Finding the content in CDNs

EIIN903-Multimedia Networking
UBINET Master 2024-25
Université Côte d'Azur
Instructor : Ramón APARICIO (Ramon.APARICIO-PARDO@univ-cotedazur.fr)

6 December 2024

1 Objective

In this laboratory, we are going to study how CDNs direct clients to a particular content in a CDN server.

2 Rules

You can work in groups of 1 or 2 students.

You have to prepare a written report (pdf) with the answers of the lab and upload it to the drop box "Report - Lab 1 : CDN" in the moodle course by December 16th at 23 :45.

3 Before the lab

In this lab, we will work with `Wireshark` sniffer, a free and open source packet analyser. You can install it in any Operating System from `https://www.wireshark.org/download.html`.

4 Finding a Web content

In this section, we are going to review the process to access to a web content hosted by a CDN server, particularly, we will study the DNS exchanges allowing to find the right server hosting the page. To do that, we will analyse using the `Wireshark` tool the sequence of DNS requests/responses exchanged when we open a web page.

4.1 Non-CDN case

Traditionally, web content was hosted in web servers directly administrated by the owner of the content (the Content Provider, CP) without resorting to a CDN. For example, the site of the I3S research laboratory at Université de Nice Sophia Antipolis (UNS) `http://www.i3s.unice.fr` is hosted by a machine physically located at the University premises and administrated by the University staff. We will use the I3S web page to review how DNS finds a web server when no CDN entity is involved.

First, we will find the IP address of the I3S web server, which will allow to find the geographical location of the machine.

Open a web browser without typing any URL. Then launch a `Wireshark` capture (`Start` button). Maybe, you will need to define some settings in `Capture` → `Options` : untick

Promiscuous mode and Resolve network names. Afterwards, enter www.i3s.unice.fr in the web browser, then stop the capture. Identify the packets corresponding to the DNS exchange started up by your web request by using the filter dns (the display filter is below the control panel and above the display). Search for the DNS packet response to the A type Resource Record (RR) query of www.i3s.unice.fr (i.e., the response of the name resolution of the domain name www.i3s.unice.fr). Double click on the packet and search first for the Domain Name System response part, and, then the Answers section of the DNS response. You will find there the searched IP address.

> *NOTE : If you are using a MAC or LINUX machine, instead using Wireshark, you can simply open a terminal window and type the UNIX command dig to send DNS requests without opening a webpage. You can use this online manual about command dig (see https://linux.die.net/man/1/dig). In our case, to generate the same DNS exchange as the one in the capture, you must type dig A www.i3s.unice.fr*

Question 1 : What is the IP address of the I3S web server ? .

Now, we will find the physical location of the I3S web server by entering its IP address in http://whois.domaintools.com and https://www.iplocation.net/.

Question 2 : From the information returned by these two web pages, where is the I3S web server and who manages it ? .

Finally, we will revisit the whole sequence of DNS exchanges that allowed to find the IP address of the I3S site. To do that, we cannot use a Wireshark capture. We need to make use of the UNIX command dig. In case that your computer is a Windows machine, you can employ the site http://www.digwebinterface.com, that simply provides a web interface to call the dig command. If you use LINUX or MAC, you can directly type the command on your terminal as before. Now, we use the command with the option +trace to "walk the tree," i.e., to make iterative queries to find the address (name resolution) following referrals from the root servers. The command returns the answer from each server that was used to resolve the lookup. If you are using the site http://www.digwebinterface.com, you have to tick the option trace. If you use a Unix-based system, simply add the +trace option to the command invocation.

> *NOTE : If you are using a MAC or LINUX machine, you will probably need to explicitly select a name server to query to force the iterative queries. A good option is to request to a Google Public DNS server (i.e., 8.8.8.8 or 8.8.4.4, see https://developers.google.com/speed/public-dns/). You can provide the Google Public DNS server by adding @8.8.4.4 to the command call.*

Question 3 : List the DNS servers (among those ones suggested in the DNS responses) that actually replied to the iterative queries.

> *NOTE : These name servers are usually identified in terms of IP address (v4 or v6) at the line starting from ;; Received .... Then, you will need to send a reverse DNS lookup to find the corresponding domain name : either select the Reverse type at http://www.digwebinterface.com or add the option -x to command dig (dig -x ...)*

4.2 CDN case

Now, we are going to repeat the previous study but with a web site hosted by a CDN. Practically, any commercial web site makes recourse to professional CDN solutions. For this activity, we will use the Huawei web site https://www.huawei.com, one of the main telecommunications equipment company, that of course, trust in a CDN (in that case, Akamai) to cache its web sites.

Then, again, we will find the IP address of the Huawei web server, which will allow to find its geographical location. Open the web site of www.huawei.com and capture the DNS exchange started up by the web request by means of `Wireshark` sniffer. Search for the response of the name resolution `www.huawei.com` : a response to an `A` type Resource Record (RR) query for `www.huawei.com`. Check as well if there is an IPv6 resolution (`AAAA` type RR) query. Analyse the DNS response and find the searched IPv4 (or IPv6) in the `Answers` section.

> *NOTE : If you are using a MAC or LINUX machine, instead using `Wireshark`, you can simply open a terminal window and type the UNIX command `dig A www.huawei.com` or `dig AAAA www.huawei.com`*

Question 4 : What is the IP address of of the Huawei web server ? .

Since most part of HTTP exchanges are encrypted, you cannot analyse the corresponding HTTP exchanges with `Wireshark`. Then, you will be forced to use the tools of the Mozilla Firefox browser. Go to `Firefox` → `More Tools` → `Web Developer Tools` → `Network Tab`, and re-type the `www.huawei.com` in the browser. On the `Network Tab`, look for the first successful HTTP GET request (code `200 OK`). Inspect the `Headers Tab` at the left panel and search for the involved IP(v4 or v6) address. This address should match the result of the previous question.

Then, we will find the physical location of the Huawei web server by entering its IP address in `http://whois.domaintools.com` and `https://www.iplocation.net/`.

Question 5 : From the information returned by these two web pages, where is the Huawei web server and who manages it ? If you have alternate possible locations, `ping` the returned IP address. If RTT times overcome 100 ms, the server is not in Europe. .

Now, search for the IP address of `www.huawei.com` by using the web site `www.whatsmydns.net` (you enter here `www.huawei.com`) and enter some of them in `http://whois.domaintools.com` and `https://www.iplocation.net/` to find the physical location and administrator of these IP addresses.

Question 6 : Who manages these IP addresses and what is the connection with the address you found at Question 4 ? Why are they different ? .

Finally, we will study again the sequence of DNS exchanges that in that case allows to find the IP address of the Huawei site. To do that, we will employ again the UNIX command `dig`. If you are using the site `http://www.digwebinterface.com`, you have to tick the option `trace`. If you use a Unix-based system, simply add the `+trace` to the command invocation. In that case, remember to add the `@8.8.4.4` to query a Google Public DNS server.

Question 7 : List the DNS servers (among those ones suggested in the DNS responses) that actually replied to the iterative queries.

> *NOTE : These name servers are usually identified in terms of IP address (v4 or v6) at the line starting from `;; Received ...`. Then, maybe, you will need to send a reverse DNS lookup to find the corresponding domain name : either select the `Reverse` type at `http://www.digwebinterface.com` or add the option `-x` to command `dig` (`dig -x ...`)*

Question 8 : What is the "canonical" name (i.e., the "real" name) of the Huawei server returned in the `CNAME` RR ?

Come back to the `Wireshark` capture, and, find the `CNAME` RRs at the `Answers` section of the same DNS response packet as before. Alternatively, you can obtain the same information if you type at the UNIX command terminal `dig www.huawei.com`. You will obtain a nested sequence of "canonical" name RRs. The last RR points out to an IPv4 (or IPv6) address.

Question 9 : What is the last "canonical" name ?

Now, you will see how the nested sequence of "canonical" name RRs was actually built. To do that, you will resolve the list of canonical names one by one by using again the UNIX command `dig` with the option `trace` and querying a Google Public DNS server (remember you can also use the `http://www.digwebinterface.com`). You will be forced to repeat this procedure several times till finding the IP address.

Question 10 : In all these queries, who is the last nameserver replying (see the last line starting from `;; Received ....`) ? Does it make sense ? Note : you can check the end of the Unit 1 about CDNs.

5 Finding a video content

In this section, we are going to study how a video stream is delivered from YouTube and we will compare the process with the Akamai method to find the server caching the content.

First, turn off the volume of your computer. Open a `Firefox` browser window and search on YouTube any clip or music group (popular enough, ex : *"With or Without You"* by *U2*). DO NOT CLICK. Before clicking the video, launch a `Wireshark` capture. Then, click on the video. If an advertisement video starts up, let it finish. Let also the video you requested play during ca. 30 seconds. After that, stop the `Wireshark` capture and pause the video.

In the `Firefox` window where the video was played, inspect the source code (click on left mouse button and select *View Page Source* or select *Inspect element*). In the `html` code, search for an expression with this format : `rX---sn-YYYYYYYY.googlevideo.com`. For example, for the official video of *'With or Without You"* by *U2* (whose URL is `https://www.youtube.com/watch?v=ujNeHIo7oTE`), this expression was `rr3---sn-4gxx-25gy.googlevideo.com` on Dec. 7th 2023 from a home connection. This is the domain name of the youtube server hosting this video. If you use an Internet connection provided by the University, the domain name may be different.

Question 11 : What is the IP address of this youtube server ? Note : you can Use the UNIX command `dig` as before (or the webpage `www.digwebinterface.com`) to resolve the name.

Let's come back to the `Wireshark` tool. Find the main TCP (or UDP, if QUICK protocole used.) sessions in terms of traffic. To do that, open `Statistics` → `Conversations` and sort the TCP sessions by `Bytes`. Normally, one of the columns (`Address A` or `Address B`) is your local IP address and the other column is the IP address of the remote server. (NOTE : To know which transport protocol is carrying the most part of traffic, check `Statistics` → `Protocol Hierarchy`).

Again, you can also use the tools of the Mozilla Firefox browser to find this IP (v4 or v6) address. Go to `Firefox` → `More Tools` → `Web Developer Tools` → `Network Tab`, and retype the url in the browser. On the `Network Tab`, look for a HTTP GET (or POST) request targeted to the `rX---sn-YYYYYYYY.googlevideo.com` server (typically the downloaded content type is `mp4` or `vnd.yt-ump`[1]). Inspect the `Headers Tab` at the left panel and search for the involved IP(v4 or v6) address. This address should match the result of the previous question.

Question 12 : What is the IP address of the remote server for the session ranked in first place ? Is the same address as in Question 11 ? Why ?

Try to confirm your response to Question 12 by finding the actual DNS exchange in the `Wireshark` capture.

Question 13 : Do you find in the `Answer` section of the DNS query response the same information as the return of the command `dig` in Question 11 ?

[1] `https://www.iana.org/assignments/media-types/video/vnd.youtube.yt`

NOTE : This DNS exchange could not appear at `Wireshark` capture depending on the status of several caches (local caches at your machine, proxy caches of your ISP, ..). In such a case, the DNS exchange did not take place because the name resolution information was already cached.

Take the IP(v4 or v6) address found at the preceding questions and look it up on `www.whois.domaintools.com` and on `www.iplocation.net` to find its physical location and its administrator.

Question 14 : Where is it located the server using this IP ?

Now `ping` this address.

Question 15 : Are the RTT obtained consistent with the response to Question 14 ?

Now enter the server name corresponding to this IP address in `www.whatsmydns.net`.

Question 16 : Does the result make any sense ? Could you can explain that ? Remember that `www.whatsmydns.net` is sending DNS requests from machines distributed all around the world.

This shows that Youtube does not manage its CDN the same way as Akamai does. To explain the result you get, review the Unit 1 slides about *Server selection* and have a look to the Section II in paper **?**, that you will find in the folder `docs` in the website.