Cloud and Network Security-C1-2026

Student Name: **Felix Webbo**

Student No: **CS-CNS11-26044**

SUNDAY, FEB 01, 2026

Week 3: Assignment 1

Class Exercise: **TryHackMe: DNS In Detail**

1 ABSTRACT

This report documented the successful completion of the *DNS in Detail* module on the TryHackMe learning platform. The objective of the study was to examine the structure, functionality, and security relevance of the Domain Name System (DNS). The module explored DNS fundamentals, domain hierarchy, record types, and the process of DNS resolution. Through guided explanations and practical tasks, foundational DNS concepts were reinforced to support cybersecurity analysis and network defense. The report presented findings, methodology, and reflections based on the completed module.

Table of Contents

2 INTRODUCTION

The Domain Name System (DNS) played a critical role in modern networking by translating human-readable domain names into machine-readable IP addresses. This assignment focused on understanding how DNS functioned, how domain names were structured, and how DNS queries were processed across multiple servers. The *DNS in Detail* module was completed to strengthen technical knowledge relevant to cybersecurity operations, including traffic analysis, threat detection, and secure network design. Understanding DNS was essential for identifying misconfigurations, malicious redirection, and DNS-based attacks.

3 METHODOLOGY

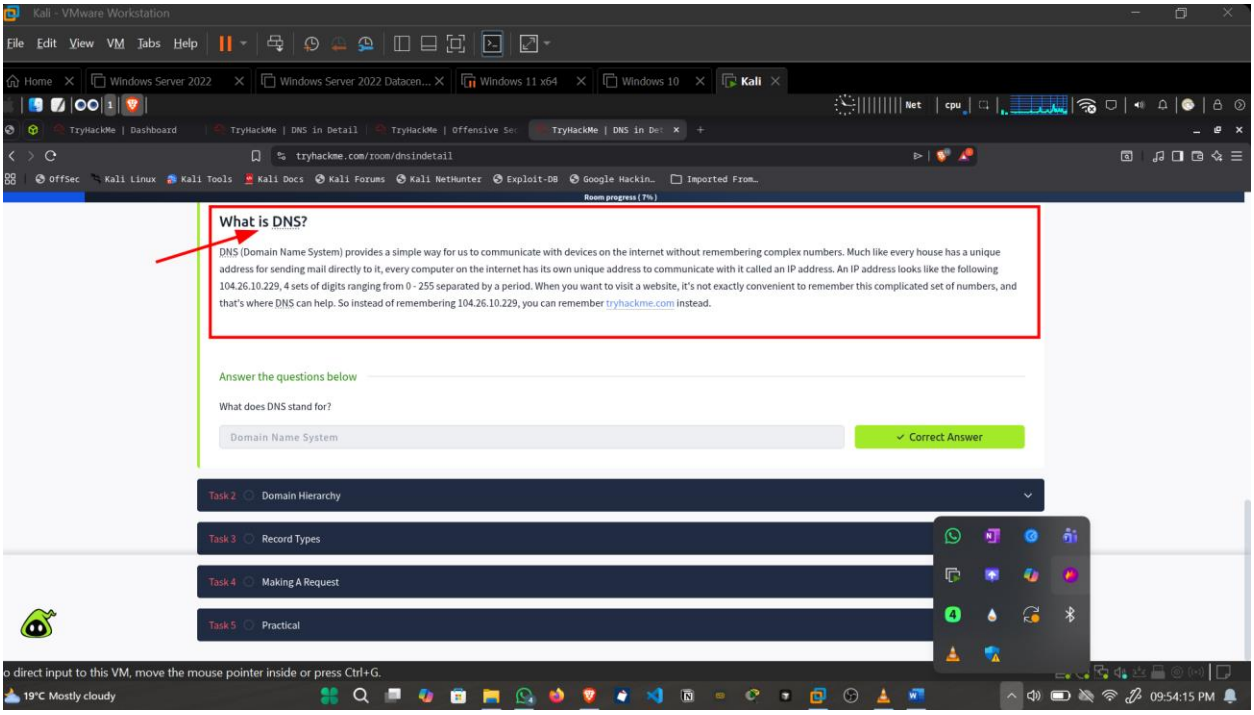
The module was completed using the TryHackMe online learning platform. Each section was studied sequentially, and embedded questions were answered based on the instructional content provided. Conceptual explanations were reinforced through diagrams and practical examples. Screenshots were captured to document task completion and verify learning outcomes. The final module completion was shared using a generated public link and posted on social media to demonstrate continuous professional development.

4 IMPORTANCE OF DNS

DNS served as a foundational service for internet communication and enterprise networks. Without DNS, users would be required to remember IP addresses instead of domain names, significantly reducing usability. From a cybersecurity perspective, DNS was often targeted by attackers through techniques such as DNS spoofing, cache poisoning, tunneling, and malicious redirection. A solid understanding of DNS behavior was therefore essential for detecting anomalies, securing infrastructure, and responding to incidents.

5 WHAT IS DNS?

DNS (Domain Name System) provides a simple way for us to communicate with devices on the internet without remembering complex numbers. Much like every house has a unique address for sending mail directly to it, every computer on the internet has its own unique address to communicate with it called an IP address. An IP address looks like the following 104.26.10.229, 4 sets of digits ranging from 0 - 255 separated by a period. When you want to visit a website, it's not exactly convenient to remember this complicated set of numbers, and that's where DNS can help. So instead of remembering 104.26.10.229, you can remember tryhackme.com instead.



6 DOMAIN HIERARCHY

6.1 TLD (Top-Level Domain)

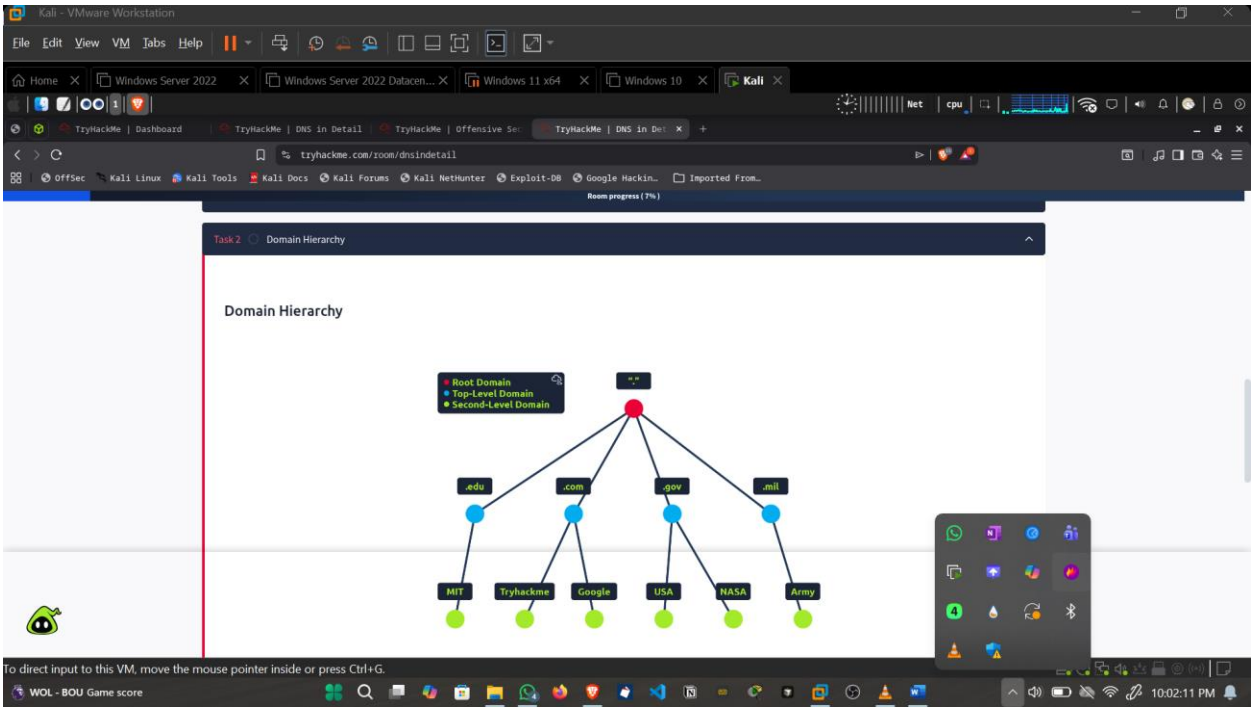
A TLD is the most righthand part of a domain name. So, for example, the tryhackme.com TLD is .com. There are two types of TLD, gTLD (Generic Top Level) and ccTLD (Country Code Top Level Domain). Historically a gTLD was meant to tell the user the domain name's purpose; for example, a .com would be for commercial purposes, .org for an organisation, .edu for education and .gov for government. And a ccTLD was used for geographical purposes, for example, .ca for sites based in Canada, .co.uk for sites based in the United Kingdom and so on. Due to such demand, there is an influx of new gTLDs ranging from .online , .club , .website , .biz and so many more. For a full list of over 2000 TLDs click here.

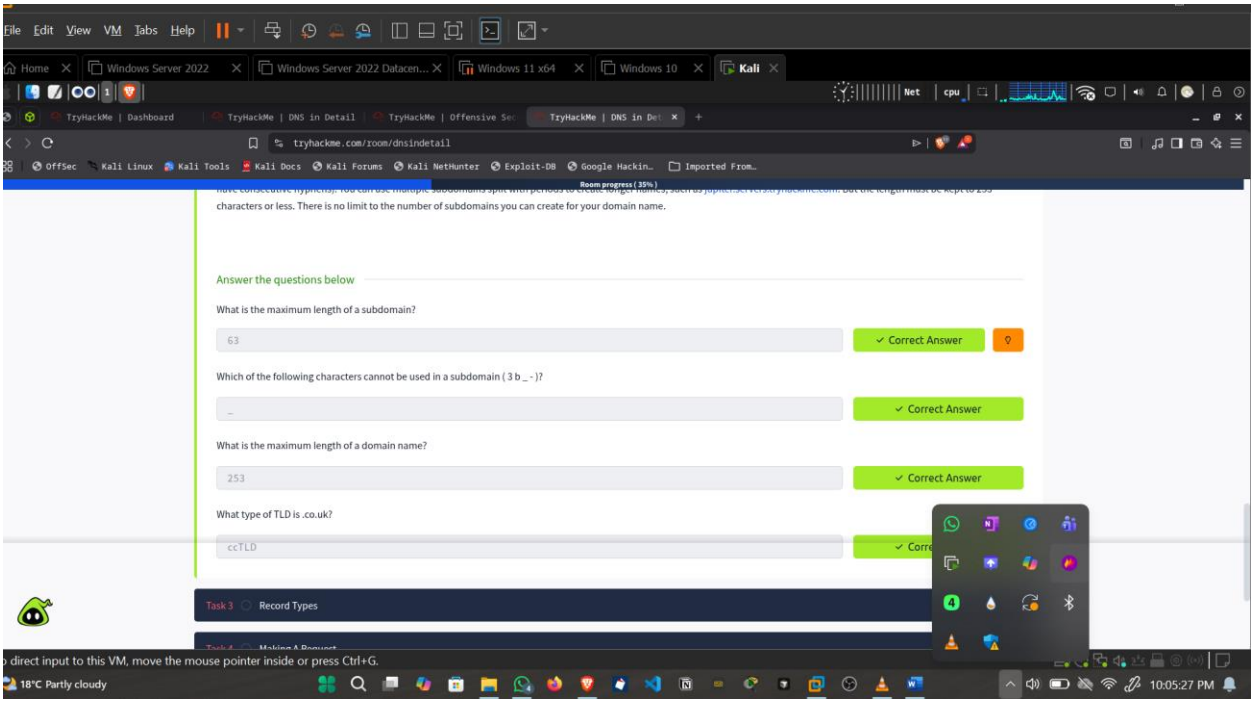
6.2 Second-Level Domain

Taking tryhackme.com as an example, the .com part is the TLD, and tryhackme is the Second Level Domain. When registering a domain name, the second-level domain is limited to 63 characters + the TLD and can only use a-z 0-9 and hyphens (cannot start or end with hyphens or have consecutive hyphens).

6.3 Subdomain

A subdomain sits on the left-hand side of the Second-Level Domain using a period to separate it; for example, in the name admin.tryhackme.com the admin part is the subdomain. A subdomain name has the same creation restrictions as a Second-Level Domain, being limited to 63 characters and can only use a-z 0-9 and hyphens (cannot start or end with hyphens or have consecutive hyphens). You can use multiple subdomains split with periods to create longer names, such as jupiter.servers.tryhackme.com. But the length must be kept to 253 characters or less. There is no limit to the number of subdomains you can create for your domain name.





7 DNS RECORD TYPES

The Domain Name System supported multiple record types beyond website resolution, each serving a specific function within network and service operations.

7.1 A Records

Were used to resolve domain names to IPv4 addresses. for example, 104.26.10.229

7.2 AAAA records

Performed the same function for IPv6 addresses. for example, 2606:4700:20::681a:be5

7.3 CNAME records

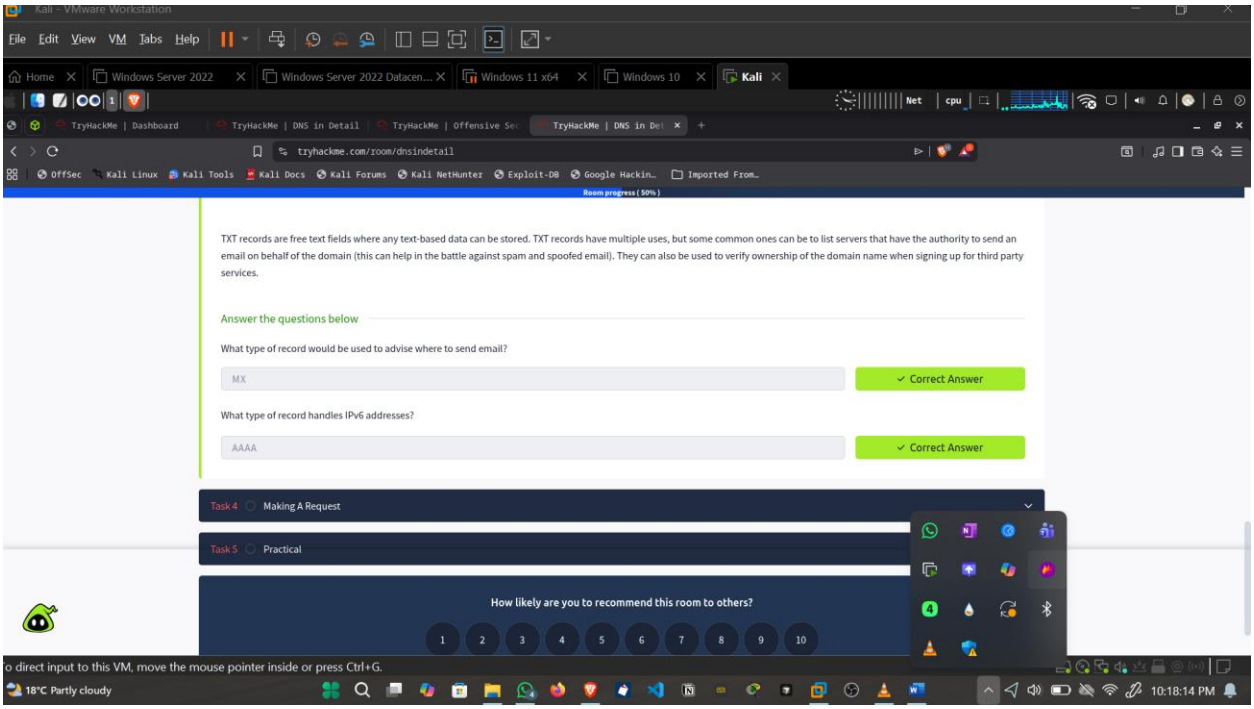
Mapped one domain name to another canonical domain, requiring an additional DNS query to resolve the final IP address. for example, TryHackMe's online shop has the subdomain name store.tryhackme.com which returns a CNAME record shops.shopify.com.

7.4 MX records

Identified mail servers responsible for handling email for a domain and included priority values to ensure redundancy and failover in email delivery. for example an MX record response for tryhackme.com would look something like alt1.aspmx.l.google.com.

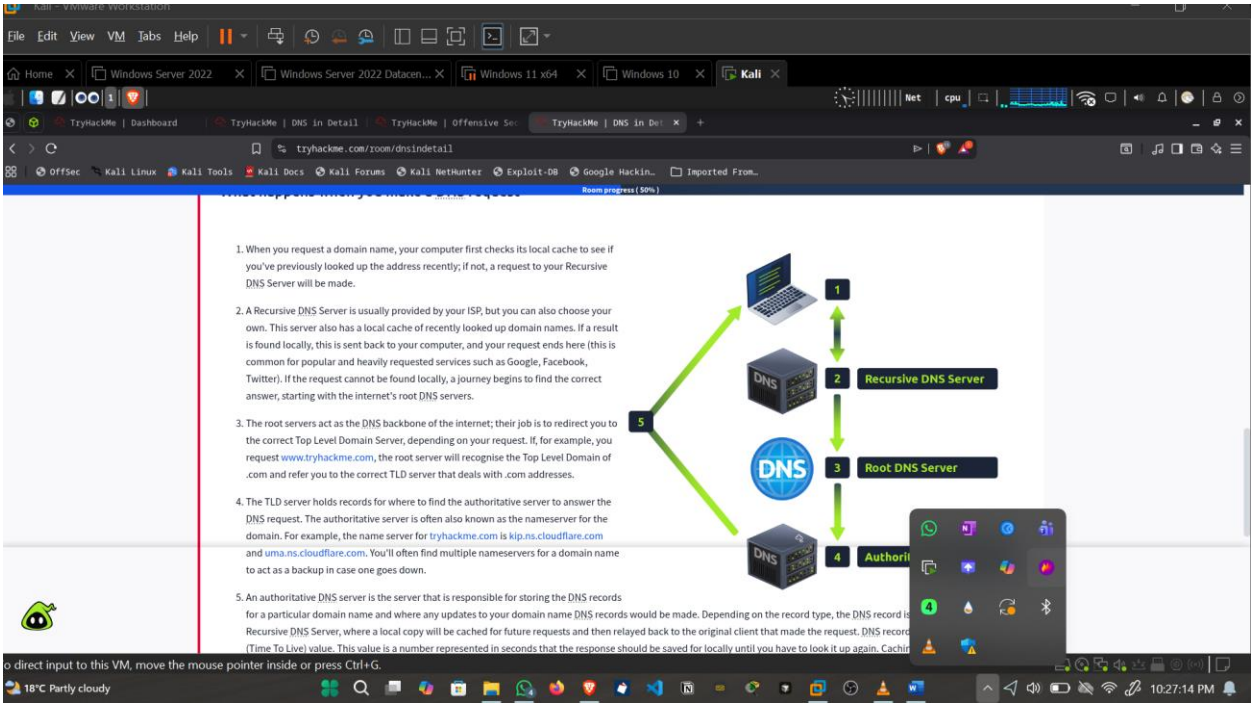
7.5 TXT records

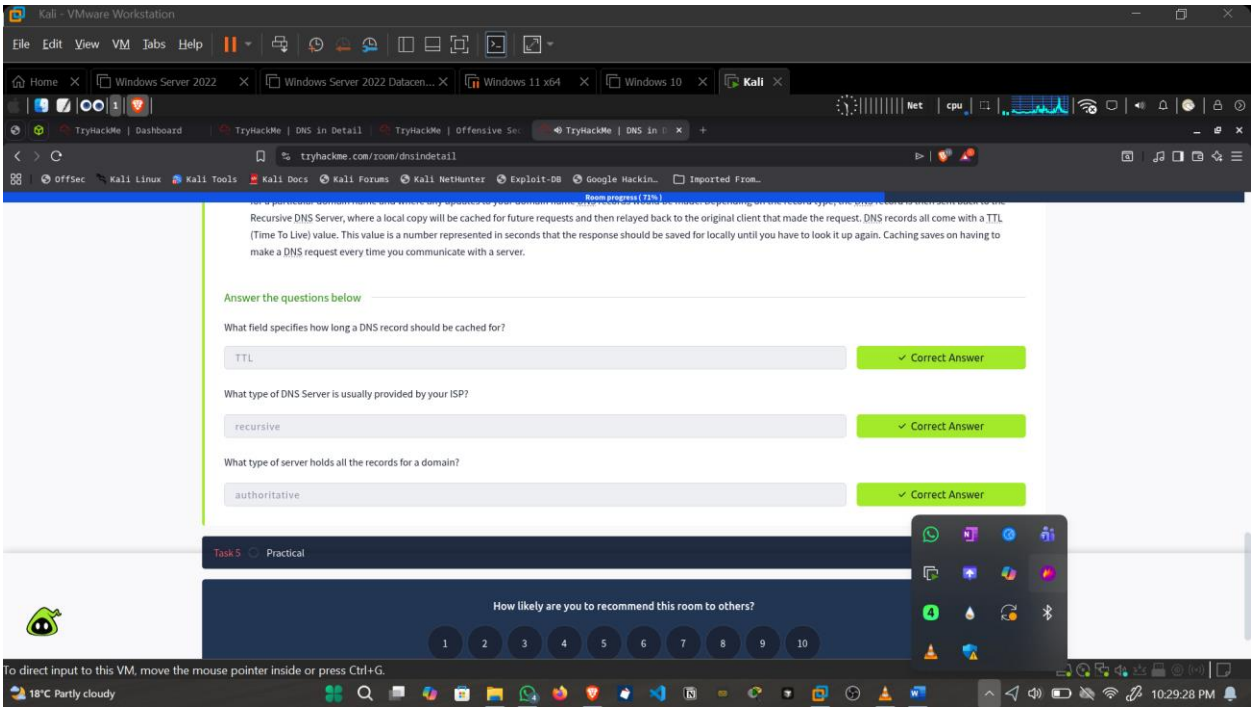
Stored text-based information and were commonly used for domain ownership verification and email security controls such as sender authorization, helping reduce spam and spoofing.

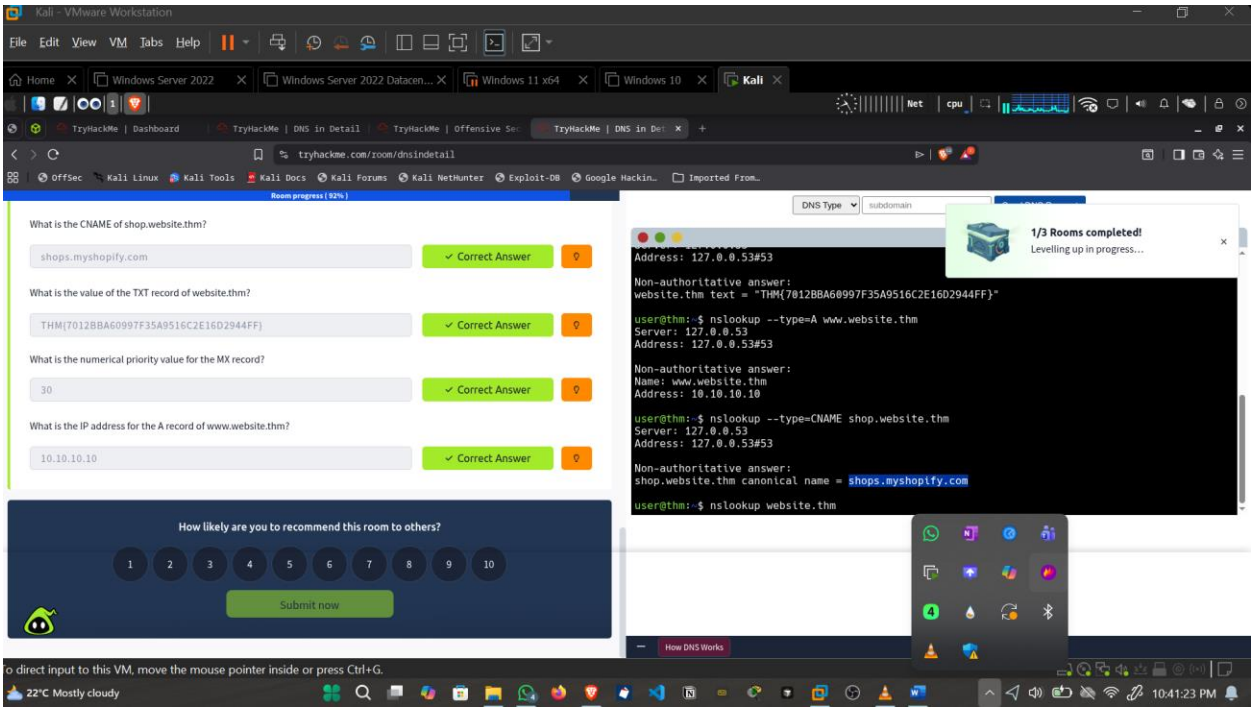


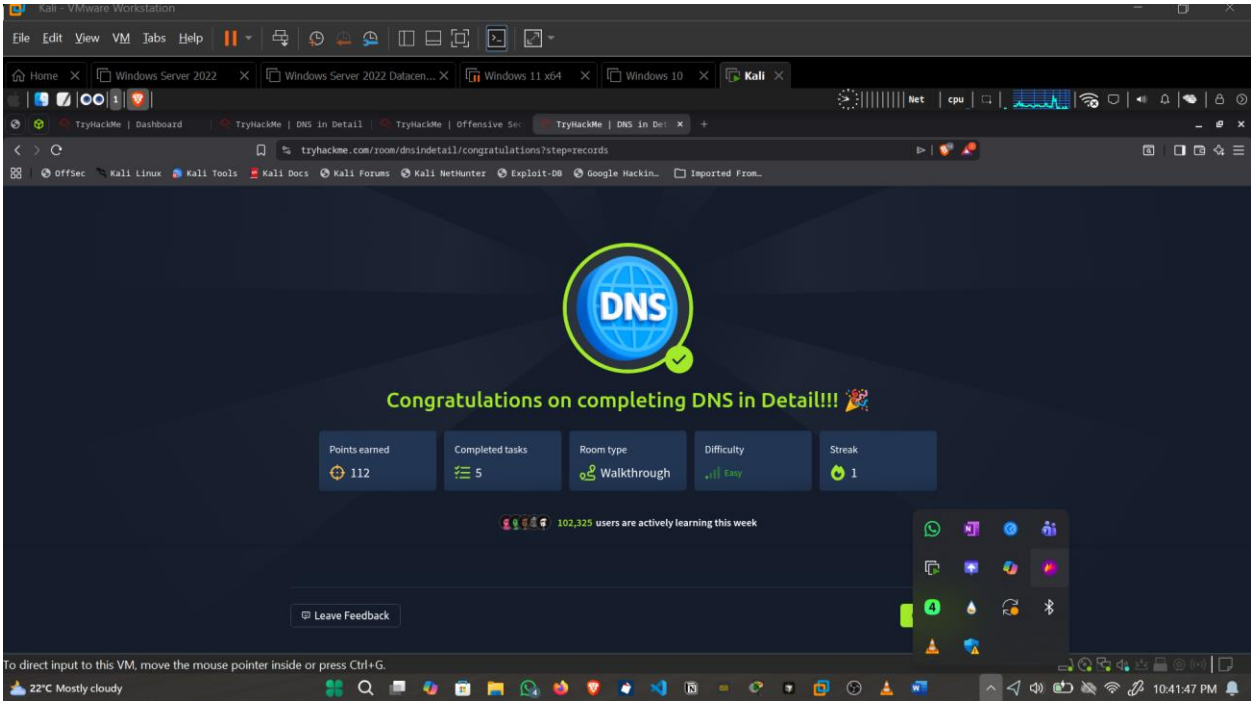
8 MAKING A DNS REQUEST

When a DNS request was made, the client first checked its local cache for a recent record. If no entry was found, the request was sent to a Recursive DNS Server, which also checked its cache. If the record was unavailable, the recursive server queried the Root DNS Servers, which directed it to the appropriate Top-Level Domain (TLD) server. The TLD server then identified the Authoritative DNS Server responsible for the domain. The authoritative server returned the requested DNS record, which was cached by the recursive server and sent back to the client.


1. When you request a domain name, your computer first checks its local cache to see if you've previously looked up the address recently; if not, a request to your Recursive DNS Server will be made.
2. A Recursive DNS Server is usually provided by your ISP, but you can also choose your own. This server also has a local cache of recently looked up domain names. If a result is found locally, this is sent back to your computer, and your request ends here (this is common for popular and heavily requested services such as Google, Facebook, Twitter). If the request cannot be found locally, a journey begins to find the correct answer, starting with the internet's root DNS servers.
3. The root servers act as the DNS backbone of the internet; their job is to redirect you to the correct Top Level Domain Server, depending on your request. If, for example, you request www.tryhackme.com, the root server will recognise the Top Level Domain of .com and refer you to the correct TLD server that deals with .com addresses.
4. The TLD server holds records for where to find the authoritative server to answer the DNS request. The authoritative server is often also known as the nameserver for the domain. For example, the name server for tryhackme.com is kip.ns.cloudflare.com and uma.ns.cloudflare.com. You'll often find multiple nameservers for a domain name to act as a backup in case one goes down.
5. An authoritative DNS server is the server that is responsible for storing the DNS records for a particular domain name and where any updates to your domain name DNS records would be made. Depending on the record type, the DNS record is
Recursive DNS Server
Root DNS Server
Authori
Recursive DNS Server, where a local copy will be cached for future requests and then relayed back to the original client that made the request. DNS record
(Time To Live) value. This value is a number represented in seconds that the response should be saved for locally until you have to look it up again. Cachi
o direct input to this VM, move the mouse pointer inside or press Ctrl+G.
18°C Partly cloudy
10:27:14 PM


Recursive DNS Server, where a local copy will be cached for future requests and then relayed back to the original client that made the request. DNS records all come with a TTL (Time To Live) value. This value is a number represented in seconds that the response should be saved for locally until you have to look it up again. Caching saves on having to make a DNS request every time you communicate with a server.
Answer the questions below
What field specifies how long a DNS record should be cached for?
TTL
Correct Answer
What type of DNS Server is usually provided by your ISP?
recursive
Correct Answer
What type of server holds all the records for a domain?
authoritative
Correct Answer
Task 5
Practical
How likely are you to recommend this room to others?
1 2 3 4 5 6 7 8 9 10
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
18°C Partly cloudy
10:29:28 PM


What is the CNAME of shop.website.thm?
shops.myshopify.com
Correct Answer
What is the value of the TXT record of website.thm?
THM{7012BBA60997F35A9516C2E16D2944FF}
Correct Answer
What is the numerical priority value for the MX record?
30
Correct Answer
What is the IP address for the A record of www.website.thm?
10.10.10.10
Correct Answer
How likely are you to recommend this room to others?
1 2 3 4 5 6 7 8 9 10
Submit now
1/3 Rooms completed!
Levelling up in progress...
Address: 127.0.0.53#53
Non-authoritative answer:
website.thm text = "THM{7012BBA60997F35A9516C2E16D2944FF}"
user@thm:~$ nslookup --type=A www.website.thm
Server: 127.0.0.53
Address: 127.0.0.53#53
Non-authoritative answer:
Name: www.website.thm
Address: 10.10.10.10
user@thm:~$ nslookup --type=CNAME shop.website.thm
Server: 127.0.0.53
Address: 127.0.0.53#53
Non-authoritative answer:
shop.website.thm canonical name = shops.myshopify.com
user@thm:~$ nslookup website.thm
How DNS Works
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.


Congratulations on completing DNS in Detail!!! 🎉
Points earned
112
Completed tasks
5
Room type
Walkthrough
Difficulty
Easy
Streak
1
102,325 users are actively learning this week
Leave Feedback
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Click Here

9 CONCLUSION

This study strengthened the understanding of how the Domain Name System operates and its importance within modern networks. By examining DNS structure, record types, and the resolution process, the module demonstrated how domain requests were efficiently translated into IP addresses. The exercise also highlighted the security relevance of DNS, particularly how misconfigurations or abuse could enable attacks such as redirection or data exfiltration. Overall, the module provided a solid foundation for applying DNS knowledge in cybersecurity analysis, network monitoring, and defensive security practices.

10 REFERENCES

1. TryHackMe. (n.d.). *DNS in Detail*. TryHackMe Learning Platform.
2. Cyber Shujaa LLM