Deployment Experience of Signed HTTP Exchanges with AMP as a Publisher

Shigeki Ohtsu
Yahoo Japan Coorpration

Abstract

Deployment experience of signed HTTP exchanges for use of AMP as a publisher was described. Its system design and results are shown with several feedbacks. Discussions were made if SXG can be a new platform to distribute trusted contents in Internet with a new policy of issuing SXG certificates and a validation.

Introduction

Yahoo! Travel[1] started to launch AMP[2] service in July 2018. The analysis showed that it got improvements of +60% traffics, -5% bounce rate and +0.5%CVR to the service[3]. Fig.1 shows that a screenshot of Android Google Chrome 74 that shows an AMP page in Yahoo! Travel.
This page was navigated form the search result pages in https://www.google.com/.

Figure 1: Yahoo! Travel AMP Page

An omnibox at the top of browser indicates that the page is served from https://www.google.com/. Its publisher name, travel.yahoo.co.jp, is displayed in the center of the additional bar below the omnibox. The omnibox is hided at the first display so that users cannot find the origin url unless they touch and scroll the browser.
The publisher's url can be found in the popup window with clicking the information icon as shown in the Fig. 2.

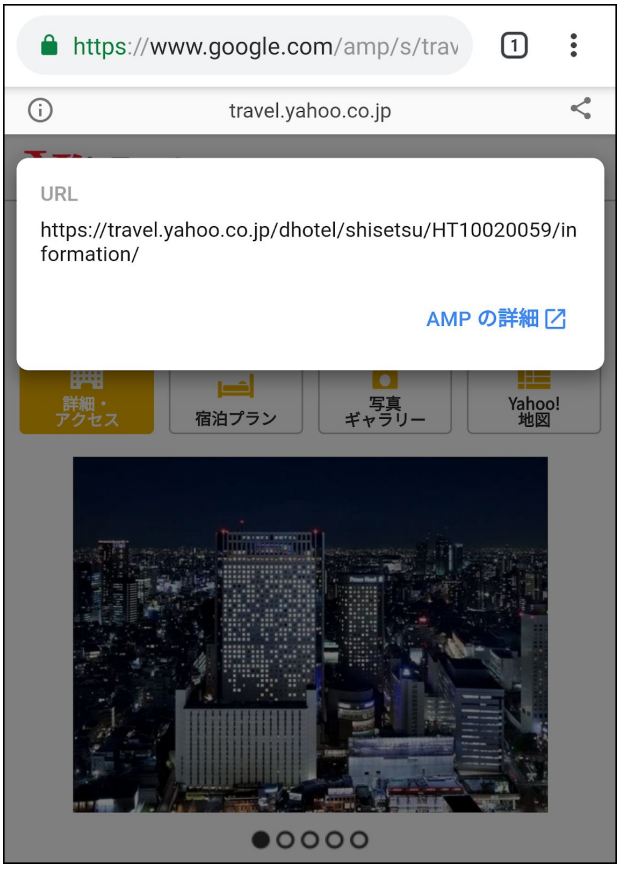

Figure 2: Popup window to show a publisher url of AMP

It is discussed that not all publishers gain the benefits of AMP pages[4] but the analysis of Yahoo! Travel indicates its positive statics owing to AMP. In order to make the most use of AMP, a new technology of Signed HTTP exchanges (SXG)[5] was deployed to Yahoo! Travel in Feb. 2019 with the following two objectives.

1. Showing a publisher's url in the browser as an origin
2. Serving a personalized AMP page to a user without having cross-origin access

At the time of writing this in June 2019, the second objective is under development. In this position paper, the deployment experience of SXG is described and its discussion is focused on the first.

SXG/AMP System Design

The SXG/AMP system was built in the exiting Yahoo! Travel service for about 3 weeks. Before built, it took about a month for investigation, evaluation and design.

In order to serving SXG to a user agent, it is needed to make a collaboration between a publisher and distributor. In the SXG/AMP system, the publisher is SXG/AMP system of Yahoo! Travel and the distributor is AMP Cache system maintained by AMP Project. A search bot plays a role to make a collaboration between SXG/AMP system and AMP Cache with crawling. Fig. 3 is a sequence diagram for serving SXG to a user agent.

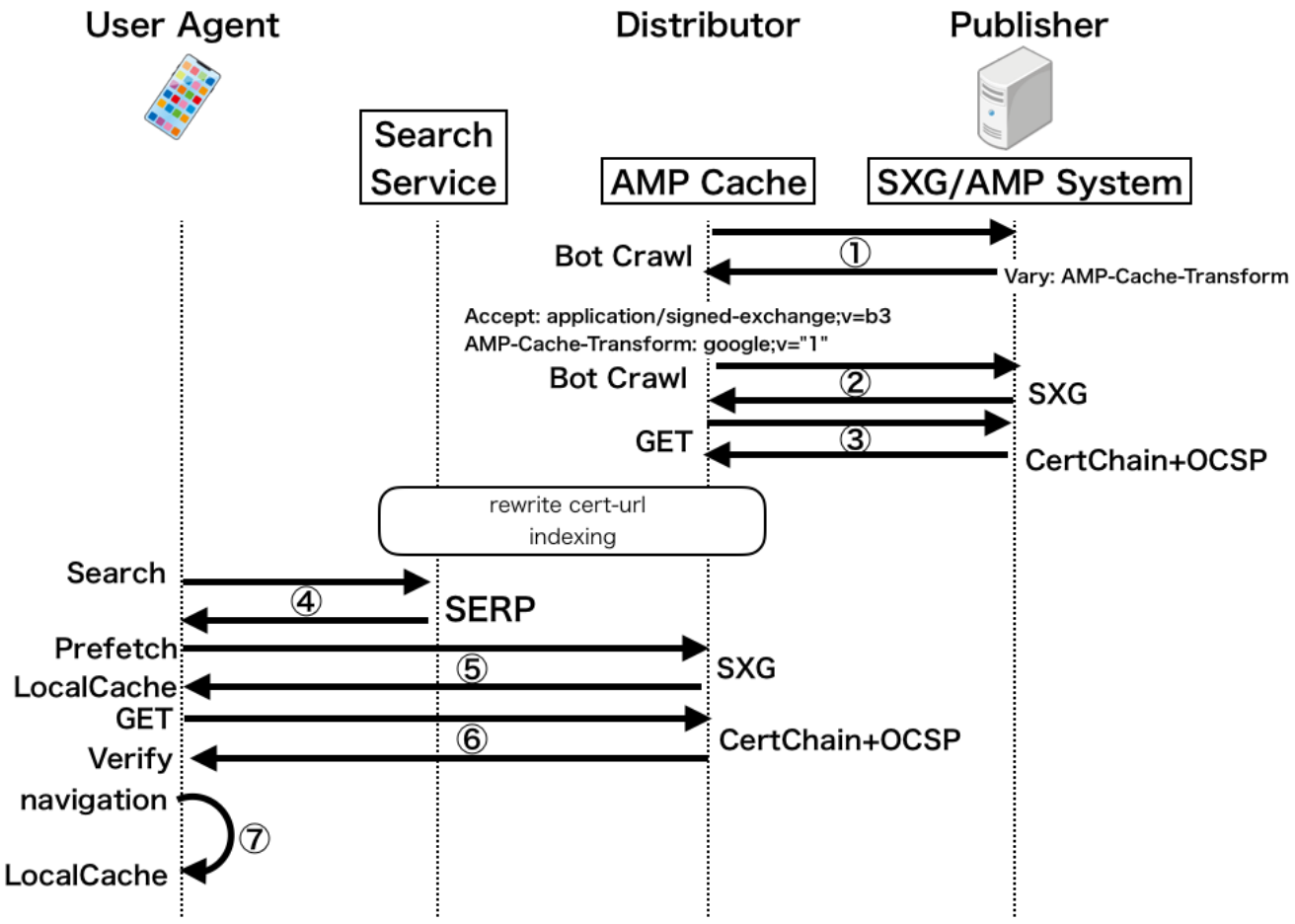

Figure 3: Sequence Diagram for serving SXG to a user agent

The following 7 steps are made until SXG is served to a user.

1. At the first crawling of the search bot, the SXG/AMP system returns ‘Vary: AMP-Cache-Transform’ header to notify the bot of SXG support.
2. The search bot requests with headers of ‘Accept: application/signed-exchange;v=b3’ and ‘AMP-Cache-Transform: google;v=”1”’ and get SXG files from the publisher. Note that “v=” is a version number and subject to be changed in the future.

3. The search bot parses the SXG file and obtain a cert-url, then get a file that includes a cert chain, a SXG certificate and intermediate certs, and OCSP(Online Certificate Status Protocol) data. After that, AMP Cache rewrite its cert-url to their cache url as it is out of signing. The search service make indexing of AMP contents.
4. A user agent submit a keyword to the search service and a SERP(Search Engine Result Page) is returned.
5. When a SXG supported AMP page is hit in the search results, prefetch tag is included in SERP. The user agent get SXG file from AMP cache and store it into the local cache during the time of showing of search results.
6. The user agent also obtains the file of cert chain and OCSP data from AMP Cache and verify the signature in the SXG and its content integrity.
7. When the user agent naviates the SXG site via search results, the contents are loaded from its local cache and its origin is substituted with the publishers url.

A overview of the SXG/AMP system in Yahoo! Travel is shown in Fig. 4.

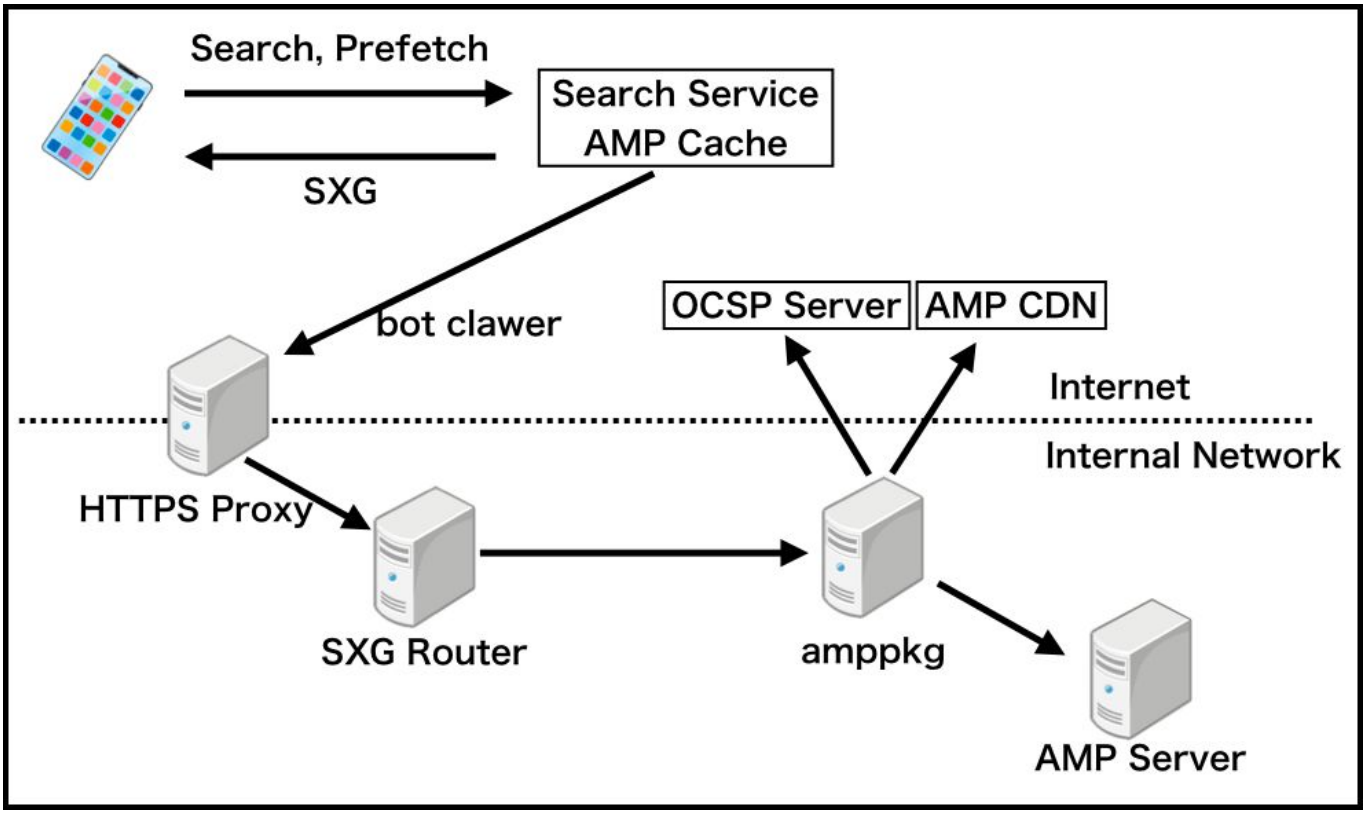

Figure 4: SXG/AMP system overview

At the boundary of internet, an HTTPS proxy terminate TLS connections and reroute various service requests to internal systems. All requests to AMP pages in Yahoo! Travel are forwarded to an SXG router.

The SXG router investigate requests and reroute SXG requests to amppkg with query parameters for signing and fetching. It always returns ‘Vary: AMP-Cache-Transform’ to notify bot crawlers of accepting SXG requests.

The amppkg is an OSS of amppackager provided by AMP project[6]. It is a main program used for signing AMP pages and serving SXG files with using an SXG certificate and its private key. The SXG certificate is an ECC certificate which has an extension for limited use for only SXG. Currently, only DigiCert issues SXG certificates[7] and its validity period is limited to 90 days according to the security requirements of SXG specification. The SXG certificate and intermediate certificates are served to bot crawlers together with OCSP data cached from OCSP server.
Before signing, the amppkg optimizes AMP contents with using transformer program where AMP runtime files periodically obtained from AMP CDN. It is needed because content data cannot be modified after signing.

An AMP server is serving AMP contents both for SXG and non-SXG requests. The SXG specification requires AMP contents must be cacheable so that we have to remove stateful headers such as `Cache-Control: private` or `Set-Cookie` for bot requests.

Results

Currently more than 10K AMP pages in Yahoo! Travel were served with SXG. The amppkg is very stable and running without any troubles.

The screenshot of SXG with Android Chrome 74 is shown in figure 5. This page is navigated with the same procedures of fig. 1.

Figure 5: Screenshot of SXG page in Yahoo! Travel with Android Chrome 74

The publisher's url, https://travel.yahoo.co.jp, is shown in the omnibox even when this contents was not served from the publisher.
To see it more clearly, figure 6 shows the network panel of devtool in Desktop Chrome 74 with simulating a mobile user agent. Red lines show prefetch requests of SXG and cert chain files from AMP Cache of https://travel-yahoo-co-jp.cdn.ampproject.org/. This means that even if prefetch requests were made in SERP, their privacy was preserved with no accessing to the origin. When the user agent was navigated to the SXG site, the contents are loaded from local cache and it achieves huge performance gains.

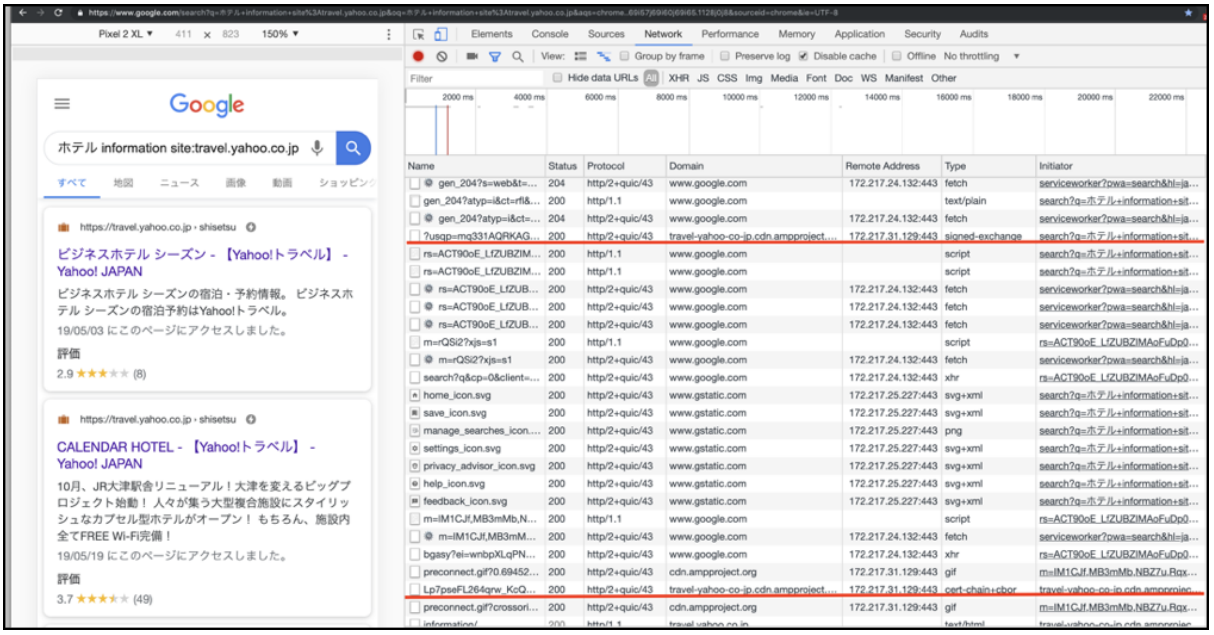

Figure 6: Network requests during showing SERP where an SXG page is included. Red lines show prefetch requests of SXG and cert chain files from AMP Cache.

Lessons and Learned from SXG Deployment

Here is a list of feedbacks from SXG deployments.

- Infrastructure impact was limited because of working on AMP url path.
- User impact was limited only for care about bot crawling.
- Testing was not completed before releasing. Bot crawling and AMP cache tests can be done after the system in the public to the internet.
- Need to know if AMP contents are really cacheable.
    - Stateful headers such as "Cache-control: private" or "Set-Cookie" are NOT included. The amppkg can check these headers but they need to be cared before deployments.
    - Confidential data is NOT included in the contents or headers.
- Check if contents work well on both same and cross origin for not all browsers support SXG yet.

These issues would be resolved when we have several best practices and good deployment tools.

Discussions

SXG with AMP gives benefits to the service with the publisher's origin and performance gains together with privacy preserving. It also enables us to distribute the content from any CDN without worrying about their origin because it is authenticated and verified.

As seen in the figure 1, it is sure that the current AMP makes users confused by showing different origin from that of a publisher. It would get worse if they tend to ignore to confirm the origin because they would not be careful of phishing sites.

Comparing two screen shots with showing the same page in figure 7. One is SXG and the other is HTTPS. There are no difference between two unless finding issued CA in the certificate(Fig. 8).

The features compared between HTTPS and SXG is listed in table 1.

Figure 7: Comparison of screen shot between SXG and HTTPS

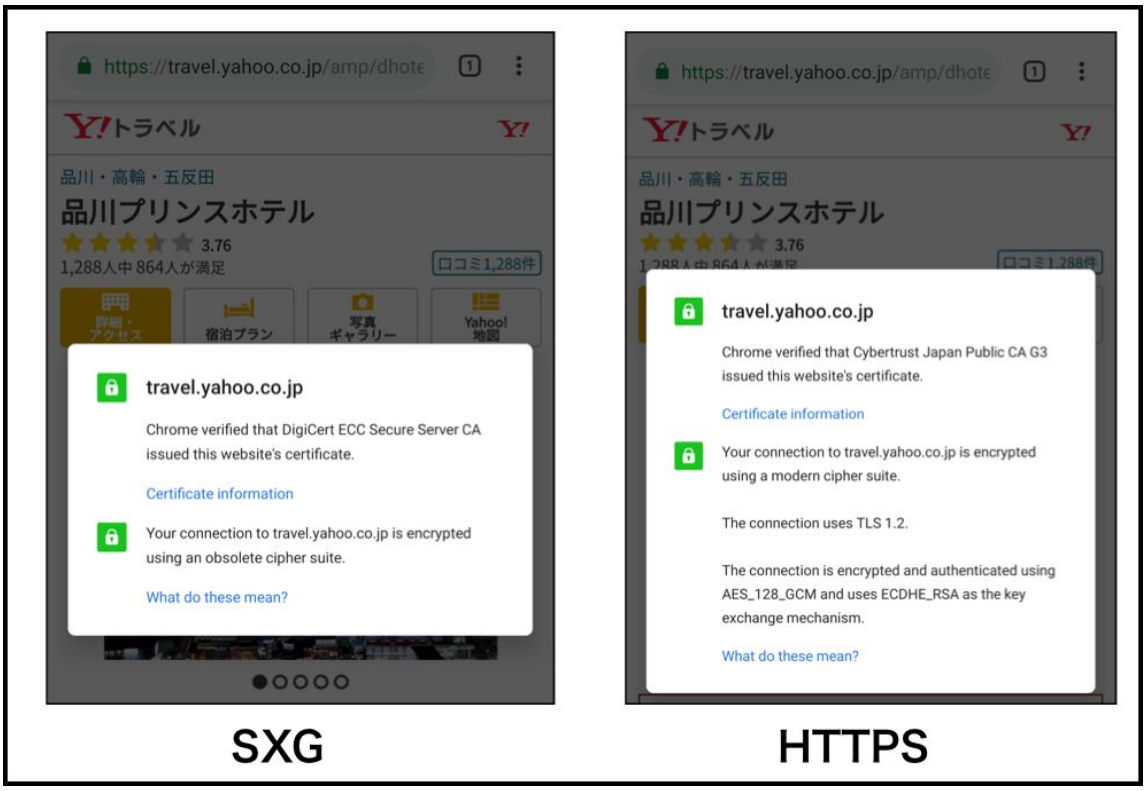

Figure 8: Finding distinguishments between SXG and HTTPS when certificates are issued from different CAs.

Table 1: List of features compared between HTTPS and SXG

	HTTPS	SXG
Confidentiality	✓	--
Integrity	✓	✓
Authentication	✓	✓
Non-Repudiation	--	✓
Origin	Peer	Publisher
Protected Period	during connection (about hundreds msecs)	until signature expires (maximum 7days)
Constraints	--	Cacheable Contents (GET only, Header Restrictions)

HTTPS protects its connection while SXG protects its contents. SXG has different security features from HTTPS so that security requirements of SXG are much severer than that of HTTPS such that SXG certificate limited to its validated period within 90 days.

In Nov. 2014, IAB published “IAB Statement on Internet Confidentiality”[8]. Firefox telemetry shows 78% of web pages loaded by using HTTPS in April 2019[9].

On the other hand, 58% of phishing websites use SSL/TLS and HTTPS in Q1 of 2019 with a rapid increase in these years[10]. According to the statics of phishbank.org, more than 90% of phishing sites are DV certificates. But it was proved that even EV certificates can be weak for phishing if its costs are ignored[12]. In 2018, the company name was removed in Safari UI when EV certificate was used[13].

In order to take an action against HTTPS phishing, CASC(Certificate Authority Security Council) started to make London Protocol[14] in 2018, which is not finished yet. In 2018, Symantec was distrusted by major browsers[15], customers needed to renew their certificates.

It is obvious that the current Web PKI and certificate validations of DV, OV and EV have issues not to be trusted blindly in the future of HTTPS everywhere. The policy to issue SXG certificates is not defined in CAB forum[16] yet.

SXG authenticates its contents. Its contents can be trusted by users if SXG certificate is issued by a trusted CA to a trusted publisher with a trusted validation against its publishment. They can be distributed from any distributors via secure channel in HTTPS.

I believe that SXG can be a new platform to distribute trusted contents if the specification covers not only a protocol format but also a validation procedure of issuing SXG certificate together with CAB forum. If it can be achieved, a new UI to show the contents validated with SXG would lead users for the safe use of Internet.

References

1. https://travel.yahoo.co.jp/
2. https://amp.dev/
3. https://www.slideshare.net/techblogyahoo/maximize-yahoo-japans-ux-with-amp-and-signed-http-exchanges-ampconf-141537298
4. https://cdn2.hubspot.net/hubfs/3794894/PDF/Chartbeat%20AMP%20Study_082318%20.pdf
5. https://github.com/WICG/webpackage
6. https://github.com/ampproject/amppackager
7. https://docs.digicert.com/manage-certificates/certificate-profile-options/get-your-signed-http-exchange-certificate/
8. https://www.iab.org/2014/11/14/iab-statement-on-internet-confidentiality/

9. https://letsencrypt.org/stats/#percent-pageloads
10. https://www.thesslstore.com/blog/58-of-phishing-websites-now-use-https/
11. https://cabforum.org/wp-content/uploads/London-Protocol-Presentation.pdf
12. https://stripe.ian.sh/
13. https://cabforum.org/wp-content/uploads/201806AppleCABF.pdf
14. https://casecurity.org/2018/06/27/casc-announces-launch-of-london-protocol-to-improve-identity-assurance-and-minimize-phishing-on-identity-websites/
15. https://security.googleblog.com/2018/03/distrust-of-symantec-pki-immediate.html
16. https://cabforum.org/