Shedding Light on Hidden Dangers: A New Perspective on DNS Leaks

Peter Membrey
ExpressVPN
pete.m@expressvpn.com

Abstract

In this paper, we introduce a novel categorization of DNS leaks into Type 1 *and Type 2, underscoring the privacy and security risks these pose to VPN users. A critical examination reveals how Stealth DNS servers, by remaining hidden from traditional DNS leak detection tools, contribute to a false sense of security, particularly with Type 2 leaks. We then highlight the challenges associated with detecting such leaks. The paper also explores proposed mitigations and best practices for VPN providers, emphasizing server-side protections and the strategic whitelisting of DNS servers to enhance online privacy and security.*

Introduction

In this paper we introduce a new form of DNS leak (imaginatively named Type 2) and highlight the risk it poses to the privacy and security of VPN users. Additionally we identify a novel way in which DNS servers can hide themselves from the traditional DNS leak test tools and explore how this can give users a false sense of security when it comes to their VPN of choice.

> The issues outlined in this paper have been seen "in the wild", specifically on devices running Microsoft Windows. All of the VPN providers where this issue was detected quickly and responsibly applied one of the recommended solutions, effectively securing their platforms. We thank them all for working with us.

The rest of the paper is organized as follows. In the Background section, we provide an overview of DNS and VPN technologies, essential for understanding the significance of DNS leak prevention. The Expanding the Definition of DNS leaks section introduces the nuanced differences between Type 1 and Type 2 DNS leaks, then in Threat Scenarios emphasizes the novel risks associated with them. Challenges in Detecting Stealth DNS Servers discusses the limitations faced by existing DNS leak detection methods in identifying stealth servers. Proposed Mitigations and Best Practices offers early recommendations for VPN providers to mitigate the risks of Type 2 DNS leaks. We conclude with Discussion and Conclusion, reflecting on the implications of our findings and the path forward for enhancing online privacy and security.

Background

The Domain Name System (DNS) plays a pivotal role in the internet's functionality, acting as the translator between human-readable web addresses and their corresponding IP addresses. This system ensures that users can easily access websites without needing to memorize complex numerical addresses. Virtual Private Networks (VPNs), on the other hand, serve as a critical tool for enhancing online privacy and security. By creating a secure and encrypted connection over the internet, VPNs allow users to mask their real IP addresses, effectively shielding their online activities from unauthorized surveillance and data interception. Despite these protections, the phenomenon known as a DNS leak can undermine the privacy guarantees of VPNs.

A traditional DNS leak occurs when DNS queries, which should be securely routed through the VPN's encrypted tunnel to the VPN provider's DNS servers, are instead sent directly to the user's Internet Service Provider (ISP) or other external DNS servers. This inadvertent exposure of DNS requests outside the encrypted VPN connection can reveal detailed information about the user's internet activities, compromising their anonymity and privacy. Understanding the dynamics of DNS and VPN technologies, and the inherent risks of DNS leaks, is essential for developing more robust security measures to safeguard online privacy.

Expanding the Definition of DNS leaks

When considering the concept of a DNS leak within the context of VPN security, it's important to refine and expand upon the traditional definition to encompass a broader range of vulnerabilities. The conventional understanding of a DNS leak is often limited to scenarios where a user's public IP address is inadvertently exposed to a DNS server. This perspective suggests that as long as DNS requests are routed through the VPN (and hence the user's actual IP address is masked) no leak has actually occurred. However, our research indicates that this view is overly simplistic and fails to account for the complexities and potential dangers associated with DNS requests under certain conditions.

> We do not address leaks with DoH in this paper due to its very different nature. However, it too is potentially vulnerable to a Type 2 leak if it is either set in the operating system or automatically enabled by the browser.

Type 1 DNS leak

A Type 1 DNS leak is identified by the following characteristics:

```
The VPN is active, yet due to configuration errors or a lack of protective measures,
DNS requests bypass the VPN tunnel, directly reaching a public DNS server using the
user's genuine IP address.
```

This type of leak directly compromises the user's anonymity by revealing their IP address to the DNS server, thereby providing a straightforward method to track the user's online activities back to their physical location.

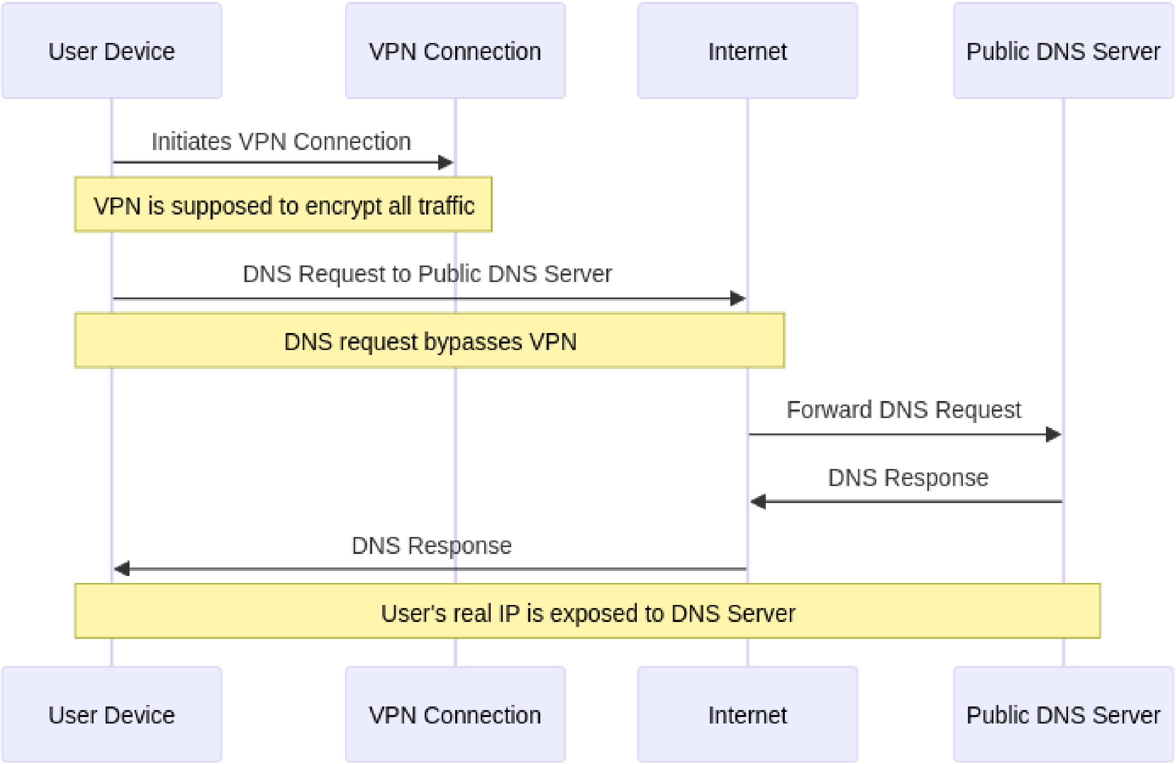

Figure 1. Sequence diagram for the flow in a Type 1 DNS leak

Type 2 DNS leak

A Type 2 DNS leak, on the other hand, presents a subtler but no less significant risk:

```
The VPN is operational, and DNS requests are channeled through the VPN. However, these
requests are directed to DNS servers specified by the user's ISP or the local network,
instead of secure, privacy-focused DNS servers.
```

While the user’s real IP address is not immediately disclosed, the fact that DNS queries are handled by ISP-linked servers introduces a vulnerability. These servers can log the requests, potentially allowing for user tracking and profiling. Moreover, as these servers can also reply to the user, in conjunction with targeted attacks such as a MITM (Man in the Middle) attack, this information can be exploited to infer the user’s identity or location.

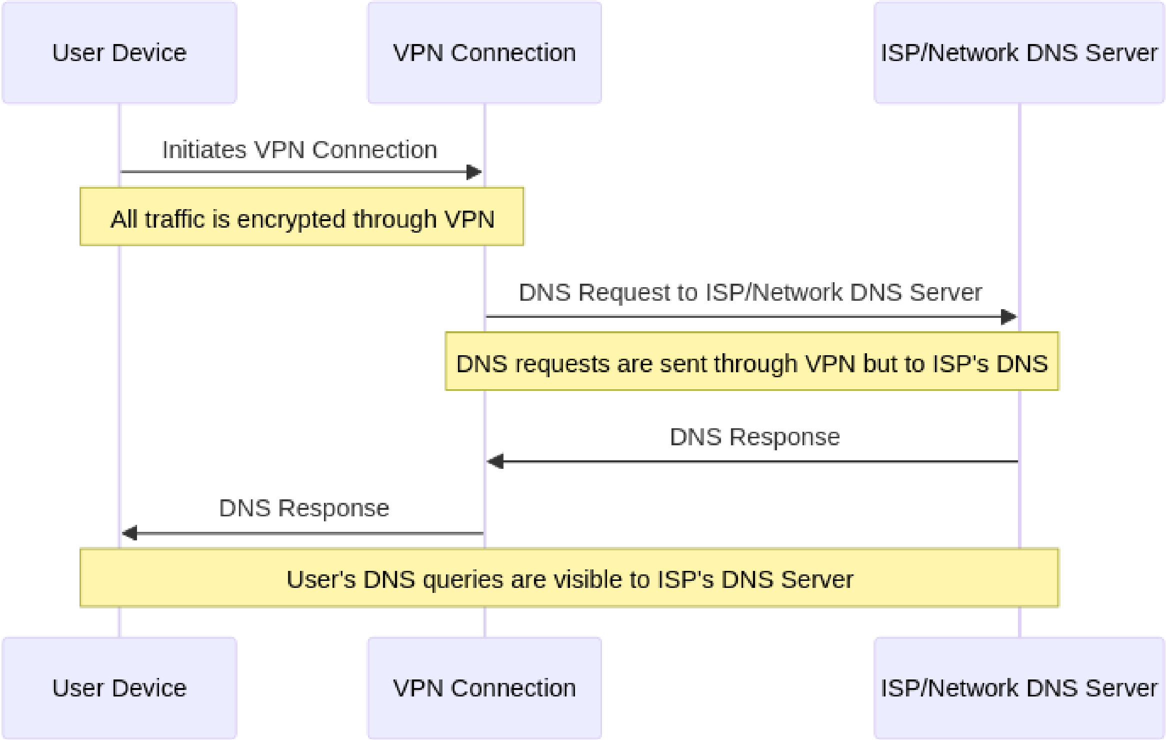
User Device
VPN Connection
ISP/Network DNS Server
Initiates VPN Connection
All traffic is encrypted through VPN
DNS Request to ISP/Network DNS Server
DNS requests are sent through VPN but to ISP's DNS
DNS Response
DNS Response
User's DNS queries are visible to ISP's DNS Server
User Device
VPN Connection
ISP/Network DNS Server

Figure 2. Sequence diagram for the flow in a Type 2 DNS leak

> A Type 2 DNS leak occurs when DNS requests are directed to DNS servers not deliberately chosen by the user. For instance, if a user intentionally sets their system to use a specific DNS provider like Cloudflare, this action is deemed a matter of personal preference rather than a leak.

This refined definition emphasizes the necessity of addressing both types of DNS leaks. The focus should not solely be on preventing the exposure of the user's IP address but also on ensuring that DNS requests are securely managed and routed to prevent inadvertent data leaks or exploitation by malicious actors. By broadening the understanding of what constitutes a DNS leak, VPN providers can implement more comprehensive safeguards against a wider array of threats.

Direct Threats Posed by DNS leaks

Before delving into specific threat scenarios, it's important to understand the immediate dangers associated with Type 2 DNS leaks. These vulnerabilities can compromise user privacy and security in several ways:

- **Privacy Exposure:** By routing DNS requests to an ISP's DNS server, the ISP gains visibility into every website a user attempts to visit, as they receive the hostnames needing resolution.
- **Activity Correlation:** ISP's can also correlate incoming DNS requests with other traffic on their network, potentially identifying users' online behaviors and patterns.
- **Censorship through DNS Poisoning:** An ISP's DNS server can undermine the user's ability to freely browse the internet by poisoning DNS responses, effectively blocking access to specific websites.

Note: While these threats are also relevant to Type 1 DNS leaks, the direct exposure of the user's IP address in such cases often makes additional exploitation redundant.

It’s crucial to recognize that DNS server settings can be manipulated by DHCP (Dynamic Host Configuration Protocol), meaning the source of a leak isn’t limited to ISPs. Schools, coffee shops, hotels, and other networks offering Wi-Fi can equally be points of compromise by injecting their DNS servers, thus broadening the potential environments where users might face these threats. This sets the stage for the Threat Scenarios section, where we’ll explore the application of these threats in real-world contexts.

Threat Scenarios

The scenarios presented below illustrate the potential risks and vulnerabilities associated with Type 2 DNS leaks. These examples highlight the consequences of such leaks, particularly when VPN providers fail to adequately secure DNS traffic.

> ℹ The assumption for these scenarios is a user operating on a Windows laptop, with a VPN provider that routes all DNS traffic over the secure VPN tunnel. These conditions set the stage for various exploitative scenarios.

Scenario 1: Coffee Shop Marketing Data Collection

A coffee shop aims to enhance its marketing efforts by analyzing the online browsing habits of its customers. Traditionally, this involved tracking internet usage through WiFi access linked to loyalty cards. However, the rise of VPN usage has diminished the value of this data. To counteract this, the coffee shop deploys DNS servers with public IP addresses, assigning them to customers’ devices via DHCP. This setup allows the shop to monitor all domain lookup activities, inadvertently gathering detailed information on customers’ online behavior. Over time, with basic analysis, the shop could potentially link specific website visits to individual customers.

Scenario 2: Coffee Shop and Sensitive Content

In this scenario, a coffee shop owner seeks to identify customers accessing content deemed sensitive or controversial. By setting up a public DNS server, the owner can monitor DNS traffic and, through physical observation, correlate online activities with specific customers. For example, in regions with strict regulations on certain topics, this could lead to users being reported to authorities based on their browsing habits, illustrating the grave privacy implications of DNS leaks.

Scenario 3: Targeted Stalking

Here, a benign coffee shop environment is exploited by a malicious individual intent on stalking. Using a laptop equipped with the necessary tools, the attacker manipulates DHCP responses to redirect a target’s DNS queries to a server under their control. This enables the attacker to monitor the victim’s online activities without requiring advanced hacking skills or privileges.

Scenario 4: Hotel Guest Monitoring

Hotels, especially those in cooperation with state entities, are in a prime position to exploit DNS leaks. By assigning unique DNS servers to each guest, the hotel can directly link DNS requests to specific individuals, thereby compromising their privacy and security on a highly personalized level.

Introducing Stealth DNS Servers

Type 2 DNS leaks present a nuanced challenge in network security, exposing users to potential privacy breaches and security risks without the direct exposure of their real IP address. This type of leak occurs when DNS requests, though routed through a VPN, end up being processed by DNS servers not explicitly chosen by the user - often those provided by the user's ISP or a default network. The complexity of Type 2 leaks further escalates with the introduction of Stealth DNS servers, adding a layer of obscurity and risk to the environment.

What is a Stealth DNS Server?

A Stealth DNS Server is a DNS server that does not show up during a leak test. A Stealth DNS Server may not be malicious, but even an ISP's DNS servers may log resolution requests which could put the user's privacy at risk. It is also possible for a Stealth DNS server to be malicious in that it specifically records user activity, or even potentially responds with fake information, making a potential MITM attack possible. Again, the key feature of a Stealth DNS server is that it does not show up during a leak test, even though it has been set without the user's knowledge or consent.

Stealth DNS Servers and Type 2 leaks

Stealth DNS servers exacerbate the vulnerabilities associated with Type 2 DNS leaks by operating under the guise of legitimacy. These servers can be silently injected into a network, either through DHCP configuration manipulation or other means, without the user's explicit consent. This silent introduction can happen across various networks, be it ISPs, educational institutions, or public Wi-Fi hotspots, making any network susceptible to such interventions.

One critical aspect that ties Stealth DNS servers closely to Type 2 DNS leaks is their capability to see and potentially log all user DNS requests. Typical ISP DNS servers, which generally do not respond to DNS requests originating outside their customer base (such as those coming from a VPN network) are also acting as Stealth DNS servers. This characteristic means that even non-malicious ISPs inadvertently contribute to the obfuscation of DNS traffic monitoring, as their servers, while not appearing in standard DNS leak tests, continue to process user DNS queries unseen.

> ISP DNS Servers can be classified as Stealth DNS Servers when they fail to respond to queries originating from external networks, not within the ISP's own network. This behavior is generally deemed as best practice, aiming to restrict DNS services solely to the ISP's clientele. However, this characteristic means that during DNS leak tests, these ISP DNS servers may not be detected, even though they could be receiving, logging, and processing the requests. This nuance is critical in

> understanding the subtleties of DNS leak tests and the potential for overlooked vulnerabilities.

Stealth DNS Servers have a notable relationship with Type 2 DNS leaks, impacting their detection and the potential risks involved. While similar implications can be observed with Type 1 Leaks, the nature of these interactions differs significantly. In Type 1 leaks, DNS requests originate directly from the user’s device and are sent through their ISP, making it more likely for these requests to receive responses from DNS servers. Consequently, such requests are typically identified in DNS leak tests as anticipated.

Challenges in Detecting Stealth DNS Servers

The difficulty in identifying Stealth DNS Servers, particularly in the context of Type 2 DNS leaks, stems from a number of factors that mask their presence and operation. This section delves into why these server remain elusive to standard detection methods.

Inherent Stealthiness

Stealth DNS Servers operate under the radar, blending seamlessly with legitimate network traffic. They do not overtly alter network behavior in a manner that is easily detectable by traditional security tools or DNS leak tests. This subtlety in operation means that they avoid triggering the usual red flags associated with network anomalies.

Complex Interactions with VPNs

When VPNs are employed, particularly with features like split tunneling activated, the intricate interactions between the VPN software and the operating system complicate the detection landscape. These interactions can inadvertently allow DNS requests to get routed to Stealth DNS Servers without clear indicators of a leak. The nuanced behavior of these leaks makes them challenging to detect through standard leak test websites, which typically rely on overt signs of DNS misrouting.

Behavioral Mimicry

Stealth DNS Servers can mimic the operational characteristics of legitimate DNS servers, responding to queries in expected manners and times. This behavioral mimicry means that even when DNS requests are routed differently, the responses received do not deviate significantly from what would be expected from a non-stealthy server, thus avoiding suspicion.

Variable Response Patterns

The detection difficulty is further compounded by the variable response patterns of DNS servers across different networks. ISPs' DNS servers, for example, may not respond to requests originating from outside their network (e.g., from a VPN), masking potential leaks. This characteristic means that Stealth DNS Servers operating within these parameters can remain undetected, as their lack of response to external queries is not out of the ordinary.

Summarizing the challenge

The detection of Stealth DNS Servers, especially within the context of Type 2 DNS leaks, presents challenges that impact online privacy and security significantly. These servers, by their very design, blend seamlessly into network configurations, mimicking the behavior of legitimate DNS servers while potentially intercepting or manipulating user DNS requests. This capability to remain inconspicuous, coupled with their complex interactions with VPNs and network settings, makes them formidable threats that exploit the mechanisms intended to safeguard user data.

Proposed Mitigations and Best Practices

When it comes to mitigating type 2 DNS leaks, service providers are presented with choices that hinge on their policy towards third-party DNS servers. This section highlights best practices and delves into tailored solutions that align with the varying stances of VPN providers on this matter. At the core of both of these strategies is the desire to safeguard user privacy and ensure the integrity of VPN connections, without inadvertently compromising the flexibility and preferences of the end-user.

> The solutions here focus on those deployed on the server so as to provide coverage regardless of the protocol or platform used to connect. Client side protection is of course vital and should be considered the first line of defence. Indeed such protections are necessary to effectively prevent Type 1 leaks. However, as providers cannot guarantee that clients are being used as expected, additional protection on the VPN servers is highly recommended.

> Specific clientside defenses are not discussed here due to the wide range of implementations and related issues. Regardless, VPN providers should prioritise ensuring clientside defenses, even after adding protections to the server.

For VPN providers opting not to support third-party DNS servers, the recommended best practice is to block DNS traffic outright. This approach serves as a straightforward method to prevent DNS leaks by ensuring that all DNS queries are either resolved within the VPN's own network or not at all. Blocking DNS requests effectively neutralizes the risk of exposing users through DNS queries to external observers. Providers can implement this measure with either `nft` or `iptables` (assuming Linux servers), creating a secure environment by default. However, to support the diverse needs of their user base, providers may choose to selectively whitelist specific DNS servers. This concession allows users who rely on certain third-party services, for reasons such as enhanced privacy or speed, to continue using these services within the VPN's network.

Conversely, VPN providers that aim to accommodate or currently support the use of third-party DNS servers must adopt a more nuanced approach to prevent DNS leaks. The use of a Transparent DNS Proxy stands out as an effective solution in this scenario. A Transparent DNS Proxy intercepts all DNS requests, regardless of their intended destination, and reroutes them to a DNS server approved and trusted by the VPN provider. This method ensures that all DNS queries are securely resolved within the provider's controlled environment, mitigating the risk of leaks. To further align with user demands and preferences, providers can also employ whitelisting. By doing so, they can cater to specific requests for third-party DNS services, ensuring those DNS queries are handled according to the user's choice without compromising the overall security of the VPN connection.

By either blocking DNS traffic outright or utilizing a Transparent DNS Proxy, providers can secure their networks against Type 2 DNS leaks. At the same time, through strategic whitelisting, they can respect and accommodate the diverse needs of their users, ensuring a secure, private, and user-friendly VPN experience.

Implementing the solution

To effectively prevent type 2 leaks, a direct and efficient method involves blocking all DNS requests originating from VPN clients before they pass through the server. This precaution ensures that any such requests are intercepted and halted by the VPN provider, eliminating the risk of them leaking out to external networks. The implementation of this protective measure can be accomplished through the use of `nft` or `iptables`, depending on the Linux distribution and kernel version deployed.

With `nft`:

Commands to block DNS traffic being forwarded on a Linux based VPN server using `nft`

```
nft add rule ip filter forward udp dport 53 counter drop
nft add rule ip filter forward tcp dport 53 counter drop
```

Similarly with `iptables`:

Commands to block DNS traffic being forwarded on a Linux based VPN server using `iptables`

```
iptables -A FORWARD -p udp --dport 53 -j DROP
iptables -A FORWARD -p tcp --dport 53 -j DROP
```

The principal drawback of blocking all DNS requests lies in the disruption it causes to users relying on third-party DNS providers, like Google (`8.8.8.8`) or Cloudflare (`1.1.1.1`). These users might face connectivity problems and won't regain network access until they adjust their DNS settings to comply with the new restrictions. To mitigate such issues, an alternative solution involving the use of a Transparent DNS Proxy is explored in the following section, offering a more seamless transition without compromising network integrity or user experience.

Transparent DNS Proxy

This method proves beneficial for VPN providers permitting DNS requests to the internet, accommodating users reliant on this feature. A transparent proxy functions by altering the destination of DNS requests, ensuring they are processed by the provider's DNS servers. Such a mechanism effectively prevents leaks, as irrespective of the initially configured DNS server, all traffic is redirected and handled by the VPN provider, offering a more secure alternative to the inadvertent use of third-party DNS servers.

With `nft`:

Commands to rewrite DNS requests to a specific trusted DNS server with `nft`

```
nft add rule ip nat prerouting udp dport 53 dnat to 10.0.0.1:53
nft add rule ip nat prerouting tcp dport 53 dnat to 10.0.0.1:53
```

Similarly with `iptables`:

Commands to rewrite DNS requests to a specific trusted DNS server with `iptables`

```
iptables -t nat -A PREROUTING -p udp --dport 53 -j DNAT --to-destination 10.0.0.1:53
iptables -t nat -A PREROUTING -p tcp --dport 53 -j DNAT --to-destination 10.0.0.1:53
```

Whitelisting DNS Servers

It’s feasible to whitelist specific DNS providers to exempt their traffic from being either blocked or rerouted. For instance, Cloudflare’s DNS servers can be whitelisted, allowing users preferring their service to continue without interruption. This approach effectively covers the majority of scenarios where users desire the use of custom DNS servers.

Blocking DoH (DNS over HTTPS)

Although DoH is supposed to improve user security, the fact that both Firefox and Chrome in some cases can automatically enable this feature and send DNS requests to the services of their choice is very worrying. It is fortunately possible to disable this automatic feature by ensuring that the canary domain (that is a domain that exists just for checking if DoH is allowed) is blocked. To implement this, VPN providers must block the following domain:

```
use-application-dns.net
```

This should be blocked by returning the error `NXDOMAIN`. See Mozilla’s support pages here for more information.

> ⚠ The canary domain only affects automatic DoH usage - if the user has specifically chosen to use DoH, blocking this domain will have **no effect**.

Discussion and Conclusion

This paper has explored the concept of DNS leaks, specifically introducing the concept of Type 2 DNS leaks and the elusive nature of Stealth DNS servers. The discussion highlighted the inherent risks these leaks pose to VPN users' privacy and security, challenging the effectiveness of traditional DNS leak tests. We have also identified the means by which Stealth DNS servers can evade detection, thus providing users with a false sense of security regarding their online privacy.

Additionally we highlight a critical gap in the current understanding and detection of DNS leaks. By expanding the definition of DNS leaks to include Type 2 leaks and introducing the concept of

Stealth DNS servers, we shed light on the hidden dangers that threaten users' online privacy and security.

As we move forward, it is imperative that we continue to refine our detection methodologies, enhance the security measures of VPN services, and foster an environment of transparency and cooperation within the cybersecurity community. Only through these concerted efforts can we hope to stay one step ahead and safeguard the privacy and security of our users.