****

**Table of Contents**

[**Customer Information**](#_heading=h.30j0zll) **2**

[Customer Contact Information](#_heading=h.1fob9te) 2

[**Executive Summary**](#_heading=h.3znysh7) **2**

[What type of DNS threats are in this summary and why?](#_heading=h.2et92p0) 2

[Observed events](#_heading=h.tyjcwt) 3

[Detailed information about the different categories](#_heading=h.3dy6vkm) 3

[Data exfiltration details](#_heading=h.1t3h5sf) 4

[Categories](#_heading=h.4d34og8) 4

[Malware & Ransomware Details](#_heading=h.2s8eyo1) 4

[DoH events](#_heading=h.17dp8vu) 5

[Recommended Actions](#_heading=h.3rdcrjn) 5

[Addendum](#_heading=h.26in1rg) 6

[DNS Reconnaissance](#_heading=h.lnxbz9) 6

[Cache Poisoning](#_heading=h.35nkun2) 6

[Data Exfiltration](#_heading=h.1ksv4uv) 6

[DNS Tunneling](#_heading=h.44sinio) 6

[Volumetric Attacks (DDoS, Amplification)](#_heading=h.2jxsxqh) 6

[Malware Command and Control](#_heading=h.z337ya) 7

# 

# Customer Information

## Customer Contact Information

| **Customer Name** | {{ customer }} |
| --- | --- |
| **Contact Name** | {{ cus\_name }} |
| **Phone Number** | {{ cus\_phone }} |
| **E-mail Address** | {{ cus\_email }} |

# Executive Summary

This document provides a summary for your review. It calls out specific events that have been observed in your environment during the evaluation period and should serve as a launchpad to investigate gaps and a potential solution. This is not a complete assessment of your security posture. It focuses on the DNS specific vulnerabilities and gaps that may exist.

Graph

{{ myimage }}

Why the focus on DNS you may ask? A recent Forrester paper [reference – bibliography] highlights the fact that 91% of malware uses DNS in various phases/stages of the attack.

With recent high profile attacks on the rise (Colonial, JBS etc) it has become clear that DNS is not just a simple name resolution protocol, but rather a critical choke point to interrupt/disrupt the attack.

This has become a big enough problem that the NSA & CISA issued a stark warning [reference – bibliography], that DNS requires a strategic approach, labeled PDNS or Protective DNS. In their estimation, 90% of breaches could have been prevented by disrupting the attack at the DNS level.

### What type of DNS threats are in this summary and why?

For the purpose of this document, we limit the scope to Command & Control, Data Exfiltration, Ransomware and DNS over HTTP. These constitute the most serious threats to a customer:

**Data Exfiltration**: Used to steal data or other IP from you. The ability to detect and block a data exfiltration attempt in progress is critical.

**Ransomware**: This has become the method of choice (so it seems) to attack organizations and extract ransom from them. As of late, it is not just encrypting machines anymore (demanding a ransom to unlock), but also threatening organizations with the release of confidential data (obtained via data exfiltration).

**Command & Control**: Used in both ransomware and data exfiltration attacks. It’s a critical piece to the attacker’s infrastructure. If communication with Command & Control can be interrupted at the DNS level (when an infected host tries to establish communication) would be the best time to disrupt the attack.

**DNS over HTTP (DoH)**: Whilst not an attack in itself, there are 2 major problems DoH presents to IT organizations:

* Loss of visibility: Since DoH uses public DoH servers, there is no record of the request.
* Workflow disruption: Public DoH servers cannot resolve internal domain names (for us in internal applications)
* Encryption: DoH uses TLS. Decryption is expensive and most traditional network perimeter solutions struggle with.

There are many more types of DNS attacks. These are listed in the addendum for your reference.

## Observed events

In total **{{ total\_events }}** events with a HIGH threat - and HIGH confidence level were recorded during the evaluation period.

The following table breaks down the events by category:

| Category | Count |
| --- | --- |
| Data Exfiltration | {% if total\_dex\_count is not none %}{{ total\_dex\_count }}  {% else %}  NONE  {% endif %} |
| Malware/Ransomware | {% if total\_mal\_count is not none %}{{ total\_mal\_count }}  {% else %}  NONE  {% endif %} |

## Detailed information about the different categories

A few notes on the structure of tables and how to read them:

***IP address****: Depending on how the DNS queries were forwarded you may see the actual private source IP of the host (if B1DDI/DFP or an agent was used) or you may see the same public WAN IP address (if the events were forwarded to the B1TD Anycast IPs using the External Network method)*

***IoC****: Indicator of Compromise. You can get more intelligence about a specific IoC by clicking on it. The link will take you to our Dossier research tool (inside B1TD) and display what is known about the specific IoC.*

***Username****: If the data was forwarded by our agent, information about the user logged in at the time is automatically recorded. Please note that the username can also be obtained by correlating/integrating with an existing IAM solution (such as Azure AD, Okta etc). However, that level of integration typically exceeds the scope of an eval.*

### Data exfiltration details

The following tables lists all the recorded data exfiltration events aggregated by IP address. We chose to aggregate the events by source IP address, because data exfiltration is never a single DNS query. Rather it is a series of events all originating from an infected host(s). It is useful to know the number of events per source IP address, as it may be an indication of the amount of data being exfiltrated. The information is being sorted by

| Data Exfil | |
| --- | --- |
| {%tr for data in data\_exfil[‘results’]%} |  |
| {{data[‘key’]}} | {{data[‘count’]}} |
| {%tr endfor %} |  |

### Categories

The following table lists all recorded Malware & Ransomware events:

{% for data in data\_cat[‘results’][0]['sub\_bucket'] %}

{{data[‘key’]}} - {{data[‘count’]}}

{% for da in data[‘sub\_bucket’]: %}{{ da[‘key’]}} - {{da[‘count’]}}

{% for d in da[‘sub\_bucket’]: %}{% if da[‘key’] != ‘feed\_name’: %}{{ d[‘key’] }} - {{d[‘count’]}}{% endif %}

{% endfor %}

{% endfor %}

{% endfor %}

### Malware & Ransomware Details

The following table lists all recorded Malware & Ransomware events:

{% for data in data\_mal[‘results’][0]['sub\_bucket'] %}

{{data[‘key’]}} - {{data[‘count’]}}

{% for da in data[‘sub\_bucket’]: %}{{ da[‘key’]}} - {{da[‘count’]}}

{% for d in da[‘sub\_bucket’]: %}{% if da[‘key’] != ‘feed\_name’: %}{{ d[‘key’] }} - {{d[‘count’]}}{% endif %}

{% endfor %}

{% endfor %}

{% endfor %}

### DoH events

The following table lists all recorded DoH events. As stated earlier, DoH by itself is not a threat. We opted to include this information into the report to make customers aware of DoH. Oftentimes, DoH is entered into the mix unbeknownst to the client. As of late Apple products started using DoH, and Firefox browsers use DoH by default.

| DoH | |
| --- | --- |
| {%tr for data in data\_doh[‘results’][0]['sub\_bucket'] %} |  |
| {{data[‘key’]}} | {{data[‘count’]}} |
| {%tr endfor %} |  |

## Recommended Actions

We recommend consulting with your solutions partner. There are however some best practices we recommend to look into (if not already in place).

Detection is a good start, but using the information to help security operations to be more efficient is the key to deriving value. Following the NIST framework, there are 4 areas, this type of information can be used.  


***Specific to DoH***:

It is recommended to set up a RPZ (Response Policy Zone), that blocks access to public DoH servers. Applications will have to fall back to system level DNS.

## Addendum

There are many methods of attacking a DNS infrastructure to either learn information about a network infrastructure for a later attack; to disrupt, overwhelm, and shut down network services and accessibility; to exfiltrate sensitive data; or even utilize DNS infrastructure from one organization to launch an attack against a third-party entity. While not completely comprehensive, below is the most common types of DNS threats that are typically leveraged against an environment.

### DNS Reconnaissance

*DNS Reconnaissance* is part of the initial information gathering stage and part of penetration testing that a malicious actor will engage in, to obtain information regarding the DNS servers and the DNS records. This information is often publicly accessible and provides better understanding about an organization's network infrastructure. Typically, unrecognized and unmonitored, there are multiple methods of DNS reconnaissance that are attempted with varying degrees of success, including zone transfers, brute force record resolution, reverse lookup of PTR records, DNS record cache snooping, and zone walking of improperly configured zones. Information obtained through DNS reconnaissance can help the bad actor map network hosts by enumerating the contents of a zone.

### Cache Poisoning

*DNS Cache Poisoning* or DNS spoofing is a type of attack that diverts traffic away from legitimate servers and towards fake or spoofed ones. DNS cache poisoning, as the name implies, introduces an invalid or compromised IP address as part of the local cache table of a device and reroutes traffic away from authoritative DNS servers. More sophisticated DNS cache poisoning attacks stand up proxy websites and domains that look similar to the legitimate site in an attempt to trick a user into providing personal information, usernames, and passwords or to cause them to click on an invalid link in order to infect the end host.

### Data Exfiltration

*DNS Data Exfiltration* is the unauthorized transfer of data from a computer through the DNS protocol. This transfer of data can be manual triggered by someone with physical access to the computer, or automated and carried out through malware across a network. A bad actor will compromise or infect an endpoint to encode any sensitive data the client has access to, break it into chunks, and send it out masked as legitimate DNS queries, and reassemble this data offsite away from security controls. This method of data exfiltration through the DNS protocol circumvents next-generation firewalls (NGFWs), IDSs, and IPSs, and other traditional security solutions.

### DNS Tunneling

*DNS Tunneling* is a method of attack that encodes data of other programs or protocols in DNS queries and responses. DNS tunneling often includes data payloads that can be added to an attacked DNS server and used to control a remote server and applications. DNS tunneling requires the compromised system to have external network connectivity, as DNS tunneling requires access to an internal DNS server with network access. DNS tunneling is often also associated with insider threats, as one of the primary methods of circumventing network and security policy to exfiltrate data to an offsite domain.

### Volumetric Attacks (DDoS, Amplification)

*DNS Flood* attacks overwhelms an organization’s DNS servers in an attempt to disrupt DNS services. By disrupting DNS resolution, a DNS flood attack compromises a website, API, or web application, preventing the ability to respond to legitimate traffic. These types of attacks can be difficult to distinguish from normal heavy traffic because the large volume of traffic often is generated from botnets and is geographically dispersed, to mimic legitimate DNS traffic while overwhelming infrastructure.

*DNS Amplification* attacks are reflection-based volumetric distributed denial-of-service (DDoS) attack in which an attacker leverages the functionality of open DNS resolvers in order to overwhelm a target server or network with an amplified amount of traffic, rendering the server and its surrounding infrastructure inaccessible.

DNS Amplification attacks exploit an imbalance in bandwidth consumption between an attacker and the targeted resource (querying TXT records or unsecured zone transfers). By sending small queries, the attacker is able to trigger large responses, exploiting this disparity cost against the existing architecture. This is magnified across many requests, resulting in a volume of traffic that can disrupt network infrastructure and take it offline. By using a botnet to make similar requests, an attacker can both avoid detection, and significantly increase attack traffic against a network.

### Malware Command and Control

*Malware Command and Control via DNS* can be achieved by hijacking DNS queries and responding with malicious IP addresses or domains. By using spear phishing attacks to convince users to click on infected links by using typo squatting, look alike domains, domain generating algorithms, dictionary domain generating algorithms names, all are methods to create links or domains that look or sound similar to a legitimate site and bypass traditional blacklists. Malware is then installed on the compromised client and communication to the malware command and control server is maintained through DNS to perpetuate an attack against an organization.