POSTER: An Empirical Measurement Study on Multi-tenant Deployment Issues of CDNs

Zixi Cai
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
caizixi@iie.ac.cn

Zigang Cao*
Institute of Information Engineering, Chinese Academy of Sciences;
School of Cyber Security, University of Chinese Academy of Sciences
caozigang@iie.ac.cn

Gang Xiong
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
xionggang@iie.ac.cn

Zhen Li
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
lizhen@iie.ac.cn

Wei Xia
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
xiawei@iie.ac.cn

ABSTRACT

Content delivery network (CDN) has been playing an important role in accelerating users' visit speed, bring good experience for popular web sites around the world. It has become a common security enhance service for CDN providers to offer HTTPS support to tenants. When several tenants are deployed to share a same IP address due to resource efficiency and cost, CDN providers should make comprehensive settings to ensure that all tenants' sites work correctly on users' requests. Otherwise, issues can take place such as denial of service (DOS) and privacy leakage, causing very bad user experience to users as well as potential economic loss for tenants, especially under the situation of hybrid deployment of HTTP and HTTPS. We examine the deployments of typical multi-tenant CDN providers by active measurement and find that CDN providers, namely Akaimai and ChinaCenter, have configuration problems which can result in DOS by certificate name mismatch error. Several advices are given to help to mitigate the issue. We believe that our study is meaningful for improving the security and the robustness of CDN.

KEYWORDS: CDN, HTTPS, certificate

1 INTRODUCTION

HTTPS is a widely used network protocol for secure web communications over public networks. More and more websites are deployed in the form of HTTPS. Until July 2017, according to Letsencrypt, the percentage of web pages loaded using HTTPS has exceeded half of the total, and the number of digital certificates has reached more than 40 million [1].

CDN or content distribution network is a geographically distributed network of proxy servers and their data centers to optimize security, performance and reliability. Web sites using HTTPS are deployed on CDNs in different ways, which can be divided into two modes according to whether the CDN node is shared, namely the dedicated IP and the shared IP. Many small and medium-sized sites will choose the latter due to economic reasons. For tenants, it is critical to understand how CDNs treat these websites deployed on CDNs which share a same IP. CDN acts as a proxy for those sites, therefore it assumes responsibility for providing secure access. For sites, it is important to understand the deployment of these multi-tenant CDNs, so they can select a more robust and secure CDN service provider, and try to avoid those CDN services with misconfigurations.

Although a lot of works have been carried out on HTTPS and CDN, we have not seen any systematical measurement and analysis that focus on the shared IP deployment mode in CDN and its impact on security, privacy and robustness of service yet. While the widely seen deployment approach may bring issues besides its convenience and higher exploitation of the machine resources.

In this paper, we measured the sites which were deployed on CDNs by different probing requests, summarized the typical characteristics among CDNs providers on the shared IP deployment mode, and discovered some misconfiguration problems with CDNs providers which may cause severe problems, especially when a node is deployed with HTTP and HTTPS sites, affecting users' experience on top popular sites. Some suggestions are offered to help CDN providers to mitigate the issues.

2 HOW MULTI-TENANT CDN PROCESS HTTPS

It is quite common that several web sites are deployed on one server in CDN for economic reasons. When a HTTPS ClientHello request arrives at the CDN server, a corresponding server certificate is required. If a mismatched certificate is returned to user, it will cause DOS or severe security warning. In practice, there are three common solutions exploited by CDN providers.The first solution is to assign a dedicated IP to each site to solve the conflict, which means one site monopolizes an IP. Wildcard SSL and Dedicated SSL are using this approach. The second is to add the Server Name Indication (SNI) support [2] , so the server can return the corresponding certificate according to the host name in SNI extension in ClientHello. Currently, SNI is supported in all popular browsers, so the solution is easy and

* corresponding author.

CCS '17, October 30-November 3, 2017, Dallas, TX, USA

ACM ISBN 978-1-4503-4946-8/17/10. https://doi.org/10.1145/3133956.3138852

economic. The third is to use a shared SSL certificate, which requires to put all sites' domains into the certificate's Subject Alternative Name (SAN) field [3]. However, security issues or privacy leakage may happen due to improper deployment in the real world. The general work mode of processing HTTPS request in CDN is shown in Fig. 1.

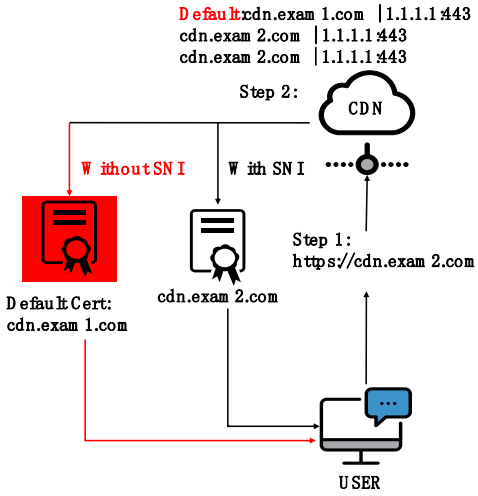

Figure 1: How CDN return certificate

3 OUR MESUREMENT

We try to identify the deployment problems of CDN through active probing techniques. Based on preliminary investigation, we choose the Alexa top 25,000 sites as our experiment objects since these sites are the most probably to exploit CDN to improve their users' experience.

To simulate the situation of accessing the site under different circumstances, we use different SSL/TLS protocol versions, different SNIs to establish SSL/TLS handshakes with the host of Alexa top 25,000 sites to obtain certificates [4]. The CDN node should select the appropriate certificate and return it to the client after it sends back the ServerHello message during the handshake. When multiple sites share one same IP, different certificates shall be return according to SNIs. In previous researches, configuration problems can be seen in the certificate properties [3-5], such as domain mismatches between certificates and websites [6].

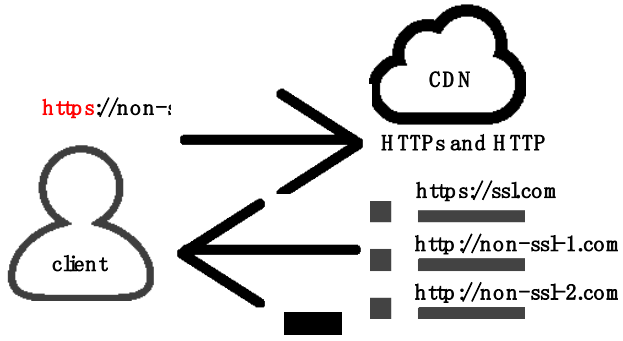

Figure 2: A user utilizes HTTPS to access websites

Fig. 2. shows a way for users to access the Internet through HTTPS. By entering "https" before a URL in the address bar of a browser, they want to establish secure connections with sites that may not support HTTPS. We call it forcing HTTPS visiting. Generally, CDN must provide the appropriate response on this condition. However, different CDN service providers may have different responses in the real world. We analyze the certificates returned by CDNs, as well as the server's responses to reveal the typical issues in current CDN service.

4 RESULTS AND DISSCUSSIONS

4.1 Multi-tenant CDN deployment situation

We summarize the differences in the behavior of these CDN providers in Table 1. We tick the checkbox if we can obtain certificate in that case; otherwise we cannot. Results shown that SNI is widely deployed by multi-tenant CDNs. Most CDN service providers only offer a high version of the TLS protocol. Including Distilnetworks, Gannett, Fastly etc.

Table 1: The inconsistent behavior of returning certificates by CDN providers

CDN provider	TLS1.0	TLS1.2	SNI	NULL
cloudflaressl	√	√	√	√
akamai	√	√	√	
chinanetcenter	√	√	√	
edgecastcdn		√	√	
myqcloud		√	√	
incapsula		√	√	
azurewebsites		√	√	
wpengine		√	√	
alicdn		√	√	
baishancloud		√	√	
jiasule		√	√	
distilnetworks		√	√	√
fastly		√	√	√
yunjiasussl	√	√		√
insnw	√	√	√	√
sucuri	√	√	√	√
cloudfront	√	√	√	√

4.2 Cases of forcing HTTPS visiting

CDN providers should response reasonably to users visiting tenants' sites using HTTPS. If tenant's site supports HTTPS, the CDN should reply with the correct content. If tenant's site only supports HTTP, the CDN should make an automatic jump from HTTPS to HTTP and provide the web content. However, several CDN providers do not properly handle the subsequent steps, causing problems of providing the wrong certificate, DOS, even provide the HTTP web page using a mismatched certificate which belongs to another tenant. The responses of CDN are summarized as the three cases below.

4.2.1 *Case of automatic jump*

It is a common solution for CDN providers that website will automatically jump from HTTPS to HTTP to ensure the visibility

of the content of the site. Many large CDN have adopted this approach, such as Cloudflaressl, and Cloudfront.

4.2.2 *Case of privacy leakage*

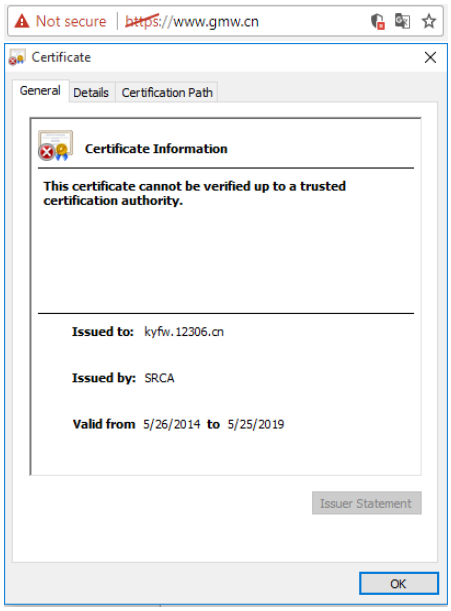

Figure 3: **Example of host name mismatch error. Using HTTPs to visit www.gmw.com and return certificate belong to 12306**

The process that CDN provides HTTPS services includes an SSL handshake, processing decrypted requests and returning content. Any of these steps can cause problems that lead to inconsistencies. Our experiment only focuses on the ClientHello and the certificate during TLS handshake. In our probing, a ClientHello of TLS version 1.2 carrying the correct SNI is sent to server, which imitates the normal behavior of visiting the site by modern mainstream browsers.

In Fig. 3, we illustrate this problem further by an example. When we force to visit one site by HTTPS that returns inconsistency through the Chrome browser, such as www.gmw.cn, we receive a certificate that belongs to kyfw.12306.cn (the official railway ticket selling site in China). After ignoring the wrong browser error message ERR_CERT_COMMON_NAME_INVALID and proceed to visit, web content of www.gmw.cn can still be provided through HTTPS. We think there are at least two configuration issues. First, after the correct SNI is provided, the server returns the wrong certificate. It is reasonable to suspect that CDN provider does not properly handle these SNIs in requests, especially considering that the client cannot obtain certificates from those servers when the SNI is not carried. Second, this is not just a problem with the SNI mechanism, because the server gets the corresponding host after decrypting the HTTPS request. This is the second chance that the CDN server can block the wrong service. However, we observe that the servers are still providing mismatched content.

3.2.2 *Case of DOS*

DOS can also be understood as a connection rejection on the page after ignoring browser certificate security alerts. Compared to privacy leakage, DOS has made progress by interrupting web services after decrypting the HTTP request.

4 CONCLUSIONS

Many websites deliver content through CDN. When multiple sites share the same CDN node, especially when the HTTPS site and HTTP site are deployed together, the management of CDN becomes rather complex.

We investigate the deployment of multi-tenant CDN providers using different SSL/TLS versions and SNI hosts. By analyzing certificates, we find that CDN service providers have different strategies in dealing with SNI and TLS version, resulting in different compatibility and even practical problems.

Based on an in-depth analysis of the measurements results, we also find that sites preferring to use SNI SSL are more likely to have problems with inconsistency in certificates and domains. One main reason is that these CDN providers do not take all possible SNI scenarios into account and verify whether those sites on the CDN can provide HTTPS service and working properly. Especially when SSL certificate name mismatch error occurs, sites can still continue to provide access. We expect our work can raise the awareness of this problem in the community. We believe our study will be useful for improving transparency, privacy and security, as well as strengthening Robustness of CDN.

5 SUGGESTIONS AND FUTURE WORK

The short-term approach is to avoid hybrid deployment of HTTP and HTTPS which requires more complex processing mechanisms. The long-term strategies are to strengthen CDN's ability to deal with all kinds of situations and provide a wider range of HTTPS support.

We guess that some sites offering both HTTPS and HTTP are deployed on CDN nodes that only support HTTP speed up service, leading to problems. We will do further study to verify it.

ACKNOWLEDGMENTS

This work is supported by The National Natural Science Foundation of China (No. 61602472, No. U1636217) and The National Key Research and Development Program of China (NO. 2016YFB0801200).

REFERENCES

[1] Letsencrypt, HTTPS://letsencrypt.org/stats/. July 13, 2017.

[2] D. Eastlake 3rd and Huawei, Transport Layer Security (TLS) Extensions: Extension Definitions, IETF RFC 6066, January 2011; www.rfc-editor.org/rfc/rfc6066.txt.

[3] GB/T 7714 Housley R, Ford W, Polk W, et al. Rfc 5280: Internet x. 509 public key infrastructure certificate and crl profile[J]. 2008.

[4] Alexa Internet Inc., "Top 1,000,000 sites (updated daily)," HTTP://s3.amazonaws.com/alexa-static/top-1m.csv.zip, 2009–2011, [online; last retrieved inMay 2011].

[5] GB/T 7714 Durumeric Z, Kasten J, Bailey M, et al. Analysis of the HTTPS certificate ecosystem[C]//Proceedings of the 2013 conference on Internet measurement conference. ACM, 2013: 291-304.

[6] GB/T 7714 Amann B, Vallentin M, Hall S, et al. Revisiting SSL: A large-scale study of the internet's most trusted protocol[J]. ICSI, Tech. Rep., 2012.

[7] N. Vratonjic, J. Freudiger, V. Bindschaedler, and J.-P. Hubaux. The inconvenient truth about web certificates. In 10th Workshop on Economics in Information Security, 2011.

[8] R. Holz, L. Braun, N. Kammenhuber, and G. Carle. The SSL landscape: A thorough analysis of the x.509 PKI using active and passive measurements. In 11th ACM Internet Measurement Conference, Nov. 2011.