Subdomain Enumeration

Overview

Room URL: https://tryhackme.com/room/subdomainenumeration

Difficulty: Easy

Category: Reconnaissance/Subdomain Enumeration

Date Completed: 01/21/2026

Objective

Learn and practice three different subdomain enumeration methods to expand the attack surface and discover potential points of vulnerability: Certificate Transparency logs, DNS Brute Force, OSINT tools, and Virtual Host enumeration.

Table of Contents

Introduction

What is Subdomain Enumeration?

Subdomain enumeration is the process of finding valid subdomains for a domain. The purpose is to expand our attack surface to discover more potential points of vulnerability that may not be immediately visible from the main domain.

Why Subdomain Enumeration Matters

Organizations often have multiple subdomains for different purposes:

- Development and staging environments
- Administrative portals
- API endpoints
- Regional or departmental sites
- Legacy applications

These subdomains may have different security configurations, and some may be less secured than the main domain, making them valuable targets for security testing.

Method 1: Certificate Transparency (CT) Logs

What are CT Logs?

When an SSL/TLS (Secure Sockets Layer/Transport Layer Security) certificate is created for a domain by a CA (Certificate Authority), CAs participate in what's called "Certificate Transparency logs". These are publicly accessible logs of every SSL/TLS certificate created for a domain name.

Purpose:

The purpose of Certificate Transparency logs is to stop malicious and accidentally made certificates from being used. We can use this service to discover subdomains belonging to a domain.

Tool:

Sites like https://crt.sh offer a searchable database of certificates that shows current and historical results.

Method 2: DNS Brute Force Enumeration

What is it?

Brute force DNS enumeration is the method of trying tens, hundreds, thousands, or even millions of different possible subdomains from a pre-defined list of commonly used subdomains.

Why Automate?

Because this method requires many requests, we automate it with tools to make the process quicker.

Common Tools:

- `dnsrecon` - DNS reconnaissance tool
- `dnsenum` - DNS enumeration tool
- `fierce` - DNS scanner

How it Works:

The tool takes a wordlist of common subdomain names (e.g., www, mail, ftp, admin, api, dev) and tests each one to see if it resolves to an IP address.

Method 3: OSINT - Automated Discovery

What is OSINT?

Open-Source Intelligence (OSINT) refers to using publicly available information sources to gather intelligence about a target.

Tools:

Tools like Sublist3r automate the OSINT subdomain discovery process by:

- Searching multiple search engines (Google, Bing, Yahoo)
- Querying DNS databases
- Using certificate transparency logs
- Checking web archives
- Searching threat intelligence platforms

This approach speeds up discovery by aggregating results from multiple sources automatically.

Method 4: Virtual Host Enumeration

The Concept:

Some subdomains aren't always hosted in publicly accessible DNS results. These might include:

- Development versions of web applications
- Administration portals
- Internal testing environments

Where These Records Live:

- Private DNS servers
- Local `/etc/hosts` file (Linux/Mac)
- `c:\windows\system32\drivers\etc\hosts` file (Windows)

How Web Servers Handle Multiple Sites:

Web servers can host multiple websites from one server. When a website is requested from a client, the server knows which website the client wants from the **Host header**.

The Attack Method:

We can utilize this Host header by making changes to it and monitoring the response to see if we've discovered a new website. Like with DNS brute force, we automate this process using a wordlist of commonly used subdomains.

Tool: ffuf (Fuzz Faster U Fool)

`ffuf` is a fast web fuzzer that can be used for virtual host discovery by fuzzing the Host header.

Basic Syntax:

```
ffuf -w [wordlist] -H "Host: FUZZ.[domain]" -u http://[IP]
```

Key Flags:

- `-w`: Specifies the wordlist to use
- `-H`: Adds/edits a header (in this case, the Host header)
- `FUZZ`: Keyword that will be replaced with each word from the wordlist
- `-u`: Target URL
- `-fs`: Filter by size - tells ffuf to ignore results of a specified size

Why Filtering is Important:

The command will always produce a result (even for non-existent subdomains) because the web server responds. We need to filter out false positives by excluding the most common response size.

Walk Through

Task 1: Certificate Transparency Logs

1. Navigate to https://crt.sh
2. Search for `tryhackme.com`
3. Look through the results for entries logged on `2020-12-26`
4. Identify the subdomain

Answer: `store.tryhackme.com`

Task 2: DNS Brute Force with dnsrecon

1. Click the "View Site" button on the TryHackMe page
2. Press the "Run DNSrecon Request" button to start the simulation
3. Observe the output from the dnsrecon tool
4. Identify the first subdomain found

Answer: `api.acmeitsupport.thm`

Task 3: OSINT with Sublist3r

1. Click the "View Site" button on the TryHackMe page
2. Run the sublist3r simulation
3. Review the discovered subdomains
4. Identify the first subdomain found

Answer: `web55.acmeitsupport.thm`

Task 4: Virtual Host Enumeration with ffuf

Step 1: Initial Scan (Unfiltered)

```
ffuf -w /usr/share/wordlists/SecLists/Discovery/DNS/namelist.txt -H "Host:
FUZZ.acmeitsupport.thm" -u http://MACHINE_IP
```

- This command will return many results
- Note the most common response size value

Step 2: Filtered Scan

```
ffuf -w /usr/share/wordlists/SecLists/Discovery/DNS/namelist.txt -H "Host:
FUZZ.acmeitsupport.thm" -u http://MACHINE_IP -fs {size}
```

- Replace `{size}` with the most common size from the previous results
- This filters out false positives
- The filtered results reveal two new subdomains

Answers:

First subdomain discovered: `delta`

Second subdomain discovered: `yellow`

Lessons Learned

- Certificate Transparency logs are publicly accessible and provide historical SSL/TLS certificate data useful for subdomain discovery
- DNS brute forcing automates the process of testing thousands of potential subdomains from wordlists
- OSINT tools like Sublist3r aggregate data from multiple sources simultaneously for comprehensive results
- Virtual host enumeration can discover subdomains not listed in public DNS records
- Filtering false positives (using `-fs` in ffuf) is essential when fuzzing to identify legitimate results
- Multiple enumeration techniques should be combined for comprehensive subdomain discovery
- Hidden subdomains (dev environments, admin panels) may only be accessible through virtual host enumeration
- Pre-defined wordlists (like SecLists) are crucial for effective brute force enumeration

---

Resources

TryHackMe

crt.sh - Certificate Transparency Search

SecLists - Security Testing Wordlists

ffuf - Fast Web Fuzzer

[dnsrecon - DNS Enumeration Tool]

[Sublist3r - Subdomain Enumeration Tool]

[OWASP - Subdomain Enumeration]

Revision #2

Created 2026-01-21 22:59:36 UTC by David Rizzo

Updated 2026-01-21 23:13:21 UTC by David Rizzo