Lab Tutorial 2- DNS Anomaly Detection Using Wireshark

FSCT 8540 Lab: Analyzing anomalous DNS traffic

Author: Maryam R. Aliabadi

Date: Sep 17th, 2025

Introduction

DNS (Domain Name System) is a critical component of the Internet that translates human-readable domain names into IP addresses. Attackers often exploit DNS vulnerabilities to redirect users to malicious sites through DNS spoofing or poisoning. This lab will introduce students to Wireshark for network analysis, focusing on DNS traffic and anomaly detection.

Learning Objectives

By the end of this lab, students will be able to:

- Understand the role of DNS and common DNS attacks.
- Install and use Wireshark to capture and analyze network traffic.
- Identify normal DNS query-response patterns.
- Detect anomalies in DNS packets (TXID mismatch, suspicious source ports, early responses).
- Understand how attackers execute DNS poisoning attacks.

Section 1: Wireshark Installation and Introduction

Download and Install Wireshark

- Go to Wireshark Official Website.
- Download the appropriate installer for your OS (Windows, Linux, macOS).
- Follow installation instructions, including installing WinPcap/Npcap if prompted (required for packet capture on Windows).

Introduction to Wireshark

- **Open Wireshark**.
- **Explore the interface**: Menu, Capture Interfaces, Packet List, Packet Details, and Packet Bytes.
- **Select the network interface** (or virtual interface) that carries DNS traffic (e.g., Wi-Fi or Ethernet).

Section 2: Capturing DNS traffic

Settings and filters

i) **Apply a display filter in Wireshark to view only DNS packets**

ii) **Set a capture filter (optional, before starting capture):**

- UDP DNS only (standard): udp port 53
- TCP DNS (large responses / DNS over TCP): tcp port 53
- Both UDP and TCP: port 53
- Specific resolver: host 1.2.3.4 and port 53
  Why? Capture filters reduce the volume and keep captures focused. If you want to capture everything, skip this and use display filters later.

iii) **Start the capture.**

iv) **Reproduce the behavior you want to observe** (e.g., open a browser and visit some domains. You can also use **dig** or **nslookup** to manually query domains and observe DNS responses.)

v) **Stop the capture** once enough traffic is collected (avoid running indefinitely).

Understand DNS Packet Structure

Expand a DNS packet and examine key fields:

- **Transaction ID (TXID)** — unique identifier for each query-response pair.

- **Query/Response Flags** — identifies whether the packet is a query or response.

- **Source/Destination IP and Ports** — normal DNS responses usually come from port 53.

- **Answer Section** — contains the resolved IP address.

Section 3: Display filters to inspect DNS traffic

Use Wireshark's display filters (apply after capture):

All DNS packets: dns

DNS queries only: dns.flags.response == 0 or dns.qr == 0

DNS responses only: dns.flags.response == 1 or dns.qr == 1

DNS for a specific name: dns.qry.name == "example.com"

DNS with invalid signature or DNSSEC-related flags: dns.flags.ad == 1 (AD = authenticated data)

Show packets with suspicious UDP ports: udp.srcport != 53 && dns

Show responses where answer count > 0: dns.count.answers > 0

Packet bytes / raw DNS layer: click packet → expand Domain Name System section

Tip: use Follow UDP stream or Follow TCP stream for a particular query/response pair if you want the sequence.

Section 4: Using Wireshark Statistics for DNS Analysis

- **Statistics → DNS**: shows top queried names, response codes.
- **Statistics → Protocol Hierarchy**: see volume of DNS traffic.
- **Statistics → Conversations**: select UDP/TCP to see endpoints and unusual partners.
- **Statistics → IO Graphs**: plot DNS query/response rates over time.
- **Analyze → Expert Information**: highlights malformed or suspicious packets.

These views help detect spikes in DNS responses or many responses from unexpected IPs.

Section 5: Detecting DNS Anomalies

Common Anomalies

- **TXID Mismatch** – Response TXID does not match the query.
- **Suspicious Source Ports** – Responses coming from non-standard source ports.

- **Suspicious Source IPs** – Responses coming from non-standard source IP addresses.
- **Early Responses** – Responses received before the query (often indicative of a man-in-the-middle).
- **DNS Poisoning** – Fake DNS responses redirecting traffic to malicious IP addresses.

Understanding DNS Poisoning Attacks

- Attackers can forge DNS responses and inject them into a victim's cache.
- Characteristics of a DNS poisoning attack:
    - Fake IP address for legitimate domains.
    - TXID mismatch or unusual source IP.
    - Often combined with ARP spoofing or MitM techniques.

Example Scenario

- User queries `www.bank.com`.
- Attacker sends a forged response with TXID mismatch, pointing to a malicious IP.
- User unknowingly visits the fake site.

Ethics & safety (must read)

Never attempt poisoning or packet spoofing on networks you do not own or explicitly have written permission to test. Performing DNS spoofing on public networks is illegal and harmful.

Section 6: Signs of DNS Poisoning or Spoofing in Captures

Use this checklist to flag suspicious behavior:

- Unexpected authoritative IP.

- Mismatched TXID or unsolicited responses.

- Responses arriving before or nearly simultaneous with queries.

- Multiple conflicting responses to the same query with different answer IPs or TTLs.

- Unusually long/short TTLs.

- Extra records in additional/authority sections.

- High volume of NXDOMAIN/SERVFAIL responses.

- Lack of DNSSEC validation (missing AD flag or invalid RRSIG).

- Responses from many different source IPs for the same query.

- Packet anomalies: malformed headers or inconsistent lengths.

Section 7: Lab Exercise

7-1: Identify the Anomalies

1. Open the provided anomalous DNS packet captures (pcap files in ex1 folder) in Wireshark.
2. Analyze each packet.
3. Identify the type of anomaly in each case.

For each file, fill in the table:

Packet No	Type of Anomaly	Evidence in Packet	Suggested Mitigation
1			
2			
3			
4			
5			

7-2: Inspect DNS traffic with filters

You are provided with a **sample packet capture** (dns_filters.pcap in ex2 folder) simulating DNS traffic:

Use these **display filters** in Wireshark to inspect DNS traffic (Section 3) and answer the following questions:

1. Using dns, how many DNS packets are in the capture?

2. Apply dns.flags.response == 0. Which domains were queried?

3. Apply dns.flags.response == 1. What were the responses?

4. Look at the TXID values. Do all queries and responses match correctly?

5. Use udp.srcport != 53 && dns. Which packets appear suspicious? Why?

6. Which response looks like a **DNS poisoning attempt**? Explain.

7. How could DNSSEC prevent this type of attack?

Deliverables

Submit the screenshots and the your answers in a word file. *You have to upload your work in the following format:*

- ***Filename: Lab2-FirstName-Lastname-StdNo.PDF***