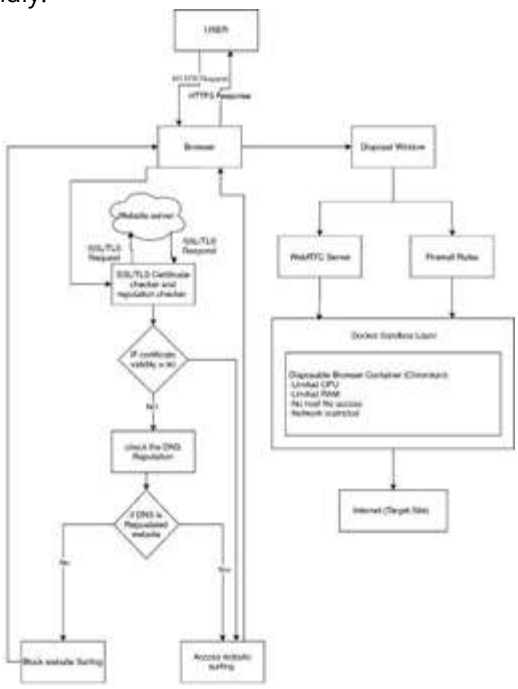

Figure 2: AUDS System Architecture

The AUDS architecture follows a layered security approach. Which means it works directly within the browser and performs real-time checks. It controls the access to websites until security verification is completed. When a user sends an HTTPS request, the browser extension first intercepts it before the webpage loads and begins the validation process. First the system connects to the web server in order to retrieve and verify the SSL/TLS certificate. It then checks, whether the certificate is valid, issued by a trusted authority, properly linked in the trust chain, and matches the domain name. If the certificate fails validation then the access to the website is immediately blocked to prevent phishing or spoofing attacks. Incase if the certificate is valid, the system then performs DNS monitoring and reputation analysis. At this stage, it checks for DNS spoofing, suspicious hosting behavior, or unusual domain resolution patterns. This is done by examining DNS integrity and the domain's previous reputation history.

The Security Decision Engine uses the results from certificate verification and DNS analysis. It decides whether a website should be allowed, blocked, or opened safely. If a website appears suspicious but is not confirmed as harmful then the AUDS activates the Disposal Window mechanism. This mechanism mainly runs the website inside a lightweight Docker-based sandbox using an isolated Chromium container.

The container has limited CPU and memory usage, no access to host system files, and restricted network permissions. Also additional firewall rules and WebRTC restrictions prevent unauthorized communication between the sandbox and the main system. So even if the malicious script runs inside this isolated environment, it remains confined within the sandbox. This protects user sessions, passwords, and system resources. And keep the performance impact minimal and the browsing experience smooth.

	exisiting extrension	AUDS
Certificate Date validity	YES	YES
Warning	YES	YES
Reputation checking	NO	YES
Certificate overall information use	NO	YES
sandboxing	NO	YES

Figure 3: Comparison of Existing System and the Proposed AUDS Framework

The comparison table shows the difference between the proposed Agentless Unified Defense System (AUDS) and traditional browser security extensions in terms of features and design. Both systems can check whether an SSL/TLS certificate is valid and display warning messages. However, regular extensions usually perform only basic checks, such as verifying certificate expiry or configuration errors, without deeper analysis. While in AUDS there is more advanced validation . It consist of reputation based checks and detailed analysis of certificate information. It mainly contains issuer authenticity, domain matching, and trust chain verification. As a result this layered approach reduces the risk of phishing and certificate spoofing attacks and also bypass the simple expiration checks.

Traditional browser extensions mainly work in a reactive manner and it do not include advanced domain knowledge. They usually do not check DNS reputation or isolate the browser environment when a website appears suspicious. Because of this, the users ignore the warnings or continue browsing at their own risk. This can expose the sensitive data such as login credentials and session information of the users. The AUDS addresses these limitations through DNS reputation monitoring and the Disposal Window mechanism. Suspicious websites are opened in a controlled, sandboxed environment which prevents the harmful scripts from accessing the system resources, cookies, or any other active sessions. The key difference lies in the security approach. The traditional tools rely on the alert-based, single-layer protection, while the AUDS provides a proactive, multi-layered defense with real time validation. As a result, AUDS improves detection accuracy, reduces false positives, enhances user safety. It is more flexible and reliable browser-level security solution.

VII. RESULTS

This interface demonstrates the real-time certificate validation feature of the Agentless Unified Defense System (AUDS). When a user visits a website, the browser extension intercepts the HTTPS request and displays the current website name along with an option to view certificate details. The system mainly checks the important SSL/TLS parameters such as the certificate validity period, issuer authenticity, and trust chain verification.

Figure 4: Certificate Checker Popup Interface

If the certificate satisfies all required security checks, the extension clearly displays a “SAFE” status to the user. This validation takes place before the user fully interacts with the webpage. Hence it prevents phishing and certificate spoofing attacks. By presenting certificate information in a simple and easy-to-understand format, AUDS improves user awareness while maintaining a smooth browsing experience.

Figure 5: Certificate Details Page

This page displays the detailed information about the SSL/TLS certificate. Because of this the user can verify the website’s security credentials. It shows all the important details such as domain name, issuer authority, subject name, certificate validity period (valid from and valid to), current status, and Subject Alternative Names (SANs). Here the system checks whether the certificate is active, correctly issued, and

trusted, and clearly indicates if it is valid. Hence by presenting the certificate details in a structured way the AUDS helps the users to review all the security information without needing deep technical knowledge. This helps the users to detect possible certificate misuse or spoofing. It also allows the user to make informed decisions before continuing to use the website.

Figure 6: Blocking Page

This interface mainly shows how the AUDS protects the user when a website is detected as unsafe. If a security risk is found then the extension immediately blocks the access and displays a warning message. The message mainly contains "Access to this website was blocked" along with clear UNSAFE status. The page also shows the details such as the domain name, risk level, confidence score. Users are provided with limited option like "COPY URL" , "View Certificate" and a restricted "Go Back" option is given to prevent unauthorized access. A warning note clearly informs users that they should proceed only if they understand the risks. Hence this structured warning interface improves user awareness while maintaining the control over dangerous access.

VIII. CONCLUSION

The Agentless Unified Defense System (AUDS) is a browser- based solution designed to handle modern cybersecurity threats such as phishing attacks, DNS spoofing, malicious advertisements, and unsafe redirects. It combines domain reputation analysis, certificate verification, DNS monitoring, ad inspection, and intelligent URL checking to detect and block threats before they affect the user. Unlike traditional security tools that rely on static blacklists and delayed responses, the AUDS mainly operates in real time. This ensures that validation and protection take place immediately. Hence the users get a smooth and uninterrupted browsing experience.

The adaptive design of AUDS allows the system to improve its threat detection capabilities over time. It can handle changing phishing techniques, AI-based attack patterns, and also the hidden malicious content. Because of its modular structure, new features can be easily added without affecting the existing system. Hence it which makes it scalable and flexible. Since validation and analysis are performed directly inside the browser, AUDS reduces latency. It also avoids the dependence on external security agents, while also supporting user privacy. The browser-level isolation mechanism further strengthens security by preventing suspicious websites from interacting with trusted sessions or system resources. Overall, AUDS bridges the gap between traditional reactive security systems and proactive defense approaches. It mainly provides a lightweight and user- friendly protection framework that enhances web browsing safety without adding complexity.

REFERENCES

1. K.Sakai,K. Takeshige,SMatsugaya, M. Shimamura, and M. Hashimoto, "Phishing Prevention Focusing on Certificates and Input," IEEE, 2025.
2. J. Martinho, D. Mendes, and P. Pinto, "An Open Redirect Analysis Tool," IEEE, 2024.
3. S. Abirami and R. Naresh, "DNS Enhancement with DNSSEC and DoT for Enhanced Online Security," IEEE, 2024.
4. S. K. Singh and P. K. Roy, "Detecting Malicious DNS over HTTPS Traffic Using Machine Learning," IEEE, 2020.
5. R. Hegde, A. More, D. Hendre, S. Chafle, and L. Ragha, "Prompt-to-SQL Injections in LLM-Integrated Web Applications: Risks and Defenses," arXiv, 2024.

6. PhishLang - A Real Time, Fully Client-Side Phishing Detection Framework Using MobileBERT, arXiv, 2025.
7. Efficient Chrome Extension for Phishing Detection Using Machine Learning Techniques, Springer Nature (Preprint), 2024.
8. Detection of Malicious DNS and Web Servers Using Graph-Based Approaches, IEEE, 2021.
9. Enhanced Link Redirection Interface for Secured Browsing using Web Browser Extensions, IJACSA, 2013.
10. Toward Automated DNS Tampering Detection Using Machine Learning, FOCI, 2024.
11. LogiTriBlend - A Novel Hybrid Stacking Approach for Enhanced Phishing Email Detection Using ML Models and Vectorization Approach, IEEE, 2024.
12. Probabilistic Matching-Based Compression to Mitigate Compression Side Channel Attacks Against HTTPS, IEEE, 2024.