INGI1341: Project 2
Analysis of a website

Sergio Alejandro Chávez Cárdenas

Abstract—The details of a remote website sopitas.com are analysed and its responses over the Internet are observed. This report describes in details the key technical elements that have been found while interacting with this server, namely DNS queries, TCP segments, and HTTP requests.

December, 2014

1 INTRODUCTION

This report analyses the key technical elements about the functioning of the website http://sopitas.com/, which is a very popular site in Mexico, that provides news about sports, music, technology and general topics. It ranks 29th in number of visits within the country, according to Alexa statistics service [2]. Its main audience is young and adult people, including links to other related news’ websites, social networks and advertisement. Most of the advertisement in the website come from the Google AdSense web-service.

2 INTERACTION WITH WEBSITE

The tests with the website were performed mainly with a laptop running Ubuntu 14.04 and Firefox browser. Some other tests were performed in the Windows 8.1 environment and Google Chrome browser. The software to capture the packets is Wireshark.

2.1 DNS

To first look into the website functioning, the DNS queries that are performed by the browsers were analysed. Both, Firefox and Chrome query for IPv4 and IPv6 addresses with A and AAAA resource records (RR). For the first case, two queries are sent to different servers and they respond with the IP of the website and a list of authoritative name servers located close to the hosting company in the US. The Time-To-Live (TTL) of the response is very large (more than 100 days). In the case of the IPv6 query, the DNS protocol responds only with the CNAME (sopitas.com) an the name of an authoritative name server. In fact, the site does not yet support the IPv6 protocol. This can be confirmed in Fig. 1, where the DNS chain of the site according to RIPE [7] is shown. From this schematic, and from the interaction with the website, it is observed that the site is hosted by a server that uses sopitas.com as an alias.

It is also important to remark that the query may return different name servers depending on the location of the connecting host. For instance, when the query is performed in Belgium, the following name servers appear in the authority section of the message:

```plaintext
;; AUTHORITY SECTION:
sopitas.com. 163757 IN NS ns4.next.la.
sopitas.com. 163757 IN NS ns2.next.la.
```

Fig. 1. DNS chain for sopitas.com.
Meanwhile, the same query done in Mexico City returns the following name servers:

```
;; AUTHORITY SECTION:
com. 54973 IN NS f.gtld-servers.net.
com. 54973 IN NS g.gtld-servers.net.
...```

Now, if we consider the other web servers that are queried while loading sopitas’ site, it is observed that the browser request access to resources located in content delivery networks (CDNs) of companies such as Akamai, Edgecast and Amazon. These resources and queries are related with contents provided by companies like Google, Facebook and Twitter. In these queries, it can be observed that the CDNs offer larger responses including multiple IP addresses that are chosen differently each time a query is done. This is behaviour is shown in Fig. 2, where we observe that some CDNs already support IPv6 connections. In addition, these responses have short TTL values that may be used to spread the load over servers by forcing the hosts to send new queries within few seconds or minutes.

### 2.2 TCP connections

A considerable part of sopitas’ contents are hosted, so a number of TCP sessions to different servers are opened in order to show all the contents. Just by requesting access to the website, more than a hundred TCP sessions are registered to download all the contents of sopitas.com site. From these sessions, each browser opens up to six simultaneous connections per web server.

The connection options are negotiated in the three-way handshake. It has been observed during the tests, that the client uses some of the high performance extensions established in RFC 7323 [4]. The options are timestamps (Firefox only) and TCP window scaling. An interesting discovery while analysing the session establishment with Firefox and Chrome browsers, is that both send the SACK Permitted option enabled in all outcoming SYN packets, but many servers that use this option only respond to this flag when using the last. The same happens for the TCP window scaling addition, that is only present in both ends when packets are exchanged with the Chrome browser. So, if sopitas’ server does not respond to these extra features, it sets its congestion window to 65535 bytes and its MSS to 1460 bytes. In the other case, the congestion window has an initial scaling factor of nine. These differences are shown in Fig. 3.

The connections to sopitas.com are finished gracefully. But another interesting feature about TCP that is used by Ubuntu and that was not present in Windows, is the usage of TCP keep-alive packets which, according to RFC1122 [3], are empty ACK packets that are sent in order not to close the connection even if there is not data available to transfer. Those TCP sessions are finished gracefully after idle
Fig. 4. TCP keep-alive packets (multiple ACKs) in a session with Ubuntu.

Fig. 5. RTT of a TCP session with sopitas.com.

intervals from around 30 seconds up to 2 minutes. This behaviour is illustrated in Fig. 4.

In several captures, the data transfer was done reliably with only sporadic retransmissions. The average round-trip time (RTT) was different for each TCP connection, but generally below 3 ms as can be seen in Fig. 5.

2.3 HTTP requests

Different types of resources are retrieved when consulting sopitas’ website. For this, the HTTP GET method is used. The initial requests they all use HTTP 1.1 version, but this seems not to be an issue for the server that replies to the request with the same protocol version. The packets are mainly interchanged via the default HTTP port (i.e. 80), except for those instances that use HTTPS (port 443). The initial interchange of messages between client and server are shown next.

Client:
GET / HTTP/1.1
Host: www.sopitas.com
Connection: keep-alive
Accept: text,application,image [T]
User-Agent: Mozilla/5.0 [Truncated]
Accept-Encoding: gzip,deflate,sdch
Accept-Language: es,en [T]

Server:
HTTP/1.1 301 Moved Permanently
Server: nginx/1.6.2
Date: Thu, 13 Nov 2014 19:44:50 GMT
Content-Type: text/html
Transfer-Encoding: chunked
Connection: keep-alive
X-Powered-By: PHP/5.3.3
Location: /site/

Client:
0 [ACK]

Client:
GET /site/ HTTP/1.1

In the previous messages, we observe that the web server redirects the client to a different location each time the web page access is requested. This impacts a little in the page performance since additional RTTs are added at the beginning of the connection. Depending on the scenario, the GET method may include cookies or not. For instance, if we use a browser without previously stored data in the cache, no cookies are sent. According to several captures the contents consist mostly in HTML, CSS, javascript and image files. However, after few seconds in navigating through the website, several third party cookies are retrieved. According to the consults made with the web browser, there are at least five servers that store some information only from loading sopitas.com. Those sites as well as a description of the detected cookies according to [6] are next enlisted:

- Google.com: Uses a cookie named SID in combination with HSID to verify a Google user account and most recent login time.
- trkjmp.com: Unknown cookie named crtV.
- youtube.com: Uses a cookie named

1. Packets forwarded to this port were only present when using third party applications embedded in the website such as comments via Facebook.
Fig. 6. Google search box in sopitas’ site.

- **use_hitbox** to keep track of the number of views of a video.
- **doubleclick.net**: Uses two cookies named **id** and **test_cookie** for advertising purposes.
- **sharethis.com**: Uses a cookie named **use_stid**, which stores an unique identifier that is used for interest profiling aimed to advertising.

Moreover, when surfing throughout the website, more information is retrieved after using certain instances. For instance, if we perform a search with the embedded Google’s search box (shown in Fig. 6), extra cookies are detected:

- **Cookie _unam**: This third party cookie is used by the ShareThis service. It is present when clicking to the share buttons on each page across various social networks such as Twitter, Facebook etc. The _unam cookie monitors web pages viewed, navigation and time spent on each page.
- **Cookie _qca**: This cookie enables the analytics service Quantcast to collect information about the user activity with the queried website. The information includes IP, referrer, the search terms that have been used and the time of usage of the site.
- **Google Analytics cookies _ga and _gat**: These cookies are associated to Google Universal Analytics, which is a significant update to Google’s more commonly used analytics service. The new service reduces the reliance on cookies in general, and only sets these two cookies, although Google also say data can be collected without setting any cookies.

To sum up about the Google’s search box, whenever one looks for some information in the website, packets are interchanged showing the referrer website with the cookies enlisted previously.

Finally, it is possible to give comments about some of the news that are posted on the website. This can be done using either a social network account (e.g Facebook) or with the embedded comments field of the website.

In some captures using the computer default browser (Chrome), some HTTP headers contain a relatively recent extension called Do Not Track header [8]. This add-in has not been yet standardised, but it enables users to express preferences about third-party web tracking. That is, if the flag is set to true, the advertising site cannot track the user to offer him targeted advertisements. Although other web browsers are compatible with this feature, it is not enough to guarantee user’s privacy. According to some news’ articles, the advertiser can just decide to ignore the flag [9].

3 Conclusion

An analysis of the functioning of a popular Internet website has been performed. After analysing the different protocols that take part into the connection establishment, the details and site behaviours have been described. From the DNS protocol point of view, the relationship between the site and its hosting company has been established. Then, and the transport level there are some options that slightly affect the site loading when using different platforms and web browsers.

We observe, that sopitas.com uses Internet cookies to generate revenue as the majority of the Internet web sites. Unfortunately, it is often difficult for users to realize the amount of personal information that is shared and exposed when consulting their favorite web sites.

**References**