

WoSign Incidents Report Update

(7th Oct. 2016)

First, WoSign like to be more transparent, this is why WoSign like to setup CT log server and decided to post all SSL certificate to Google CT log server since July 5th 2016 by our own initiative.

Second, WoSign pledges that the information in this update report is 100% based on the fact for full transparency.

Third, this update report is for WoSign only, not for StartCom. StartCom will release a separate response for near term change plan.

Fourth, due to the severity of issues noted within, the decision has been made to legally separate WoSign and StartCom as well as appoint new leadership for StartCom and WoSign.

Part I Response to issues

WoSign received an email from Mozilla for 3 incidents on August 24th 2016. Over a month, there has been many emails asking for clarification and WoSign has released two reports ([Preliminary Report](#) and [Final Report](#)) with detailed information of those incidents and some others that have been requested as indicated in the Mozilla wiki [page](#).

After Mozilla released a next action proposal, WoSign investor – Qihoo 360 realized the severity that assign CSO with team to investigate this case. Today, WoSign released this updated incident report.

For a better understanding and transparency, WoSign posted all 2015 issued SSL certificates to Google CT log server and WoSign CT log server, making a total of **101,485** certificates. And recently WoSign also published all certificates that have been issued from Jan. 1st 2016 to July 4th 2016, making a total of **94,073** certificates. Since July 5th 2016, WoSign decided to post all issued SSL certificate to CT log servers and embedded the SCT data in the certificate meaning that all WoSign issued SSL certificates are now in the CT log server for full transparency.

1. Issue D: Long-Lived SHA-1 Certs (Jan - Mar 2015)

(a.k.a. "Issue -2")

Between 16th January 2015 and 5th March 2015, WoSign issued 1,132 SHA-1 certificates whose validity extended beyond 1st January 2017. This is documented in their [BR audit](#).

WoSign Response

1.1. What happened

WoSign issued 1,132 SHA-1 certificates from Jan. 16th 2015 to Mar. 5th 2015. WoSign reported this

issue to the WebTrust auditor, and this incident is included in the 2015 WebTrust BR report that was sent to all browsers.

WoSign was aware of the issue and the reason why this was done and thus notified to the auditor to include in the report for transparency.

1.2. Why this happened

The BRs recommended all CAs for not issuing SHA-1 SSL certificates with a validity period beyond January 2017, starting on 16 Jan 2015, so WoSign started to update its PKI system when CAB Forum ballot 118 passed, but due to unexpected delays in the systems upgrade, WoSign could not finish it until Mar. 5th 2015.

Wosign should have prepared to update its systems earlier to avoid a delay in implementation.

During this time WoSign was aware to be issuing certificates beyond that end date.

1.3. What has been done

WoSign decided to contact all “effected” customers and offer a revocation and replacement to provide them a new one with SHA-2. Up to now, only 171 certificates have been revoked and replaced to SHA-2, for the remaining WoSign is waiting, knowing the issues these customers may have due to conversations with them, for example, subscriber’s web server or equipment obsolete, upgrading pending, etc.

WoSign will wait and try to replace these remaining certs but if no answer is received and following the decision taken and presented to the auditor, WoSign will revoke those certificates before Dec. 31th 2016.

1.4. Current situation

As indicated above, WoSign updated their systems at Mar. 5th 2015 that no more long-lived SHA-1 certificates can be issued.

2. Issue F: Certs Identical Except For NotBefore (Mar 2015)

WoSign issued two certificates in March 2015. These certificates are identical in all ways (including their serial numbers) except for their notBefore dates, which are 37 seconds apart.

- [Cert 1](#)
- [Cert 2](#)

WoSign Response

2.1 What happened

This incident is reported in the “Incidents involving the CA WoSign” mail list thread.

-----Original Message-----

From: Peter Bowen [mailto:peterbowen@gmail.com]

Sent: Friday, September 2, 2016 11:21 PM

To: Richard Wang <richard@wosign.com>

Cc: Ryan Flood <ryan@elewin.com>; mozilla-dev-security-policy@lists.mozilla.org

Subject: Re: Incidents involving the CA WoSign

Richard,

It seems then there is a newly exposed bug.

<https://www.censys.io/certificates/e2665bb07940b5bee73145f47c99dcf5781edbe9d78f9cada8f1d702d5e340ad> shows a certificate issued by your CA that has a notBefore in March 2015. It does not appear in the CT log. However another certificate with identical serial number and subject, but different Validity, does appear in the log.

Are you aware of a bug where you were issuing certificates identical except for validity period?

This issue with same serial number in certificates with same subject information but different signing time was detected internally time ago, in fact this is not the unique case because there are 16 similar issues. This was caused by the CMS (Certificate Management System), when it sent the signing request of the certificate to the signing server A, which had no response, then the CMS sent it to the other newly added signing server B. After a while the signing server A signed the certificate and sent to the CMS and also to the subscriber, then the subscriber installed the cert in its website and that's why Censys recorded this certificate; in the meantime, the signing server B also signed this certificate some time later (in seconds) and sent it to the CMS, the CMS accepted it and rewrote it in the DB.

Of course the subscriber didn't know this issue, and only the first signed one, cert 1, is known by the subscriber and public. But internally, cert 2 replaced cert 1, so in the CMS and PKI DB only cert 2 appeared. When decided to publish all issued certificates to CT log server then only cert 2 was published because replaced cert 1, which is the one in the subscriber website. WoSign only had internally cert 2.

2.2 Why this happened

This issue happened after adding another signing server on **Jan 5th 2015**, and found it on **April 9th 2015**. When had the two signing servers added a load balancer, but the configuration was not properly done because it didn't lock the request.

This case also exposed a bug in the CMS that didn't lock the order's record after getting the signed certificate.

Here is the crt.sh link for all certificates with apart time (seconds) and issued time:

Crt.sh link	Issued time 1	Issued time 2	Apart time
https://crt.sh/?serial=6c5f294a0b7838a51b96d33adf3fb774	2015/1/5 8:08:23	2015/1/5 8:08:30	7
https://crt.sh/?serial=25e5400b6bffdbd74c8ce82878b44188	2015/1/10 15:07:41	2015/1/10 15:09:08	87
https://crt.sh/?serial=1e3b88dac8846ea1a268f93c911a19e4	2015/1/21 13:25:24	2015/1/21 13:25:30	6
https://crt.sh/?serial=12fe10c1ea652c1472eb6e17a42e5ecd	2015/3/10 7:30:38	2015/3/10 7:30:43	5
https://crt.sh/?serial=1a8121304718800638ea6899ede21d45	2015/3/10 7:35:09	2015/3/10 7:35:15	6
https://crt.sh/?serial=320a71d69cff608e87766b918ecd07a	2015/3/10 12:33:28	2015/3/10 12:33:36	8
https://crt.sh/?serial=177f1598e211b6ad3877d2a8729a48e1	2015/3/11 13:52:21	2015/3/11 13:55:45	204
https://crt.sh/?serial=50bbf6f23fb69c2b541bc75e09f02ff7	2015/3/13 14:32:20	2015/3/13 14:33:07	47
https://crt.sh/?serial=32e2b62f4467a6ed69c01d87675dc93f	2015/3/13 14:34:53	2015/3/13 14:35:05	12
https://crt.sh/?serial=12df3f24d92bb30798d68e9c477a48a2	2015/3/13 14:41:51	2015/3/13 14:42:04	13
https://crt.sh/?serial=46a20ef79b21d30b015aeac84b22747b	2015/3/13 14:42:27	2015/3/13 14:42:50	23
https://crt.sh/?serial=1115b96b885686f5dbb014150dc23b91	2015/3/13 14:43:24	2015/3/13 14:43:55	31
https://crt.sh/?serial=12529e7bef44097e49c5d2800070f0ea	2015/3/13 14:44:20	2015/3/13 14:44:46	26
https://crt.sh/?serial=4c139e39e6050269681ca264f1429fac	2015/3/13 14:46:56	2015/3/13 14:47:05	9
https://crt.sh/?serial=4f6b5d70f0d9fcdda6347a006f3e98c5	2015/3/13 14:50:43	2015/3/13 14:51:20	37
https://crt.sh/?serial=4b6e1b196ad9e713122a34f118605683	2015/4/9 7:24:07	2015/4/9 7:24:08	1

2.3. What has been done

WoSign fixed the load balance system and changed the signing mechanism, and updated the configuration of the load balancer in such a way that now, all signing request from CMS will write to database first, and the signing server will get the signing task from the database. If one request is assigned to one signing server, it will be locked exclusively.

In the CMS side, it will reject the signed certificate return from PKI system after received, this can prevent the PKI signing server still send the signed-again certificate in any case. The idea behind this is that the CMS will reject the cert 2 because the CMS already sent the cert 1 to the subscriber

2.4. Current situation

This is an incident caused by system bug and wrong configuration, but all certificates were well validated

WoSign considers that there's no reason to revoke these certificates because cert 2 was never used in public, and only aware of it when published.

3. Issue H: Duplicate Serial Numbers (Apr 2015)

(a.k.a. "Issue X")

Between 9th April 2015 and 14th April 2015, WoSign issued 392 certificates with duplicate serial numbers, across a handful of different serial numbers. Here is one example. This is documented in their most recent BR audit.

WoSign Response

3.1. What happened

WoSign issued 392 certificates with several duplicated serial numbers from April 9th 2015 to April 14th 2015. WoSign reported this to the WebTrust auditor, and this incident is included in the 2015 WebTrust BR report that was sent to all browsers.

3.2. Why this happened

This issue can be divided into 2 different ones because the source of the issue is different even the result is the same.

Firstly 313 certificates and secondly 27 certificates were affected by a system bug with the serial number generation, generating a serial number starting with "0" in the first left position. The signing system had a bug that didn't know how to deal with this kind of serial number.

The others were due to a load balancer configuration issue with the two signing servers, different from the above one, because this time signed two certificates with the same time using the same serial number, for example: <https://crt.sh/?serial=112A93A547BC6A7701A2BBDD0B4E67FF>, the issuing time is as "Apr 9 08:12:40 2015 GMT"

3.3. What has been done

One of our customers notified us of this incidence on April 14th 2015, then after internal checks and reviews an email was sent to these subscribers offering a replacement within one week. Those certificates with duplicated serial numbers were revoked from April 21st 2015 till to April 28th 2015.

WoSign revoked the 392 certificates. See below details:

CT log URL in crt.sh	Quantity
https://crt.sh/?serial=56D1570DA645BF6B44C0A7077CC6769	313
https://crt.sh/?serial=D3BBDC3A0175E38F9D0070CD050986A	27
https://crt.sh/?serial=112A93A547BC6A7701A2BBDD0B4E67FF	2
https://crt.sh/?serial=12E9B15E3FF1CDED3EC86BF132063AB8	2
https://crt.sh/?serial=16E60CC1BEFE5C243F65AD7D85F9328D	2

https://crt.sh/?serial=1FD8B07664CEEE58E779AA200AF0A9A6	2
https://crt.sh/?serial=20AD94DE69A0EE25428BB6CA5EAF0395	2
https://crt.sh/?serial=23876A3F46E65EB83500914B87409C2C	2
https://crt.sh/?serial=29C6BC8782E004B26BED594FBEF3122B	2
https://crt.sh/?serial=2F7099A8DF3EBE2E0DEBD6DC9BB908B9	2
https://crt.sh/?serial=302388004A3479660553EF2A79A44B78	2
https://crt.sh/?serial=3045B73FE2FCCB51CC5DA9887B4C6ED0	2
https://crt.sh/?serial=318FE0819849A5F9C1D1854C7465D9A5	2
https://crt.sh/?serial=35CE1A681804BC007A30513A4AA042AE	2
https://crt.sh/?serial=390DE9D519163B3D1976A20CF484C515	2
https://crt.sh/?serial=3B9B2E8670B65ECFABAF06345346CAD8	2
https://crt.sh/?serial=4333FA3C03F27340425E3E44C62DF6F4	2
https://crt.sh/?serial=44E44D23471518D4A9E1C91AC30DA77B	2
https://crt.sh/?serial=47E3642E70C6463C03A6A23B4A7B98A9	2
https://crt.sh/?serial=49D978B3CF1229BE1B32B537E53C8972	2
https://crt.sh/?serial=4CEDEAE1148775B749EACAF890DAB4C0	2
https://crt.sh/?serial=4F8226EF661F99B282561B3B308B6186	2
https://crt.sh/?serial=504261656EE8447B56166E77BFE52548	2
https://crt.sh/?serial=527D513DC9806DAEC778D1B876B3B764	2
https://crt.sh/?serial=5F1AAF998410578D64D0A783BCDFA8C1	2
https://crt.sh/?serial=67034EA2ED2A1DF42C563F577911FD45	2
https://crt.sh/?serial=6B87C339A14F8CEA541421883BB72085	2
https://crt.sh/?serial=6C311A6B744F8CC66794F53ADF99CE60	2

3.4. Current situation

The 392 certificates are revoked. For the first case scenario problem, WoSign fixed the code that can recognize the “0” at first in the serial number.

For the second case, WoSign added more checking and verification in the system’s code before signing for each signing server.

At the same time WoSign have improved its internal quality control system that all certificates must pass these checks before sending to subscribers.

4. Issue J: Various BR Violations (Apr 2015)

(a.k.a. "Issue -1")

On April 3rd 2015, WoSign was contacted by Google, who were concerned about Baseline Requirements violations in recently-issued certificates from WoSign. Instead of specifying the violations directly, Google asked WoSign to check their certificates against their CPS.

WoSign Response

4.1. What happened

WoSign was notified by email from Google on Apr. 4th 2015 09:25AM about an issue regarding some violations of the BRs. WoSign's CEO replied Google to check it asap.

From: Richard Wang
Sent: Saturday, April 4, 2015 11:05 AM
To: richard.wang@wo-sign.com
Subject: Re: WoSign Irregularities

Hi Ryan,

For irregularities, we will check it carefully next week.
Thanks.

Regards,

Richard

> On Apr 4, 2015, at 09:25, ryan@wo-sign.com wrote:
>
> Hi Richard,
>
> It's come to our attention that WoSign may be issuing certificates that are
> not conforming to your CPS and not conforming to the Baseline Requirements.

The main problem was due to adding a description in the free SSL certificate in two languages:

CN = wosign@wo-sign.com
Description = Free SSL Cert apply URL: https://buy.wosign.com/free

CN = woopah.org
Description = Apply Free SSL Certificate at https://buy.wosign.com

CN = raf.lcz.de
Description = 免费SSL证书 申请网址 : https://buy.wosign.com/free

```
CN = cle...ky.me
Description = 免费SSL证书 申请网址 : https://buy.wosign.com
```

Google said these certificates “are not conforming to your CPS and not conforming to the Baseline Requirements.”

4.2. Why this happened

WoSign started to provide free SSL certificates on Jan. 1st 2015, and decided to add an advertisement in the subject of the certificate having in mind what other CA had been doing adding some additional content in the OU field of the subject.

When notified, searched our system and found 15,211 certificates affected from Jan. 1st 2015 to April 7th 2015.

4.3. What has been done

WoSign updated their CPS accordingly in the specific section and checked and updated the certificate profile affected on April 8th 2015, 8:23 A.M.

From: Richard Wang
Sent: Wednesday, April 8, 2015 8:23 AM
To: '...'@google.com>
Subject: RE: OU RE: WoSign Irregularities

OK, thanks.

I think we corrected all, thanks for your help.

4.4. Current situation

As mentioned above WoSign is committed to follow all the standards, best practices and CAB Forum documentation, having said this, WoSign decided to start some marketing practices following what other CAs were doing but unfortunately not well. On the other hand, WoSign considers that it is no necessary to revoke these certificates since all of them were correctly validated.

No more certificates have been issued with this additional information since April 8th 2015.

WoSign following CPS very strictly, but if we found the policy is not so good for products, we must update the CPS. If any violated the CPS, we have two solutions: update CPS or update system.

WoSign should have resolved this in a more timely manner.

5. Issue L: Any Port (Jan - Apr 2015)

(a.k.a. "Issue 0")

From Jan 10th 2015 to April 23rd 2015, WoSign's certificate issuance system for their free certificates allowed the applicant to choose any port for validation. Once validation had been completed, WoSign would issue certificates for that domain. A researcher was able to obtain a certificate for a university by opening a high-numbered port (>50,000) and getting WoSign to use that port for validation of control.

This problem was reported by Google, and WoSign resolved. Mozilla only became aware of it recently.

WoSign Response

5.1. What happened

WoSign got report from Google at 8:55 AM April 24th 2015 that point out this high port problem that allowed the applicant to choose any port for website control validation. Richard Wang replied Google email within **2 minutes**, and promised to fix this bug within **1 hour**. Richard sent email to Google at 10:09AM after fixed the bug.

We searched our certificates orders from January 10th 2015 to April 24th 2015, there were 72 certificates issued using higher numbered ports website control validation, those certificates were validated by website control validation* method that using other port instead of 80 and 443.

* "Website Control Validation" means subscriber must upload the html file with verification code into its website root directory.

5.2. Why this happened

WoSign was aware of some customers couldn't use the 80 or 443 ports for performing the website control validation and requested a change to use any port for this validation. This change was made on January 10th 2015.

5.3. What has been done

WoSign changed their system to fix the problem and closed all ports except 80 and 443. So the high port validation allowed period is from Jan. 10th, 2015 to April 24th, 2015.

WoSign posted all those certificates to WoSign CT log server at Aug. 26th 2016 and Google CT log server at Sept. 03rd 2016.

WoSign logged the domain validated time and method, don't log the website control validation method port, we think this is not important, the importance is if this domain is validated or not validated, using which method to validated at what time. WoSign will log the port info if WoSign decide to re-open the website control validation method.

5.4. Current situation

WoSign fixed the bug and disabled the website control validation for ports different of 80 or 443. Also have investigated every certificate and decided to not revoke these certificates. The certificates were not violating the BRs.

On the other hand, it's WoSign fault not having notified the WebTrust auditor of this issue and hence not communicated to the browsers.

6. Issue N: Additional Domain Errors (June 2015)

(a.k.a. "Issue I")

In June 2015, an applicant found some problems with WoSign's free certificate service. There were actually two bugs, which we will denote N1 and N2.

Bug N1 was an issue where someone proving control of <subdomain>.example.tld also was given a cert covering example.tld.

Bug N2 was an issue where arbitrary domains can be added to an existing request after validation.

WoSign Response

6.1 What happened

6.1.1 Bug N1

This is a system bug come from website control validation that when a subscriber passes the subdomain validation, then our system added the top domain in the certificate automatically. We searched our database and there were 21 mis-issued certificates of this type, all certificates were revoked and posted to CT log servers.

6.1.2 Bug N2

This is another system bug that when the subscriber finished the domain control validation, he/she can use a special professional method to add other un-validated domain to the order, then our system issued the certificate including all domains in the order.

We searched our database and found 12 mis-issued certificates with this bug including the certificate issued to the domain "github.com, all certificates were revoked and posted to CT log servers.

6.2 Why this happened

6.2.1 Bug N1

This mis-issued case was caused by the engineer misunderstood the adding of an additional domain rule. The rule is; if you validate the domain: wosign.com, and you apply for a certificate for wosign.com, then the system will add a subdomain www.wosign.com in the SAN for free, this is for the subscriber convenience and there's no problem if the site visitor visits https://wosign.com and https://www.wosign.com.

This is not a problem in Domain Whois Control Validation*, but for website control validation method, it would be a problem if the subscriber validated a subdomain that added the top domain to the certificate. This bug is fixed completely at Aug. 10th, 2015 system update since we changed the order procedure that subscriber submit all the domains first to database, then validate it one by one, so the vulnerability was fixed.





* "Domain Whois Control Validation" means "BR - 3.2.2.4.4 Constructed Email to Domain Contact" that system send verification code to domain name whois admin email, subscriber must input this code in the application process.

6.2.2 Bug N2

These mis-issued certificates were a system bug that when the subscriber finished the domain validation, they can add any other domain before submitting this order to system., the vulnerability got fixed on the August 10th 2015 system upgrade, this upgrade changed the order procedure that subscriber submit the all domains first to database, then validate it one by one, the vulnerability was fixed.

The reason that we found the github issue but did not found others is we have a protected domain list that github is in the list, other mis-issued certificate is not recognized as a famous brand that not in the list and was issued automatically.

The following screenshot is the current keyword setting for github, "f"=flag; "r"=reject, we changed the class 1 certificate from "f" to "r" after we found out the mis-issued certificate case for github.

Keyword	Level	Enable	Sub Domain	Top Domain	Root Domain
github	Class3		f	f	r
github	Class4		f	f	r
github	Class2		r	r	r
github	Class1		r	r	r

6.3 What has been done

The two bugs were caused by the unreliable order procedure that our system needed to verify every parameter in the server side. So we changed the order procedure that all orders info, including the domain list, write into database first, then the subscriber need to validate the domain name one by one, with no chance for the subscriber to modify the order data.

And the problem with the website control validation, we think it is not a secure method for validation, so finally decided to disable this method to prevent this case happen in the future.

6.4 Current situation

WoSign fixed the bug and disabled the website control validation since Aug. 27th 2015 even the BR allows this method.

Why we want to change our order procedure since we know we issued the wrong certificate caused by the wrong order procedure, it is definitely not the unrelated system upgrade.

It is not “issue first, validate later”, it is “Issue after Validation, Review later”, every order is validated according to our CPS, for the issue N2, subscriber finished the domain control validation or website control validation, the certificate is mis-issued by the subscriber used a special technical way to include an un-validated domain to the certificate, this is code bug, it is not “issue first, validate later”. WoSign do the “review later” in the next day, found the misissuance problem, then revoked the certificate.

7. Issue P: Use of SM2 Algorithm (Nov 2015)

In November 2015, WoSign issued two certificates that have subject public keys which are for the [SM2 algorithm](#). SM2 is an elliptic-curve-based algorithm but it does not use the US NIST P-256, P-384, or P-521 curves. The CA/Browser Forum Baseline Requirements section 6.1.5 requires that only these three curves be used for elliptic curve keys in certs covered by the BRs.

In addition to including subjects keys using unapproved parameters, it seems these each share their serial number with another certificate for the same subject.

- [1st SM2 cert in crt.sh](#); [cert with same serial number in crt.sh](#)
- [2nd SM2 cert in crt.sh](#); [cert with same serial number in crt.sh](#)

Secondly, for the first pair of certs, the validity period is 4 years, which is 9 months longer than allowed by the BRs.

WoSign Response

7.1 What happened

WoSign issued two SM2 algorithm SSL certificates for testing in 2015, and also issued 4 SM2 algorithm certificates in 2016 for testing again.

Here is the 2016 issued SM2 certificate in crt.sh:

2016-01-13 sm2 signature: <https://crt.sh/?id=31753567>

2016-01-13 sm2 encryption: <https://crt.sh/?id=31753575>

2016-01-25 sm2 signature: <https://crt.sh/?id=31753571>

2016-01-25 m2 encryption: <https://crt.sh/?id=31753573>

7.2 Why this happened

For year 2015 issue, these two certificates were issued manually in the test lab since the CMS and PKI system can't issue this SM2 algorithm certificate, and doing manually, a human mistake issued these certificates exceeding the 39-month limit.

We used the same serial number as the RSA certificate (same subject) to test if we can setup a server side gateway that install this two type certificates, it can make the handshake automatically using different certificate based on the browser algorithm support.

The reason why these certificates were issued from a trusted root was for testing the real scenario in the Internet using standard browser without SM2 support and browser with SM2 support, and for the effect and probing if Windows can display the certificate path correctly.

For year 2016 issue, this is a small change of the previous one but this time not using the same serial number with the RSA certificate, these certificates were issued in the test lab manually again because the CMS and PKI system can't issue this SM2 algorithm certificate. These 4 SM2 certificates were for testing the effect for different key usage in browser side and in server side.

7.3 Current situation

The test is finished and there's no need to test any more.

To avoid future testing incidents, WoSign updated the internal test systems to avoid issuing test certificates from public trusted root that violate the BR.

If we need to issue SM2 certificate in the future, we will follow the BR policy to comply with BR.

8. Issue R: Purchase of StartCom (Nov 2015)

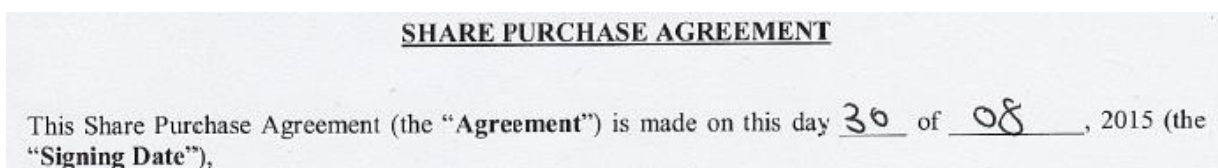
WoSign purchased the CA "StartCom" and did not disclose the transaction as a change of ownership, which we believe violates section 5 of the [Mozilla CA Certificate Maintenance Policy](#). Furthermore, when this clause was brought to their attention, they denied that any changes fell under it, and they attempted to suppress further information about the ownership transfer when it was brought to the community's attention.

Full details can be found in [the post in mozilla.dev.security.policy](#).

WoSign Response

WoSign is a subsidiary of Qihoo 360. In 2015, Richard noted to Qihoo 360 the intention of WoSign acquiring Startcom, which Qihoo 360 supported. The transaction was signed in August 2015. The final payment (for the first phase) was in September 2016:

- (1) The SPA signed at 30th August 2015:



- (2) The last payment must be done before 30th June 2016 (the "Release date"):

4.4.1 all funds

//removed confidential info//

shall be distributed on June 30, 2016 (the “**Release Date**”); and

- (3) But if some condition is not satisfied, then the last payment can be postponed till 31 Dec. 2016 or even later after 31 Dec 2016:

4. If until 31 December 2016, the Full Sale Condition has not been satisfied, the Purchaser shall

//removed confidential info//

and such proceedings are still pending on 31 December 2016, the Purchaser is entitled in its sole discretion to postpone the application of this Clause 4 until after 31 December 2016.

- (4) Due the early expose of the transaction, both parties agreed to end this deal and released the [news](#) at Sept 19th 2016. The final payment is paid at Sept. 13th 2016 that we can provide the bank transfer screenshot to Mozilla as an evidence.
- (5) This is the deal Phase One; the Phase Two will be finished till Dec. 31st 2018.

WoSign chose to announce the transaction after completion of its first phase of payment “Full Sale Condition” was completed.

WoSign understands that that there could have been an earlier date of announcement, before all payment milestones (and underlying handoff milestones) were met. Additionally, WoSign could have been more forthcoming in the interim process.

9. Issue S: Backdated SHA-1 Certs (January 2016)

WoSign has issued certificates after January 1st 2016 but backdated the notBefore date to be in December 2015. This has the effect of avoiding the blocks in browsers regarding SHA-1 certs issued after January 1st 2016. The number of certs affected is probably 67, but may be a few more or less.

WoSign Response

9.1. What happened

WoSign acknowledges it made a serious mistake of issuing 64 backdated certificates. It is the responsibility of the WoSign CEO to maintain technical and operational veracity according to CA standards (including no backdating) and there was a failure to do so.

9.2. Why this happened

WoSign was contacted by customers requesting SHA-1 and WoSign made a mistake to approve of backdated certificates. During mid 2016, StartCom was contacted by Tyro for a SHA-1 certificate and Richard Wang approved the issuance, which was a mistake.

We know we can't issue SHA-1 certificate from Jan 1st 2016, so our buy system closed it at Dec 30th 2015 according to Richard's SHA-1 deadline email, but don't close it in PKI system in case of some customer need it that can force to issue SHA-1 certificate if approved by someone that backdated to legal data (Dec 19th or Dec 20th 2015). It was a mistake to create a mechanism to create backdated certificates.

For the total 64 backdated certificates, there are 4 types:

- (1) The charged 42 backdated certificates are an intentional activity that we try to help the desperate customers since there are more than 3M users still using Windows XP sp2 in China. We like to make things simple that don't realize how serious this solution was.
- (2) The 16 free SSL certificate is ordered in 2015 with SHA-1 request, due to a bug that the CMS still post SHA-1 signing request to PKI, then the PKI used a SHA-1 creation mechanism to issue the certificate as NotBefore date to Dec 19th or Dec 20th 2015. This mechanism should not have been allowed or created;
- (3) The 4 certificates real signing date is Dec. 31st 2015, but due to the CMS and PKI closed the SHA-1, PKI system is triggered to SHA-1 creation mechanism to issue the certificate NotBefore date to Dec. 20th 2015;
- (4) The 2 backdated SHA-1 from StartEncrypt API is caused by Computest setup a SHA-1 parameter in the API that no any API document mentioned this SHA-1 parameter, then PKI system is triggered to SHA-1 creation mechanism to issue the certificate NotBefore date to Dec. 20th 2015.

64 backdated SHA-1 certificates crt.sh list		Issued Date	NotBefore	Ordered Date
Type (1) 6 EV SSL Certificates:				
1	https://crt.sh/?serial=6D24E483E27F55479C5C555B37745353	20160104	2015/12/19	2015/12/22 9:56
2	https://crt.sh/?serial=179A6D058F50116D62E422F49ABB8686	20160105	2015/12/20	2015/12/30 10:20
3	https://crt.sh/?serial=5ACF9A707E8E32D0A36F947ACD6C8981	20160118	2015/12/20	2016/1/18 9:49
4	https://crt.sh/?serial=15AE547B1136CA1074EEBADE368F9054	20151231	2015/12/20	2015/12/24 10:15
5	https://crt.sh/?serial=5DF26F6A29304CE8C559DBFFABBB37D1	20151231	2015/12/20	2015/12/29 12:45
6	https://crt.sh/?serial=5A47B7074267A7D44441618D84686547	20151231	2015/12/20	2015/12/28 15:47
Type (1) 25 OV SSL Certificates:				
7	https://crt.sh/?serial=64A5CB1252D1815DCDA80F6A8AB4B2D6	2016/1/14	2015/12/19	2015/12/15 11:22
8	https://crt.sh/?serial=5B76F7AC200C7DC63D90EEE8FF097438	2016/1/14	2015/12/19	2015/12/22 9:21
9	https://crt.sh/?serial=4704F1310520260DEEEEE8B8A25A4203C	2016/1/20	2015/12/19	2015/12/23 14:43
10	https://crt.sh/?serial=118C85C3A1227F5079DEE182CC300773	2016/1/28	2015/12/19	2016/1/27 19:20
11	https://crt.sh/?serial=41880FA83B3B6C1305065E97A7E277F3	2016/1/25	2015/12/19	2015/12/11 14:08
12	https://crt.sh/?serial=28768714C609EE69F5FA2228454F386E	2016/1/19	2015/12/19	2015/12/10 9:48
13	https://crt.sh/?serial=4AE2A808A61A75AB2C22D599B95D081C	2016/1/20	2015/12/20	2015/12/29 17:58

14	https://crt.sh/?serial=1C7BCBF96B98C06323F088E3CD246C6E	2016/1/5	2015/12/20	2015/12/18 15:07
15	https://crt.sh/?serial=21ED34DD27F68AF1C4B0127CCFF3F9F3	2016/1/5	2015/12/19	2015/12/11 17:28
16	https://crt.sh/?serial=3C87926F68D1554425BDA1D41DA40AFC	2016/1/7	2015/12/19	2015/12/25 22:30
17	https://crt.sh/?serial=23D95CE06F4EF0B2594FCC710ACA96C0	2016/1/29	2015/12/20	2016/1/26 15:20
18	https://crt.sh/?serial=10417A1677F41C7F56C3F56C0F769D85	2016/6/21	2015/12/20	2015/9/2 10:06
19	https://crt.sh/?serial=2F746B4FE99413AF382F8BC79C81A9FD	2016/1/26	2015/12/19	2016/1/25 12:32
20	https://crt.sh/?serial=5775779E2F6643E6FB894C3AC1AED637	2016/1/18	2015/12/20	2015/12/29 14:35
21	https://crt.sh/?serial=556E8E37CD8E998EAB77C456EA60D6F6	2016/1/25	2015/12/19	2015/12/23 15:03
22	https://crt.sh/?serial=3499767FDB189DEBD0533B2AAC8C14CE	2016/1/5	2015/12/20	2015/12/18 14:34
23	https://crt.sh/?serial=19EF10B37EBF22160A5D83BD71B2142F	2016/4/6	2015/12/20	2015/12/30 10:01
24	https://crt.sh/?serial=19393C09BEC4E34A7A60150C88501077	2016/1/6	2015/12/19	2015/11/30 10:10
25	https://crt.sh/?serial=1C8C6D346B2D870D3787CD2951BE9E37	2016/1/18	2015/12/20	2015/12/18 17:07
26	https://crt.sh/?serial=2EEFC08171E90244D165E1111E8026F	2016/1/18	2015/12/20	2016/1/18 15:37
27	https://crt.sh/?serial=243A148905774CBE7635DCFD4F750336	2016/1/27	2015/12/20	2015/12/24 15:52
28	https://crt.sh/?serial=391F8E0AB671DC17F75BA38B9A249D7D	2016/1/7	2015/12/20	2015/12/29 14:32
29	https://crt.sh/?serial=3351B3E3CDBEB2D7BACD201912A4B364	2016/1/19	2015/12/20	2015/12/29 11:30
30	https://crt.sh/?serial=32CA5614E8DB62B5871147DDF222E3B7	2016/1/6	2015/12/20	2015/12/23 11:42
31	https://crt.sh/?serial=18EF9C002EBD526F0F99B3C6DF2E166B	2016/1/27	2015/12/19	2015/12/28 14:45

Type (1) 11 Charged DV SSL Certificates:

32	https://crt.sh/?serial=5D50B970FC877982A783BB7356786C27	2016/01/28	2015/12/19	2015/12/29 10:44
33	https://crt.sh/?serial=4BEA1ACEF4D18D89BD3D9253FED1ABBF	2016/01/19	2015/12/20	2016/1/18 9:25
34	https://crt.sh/?serial=56A69DC69C971820C15546E50B0AADED	2016/01/19	2015/12/20	2016/1/18 9:20
35	https://crt.sh/?serial=2A2FD5FC4322D1E42FF0DBD108834157	2016/01/04	2015/12/20	2015/12/31 16:04
36	https://crt.sh/?serial=2D46C1357172F4012EBC06DC3374A1DE	2016/01/13	2015/12/20	2016/1/13 8:59
37	https://crt.sh/?serial=4A6D683F89B9095D6C120F6E7CAC374B	2016/01/22	2015/12/20	2016/1/22 15:28
38	https://crt.sh/?serial=3DFD3BEC6E39797D68C60D10F6634BF2	2016/01/18	2015/12/19	2016/1/16 17:35
39	https://crt.sh/?serial=3BE9494816A5C5F3B138B524CEFEB9F2	2016/01/18	2015/12/19	2016/1/18 14:33
40	https://crt.sh/?serial=4E0B75834F94A7722E0F927634DA7599	2016/01/05	2015/12/20	2015/12/29 10:18
41	https://crt.sh/?serial=35244A5CB3EE9404B9C61068617B408A	2016/01/04	2015/12/20	2016/1/4 16:09
42	https://crt.sh/?serial=523836668FF829B8BD6BF0F6609052A2	2016/01/26	2015/12/20	2015/12/25 14:34

Type (2) 16 Free DV SSL Certificates:

43	https://crt.sh/?serial=32B37DE8C629127E76434054531A6347	2016/01/20	2015/12/19	2015/11/28 13:05
44	https://crt.sh/?serial=2170445396884CDBB535ECC5D82C51ED	2016/04/29	2015/12/20	2015/9/5 22:42
45	https://crt.sh/?serial=4449C5A97BE5643FE97207EEF23318D4	2016/02/15	2015/12/20	2015/9/12 18:33
46	https://crt.sh/?serial=53526967E317332EFA386DEA051AD77E	2016/01/27	2015/12/20	2015/12/1 18:43
47	https://crt.sh/?serial=19C6D80C762AC585EB5C06D9E9AF835A	2016/01/27	2015/12/20	2015/11/16 20:32
48	https://crt.sh/?serial=3423292E4FCFFA22F982D470CC27E5EE	2016/01/22	2015/12/20	2015/12/3 19:43
49	https://crt.sh/?serial=5B635A4000D5A0692302455FCF1C7311	2016/02/01	2015/12/20	2015/11/2 13:57

50	https://crt.sh/?serial=1541691013567264F5B94BBD330F492A	2016/04/21	2015/12/20	2015/10/28 13:53
51	https://crt.sh/?serial=4F588C826570809095DB91CB31C3CE3F	2016/03/13	2015/12/20	2015/12/12 13:44
52	https://crt.sh/?serial=5C20933F08A6056B65EBEA291A1FB2ED	2016/04/23	2015/12/19	2015/11/26 9:19
53	https://crt.sh/?serial=5EEFC74E89A0BC2B7F413A5A55887C3C	2016/01/28	2015/12/20	2015/11/25 19:28
54	https://crt.sh/?serial=100FD9B985D145EA7FA14C57A59FFC5F	2016/06/02	2015/12/19	2015/11/30 16:30
55	https://crt.sh/?serial=6E937C2D656633FEDAE8111695958BC7	2016/03/15	2015/12/19	2015/11/25 19:43
56	https://crt.sh/?serial=6A87B982E4478E4CF4DF49E396AB5573	2016/03/21	2015/12/19	2015/11/17 21:35
57	https://crt.sh/?serial=47F9762177383469123846D22B1929A6	2016/01/24	2015/12/20	2015/10/30 14:03
58	https://crt.sh/?serial=4C0003167CB825D421DAA289BA6097B7	2016/01/22	2015/12/19	2015/10/17 14:07

Type (3) 4 Certificates issued at Dec. 31

59	https://crt.sh/?serial=383D5C00F511AD0BEE3A83DCA382FC8D	2015/12/31	2015/12/20	2015/12/31 15:00
60	https://crt.sh/?serial=4153D33AB18525012B5D461778E32327	2015/12/31	2015/12/20	2015/12/25 15:20
61	https://crt.sh/?serial=5EE14DC27F910CDB2BCDF39A8635AD11	2015/12/31	2015/12/20	2015/12/31 14:36
62	https://crt.sh/?serial=53ED5BC73D09C2A838482230EE552D4F	2015/12/31	2015/12/20	2015/12/31 14:43

Type (4) 2 certificates from StartEncrypt API

63	https://crt.sh/?serial=6565E1710A48FBBE1E2B61835C789C39	2016/06/23	2015/12/20	2016/06/23
64	https://crt.sh/?serial=6745ED57FE25880FB7D93A774310CF59	2016/06/28	2015/12/20	2016/06/28

9.3. What has been done

360's Corporate Development team has been notified to execute the process to legally separate Wosign and Startcom and to begin executing personnel reassignments. StartCom's chairman will be Xiaosheng Tan (Chief Security Officer of Qihoo 360). StartCom's CEO will be Inigo Barreira (formerly GM of StartCom Europe). Richard Wang will be relieved of his duties as CEO of WoSign.

For type (2)'s bug, we fixed it at Jan. 18th 2016, and we added more certificate parameter check before posting to CT log server. For type (4), we deleted the API bug code, see issue #11.

We closed the SHA-1 signing in the whole system at July 2nd 2016 after the issue #11 happened. To make transparency of this kind of case, WoSign decided to log all issued SSL certificate to Google CT log server at **July 4th 2015** that released a news:

https://www.wosign.com/english/News/2016_wosign_CT.htm, promised to all browsers that if the certificate issued after July 5th 2016 without SCT data embedded in the certificate, browsers can distrust this certificate.

9.4 Current situation

Wosign is in process of making legal and personnel changes in both WoSign and StartCom to ensure that both WoSign and StartCom have leadership that understand and follow the standards of running a CA.

WoSign fixed the bug and disabled the SHA-1 signing for SSL certificate, no more issued since July 2nd 2016.

10. Issue T: alicdn.com Misissuance (June 2016)

A certificate has been issued in June 2016 to alicdn.com which, it is claimed, was not requested by the owner of that domain. However, it has not yet been possible to confirm that this cert has been mis-issued because the owner of the private key has not been located. The domains in question currently use certificates from Symantec.

- [Cert on Github Gist](#)
- [Cert on crt.sh](#)

WoSign Response

10.1 What happened

This certificate is reported at the Mozilla mail list at August 26, 2016 1:13 PM that it claimed it is misissued certificate for “alicdn.com”

-----Original Message-----

From: dev-security-policy [mailto:dev-security-policy-bounces+1.1.1.1@wo-sign.com@lists.mozilla.org] On Behalf Of 233sec Team
Sent: Friday, August 26, 2016 1:13 PM
To: mozilla-dev-security-policy@lists.mozilla.org
Subject: Re: Incidents involving the CA WoSign

Wosign's Issue mechanism is high risking for large enterprise.
This is one prove:

<https://gist.github.com/xiaohuilam/8589f2dfa4c435bae4bf8dfe0984f69e>

Alicdn.com is the cdn asset domain name of Taobao/tmall who belong to alibaba, which are Chinese biggest online shopping websites.
With the fake cert's middle man attack, password stealing, information leaking...

dev-security-policy mailing list
dev-security-policy@lists.mozilla.org
<https://lists.mozilla.org/listinfo/dev-security-policy>

10.2 Why this happened

We checked our system, there were two orders related to domain “alicdn.com”, both orders passed the website control validation, since it is free DV SSL certificate that issued after it is a well-validated, no more manual check took.

The two certificates were post to CT log server, here is the crt.sh link:

<https://crt.sh/?id=31104164>

<https://crt.sh/?id=29884704>

10.2.1 Website control validation log

(1) Certificate: <https://crt.sh/?id=31104164>

2016-06-23 01:34:39, validation system received domain "alicdn.com" website control validation request, the URL is <http://alicdn.com/alicdn.com.html>, the domain random ID is 2e3baabe989fad9f143517796ed4941c13e7177b.

Validation system used GET/alicdn.com.html HTTP/1.1 to host: <http://alicdn.com:80/alicdn.com.html>, the server returns "HTTP/1.1 400 Bad Request". Then the validation system used POST to <http://alicdn.com:80/alicdn.com.html>, the sever returns "HTTP/1.1 200 OK", then system get the correct verification code that passed the website control validation, then issued the certificate. Here is the screen shot from validation system log:

```
2016-06-23 01:34:39 v_random=2e3baabe989fad9f143517796ed4941c13e7177b
2016-06-23 01:34:39 v_domain=alicdn.com
2016-06-23 01:34:39 verify file url=http://alicdn.com/alicdn.com.html
2016-06-23 01:34:39 Open connection to alicdn.com:80
2016-06-23 01:34:39 "GET /alicdn.com.html HTTP/1.1[\r][\n]"
2016-06-23 01:34:39 Adding Host request header
2016-06-23 01:34:39 "Content-Type: text/html;charset=utf-8[\r][\n]"
2016-06-23 01:34:39 "User-Agent: Jakarta Commons-HttpClient/3.1-rc1[\r][\n]"
2016-06-23 01:34:39 >> "Host: alicdn.com[\r][\n]"
2016-06-23 01:34:39 << "HTTP/1.1 400 Bad Request[\r][\n]"
2016-06-23 01:34:39 << "Server: Tengine[\r][\n]"
2016-06-23 01:34:39 << "Content-Type: application/xml[\r][\n]"
2016-06-23 01:34:39 << "Content-Length: 268[\r][\n]"
2016-06-23 01:34:39 << "Connection: keep-alive[\r][\n]"
2016-06-23 01:34:39 << "Date: Wed, 22 Jun 2016 17:34:37 GMT[\r][\n]"
2016-06-23 01:34:39 << "x-oss-request-id: 576ACC2DCDF474A7DFFA1A9F[\r][\n]"
2016-06-23 01:34:39 << "Via: cache18.l2et15-1[9,400-0,M], cache26.l2et15-1[9,0], cache2.cn297[66,400-0,M], cache4.cn297[67,0][\r][\n]"
2016-06-23 01:34:39 << "X-Cache: MISS TCP_MISS dirn:-2:-2[\r][\n]"
2016-06-23 01:34:39 << "X-Swift-SaveTime: Wed, 22 Jun 2016 17:34:37 GMT[\r][\n]"
2016-06-23 01:34:39 << "X-Swift-CacheTime: 1[\r][\n]"
2016-06-23 01:34:39 << "Timing-Allow-Origin: *[\r][\n]"
2016-06-23 01:34:39 << "EagleId: 8ccd4dcd14666168778245767e[\r][\n]"
2016-06-23 01:34:39 get failure!
2016-06-23 01:34:39 << "<?xml version='1.0' encoding='UTF-8'?>[\n]"
2016-06-23 01:34:39 << "<Error>[\n]"
2016-06-23 01:34:39 << "<Code>InvalidBucketName</Code>[\n]"
2016-06-23 01:34:39 << "<Message>The specified bucket is not valid.</Message>[\n]"
2016-06-23 01:34:39 << "<RequestId>576ACC2DCDF474A7DFFA1A9F</RequestId>[\n]"
2016-06-23 01:34:39 << "<HostId>alicdn.com</HostId>[\n]"
2016-06-23 01:34:39 << "<BucketName>alicdn.com.html</BucketName>[\n]"
2016-06-23 01:34:39 << "</Error>[\n]"
2016-06-23 01:34:39 Releasing connection back to connection manager.
2016-06-23 01:34:39 Open connection to alicdn.com:80
2016-06-23 01:34:39 >> "POST /alicdn.com.html HTTP/1.1[\r][\n]"
2016-06-23 01:34:39 Adding Host request header
2016-06-23 01:34:39 >> "User-Agent: Jakarta Commons-HttpClient/3.1-rc1[\r][\n]"
2016-06-23 01:34:39 >> "Host: alicdn.com[\r][\n]"
2016-06-23 01:34:39 >> "Content-Length: 0[\r][\n]"
2016-06-23 01:34:39 Request body has not been specified
2016-06-23 01:34:39 << "HTTP/1.1 200 OK[\r][\n]"
2016-06-23 01:34:39 << "Server: Tengine[\r][\n]"
2016-06-23 01:34:39 << "Content-Type: text/html[\r][\n]"
2016-06-23 01:34:39 << "Content-Length: 60[\r][\n]"
2016-06-23 01:34:39 << "Connection: keep-alive[\r][\n]"
2016-06-23 01:34:39 << "Date: Wed, 22 Jun 2016 17:34:38 GMT[\r][\n]"
2016-06-23 01:34:39 << "Via: cache61.l2nu16-1[159,200-0,M], cache52.l2nu16-1[160,0], cache2.cn297[248,200-0,M], cache8.cn297[250,0][\r][\n]"
2016-06-23 01:34:39 << "X-Cache: MISS TCP_MISS dirn:-2:-2[\r][\n]"
2016-06-23 01:34:39 << "X-Swift-SaveTime: Wed, 22 Jun 2016 17:34:38 GMT[\r][\n]"
2016-06-23 01:34:39 << "X-Swift-CacheTime: 0[\r][\n]"
2016-06-23 01:34:39 << "Timing-Allow-Origin: *[\r][\n]"
2016-06-23 01:34:39 << "EagleId: 8ccd4dd014666168779512340e[\r][\n]"
2016-06-23 01:34:39 Buffering response body
2016-06-23 01:34:39 << "bE0vdXpWdWQyQWJoT012RGx2bG51VUtBVG9JUK1ZamxzbxXlycUQxVmJDTT0[\n]"
2016-06-23 01:34:39 Releasing connection back to connection manager.
2016-06-23 01:34:39 Default charset used: ISO-8859-1
2016-06-23 01:34:39 result=bE0vdXpWdWQyQWJoT012RGx2bG51VUtBVG9JUK1ZamxzbxXlycUQxVmJDTT0
2016-06-23 01:34:39 fileContent=bE0vdXpWdWQyQWJoT012RGx2bG51VUtBVG9JUK1ZamxzbxXlycUQxVmJDTT0
2016-06-23 01:34:40 result=0||success
2016-06-23 01:34:40 Cost time=838ms
```

(2) Certificate: <https://crt.sh/?id=29884704>

2016-06-23 09:17:01, validation system received domain "alicdn.com" website control validation request, the URL is "http://alicdn.com/alicdn.com.html", domain random ID is bf5d1e3cc3f29b599c20d2280431d70b7ddc1a58.

Validation system used GET to http://alicdn.com:80/alicdn.com.html, the server returns "HTTP/1.1 400 Bad Request". Then the validation system used POST to http://alicdn.com:80/alicdn.com.html, the sever returns "HTTP/1.1 400 Bad Request".

Then system used GET **https://alicdn.com:443/alicdn.com.html**, the server returns "HTTP/1.1 200 OK", then system gets the correct verification code that passed the website control validation, and then issued this certificate.

Here is the screen shot from validation system log:

```
2016-06-23 09:17:01 v_random=bf5d1e3cc3f29b599c20d2280431d70b7ddc1a58
2016-06-23 09:17:01 v_domain=alicdn.com
2016-06-23 09:17:01 punycodeDomain=alicdn.com
2016-06-23 09:17:01 v_name=https://alicdn.com/alicdn.com.html
2016-06-23 09:17:01 verify file url=https://alicdn.com/alicdn.com.html
2016-06-23 09:17:02 get verify file succeed,strRep=RjVHeVpsdnYxdlo1U1UwWWZrV1JlN204ZkFJT1RlY3RQVTJ5U0w0OEpyWT0
2016-06-23 09:17:02 fileContent=RjVHeVpsdnYxdlo1U1UwWWZrV1JlN204ZkFJT1RlY3RQVTJ5U0w0OEpyWT0
2016-06-23 09:17:02 Call cmsWebVerification, Cost time=627ms
```

In this website control validation, system doesn't verify the domain in the certificate. Some website disabled port 80 for security, so we always try 443 if 80 fails.

10.2.2 DNS resolution record

Here is the dig record in the validation server:

```
[root@localhost ~]# dig alicdn.com +trace
```

```
; <<>> DiG 9.7.3-P3-RedHat-9.7.3-8.P3.el6 <<>> alicdn.com +trace
```

```
;; global options: +cmd
```

```
.      1225 IN    NS     j.root-servers.net.
.      1225 IN    NS     g.root-servers.net.
.      1225 IN    NS     i.root-servers.net.
.      1225 IN    NS     e.root-servers.net.
.      1225 IN    NS     l.root-servers.net.
.      1225 IN    NS     f.root-servers.net.
.      1225 IN    NS     h.root-servers.net.
.      1225 IN    NS     m.root-servers.net.
.      1225 IN    NS     d.root-servers.net.
```

```
.          1225  IN    NS    k.root-servers.net.
.          1225  IN    NS    c.root-servers.net.
.          1225  IN    NS    b.root-servers.net.
.          1225  IN    NS    a.root-servers.net.
```

;; Received 228 bytes from 101.226.4.6#53(101.226.4.6) in 52 ms

```
com.       172800 IN    NS    a.gtld-servers.net.
com.       172800 IN    NS    b.gtld-servers.net.
com.       172800 IN    NS    c.gtld-servers.net.
com.       172800 IN    NS    d.gtld-servers.net.
com.       172800 IN    NS    e.gtld-servers.net.
com.       172800 IN    NS    f.gtld-servers.net.
com.       172800 IN    NS    g.gtld-servers.net.
com.       172800 IN    NS    h.gtld-servers.net.
com.       172800 IN    NS    i.gtld-servers.net.
com.       172800 IN    NS    j.gtld-servers.net.
com.       172800 IN    NS    k.gtld-servers.net.
com.       172800 IN    NS    l.gtld-servers.net.
com.       172800 IN    NS    m.gtld-servers.net.
```

;; Received 488 bytes from 193.0.14.129#53(k.root-servers.net) in 168 ms

```
alicdn.com. 172800 IN    NS    nsp.alibabaonline.com.
alicdn.com. 172800 IN    NS    ns8.alibabaonline.com.
alicdn.com. 172800 IN    NS    nshz.alibabaonline.com.
alicdn.com. 172800 IN    NS    nsp2.alibabaonline.com.
```

;; Received 244 bytes from 192.43.172.30#53(i.gtld-servers.net) in 10158 ms

```
alicdn.com. 300  IN    A    140.205.77.240
```

```
alicdn.com. 300  IN    A    115.238.23.240
```

```
alicdn.com. 172800 IN    NS    nsp2.alibabaonline.com.
alicdn.com. 172800 IN    NS    nsp.alibabaonline.com.
alicdn.com. 172800 IN    NS    ns8.alibabaonline.com.
alicdn.com. 172800 IN    NS    nshz.alibabaonline.com.
```

;; Received 276 bytes from 140.205.2.184#53(nsp2.alibabaaonline.com) in 34 ms

Here is the nslookup using Google DNS:

```
C:\Users\Richard>nslookup - 8.8.8.8
默认服务器: google-public-dns-a.google.com
Address: 8.8.8.8

> alicdn.com
服务器: google-public-dns-a.google.com
Address: 8.8.8.8

非权威应答:
名称: alicdn.com
Addresses: 115.238.23.240
           140.205.77.240
```

10.2.3 Other related information

We noticed Aliyun (Alibaba Cloud) team to check this problem, they confirmed this order is not from Aliyun, and they checked this case that confirmed this website control validation is done successfully by short time traffic hijack, but we don't have more details for this hijack. We gave the Alibaba Cloud related person email to Mozilla to contact Alibaba Cloud directly.

And the incident reporter also confirmed this is not a validation problem, he thinks this is a problem that we must do more human validation.

-----Original Message-----

From: dev-security-policy [mailto:dev-security-policy-bounces+richard=wosign.com@lists.mozilla.org] On Behalf Of 233sec Team
Sent: Saturday, August 27, 2016 8:34 AM
To: mozilla-dev-security-policy@lists.mozilla.org
Subject: Re: Incidents involving the CA WoSign

Not vulnerabilities mentioned in this thread, but a Human-Audit weak process.
Detail you can see the reply content i send to Mr.Wang

Here is his email to Mr. Wang:

From: 蓝小灰 [mailto:incoft@gmail.com]
Sent: Saturday, August 27, 2016 8:38 AM
To: Richard Wang [mailto:richard.wang@wosign.com]>
Subject: Re: Incidents involving the CA WoSign

Yes, issued from wosign order system.
But this a high risk domain which can brings big websites' Middle-man Problem.

With important domains(alexa top 5000),
Don't your company do anything manually to review the certificate?
Don't do some phone callback mention?

10.3 What has been done

After we got report, we revoked this certificate and add keyword “alicdn” “aliyun” to our Flag-Reject system (alibaba is in the system), it will be rejected for those 3 domain for class 1 and Class 2 SSL certificate in the future.

Keyword	Level	Create Date	Enable	Sub Domain	Top Domain	Root Domain
alicdn	Class2	2016-08-29 09:53:13	✔	r	r	r
alicdn	Class3	2016-08-29 09:53:55	✔	f	f	f
alicdn	Class4	2016-08-29 09:54:16	✔	f	f	f
alicdn	Class1	2016-08-29 09:53:00	✔	r	r	r

Keyword	Level	Create Date	Enable	Sub Domain	Top Domain	Root Domain
aliyun	Class1	2016-08-29 09:55:22	✔	r	r	r
aliyun	Class2	2016-08-29 09:55:48	✔	r	r	r
aliyun	Class3	2016-08-29 09:56:06	✔	f	f	f
aliyun	Class4	2016-08-29 09:56:25	✔	f	f	f

Keyword	Level	Create Date	Enable	Sub Domain	Top Domain	Root Domain
alibaba	Class4	2016-02-01 11:52:15	✔	r	f	r
alibaba	Class3	2016-02-01 11:52:07	✔	r	f	r
alibaba	Class2	2016-02-01 11:52:00	✔	r	r	r
alibaba	Class1	2016-02-01 11:51:48	✔	r	r	r

Considering the website control validation method has potential risk, we have closed this method at **Aug. 27th 2016** even the BR allow this method. There are many famous Internet service providers provide subdomain to its customer, we can't add all of their domains to our Flag-Reject system. So we decided to close this validation method, only support domain control validation.

WoSign doesn't think this case is a misissuance mistake since it passed the website control validation. After we got report, we revoked the two related certificates, and added Alibaba related domain to Flag-Rejection system to prevent it will not happen in the future. And we even closed the website control validation method to all subscribers to prevent other Internet Service Provider's domain certificate is mis-issued.

11. Issue V: StartEncrypt (July 2016)

(a.k.a. "Issue 2")

In July 2016, it became clear that there were some problems with the StartEncrypt automatic issuance service recently deployed by the CA StartCom. This was a StartCom-branded service and was not publicized as being able to issue certificates from WoSign. However, changing a simple API parameter in the POST request on the submission page changed the intermediate/root certificate to which the resulting certificate chained up.

WoSign Response

11.1 What happened

Computest reported this bug in June 30th 2016 that using StartEncrypt API can issue SHA-1 certificates from WoSign intermediate CA backdating the certificates to Dec. 20th 2015.

11.2 Why this happened

This is caused by Computest setup a SHA-1 parameter in the API that no any API document mentioned this SHA-1 parameter, then PKI system is triggered to SHA-1 creation mechanism to issue the certificate NotBefore date to Dec. 20th 2015.

11.3 What has been done

WoSign deleted this bug code in API instantly, and closed the API service and deleted the API domain name resolution and stopped to use StartEncrypt service. Stopping this API service is the quickest way to prevent this case in the future.

12. Issue X: Unpatched Software (September 2016)

The first WoSign [incident report](#), produced in response to other issues listed on this page, has a screenshot of a dig query from their validation server. The dig program is part of the bind-utils package, and the output of dig appears to show a bind-utils version of 9.7.3-8.P3.el6. The "el6" shows that this is a version built for Red Hat Enterprise Linux 6. This version of bind-utils was released in [December 2011](#) and so is very out of date.

WoSign Response

12.1 What happened

The Domain validation system OS don't patch to newly version.

12.2 Why this happened

This is an internal server that just have outbound traffic to check the whois database, not inbound traffic is allowed.

12.3 What has been done

WoSign patched all related servers, and WoSign will enhance this OS update practice.

Part III Response to the proposed action

Mozilla:

We plan to distrust only newly-issued certificates to try and reduce the impact on web users, as both of these CA brands have substantial outstanding certificate corpuses. Our proposal is that we determine “newly issued” by examining the notBefore date in the certificates.

It is true that this date is chosen by the CA and therefore WoSign/StartCom could back-date certificates to get around this restriction. And there is, as we have explained, evidence that they have done this in the past. However, many eyes are on the Web PKI and if such additional back-dating is discovered (by any means), Mozilla will immediately and permanently revoke trust in all WoSign and StartCom roots.

This distrust would remain for a minimum of 1 year. After that time, WoSign/StartCom may be readmitted to the Mozilla trust program, under the following conditions:

- A Point-In-Time Readiness Audit (PITRA) from a Mozilla-agreed WebTrust auditor;
- A full code security audit of their issuing infrastructure from a Mozilla-chosen security auditor;
- 100% embedded CT for all issued certificates, logged to at least one Google and one non-Google log not controlled by WoSign/StartCom;
- Going through the normal Mozilla inclusion process.

Mozilla believes that continued public trust in the correct working of the CA certificate system is vital to the health of the Internet, and we will not hesitate to take steps such as those outlined above to maintain that public trust. We believe that the behavior documented here would be unacceptable in any CA, whatever their nationality, business model or position in the market.

WoSign:

WoSign and Qihoo 360 understand the severity of the issues in the Mozilla report.

Due to the severity of issues noted within, the decision has been made to address the above three areas as they fall under the areas of 1) leadership/authority in WoSign and StartCom, 2) operational/business process and 3) technology.

Additionally, we note that StartCom has been operating as a compliant, separate CA for many years and the only noted issue with StartCom (two backdated certificates issued in July 2016) was an action approved by WoSign CEO Richard Wang. Before this, the leadership, business operations, and technology have been operating in compliance for many years and for many customers.

Hence, we would like to have the impact to WoSign and StartCom be considered separately.

To address these issues, Qihoo 360 has made the the decision to legally separate StartCom from under WoSign to directly report into Qihoo 360 and additionally relieve WoSign CEO Richard Wang of his current responsibilities.

StartCom's chairman will be Xiaosheng Tan (Chief Security Officer of Qihoo 360). StartCom's CEO will be Inigo Barreira (formerly GM of Startcom Europe). Richard Wang will be relieved of his duties as CEO of WoSign.

360's Corporate Development team has been notified to initiate and execute the process to legally separate WoSign and StartCom.

StartCom is in process to provide a go-forward plan of new personnel and of separating operations and technology from WoSign.

We will provide documentation of our plan in the near term (StartCom) and also verifiable post-planning update and documentation of the changes.

We completely agree that keep the global Internet security is very important for all related stakeholders including all CAs.

Thank you to Mozilla for its consideration of WoSign and StartCom's current subscribers' benefit. We appreciate that. Many customers in China find it important to use a domestic CA for purposes of security.

Finally, we would like to thank all related parties for help WoSign to fix the problem. We will continue to strengthen our communication with international bodies, provide forthcoming communication, explore more effective cooperation mechanisms, and institute integrity and heightened quality across our leadership, operations, and technology to protect our customers' interests.

Thank you.

WoSign CA Limited