The Dark Side of Certificate Transparency

How to abuse ‘em for phun and profit

Aan Wahyu

https://petruknisme.com

Certificate Authorities

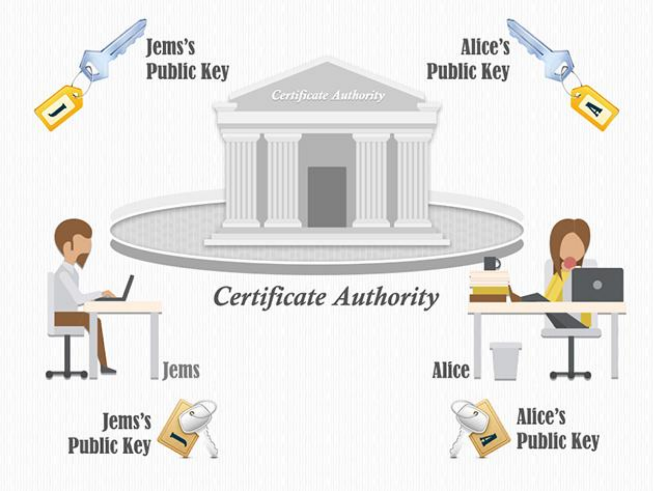
COMODO
Creating Trust Online®

digicert®

Let's Encrypt

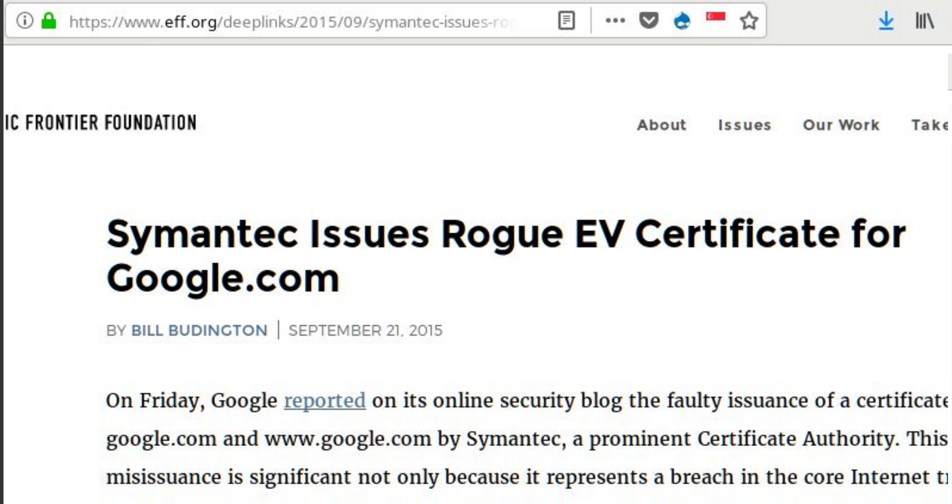
Definition of CA

A Certificate Authority is a trusted third party entity that issues digital certificates and manages the public keys and credentials for data encryption for the end user.



SSL-Chain-of-Trust is Broken & Abused

Incident



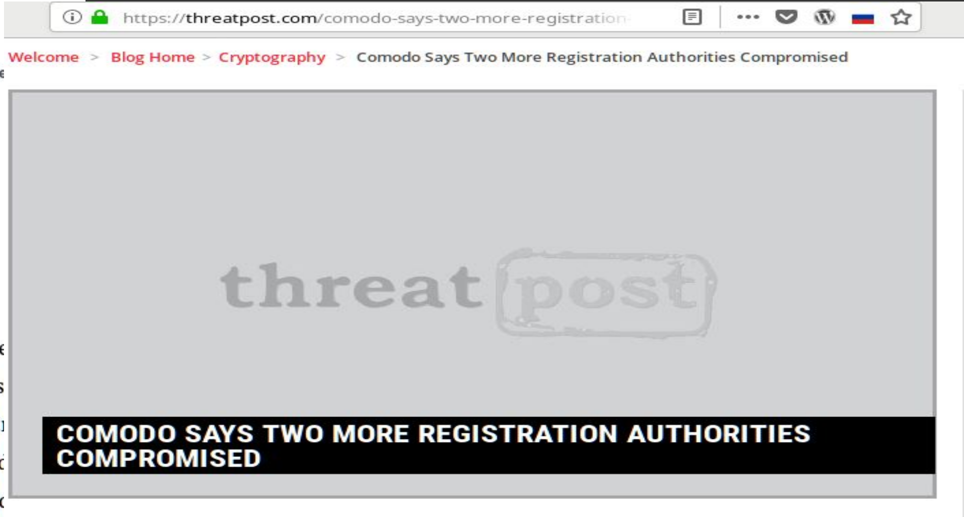
https://www.eff.org/deeplinks/2015/09/symantec-issues-ro

IC FRONTIER FOUNDATION

Symantec Issues Rogue EV Certificate for Google.com

BY BILL BUDINGTON | SEPTEMBER 21, 2015

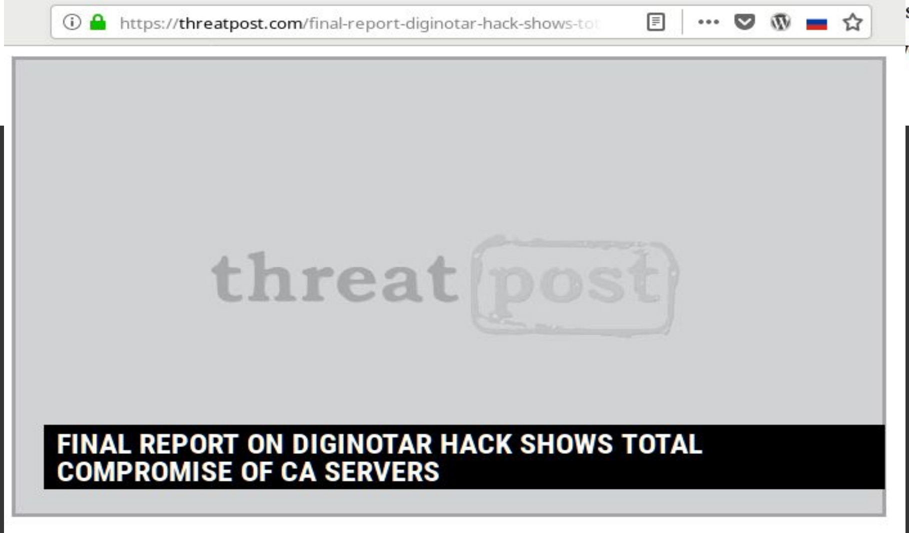
On Friday, Google reported on its online security blog the faulty issuance of a certificate google.com and www.google.com by Symantec, a prominent Certificate Authority. This misissuance is significant not only because it represents a breach in the core Internet tr



https://threatpost.com/comodo-says-two-more-registration

Welcome > Blog Home > Cryptography > Comodo Says Two More Registration Authorities Compromised
threat post

COMODO SAYS TWO MORE REGISTRATION AUTHORITIES COMPROMISED

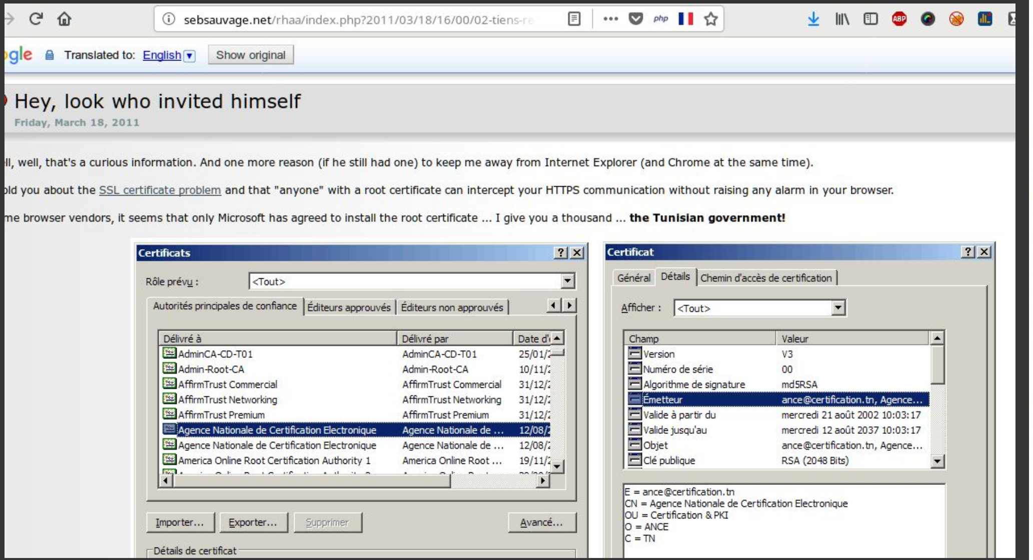


https://threatpost.com/final-report-diginotar-hack-shows-to

FINAL REPORT ON DIGINOTAR HACK SHOWS TOTAL COMPROMISE OF CA SERVERS

- https://threatpost.com/comodo-says-two-more-registration-authorities-compromised-033011/75083/
- https://www.eff.org/deeplinks/2015/09/symantec-issues-rogue-ev-certificate-googlecom
- https://threatpost.com/final-report-diginotar-hack-shows-total-compromise-ca-servers-103112/77170/

Government involved


sebsauvage.net/rhaa/index.php?2011/03/18/16/00/02-tiens-re
gle Translated to: English
Show original

Hey, look who invited himself

Friday, March 18, 2011

ll, well, that's a curious information. And one more reason (if he still had one) to keep me away from Internet Explorer (and Chrome at the same time).

old you about the SSL certificate problem and that "anyone" with a root certificate can intercept your HTTPS communication without raising any alarm in your browser.

ne browser vendors, it seems that only Microsoft has agreed to install the root certificate ... I give you a thousand ... **the Tunisian government!**

Certificats

Rôle prévu : <Tout>

Autorités principales de confiance | Éditeurs approuvés | Éditeurs non approuvés

Délivré à	Délivré par	Date d'
AdminCA-CD-T01	AdminCA-CD-T01	25/01/2
Admin-Root-CA	Admin-Root-CA	10/11/2
AffirmTrust Commercial	AffirmTrust Commercial	31/12/2
AffirmTrust Networking	AffirmTrust Networking	31/12/2
AffirmTrust Premium	AffirmTrust Premium	31/12/2
Agence Nationale de Certification Electronique	Agence Nationale de ...	12/08/2
Agence Nationale de Certification Electronique	Agence Nationale de ...	12/08/2
America Online Root Certification Authority 1	America Online Root ...	19/11/2

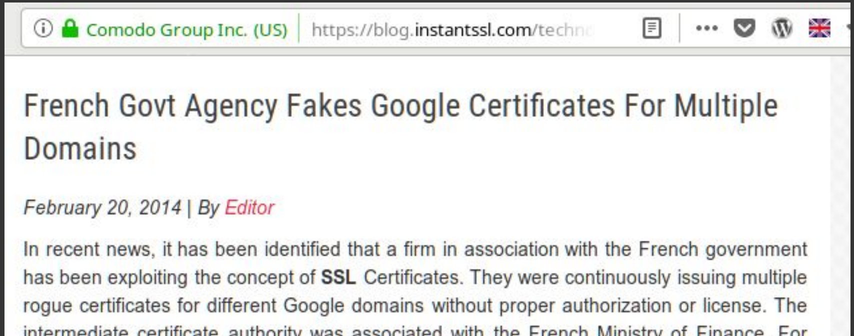
Certificat

Général | Détails | Chemin d'accès de certification

Afficher : <Tout>

Champ	Valeur
Version	V3
Numéro de série	00
Algorithme de signature	md5RSA
Émetteur	ance@certification.tn, Agence...
Valide à partir du	mercredi 21 août 2002 10:03:17
Valide jusqu'au	mercredi 12 août 2037 10:03:17
Objet	ance@certification.tn, Agence...
Clé publique	RSA (2048 Bits)

E = ance@certification.tn
CN = Agence Nationale de Certification Electronique
OU = Certification & PKI
O = ANCE
C = TN



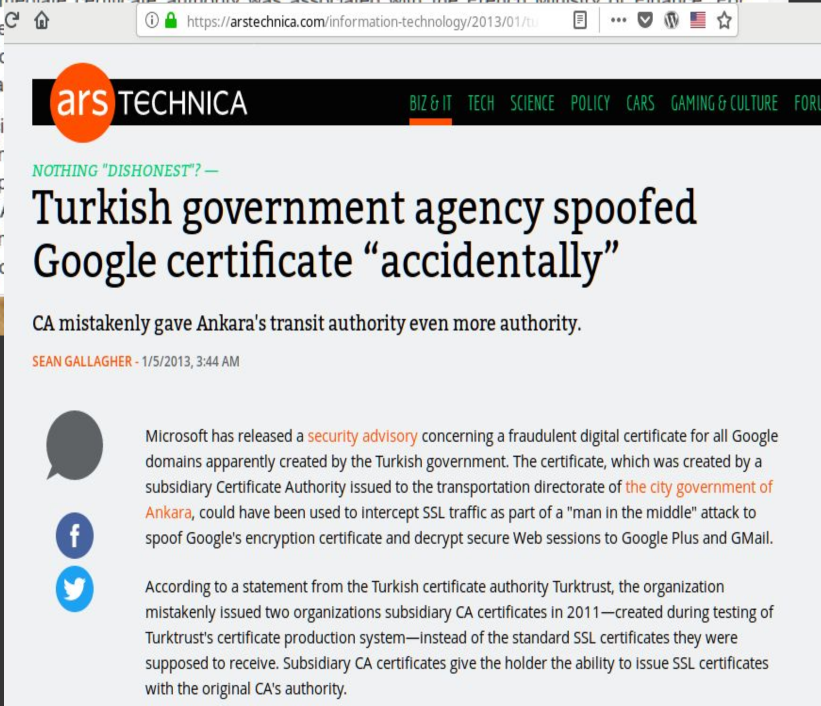
French Govt Agency Fakes Google Certificates For Multiple Domains

February 20, 2014 | By Editor

In recent news, it has been identified that a firm in association with the French government has been exploiting the concept of **SSL** Certificates. They were continuously issuing multiple rogue certificates for different Google domains without proper authorization or license. The intermediate certificate authority was associated with the French Ministry of Finance. For

some
found
bega

Offici
engir
comp
The
agair
autho



NOTHING "DISHONEST"? —

Turkish government agency spoofed Google certificate “accidentally”

CA mistakenly gave Ankara's transit authority even more authority.

SEAN GALLAGHER - 1/5/2013, 3:44 AM

Microsoft has released a security advisory concerning a fraudulent digital certificate for all Google domains apparently created by the Turkish government. The certificate, which was created by a subsidiary Certificate Authority issued to the transportation directorate of the city government of Ankara, could have been used to intercept SSL traffic as part of a "man in the middle" attack to spoof Google's encryption certificate and decrypt secure Web sessions to Google Plus and GMail.

According to a statement from the Turkish certificate authority Turktrust, the organization mistakenly issued two organizations subsidiary CA certificates in 2011—created during testing of Turktrust's certificate production system—instead of the standard SSL certificates they were supposed to receive. Subsidiary CA certificates give the holder the ability to issue SSL certificates with the original CA's authority.

- http://sebsauvage.net/rhaa/index.php?2011/03/18/16/00/02-tiens-regardez-qui-s-est-invite
- https://blog.instantssl.com/technology/french-govt-agency-fakes-google-certificates/
- https://arstechnica.com/information-technology/2013/01/turkish-government-agency-spoofed-google-certificate-accidentally/

Can we trust Certificate Authorities?

Blindly?

I think, No

CERTIFICATE TRANSPARENCY
TO THE RESCUE
imgflip.com

Certificate
Transparency

Definition

Certificate Transparency is an open framework for monitoring and auditing the certificates issued by Certificate Authorities in near real-time.

How CT works

Certificate Transparency adds three new functional components to the current SSL certificate system:

- Certificate logs
- Certificate monitors
- Certificate auditors

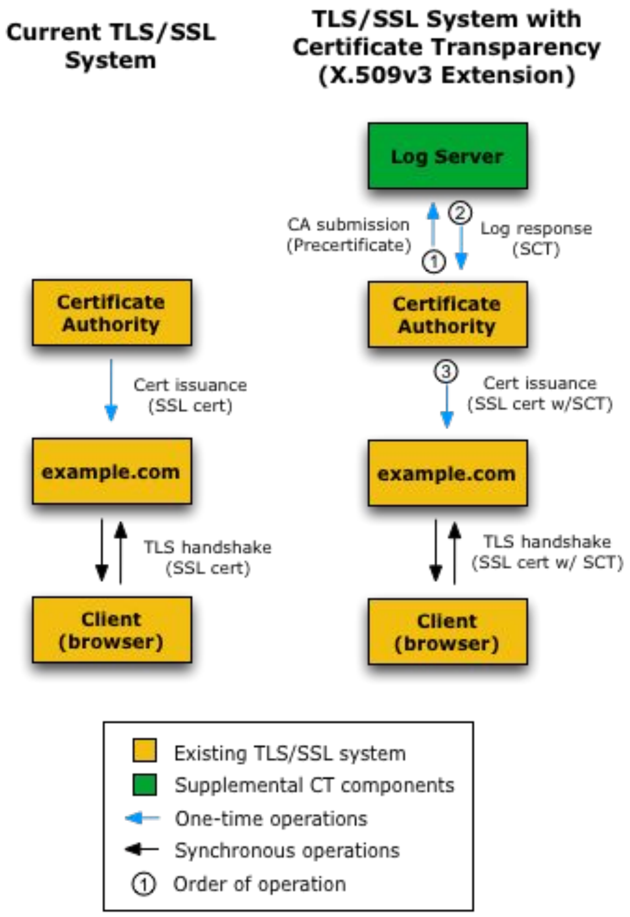
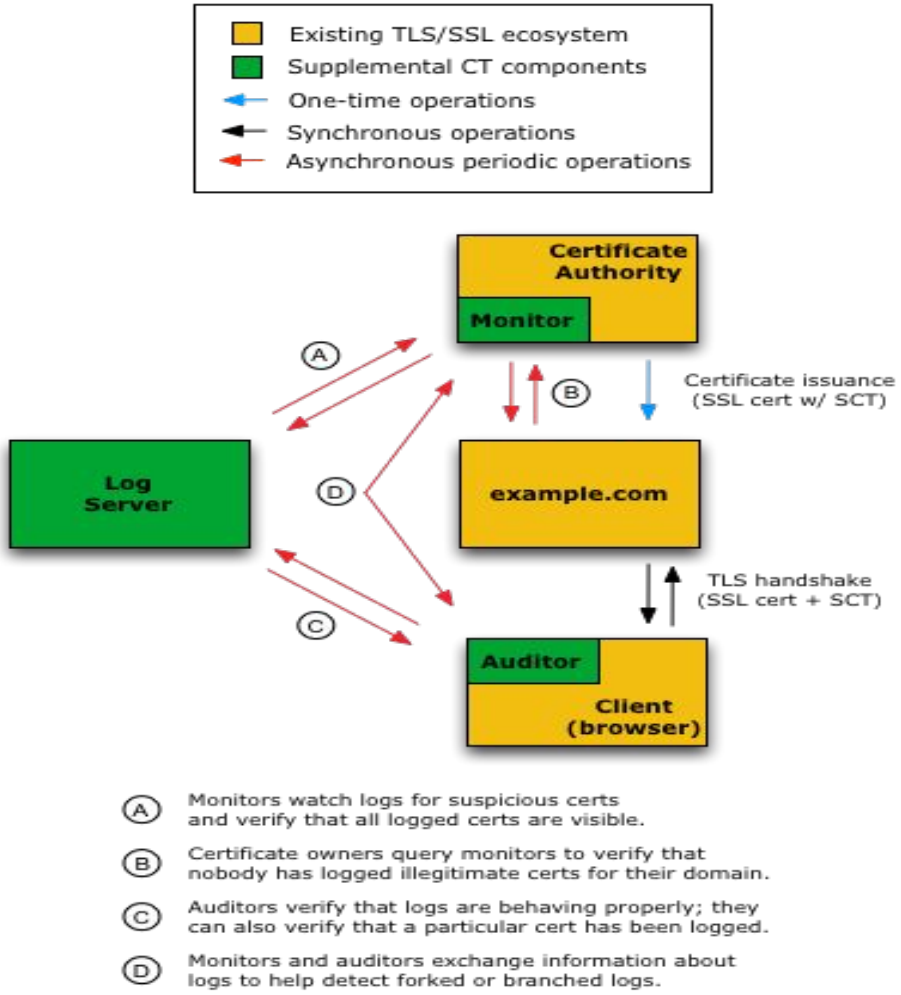


Figure 1

Typical System Configuration


🟨 Existing TLS/SSL ecosystem
🟩 Supplemental CT components
← One-time operations
← Synchronous operations
← Asynchronous periodic operations
Certificate Authority
Monitor
A
B
Certificate issuance (SSL cert w/ SCT)
Log Server
D
example.com
TLS handshake (SSL cert + SCT)
C
Auditor
Client (browser)

Ⓐ Monitors watch logs for suspicious certs and verify that all logged certs are visible.

Ⓑ Certificate owners query monitors to verify that nobody has logged illegitimate certs for their domain.

Ⓒ Auditors verify that logs are behaving properly; they can also verify that a particular cert has been logged.

Ⓓ Monitors and auditors exchange information about logs to help detect forked or branched logs.

Figure 3

RFC 6962

- Public accessible and append-only certificate logging
- Cryptography assured
- Open to All

https://tools.ietf.org/html/rfc6962

Goals

- Detect, audit, monitor rogue certificates
- Quickly identify fraudulent certificates
- Protect users from being duped by certs
- Chain of trust

https://tools.ietf.org/html/rfc6962

Server Logger

- Google:
  - 3 open for all
  - 1 let’s encrypt
  - 1 non let’s encrypt
- Cloudflare, DigiCert, Comodo
- Symantec, WoSign, CNNIC, StartCom = caught cheating
- Some smaller company

- https://www.gstatic.com/ct/log_list/log_list.js
  on
- https://lab.dsst.io/slides/33c3/slides/8167.pd

CT logs by design contain all the certificates and logs are available publicly so anyone can look through these logs.

Disadvantage

Privacy:

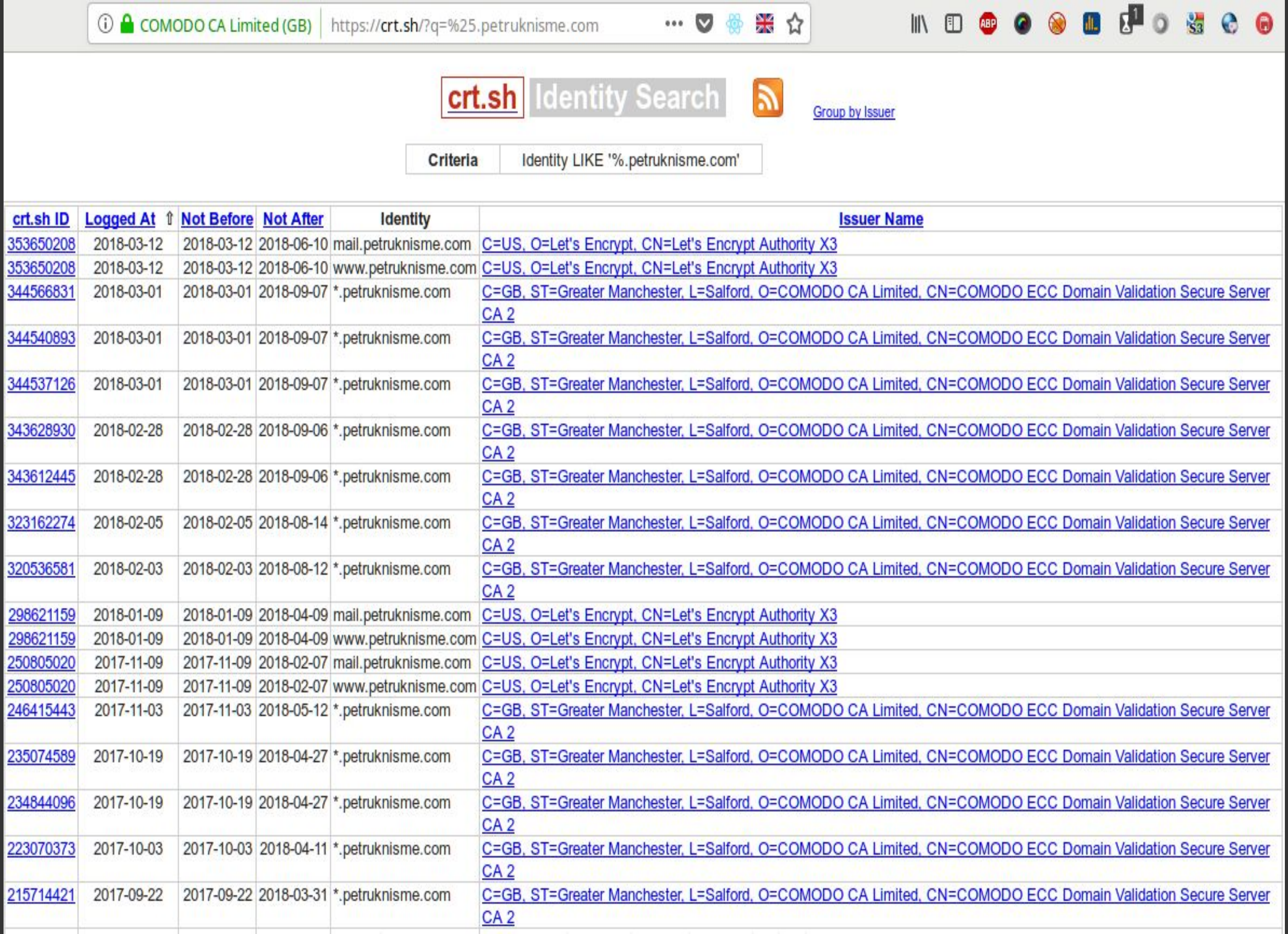
- People can mapping internal or external host
- Gather a lot of information about an organisation’s infrastructure such as internal domains, email addresses.
- Great for reconaissance(bug bounty? :P)

So, CT will becomes our source of data(?)

CT Search Engine

- https://crt.sh/
- https://censys.io/
- https://developers.facebook.com/tools/ct/
- https://www.google.com/transparencyreport/https/ct/

crt.sh



COMODO CA Limited (GB) https://crt.sh/?q=%25.petruknisme.com

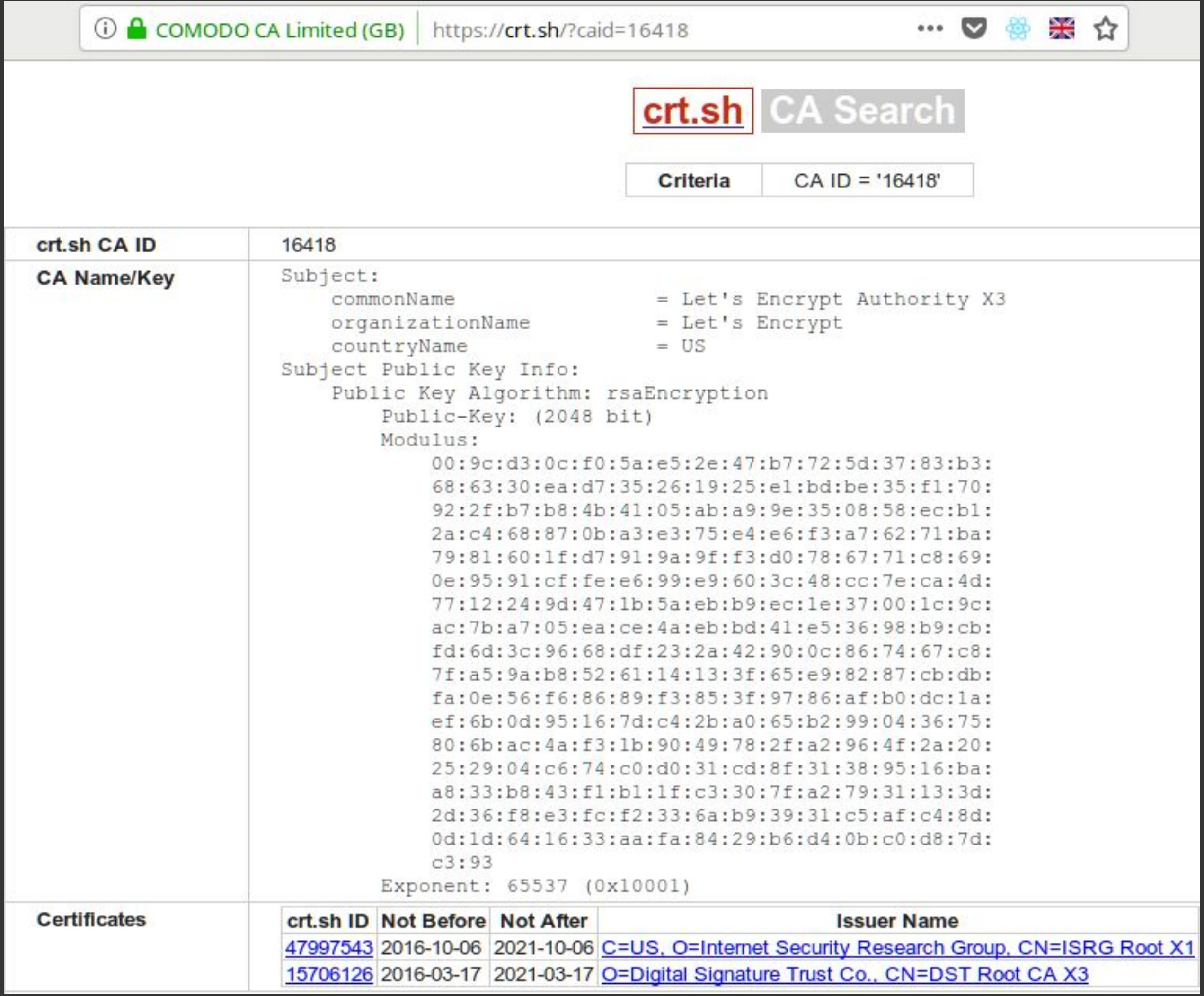
crt.sh Identity Search

Group by Issuer

Criteria	Identity LIKE '%.petruknisme.com'

crt.sh ID	Logged At ⇧	Not Before	Not After	Identity	Issuer Name
353650208	2018-03-12	2018-03-12	2018-06-10	mail.petruknisme.com	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
353650208	2018-03-12	2018-03-12	2018-06-10	www.petruknisme.com	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
344566831	2018-03-01	2018-03-01	2018-09-07	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
344540893	2018-03-01	2018-03-01	2018-09-07	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
344537126	2018-03-01	2018-03-01	2018-09-07	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
343628930	2018-02-28	2018-02-28	2018-09-06	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
343612445	2018-02-28	2018-02-28	2018-09-06	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
323162274	2018-02-05	2018-02-05	2018-08-14	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
320536581	2018-02-03	2018-02-03	2018-08-12	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
298621159	2018-01-09	2018-01-09	2018-04-09	mail.petruknisme.com	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
298621159	2018-01-09	2018-01-09	2018-04-09	www.petruknisme.com	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
250805020	2017-11-09	2017-11-09	2018-02-07	mail.petruknisme.com	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
250805020	2017-11-09	2017-11-09	2018-02-07	www.petruknisme.com	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
246415443	2017-11-03	2017-11-03	2018-05-12	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
235074589	2017-10-19	2017-10-19	2018-04-27	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
234844096	2017-10-19	2017-10-19	2018-04-27	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
223070373	2017-10-03	2017-10-03	2018-04-11	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2
215714421	2017-09-22	2017-09-22	2018-03-31	*.petruknisme.com	C=GB, ST=Greater Manchester, L=Salford, O=COMODO CA Limited, CN=COMODO ECC Domain Validation Secure Server CA 2

crt.sh



Criteria	CA ID = '16418'

crt.sh CA ID 16418

CA Name/Key

```
Subject:
    commonName                = Let's Encrypt Authority X3
    organizationName          = Let's Encrypt
    countryName               = US
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
        Modulus:
            00:9c:d3:0c:f0:5a:e5:2e:47:b7:72:5d:37:83:b3:
            68:63:30:ea:d7:35:26:19:25:e1:bd:be:35:f1:70:
            92:2f:b7:b8:4b:41:05:ab:a9:9e:35:08:58:ec:b1:
            2a:c4:68:87:0b:a3:e3:75:e4:e6:f3:a7:62:71:ba:
            79:81:60:1f:d7:91:9a:9f:f3:d0:78:67:71:c8:69:
            0e:95:91:cf:fe:e6:99:e9:60:3c:48:cc:7e:ca:4d:
            77:12:24:9d:47:1b:5a:eb:b9:ec:1e:37:00:1c:9c:
            ac:7b:a7:05:ea:ce:4a:eb:bd:41:e5:36:98:b9:cb:
            fd:6d:3c:96:68:df:23:2a:42:90:0c:86:74:67:c8:
            7f:a5:9a:b8:52:61:14:13:3f:65:e9:82:87:cb:db:
            fa:0e:56:f6:86:89:f3:85:3f:97:86:af:b0:dc:1a:
            ef:6b:0d:95:16:7d:c4:2b:a0:65:b2:99:04:36:75:
            80:6b:ac:4a:f3:1b:90:49:78:2f:a2:96:4f:2a:20:
            25:29:04:c6:74:c0:d0:31:cd:8f:31:38:95:16:ba:
            a8:33:b8:43:f1:b1:1f:c3:30:7f:a2:79:31:13:3d:
            2d:36:f8:e3:fc:f2:33:6a:b9:39:31:c5:af:c4:8d:
            0d:1d:64:16:33:aa:fa:84:29:b6:d4:0b:c0:d8:7d:
            c3:93
        Exponent: 65537 (0x10001)
```

Certificates

crt.sh ID	Not Before	Not After	Issuer Name
47997543	2016-10-06	2021-10-06	C=US, O=Internet Security Research Group, CN=ISRG Root X1
15706126	2016-03-17	2021-03-17	O=Digital Signature Trust Co., CN=DST Root CA X3

Let’s talk about something else

How about finding interesting subdomain and Hack ‘em before public release ? :)

Maybe we found git, dev, storage, firewall, etc?
:)

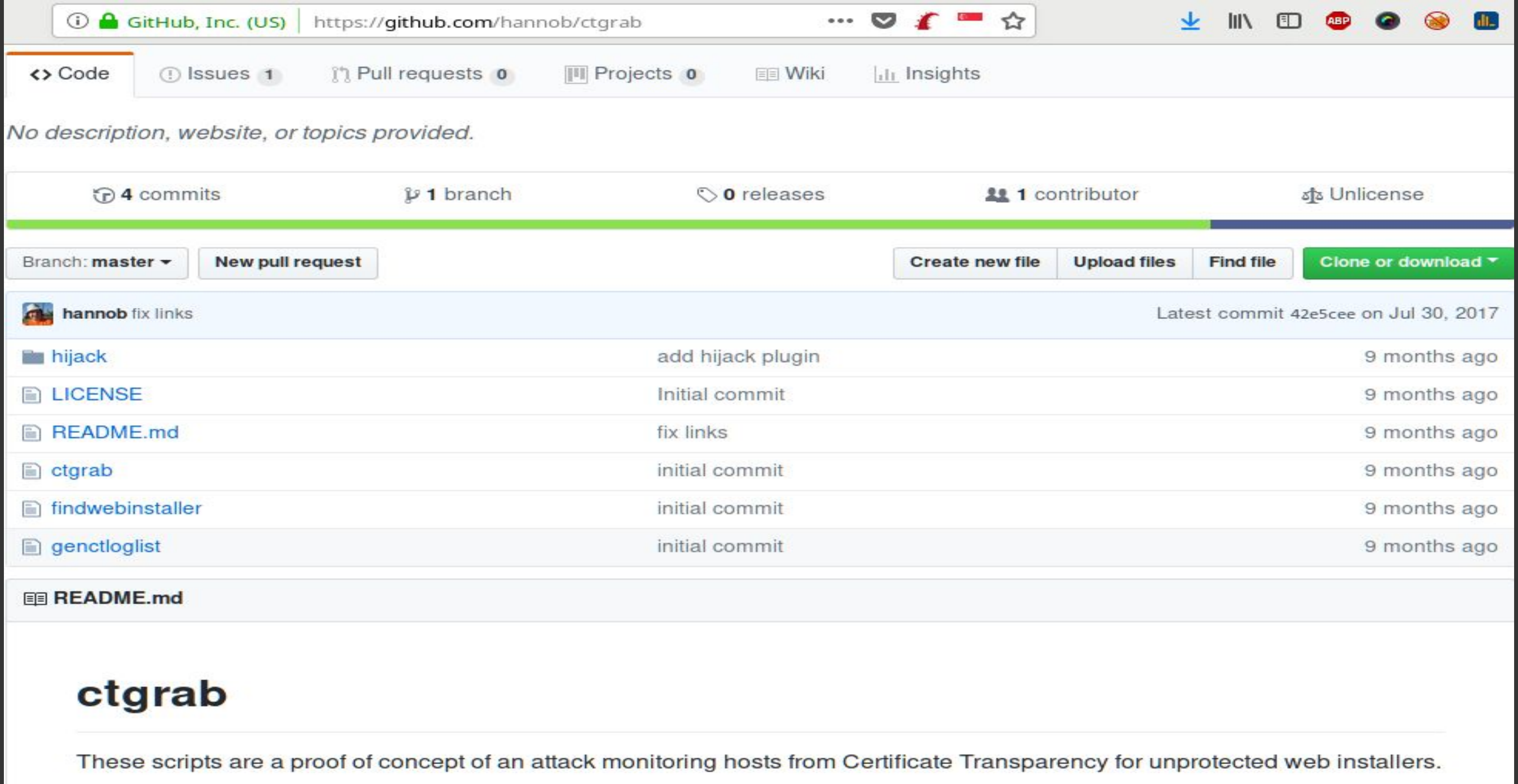
Public exploit available? Default/weak credential?

Or Attacking Content Management Systems?

I mean, CMS installer system

With no authentication

Thanks to Hanno Böck for the automation script called ‘ctgrab’



No description, website, or topics provided.

hannob fix links — Latest commit 42e5cee on Jul 30, 2017

hijack	add hijack plugin	9 months ago
LICENSE	Initial commit	9 months ago
README.md	fix links	9 months ago
ctgrab	initial commit	9 months ago
findwebinstaller	initial commit	9 months ago
genctloglist	initial commit	9 months ago

ctgrab

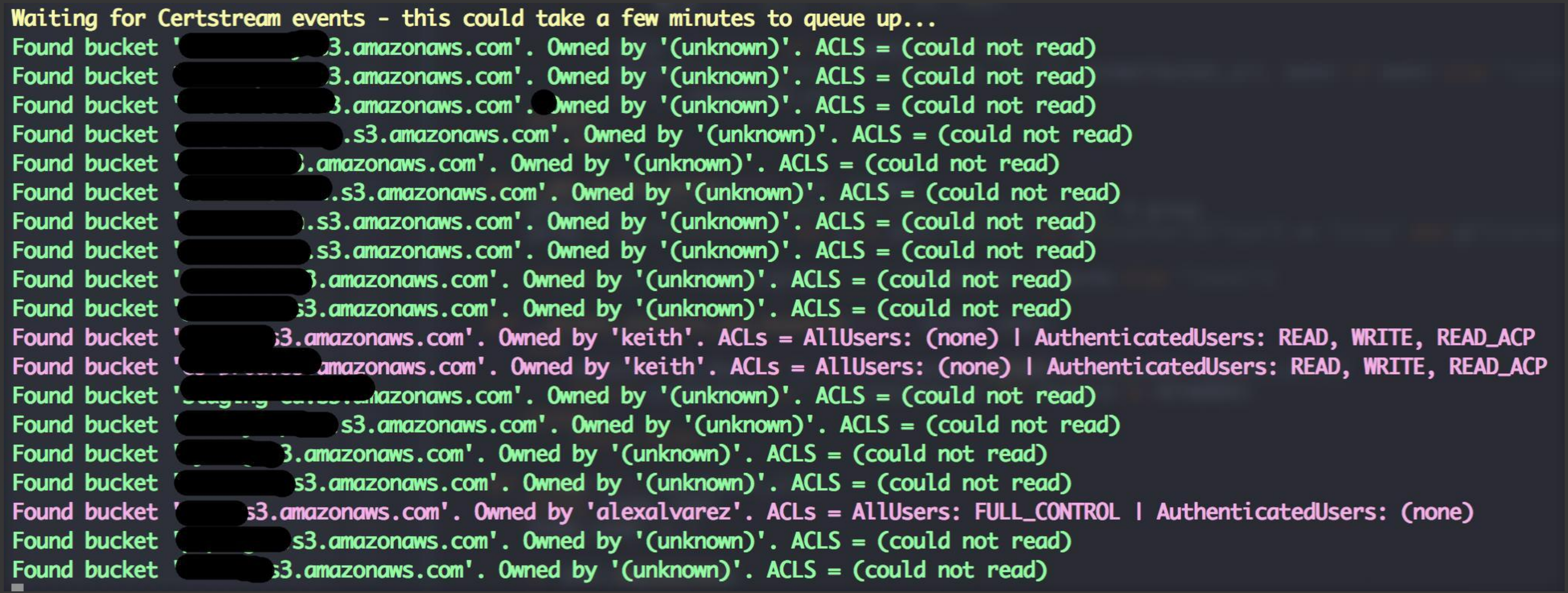
These scripts are a proof of concept of an attack monitoring hosts from Certificate Transparency for unprotected web installers.

Or finding S3 buckets from CT logs?

Watch ‘em with bucket-stream

This tool simply listens to various certificate transparency logs (via certstream) and attempts to find public S3 buckets from permutations of the certificates domain name.

bucket-stream



```
Waiting for Certstream events - this could take a few minutes to queue up...
Found bucket '[redacted]3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket [redacted]3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]3.amazonaws.com'.[redacted]wned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted].s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted].amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted].s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted].s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted].s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]3.amazonaws.com'. Owned by 'keith'. ACLs = AllUsers: (none) | AuthenticatedUsers: READ, WRITE, READ_ACP
Found bucket '[redacted]amazonaws.com'. Owned by 'keith'. ACLs = AllUsers: (none) | AuthenticatedUsers: READ, WRITE, READ_ACP
Found bucket '[redacted]azonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]s3.amazonaws.com'. Owned by 'alexalvarez'. ACLs = AllUsers: FULL_CONTROL | AuthenticatedUsers: (none)
Found bucket '[redacted]s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
Found bucket '[redacted]s3.amazonaws.com'. Owned by '(unknown)'. ACLS = (could not read)
```

https://github.com/eth0izzle/bucket-stream

So, is your server secured?

Conclusions

- Secure your servers and application end points better. Put everything sensitive behind authentication
- Restrict IP from accessing server
- Using wildcard certificates, so you are not revealing sub-domain(Not recommended)
- Accept the fact that all your SSL/TLS protected domains/sub-domains will get listed in a public CT log file.
- Deploy your own Public Key Infrastructure(PKI), to avoid CT log by public CA.
- Redact sub-domain information from CT logs when your CA support for name redaction. https://tools.ietf.org/html/draft-strad-trans-redaction-01
- Opt-out of Certificate Transparency, if your CA supports it

Thanks for your attentions!