Shining a New Light

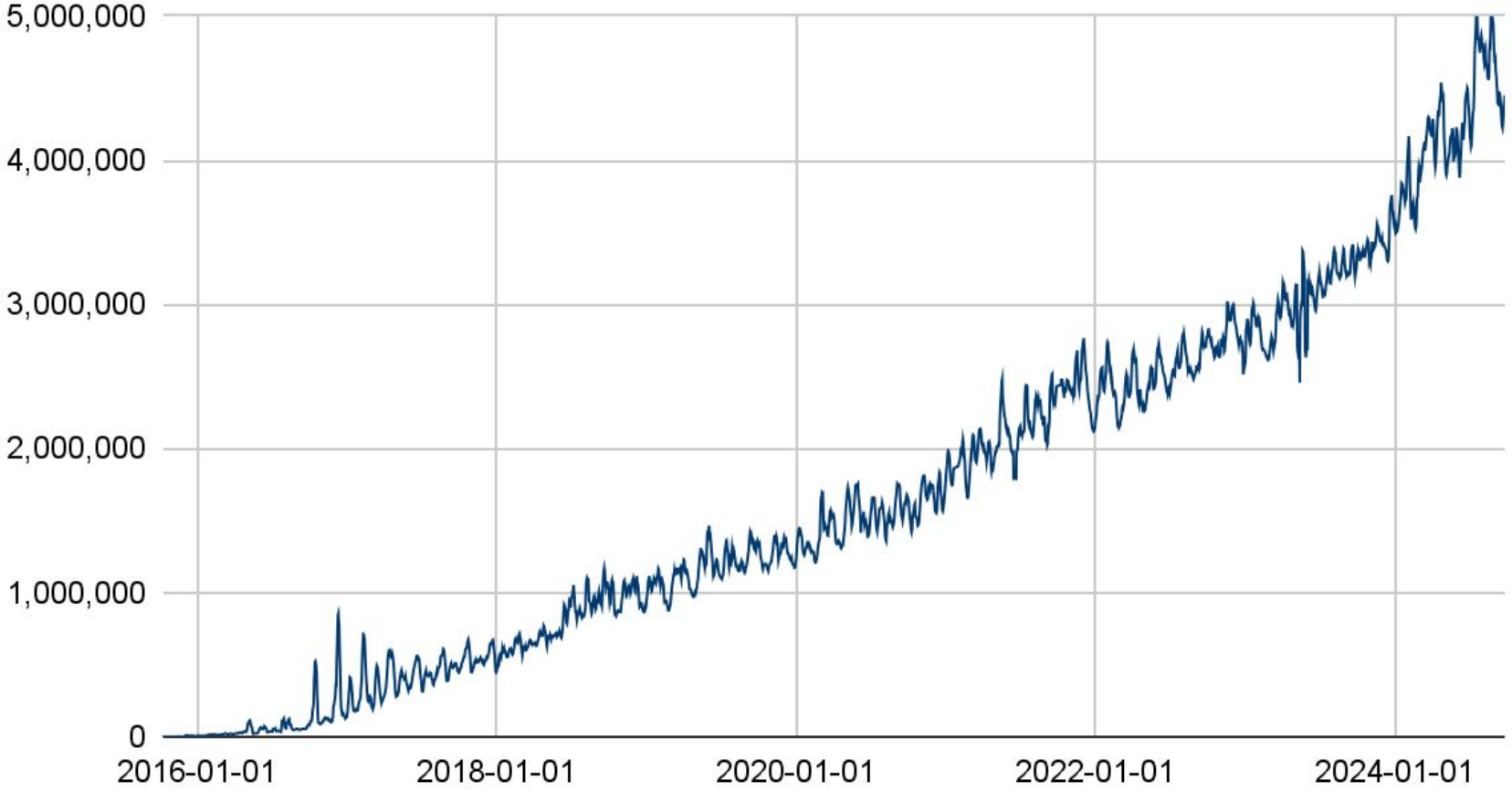
Certificate Transparency at Let's Encrypt

Matthew McPherrin

https://mcpherrin.ca/
https://infosec.exchange/@mattm
@mattm.bsky.social

Let's Encrypt is a free, automated, and open certificate authority brought to you by the nonprofit Internet Security Research Group

Let's Encrypt Certificates Per Day



Certificate Transparency

Let’s Encrypt’s Logs

Oak launched 2019

Running Trillian Implementation

Kubernetes and MariaDB on AWS

Over **4.5 Billion** Certificates Logged

Merged 1846049 Add LE OAK CT Logs and Remove DNS API Endpoints

Change Info

SHOW ALL SIGN IN

Submitted Oct 08, 2019
Owner Devon O'Brien
Uploader Commit Bot
Reviewers Ryan Sleevi +1, Commit Bot
CC certificate-tra..., martijn+crwat..., rsleevi+watch..., chromium-revi...
Repo | Branch chromium/src | master

Submit Requirements

Code-Review +1

Trigger Votes

Commit-Queue +2

```
Add LE OAK CT Logs and Remove DNS API Endpoints

Add Let's Encrypt Oak 2019, 2020, 2021, 2022 CT Logs as Qualified CT
Logs. Additionally, remove DNS API Endpoints from log_list.json now that
they are no longer used.

Bug: 963693
Change-Id: I9542ef6d6e4ecc87bdddcc59192a2bcce80f87a0
Reviewed-on: https://chromium-review.googlesource.com/c/chromium/src/
+/1846049
Reviewed-by: Ryan Sleevi <rsleevi@chromium.org>
Commit-Queue: Devon O'Brien <asymmetric@chromium.org>
Cr-Commit-Position: refs/heads/master@{#703880}
```

Comments No comments
Checks No results

Files Comments Checks

Base → Patchset 2 84b4433

File
Commit message
M components/certificate_transparency/data/log_list.json

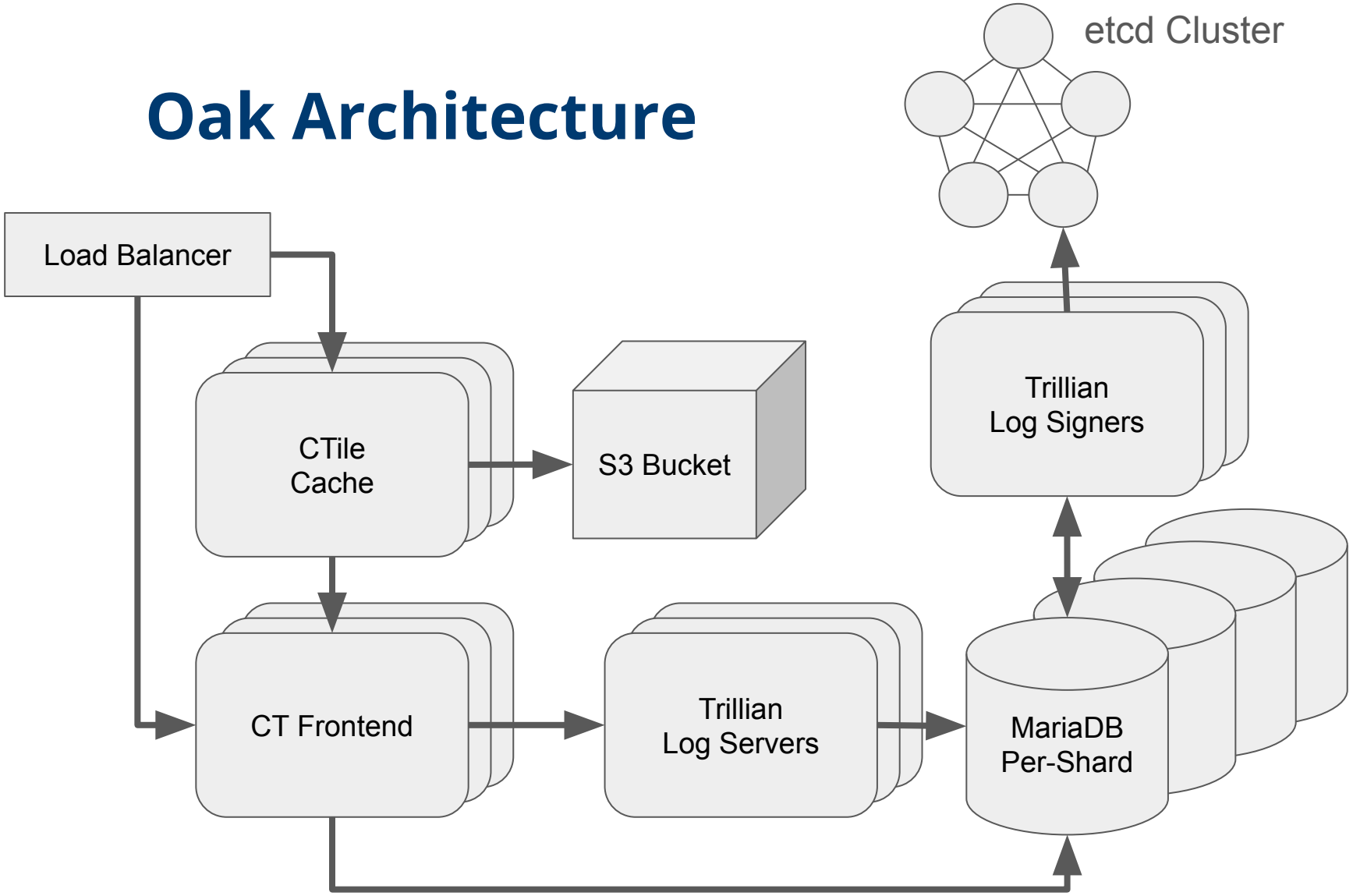
Sharding Logs

Oak isn’t one 4.5 Billion cert log!

- Oak 2019, 2020, 2021, 2022, 2023
- Oak 2024h1, 2024h2
- Oak 2025h1, 2025h2
- Oak 2026h1, 2026h2

Up to 1.2 Billion entries per shard

Oak Architecture



Oak Resources

Compute VMs: 7 x c7a.2xlarge

8 vCPU, 16 GiB RAM

Database: Per-Shard MariaDB RDS, r7g.large

4 vCPU, 32 GiB RAM

8 terabytes storage (when full)

Bandwidth: 10TB per day

CTile: S3-backed Cache

Proxy for get-entries endpoint

Offload read traffic to S3

Avoids needing much larger database

Big reliability increase!

Incidents

We've had problems!

- DB overload hurt uptime
- 16TB Limit killed TestFlume
- Near-misses:
  - No signer running
  - Multiple signers running

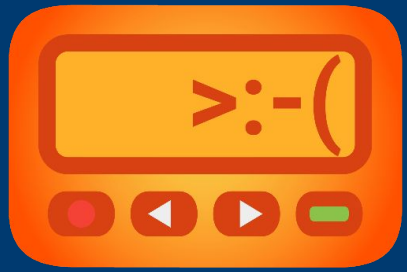
How Logs Die

- Maximum Merge Delay
  - 24 Hours

- Corruption
- Security
- Downtime

Don’t Wake Me Up!

How do we fix our problems?



A New CT Log

Wishlist for a New Log

1. Simple to run

Wishlist for a New Log

2. Eliminate the merge delay

Wishlist for a New Log

3. Single writer

Wishlist for a New Log

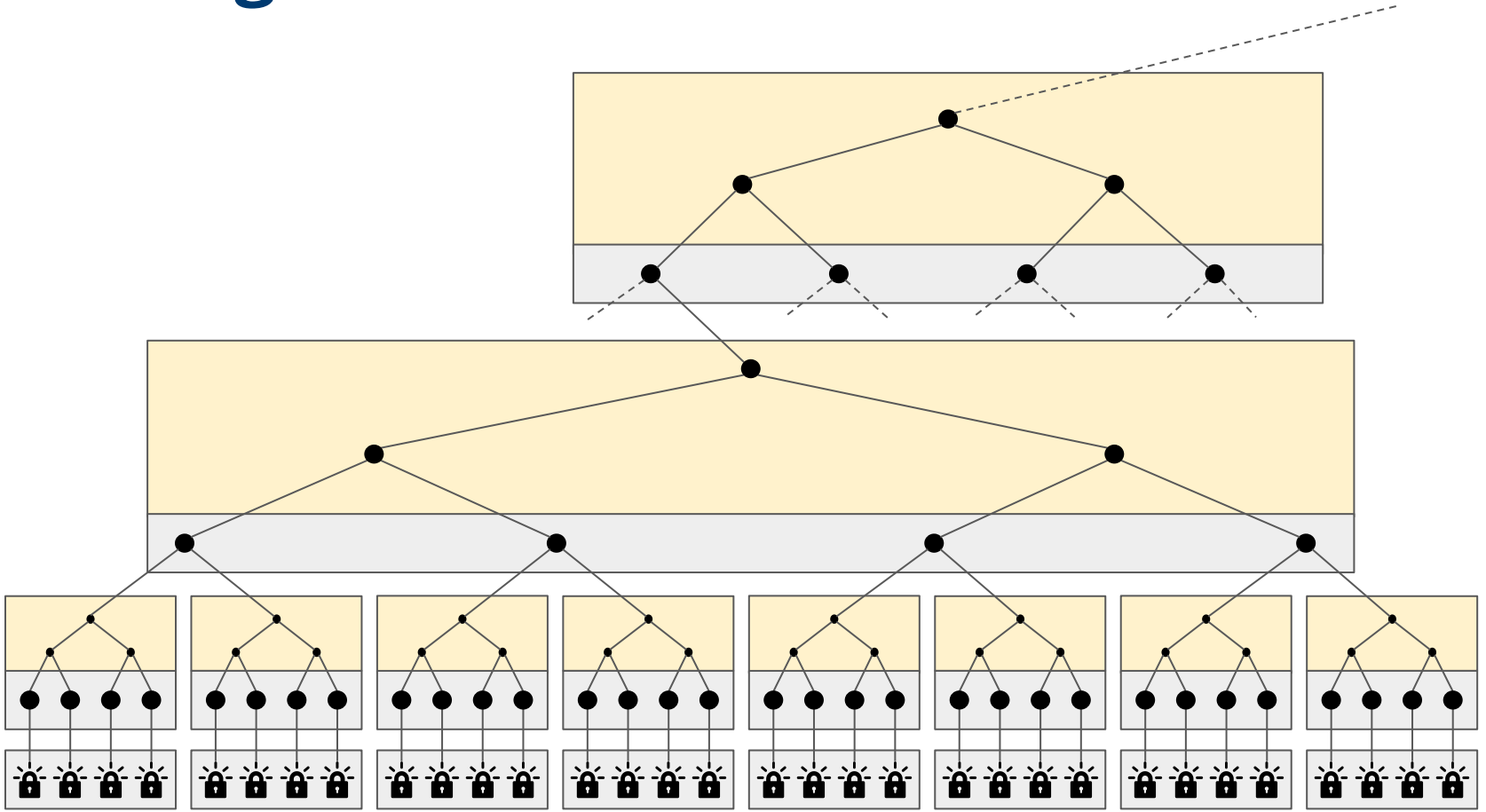
4. Object storage

Static CT API

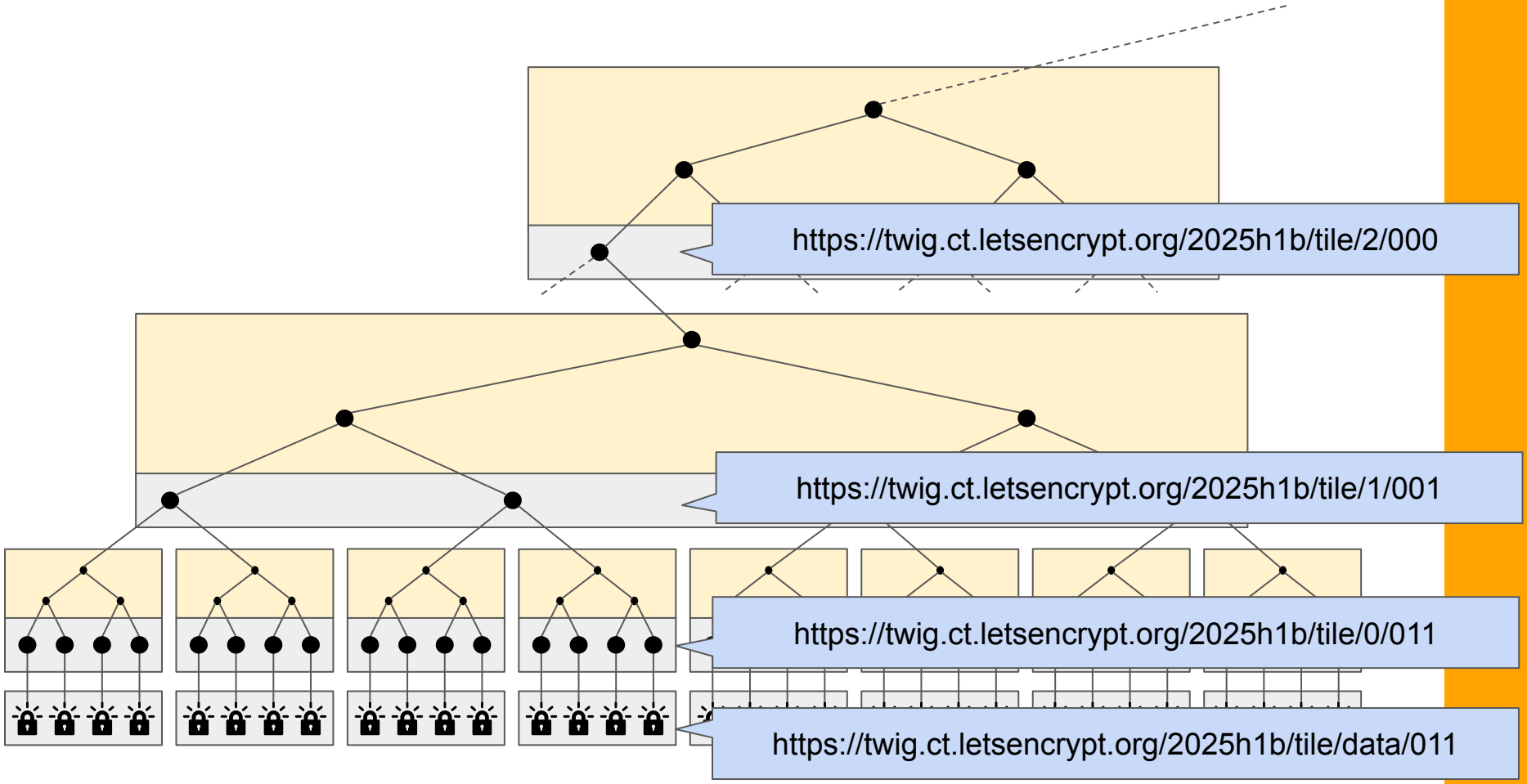
Tiles for Certificate Transparency

https://c2sp.org/static-ct-api

Log Tiles

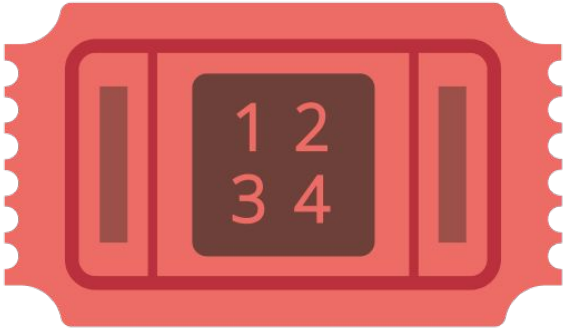


Tile URLs


https://twig.ct.letsencrypt.org/2025h1b/tile/2/000
https://twig.ct.letsencrypt.org/2025h1b/tile/1/001
https://twig.ct.letsencrypt.org/2025h1b/tile/0/011
https://twig.ct.letsencrypt.org/2025h1b/tile/data/011

SCT Index Extension

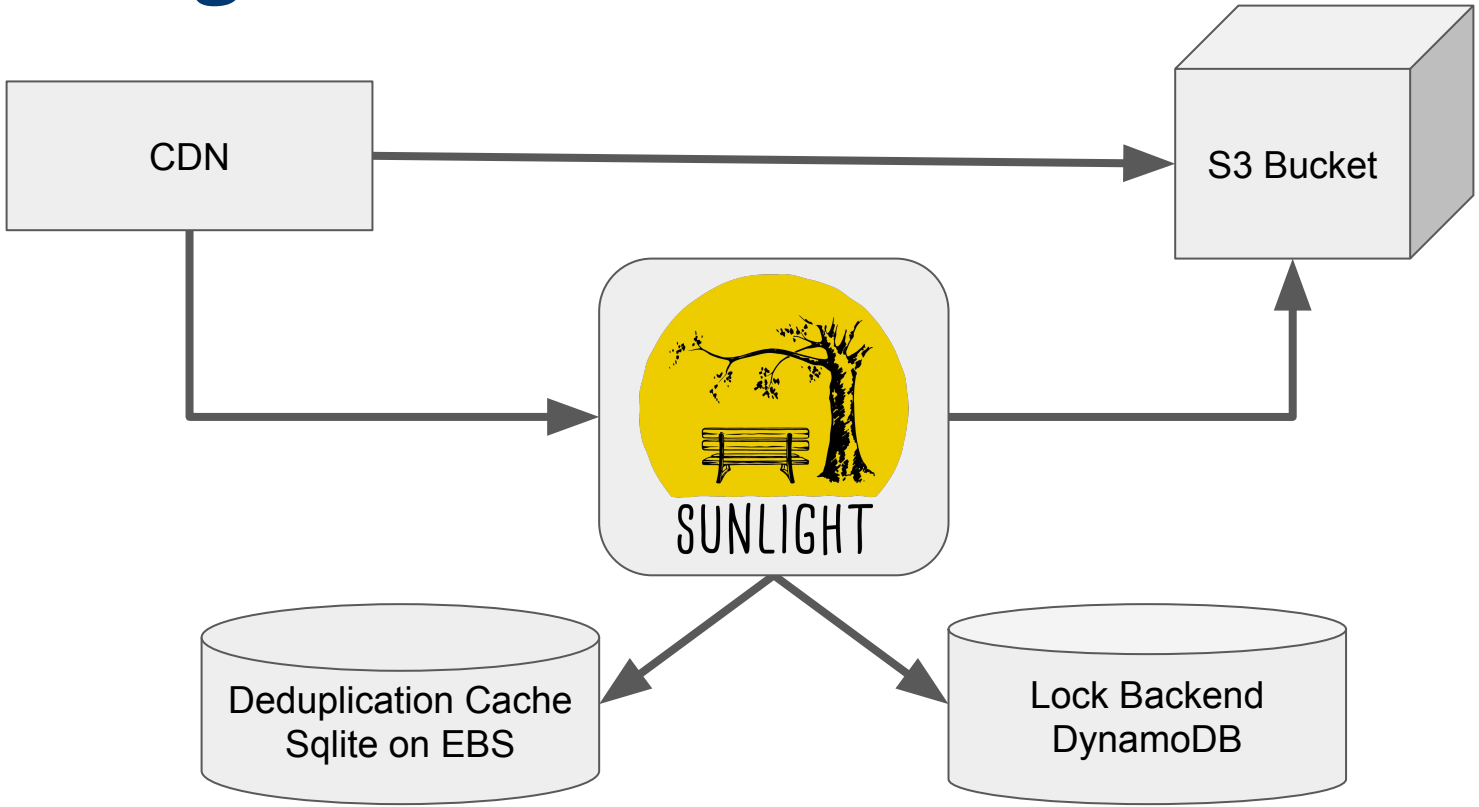
- Index of Certificate within log
- Alternative for get-proof-by-hash



Putting the pieces together

SUNLIGHT

Sunlight Architecture



Sunlight Resources

Compute VMs: 1x t4g.small

- 2 vCPU, 2 GiB RAM
- 100 GB local disk for deduplication

DynamoDB Lock Backend (tiny!)

Object Storage

CDN Bandwidth

Let’s Encrypt Sunlight logs

Test log: https://twig.ct.letsencrypt.org

Production Logs:

https://sycamore.ct.letsencrypt.org

https://willow.ct.letsencrypt.org

Docs: https://letsencrypt.org/docs/ct-logs/

Design Tradeoffs

Single-Writer Uptime

- CT log policy SLO is 99% uptime
- Even targeting 99.5%, plenty of margin
- Workload orchestrator handling rescheduling

No Merge Delay

- Corollary: Need to wait for merge!

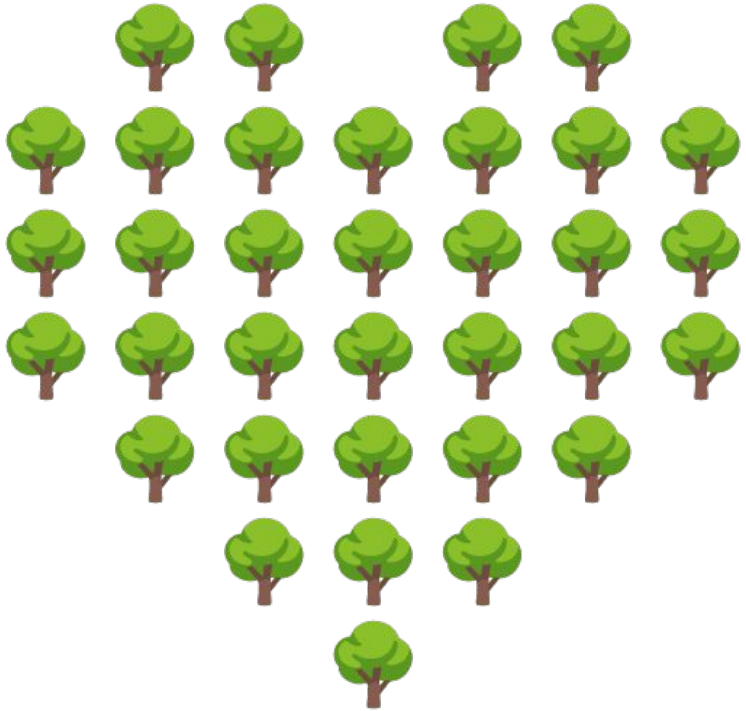
- 75 percentile latency:
  - Oak: 100ms
  - Sycamore: 1s

Static CT API

- Requires log monitors to update
- SCT Extension might cause problems

Worthwhile?

More (Smaller, Cheaper) Logs!



More Logs
=
More Robust Ecosystem

An Example Outage

- AWS S3 unavailable in us-east-2
  - 2024-10-07, 22 minutes
- Sycamore was unavailable
  - Failed to sequence: stopped accepting reqs
  - No SCTs returned for unsequenced certs
- Twig & Willow still available!
- Oak’s read path significantly degraded

Lots of Options

What's Next?

- Log Monitors supporting Static CT API
- Browsers trusting Sunlight Logs
- CAs including Sunlight SCTs

RFC 6962 Proxy: https://github.com/AGWA/sunglasses/

Let's Encrypt

Resources

https://sunlight.dev/

https://c2sp.org/static-ct-api

#sunlight on transparency.dev slack

Thanks to the organizations and individual supporters who help us make a global impact via Let's Encrypt, Sunlight, Divvi Up, and Prossimo.

You'll find more about Let's Encrypt and ISRG's other projects at abetterinternet.org.

If you're interested in becoming a sponsor, reach out! sponsor@betterinternet.org